UNIVERSIDAD POLITÉCNICA DE MADRID

ESCUELA TÉCNICA SUPERIOR
DE INGENIEROS DE TELECOMUNICACIÓN

**MONITORIZACIÓN AUTOMÁTICA DE SISTEMAS
DE SEGUIMIENTO MULTISENSOR EN
APLICACIONES DE CONTROL DE TRÁFICO
AÉREO**

TESIS DOCTORAL

ANDRÉS SOTO JARAMILLO
Ingeniero de Telecomunicación

2020

Departamento de Señales, Sistemas y Radiocomunicaciones

ESCUELA TÉCNICA SUPERIOR
DE INGENIEROS DE TELECOMUNICACIÓN

UNIVERSIDAD POLITÉCNICA DE MADRID

**MONITORIZACIÓN AUTOMÁTICA DE SISTEMAS
DE SEGUIMIENTO MULTISENSOR EN
APLICACIONES DE CONTROL DE TRÁFICO
AÉREO**

TESIS DOCTORAL

Autor:

ANDRÉS SOTO JARAMILLO

Ingeniero de Telecomunicación

Director:

GONZALO DE MIGUEL VELA

Doctor Ingeniero de Telecomunicación

2020

Departamento: Departamento de Señales, Sistemas y Radiocomunicaciones
Escuela Técnica Superior de Ingenieros de Telecomunicación
Universidad Politécnica de Madrid

Tesis Doctoral: MONITORIZACIÓN AUTOMÁTICA DE SISTEMAS DE SEGUIMIENTO
MULTISENSOR EN APLICACIONES DE CONTROL DE TRÁFICO
AÉREO

Autor: Andrés Soto Jaramillo
Ingeniero de Telecomunicación

Director: Gonzalo de Miguel Vela
Doctor Ingeniero de Telecomunicación

El Tribunal nombrado por el Excmo. Sr. Rector Magfco. de la Universidad Politécnica de Madrid,
el día de de 2021, para juzgar la Tesis arriba indicada, compuesto por los
siguientes doctores:

Presidente: _____

Vocal: _____,

Vocal: _____,

Vocal: _____,

Secretario: _____,

realizado el acto de defensa y lectura de Tesis el día de de 2021 en la Escuela
Técnica Superior de Ingenieros de Telecomunicación, acuerda otorgarle la siguiente

Calificación:

EL PRESIDENTE

LOS VOCALES

EL SECRETARIO

Madrid, a de de 2021

*A Anina, mi incansable compañera, que pese a posponer planes y más planes nunca
ha dejado de insuflarme ánimo y energía en esta larga travesía.
No dudes que nos cansaremos de hacer planes juntos.*

*A mis padres, porque les debo todo lo que soy, porque sin su apoyo incondicional
y dedicación sin medida nada de esto se hubiese materializado,
y sobre todo por lo mucho que los quiero.*

A mis profesores, porque valoro inmensamente todo el conocimiento que me han transmitido.

AGRADECIMIENTOS

Alcanzado este día, sobre el cual nunca perdí la esperanza en que llegase, pero del que en algún momento dudé si llegaría... Me resulta difícil resumir en unas pocas palabras la inmensa liberación que siento tras concluir este reto al que me lancé, hace ya mucho tiempo, sin saber muy bien dónde me metía (porque esto de investigar y tratar de innovar engancha) y que, dada mi persistencia, al final he conseguido concluir. Pero andar este camino lleno de picos y de valles contando sólo con mi persistencia, no hubiese sido suficiente. De ahí que haya sido necesario el ánimo, el cariño y la ayuda de muchas personas. Este largo período de mi vida, que ahora concluye, espero que no suponga un punto final, sino solo un punto y aparte, y dé paso a nuevos y apasionantes retos que en algunos casos llevan aparcados bastante tiempo.

Me gustaría que mis primeras palabras de agradecimiento fuesen para mi segunda familia, la del GPDS, ya que por mucho tiempo que pase, entre visita y visita, siempre me siento como en casa. Esto ha hecho que nunca se haya roto el cordón umbilical que en cierta medida ha mantenido viva la llama de mis investigaciones. Así que muchas gracias a José Ramón, Javier, Juan, Ana y muy especialmente a mi director de tesis, Gonzalo, con el que tantas ideas he intercambiado, por los buenos consejos que siempre me ha dado y porque en este frenético final nunca ha dudado que lo conseguiríamos.

También es de agradecer el apoyo que he tenido de mis compañeros y amigos de Indra, que siempre han estado ahí cuando los he necesitado. Gracias en especial a Miguel Muñoz y a Esther Delgado por proporcionarme la información relativa a las herramientas de evaluación de prestaciones de Indra. Ya sabéis que os debo una cerveza en Bruselas cuando la situación lo permita.

Otro agradecimiento muy especial es para Alex de NAV Portugal, por facilitarme los datos reales que he usado durante todo el desarrollo del prototipo que finalmente ha sustentado mis investigaciones. Sin que lo supieses, esos datos han sido el revulsivo que necesitaba para dar el empujón definitivo a este trabajo. Por eso y por otras cosas, ¡muchas gracias! Muito obrigado!

No podía dejar pasar también el apoyo que otros colegas de la industria han brindado a este trabajo suministrando información de sus productos comerciales como ha sido el caso de Arberita Shoshi y Nicolas Honore de Thales, y de muchos de mis actuales compañeros de EUROCONTROL.

Tampoco podía olvidar a mi familia: mis padres, Andrés y María, y a mi hermana, María que me han dado todo su amor, su cariño y su apoyo durante toda esta andanza.

Agradecer, como no, a Anina, su paciencia infinita, sus constantes ánimos, y sobre todo esas concienzudas revisiones que ha realizado de todos y cada uno de los capítulos y anexos de esta tesis, que han contribuido a homogenizar este documento y que tiene gran mérito al tratarse de un texto extenso y, todo sea dicho, no sencillo de leer. No dudes que serás convenientemente recompensada, que yo sé que lo estas esperando.

Por último, habrá que retomar, cuando este dichoso COVID nos deje, esa vida social que hace meses que tengo olvidada y devolver a mis amigos el afecto y los ánimos que me han transmitido durante todo este tiempo.

RESUMEN

La necesidad cada vez más extendida de evaluar las prestaciones de los sistemas para garantizar una determinada calidad de servicio (QoS) está motivando la aparición de herramientas encargadas de dicha evaluación a las que se les suele exigir adicionalmente el que sean eficientes tanto desde un punto de vista de uso de recursos, no solo materiales sino también inmateriales, como desde un punto de vista económico.

El entorno de la gestión del tráfico aéreo (ATM), que es el ámbito en el que se desarrolla esta tesis, también reproduce esta tendencia asentada desde hace ya unas décadas. Programas orientados a la evolución de los sistemas que se encargan de la gestión del tráfico aéreo, como pueda ser el europeo, SESAR, ya preveía desde sus inicios una evolución en la manera de gestionar el tráfico aéreo desde lo que era o sigue siendo una gestión basada en espacio aéreo, hacia lo que debía ser una gestión basada en trayectorias, que tiene como siguiente hito la incorporación de las prestaciones en el servicio provisto a aeronaves y a aerolíneas, es decir, a los usuarios del espacio aéreo. Estas prestaciones se entienden no solo desde el punto de vista del receptor del servicio, es decir, los usuarios del espacio aéreo, sino también del lado del proveedor del servicio, es decir, los conocidos como proveedores de servicios de navegación aérea (ANSP).

Las prestaciones ofrecidas por unos y por otros acabarán siendo las que determinen qué servicios -y con qué nivel de calidad-, pueden ser solicitados por los usuarios y ofertados por la infraestructura establecida por el ANSP. Dentro de los servicios ofrecidos por la gestión de tráfico aéreo, esta tesis se centra en los relacionados con la provisión del control de tráfico aéreo, es decir, en los relacionados en garantizar la adecuada separación de aeronaves entre ellas y con respecto al terreno. Dada la importancia de este cometido, hace que sea necesario, no solo una evaluación puntual del desempeño de los sistemas encargados de proveer dichos servicios, sino una monitorización continua y, a poder ser, automática de los mismos que permita garantizar que trabajan en unas condiciones óptimas para garantizar la eficiencia y la seguridad de las operaciones.

Dentro de los diferentes dominios que sostienen la provisión de servicios de gestión de tráfico aéreo (CNS/ATM), el que centra la atención de esta tesis es el de vigilancia (S). En concreto, es la evaluación de las prestaciones de la cadena de vigilancia la que concentra todo el interés. Ésta comprende desde los sensores que detectan a los blancos, aeronaves o vehículos aeroportuarios, hasta la presentación de los datos en las pantallas que muestran la situación actual del tráfico a los controladores, pasando por las redes y sistemas de comunicaciones (e.g. enrutadores o pasarelas de datos), continuado con los sistemas de seguimiento y finalizando con las herramientas de evaluación de prestaciones.

Esta tesis comienza (Capítulo 1) con una aproximación que parte del concepto de gestión del tráfico aéreo para ir descendiendo hasta llegar al ámbito en el que se desarrolla toda la investigación llevada a cabo en la misma y que no es otro que lo que se conoce como la función de vigilancia. Sus elementos y sistemas se presentan estableciendo la relación que existe entre ellos. Dicha función suele estar dividida en lo que se conoce como vigilancia de aire, si los blancos se encuentran volando, y vigilancia de superficie, en caso de encontrarse en tierra. En esta tesis el foco está puesto en la evaluación de las prestaciones de una cadena de vigilancia de superficie de aeropuerto, aunque en ningún momento se pierde de vista que la frontera entre ambas es ciertamente difusa y, por lo tanto, se espera cierto trasiego de información entre ambas. Dado que el entorno donde

se van a centrar las investigaciones es el aeropuerto, éste se describe junto con sus elementos integrantes ya que algunas de sus características serán aprovechadas por los algoritmos desarrollados como parte de esta tesis.

Dentro de la cadena de vigilancia, serán varios de sus elementos los que sean sometidos a la evaluación de sus prestaciones, en concreto, sensores y sistemas de seguimiento. Sus características, tipos, particularidades, modelos de medida, coordenadas en las que proporcionan sus datos y el formato usado para su envío son el objeto del Capítulo 2. Esta presentación se realiza por familias de sensores empezando por los radares, que es claramente la familia dominante a día de hoy, para también presentar la vigilancia dependiente automática (ADS), sobre todo la que se conoce como ADS-B, y la familia de sensores basados en multilateración, tanto de área extensa (WAM) como los más locales que se conocen como MLAT o LAM. No se pierde de vista que, actualmente, existe una tendencia a la incorporación de otra familia de sensores, algo más recientes que los anteriores, que buscan la detección y localización de los blancos de manera independiente y sin la necesidad de la cooperación de éste. Esta familia se conoce por sus siglas en inglés como sensores INCS. A continuación, el capítulo también presenta aspectos de la transmisión de los datos desde los sensores a los centros de procesamiento como pueda ser el formato utilizado, que no deja de ser el bien conocido formato ASTERIX, los protocolos de transmisión o las redes de comunicaciones. Finalmente, el capítulo concluye con un listado de problemas conocidos de los sensores presentados y con una visión prospectiva de lo que puede ser la vigilancia del futuro.

El siguiente capítulo (Capítulo 3) se centra en los sistemas que van a encargarse de la evaluación de prestaciones tanto de sensores como de sistemas de seguimiento. Esta evaluación es llevada a cabo por lo que se conocen como herramientas de evaluación de prestaciones (PET). Este capítulo desarrolla la historia de estas herramientas, presenta los estándares que las rigen, las clasifica en diferentes tipos y detalla los métodos que usan para realizar su evaluación. Ejemplos representativos del estado del arte de estas herramientas son incorporados incluyéndose imágenes de su interfaz gráfica. A continuación, se realiza el ejercicio de encajarlas dentro de la arquitectura definida en el programa SESAR, para acto seguido realizar una descripción pormenorizada de lo que sería una descomposición lógica y funcional de estas herramientas. Esto último constituye una primera aportación de esta tesis.

El Capítulo 4 aprovecha la descomposición funcional realizada en el capítulo anterior para identificar la sucesión ordenada de procesos que se debe de encargarse de implementar las funcionalidades necesarias para que la herramienta pueda al final de dicha sucesión proporcionar el resultado esperado. Estos procesos se agrupan en lo que se han llamado bloques de procesamiento. Excepto por el bloque encargado de la presentación de las prestaciones y aquellos que se consideran auxiliares, el resto son desarrollados en un capítulo aparte.

Así, las cuestiones relativas a la lectura y cribado inicial de los datos generados por sensores y sistemas de seguimiento se presentan en el Capítulo 5. Cabe destacar que se trata de procesos que, si bien no tienen relevancia a nivel investigador, sí que resultan de interés para comprender toda la cadena de procesamiento ya que en ellos comienza todo. Sin una buena adquisición de los datos, el resto de elaboraciones acaban siendo papel mojado.

Los procesos encargados de preparar los datos para su posterior fusión con los generados por otros sensores son detallados en el Capítulo 6. Estos procesos implementan lo que se

conoce como los prerequisites de la fusión de datos que consisten: en la conversión de todos los datos a un mismo sistema de coordenadas de referencia en el que puedan ser comparados y en la estimación y posterior cancelación de cualquier error sistemático o sesgo que pueda haber sido incorporado a los datos durante el proceso de medición del sensor o en el camino recorrido hasta llegar al centro de fusión. Esta tesis aporta una arquitectura completa no solo para la estimación de los sesgos introducidos por el sensor o de aquellos dependientes de la aeronave, sino también de los sesgos temporales derivados de una falta de sincronización entre los diferentes elementos de la cadena de transmisión. Es importante destacar que en dicha propuesta se incorpora información de contexto como pueda ser el mapa del aeropuerto o las características físicas de la aeronave, que permite mejorar los resultados del proceso. Entre las aportaciones realizadas en este capítulo cabe destacar el algoritmo que permite la identificación del segmento del aeropuerto en el que se encuentra cada uno de los datos recibidos, ya que éste supone el punto de partida de toda la estrategia de estimación de sesgos. Esta estrategia primero identifica un sensor de referencia, al cual se le corrigen los sesgos de manera independiente, para posteriormente conformar pares de medidas construidos a partir de los datos del sensor de referencia, que se considera insesgado, y de cada uno de los sensores restantes. Estos pares serán finalmente procesados por algoritmos basados en diferencias de medidas para obtener los sesgos de los otros sensores.

El Capítulo 7 representa la motivación principal de esta tesis que no deja de ser la reconstrucción de la trayectoria ideal seguida por las aeronaves ya que ésta constituye la referencia contra la que se compararán los datos generados por sensores y sistemas de seguimiento. De dicha comparación se extraerán los indicadores que permitirán valorar el desempeño de los sistemas bajo análisis. La reconstrucción de trayectorias aprovecha el conocimiento que se tiene del aeropuerto y de las aeronaves, para obtener una estimación del camino seguido por los blancos. Dicha estimación es proporcionada por un algoritmo de suavizado que es capaz de incorporar tanto la información del pasado como del futuro con respecto a un determinado instante. El diseño de dicho algoritmo también representa una aportación de esta tesis ya que no solo integra la información de contexto, sino que considera modelos de movimiento no considerados previamente en este ámbito que permiten mejorar la estimación de la trayectoria ideal.

Los resultados de las pruebas realizadas sobre datos reales del aeropuerto de Lisboa se presentan en el Capítulo 8. En ellos se puede apreciar cómo los diferentes algoritmos presentados en los capítulos anteriores desarrollan su cometido de manera muy satisfactoria. En aquellos casos en los que se aprecian diferencias con respecto a los resultados esperados, se trata de justificar la razón de los mismos y si se considera pertinente se añade dicha cuestión a la lista de temas a investigar en un futuro.

Por último, el Capítulo 9 presenta lo que se consideran como dinamizadores actuales o futuros de esta tendencia enfocada en la evaluación de prestaciones para resumir, a continuación, las aportaciones realizadas por esta tesis e identificar las áreas que podrían dar lugar a futuras líneas de investigación en este campo. El trabajo concluye con una mirada prospectiva hacia lo que podría ser el futuro de las herramientas de evaluación de prestaciones en entornos de gestión de tráfico aéreo. Futuro éste que pinta bastante prometedor y lleno de oportunidades.

ABSTRACT

The increasingly widespread need for assessing the performance of systems in order to guarantee a certain quality of service (QoS) is fostering the appearance of tools in charge of performing this assessment, which are additionally required to be efficient both, from the point of view of the use of resources, not only material but also immaterial, as well as from an economic point of view.

The air traffic management (ATM) environment, which is the field in which this thesis is developed, also reproduces this established trend for several decades now. Research and development programmes aiming at the evolution of the systems in charge of air traffic management, such as the European one, i.e. SESAR, already foresaw since the beginning an evolution in the way of managing air traffic from what was or continues to be an airspace-based management, towards what should be a trajectory-based management, whose next milestone would be the incorporation of the performance level into the service provision to aircraft and airlines, that is, to airspace users. This performance level is understood not only from the point of view of the service customer, that is, the airspace users, but also from the side of the service provider, that is, from the entities known as air navigation service providers (ANSP).

The performance level offered by both parties will ultimately determine which services - and with what level of quality - can be requested by users and offered by the infrastructure put in place by the ANSP. Within the services offered by air traffic management, this thesis focuses on those related to the provision of air traffic control, i.e. those related to ensuring adequate separation of aircraft between each other and with respect to the terrain. Given the importance of this task, it is necessary not only to evaluate the performance of the systems responsible for providing these services, but also to continuously and, if possible, automatically monitor them to ensure that they are working in optimum conditions to guarantee the efficiency and safety of operations.

Within the different domains that support the provision of air traffic management services (CNS/ATM), the one that focuses the attention of this thesis is surveillance (S). Specifically, it is the performance evaluation of the surveillance chain what keeps all the focus of interest. This chain ranges from the sensors that detect the targets, e.g. aircraft or airport vehicles, to the presentation of data on the screens that show the situational awareness picture to the controllers, also including communications networks and systems (e.g. routers or gateways), tracking systems and lastly, performance evaluation tools.

This thesis begins (Chapter 1) with a top-down approach that starts from the concept of air traffic management and goes down to the field in which all the research is developed, which is known as the surveillance function. Its elements and systems are presented and the relationship between them is established. This function is usually divided into what is known as air surveillance, if the targets are airborne, and surface surveillance, if they are on the ground. In this thesis, the focus is on the performance evaluation of an airport surface surveillance chain, although without losing sight that the border between the two is certainly diffuse and, therefore, a certain transfer of information between them is expected. Given that the environment where the research will be focused is the airport, it is described along with its building elements since some of their characteristics will be exploited by the algorithms developed as part of this thesis.

Within the surveillance chain, there are several components which might undergo a performance evaluation, in particular, sensors and tracking systems. Their characteristics, types, particularities, measurement models, coordinates in which they provide their data and the format used to send them are the subject of Chapter 2. This presentation is made per sensor families, starting with radars, which are clearly the dominant family nowadays, continuing with automatic dependent surveillance (ADS), especially the one known as ADS-B, and finally, ending with the family of sensors based on multilateration techniques, which includes both, the ones devoted to wide areas (WAM) and the more local ones known as MLAT or LAM. It is worth noting that there is currently a trend towards incorporating another family of sensors, which are more recent than the previous ones, that seeks to detect and locate targets independently and without the need for their cooperation. This family is known as independent and non-cooperative sensors (INCS). The chapter then also addresses aspects related to the data transmission from the sensors to the data processing centres, such as the data format, which is the well-known ASTERIX format, the transmission protocols, or the communications networks. Finally, the chapter concludes with a list of known problems of the presented sensors and with a prospective view of what surveillance can be in the future.

The next chapter (Chapter 3) focuses on the systems that will be responsible for performance evaluation of both, sensors and tracking systems. This evaluation is carried out by performance evaluation tools (PET). This chapter tells the history of these tools, presents the standards that govern them, classifies them into different types and details the methods they use to perform their evaluation. Representative examples of the state of the art of these tools are incorporated including images of their look like. Next, the exercise of fitting them into the architecture defined within the SESAR programme is carried out, followed by a detailed description of what would be a logical and functional decomposition of these tools. The latter constitutes a first contribution of this thesis.

Chapter 4 takes advantage of the functional decomposition made in the previous chapter to identify the ordered succession of processes that must be put in place in order to implement the necessary functionalities so that the tool can, at the end of this succession, provide the expected result. These processes are grouped in what have been called as processing blocks. Except for the block in charge of presenting the performance assessment and those considered as auxiliary, the rest are developed in a separate chapter.

Thus, the processes related to reading and initially cleaning the data received from sensors and tracking systems are presented in Chapter 5. It is pertinent to highlight that these are processes that are not relevant from a research point of view. However, they contribute to get the full picture of the entire processing chain as they represent the start of the data processing. Without a good data acquisition, the rest of the processing ends up being completely useless.

The processes involved in sensor data preparation before their fusion with the data generated by other sensors are detailed in Chapter 6. These processes implement what are known as the prerequisites of data fusion, which consist of the conversion of all data to the same reference coordinate system in which they can be compared, and the estimation and subsequent cancellation of any systematic error or bias that may have been incorporated into the data during the sensor measurement process or on the way to the data processing centre. This thesis provides a complete architecture not only for the estimation of the biases introduced by the sensors or those dependent on the aircraft, but also of the time biases derived from a lack of synchronisation between the different

elements of the transmission chain. It is relevant to highlight that this proposal incorporates contextual information, such as the airport map or the physical characteristics of the aircraft, which allows improving the results of the bias estimation. Among the contributions included in this chapter, it is worth highlighting the algorithm that allows the identification of the airport segment in which each of the data received is located as this is the starting point for the whole bias estimation strategy. This strategy first identifies a reference sensor, whose biases are independently corrected, and then builds measurement pairs with data from the reference sensor, which is considered to be unbiased, and data from each of the remaining sensors. These pairs will finally be processed by algorithms based on measurement differences to estimate the biases of the other sensors.

Chapter 7 represents the main motivation for this thesis, which is the reconstruction of the ideal trajectory followed by the aircraft, since it constitutes the reference against which the data generated by sensors and tracking systems will be compared to. From this comparison, indicators will be computed that will assess the performance of the systems under analysis. The trajectory reconstruction takes advantage of the knowledge of the airport and the aircraft features to obtain an estimate of the path followed by the targets. This estimate is provided by a smoothing algorithm that exploits both, past and future information with respect to a given moment. The design of this algorithm also represents a contribution of this thesis since it not only integrates contextual information, but also relies on movement models not previously considered in this field that allow to improve the estimation of the ideal trajectory.

The testing results carried out on real data from Lisbon airport are presented in Chapter 8. They show how the different algorithms presented in the previous chapters perform their task in a very satisfactory way. In those cases where differences are found with respect to the expected results, the aim is to justify the reason for these differences and, if considered relevant, add this issue to the list of topics to be investigated in the future.

Lastly, Chapter 9 presents the current and future dynamizers of this trend that is focused on the performance evaluation. Next, it summarizes the contributions made by this thesis and, finally, identifies the areas which could give rise to future research lines in this field. The thesis concludes with a forecast concerning the future of performance evaluation tools in air traffic management environments, which looks quite promising and full of opportunities.

CONTENTS

AGRADECIMIENTOS	IX
RESUMEN	XI
ABSTRACT	XV
CONTENTS	XIX
LIST OF FIGURES	XXVII
LIST OF TABLES	XXXI
CHAPTER 1	1
INTRODUCTION AND MOTIVATION	1
1.1 Air Traffic Management	4
1.1.1 Airspace Management	5
1.1.2 Air Traffic Flow Management	6
1.1.3 Air Traffic Services	7
1.1.4 Other services	11
1.2 Air Traffic Control	12
1.2.1 Aircraft and their aircrew	12
1.2.2 Air Traffic Control facilities	12
1.2.3 Air Traffic Control systems	13
1.2.4 Air Traffic Control Operators and Ground Personnel	19
1.3 Surveillance chain	19
1.3.1 Surveillance sensors	19
1.3.2 Surveillance distribution systems	22
1.3.3 Surveillance data processing system	23
1.3.4 Surveillance data presentation	23
1.3.5 Performance Evaluation Tools	24
1.4 The airport environment	24
1.4.1 The apron or platform	25
1.4.2 The runway	25
1.4.3 The taxiway	25
1.4.4 The manoeuvring area	25
1.4.5 The movement area	25
1.4.6 Holding position and stop bars	25
1.5 Increasing air traffic capacity, sustainability and safety. The performance-based operation paradigm	26
1.6 Problem statement	29
1.7 Contribution and Methodology	29
1.8 Structure of the Document	30
CHAPTER 2	35
SURVEILLANCE SENSORS AND THEIR DATA MODELS	35
2.1 Surveillance sensors	36

2.1.1 General features	36
2.1.2 Surveillance sensor types	37
2.1.3 Sensor measurement model	39
2.1.4 Coordinate systems and transformations	40
2.1.5 Surveillance data exchange	44
2.1.6 Surveillance data transmission	52
2.2 Radar systems	55
2.2.1 General considerations	55
2.2.2 Primary surveillance radars	70
2.2.3 Surface movement radars	71
2.2.4 Secondary surveillance radars	73
2.3 Automatic dependent surveillance	78
2.3.1 General considerations	78
2.3.2 Automatic dependent surveillance - Broadcast	79
2.3.3 Automatic dependent surveillance - Contract	94
2.4 Multilateration Systems	95
2.4.1 General considerations	95
2.4.2 LAM or MLAT	104
2.4.3 Wide Area Multilateration (WAM)	106
2.5 Independent and Non-Cooperative Systems	107
2.5.1 General considerations	107
2.6 Already known issues with surveillance systems	108
2.6.1 Time offset	108
2.6.2 Data processing issues	108
2.6.3 Different reference points	109
2.6.4 Dealing with different types of height measurements	110
2.6.5 ADS-B known issues	110
2.7 The future of surveillance	111
CHAPTER 3	115
PERFORMANCE EVALUATION TOOLS FOR SENSORS AND TRACKING SYSTEMS.	115
3.1 Introduction to performance evaluation	116
3.1.1 A little bit of history. From mid-80's to nowadays	116
3.1.2 Definition of performance evaluation tools	116
3.1.3 Objectives of performance evaluation tools	117
3.2 Performance evaluation standards	117
3.3 Taxonomy of performance evaluation tools. Application domains. State of the art	119
3.3.1 Types of performance evaluation tools	119
3.3.2 Performance evaluation tools for surveillance sensors	119
3.3.3 Performance evaluation systems for centres	131
3.4 Different methods for performance evaluation based on reference trajectories	135
3.4.1 Opportunity traffic	136
3.4.2 Flight trials	137
3.4.3 Simulations	137
3.5 SESAR framework for a generic performance evaluation tool	138

3.5.1 Functional decomposition for aerodrome ATC domain system	139
3.5.2 Surveillance function architecture	140
3.6 Architecture proposal for a generic performance evaluation tool	143
3.6.1 Context	143
3.6.2 Functional decomposition. The performance evaluation tool as a data fusion system	148
3.6.3 Logical decomposition. Internal data flows	155
3.6.4 External Interfaces	160
3.7 Requirements imposed by the new paradigm based on SESAR	163
3.8 Conclusions	164
CHAPTER 4	165
PERFORMANCE EVALUATION OVERALL PROCESS	165
4.1 Performance evaluation process overview	166
4.2 Initial configuration	170
4.3 Data reading	170
4.4 Data preparation	171
4.5 Trajectory reconstruction	171
4.6 Performance assessment	173
4.6.1 Data assessment	173
4.6.2 Result presentation	174
4.7 Ancillary processing blocks	174
4.7.1 Sensor measurement model	174
4.7.2 Airport statistics	175
CHAPTER 5	177
DATA READING PROCESSING BLOCK	177
5.1 Data reading processing block overview	178
5.2 Data acquisition	180
5.2.1 Process description	180
5.2.2 Processing details	180
5.2.3 Prototype implementation details	185
5.3 Data cleaning	186
5.3.1 Process description	186
5.3.2 Processing details	187
5.3.3 Prototype implementation details	188
5.4 Data allocation	188
5.4.1 Process description	188
5.4.2 Processing details	188
5.4.3 Prototype implementation details	189
5.5 Data presentation	189
5.5.1 Process description	189
5.5.2 Processing details	190
5.5.3 Prototype implementation details	190

CHAPTER 6	191
DATA PREPARATION PROCESSING BLOCK	191
6.1 Data reading processing block overview	192
6.2 Initial data selection	195
6.2.1 Process description	195
6.2.2 Processing details	195
6.2.3 Prototype implementation details	196
6.3 Known bias cancellation	197
6.3.1 Process description	197
6.3.2 Processing details	197
6.3.3 Prototype implementation details	199
6.4 Coordinate transformation	199
6.4.1 Process description	199
6.4.2 Processing details	200
6.4.3 Prototype implementation details	202
6.5 Final data selection	202
6.5.1 Process description	202
6.5.2 Processing details	202
6.5.3 Prototype implementation details	203
6.6 Time ordering	203
6.6.1 Process description	203
6.6.2 Processing details	205
6.6.3 Prototype implementation details	205
6.7 Data association	205
6.7.1 Problem definition	206
6.7.2 Process description	208
6.7.2.1 Monosensor association	209
6.7.2.2 Multisensor association	210
6.7.2.3 Multisensor trajectory selection	210
6.7.2.4 Target information retrieval	210
6.7.3 Processing details	210
6.7.4 Prototype implementation details	211
6.8 Data location in airport	212
6.8.1 Process description	212
6.8.2 Processing details	212
6.8.3 Prototype implementation details	213
6.9 Movement model identification	220
6.9.1 Process description	220
6.9.2 Processing details	221
6.9.3 Prototype implementation details	222
6.10 Bias estimation and cancellation	226
6.10.1 Problem definition	226
6.10.2 Bias types	226
6.10.3 Bias estimation strategy	228
6.10.4 Bias estimation methods	230

6.10.5 Process description	230
6.10.6 Processing details	231
6.10.7 Prototype implementation details	232
6.11 Noise estimation	257
6.11.1 Process description	257
6.11.2 Processing details	258
6.11.3 Prototype implementation details	258
6.12 Time re-ordering	259
6.12.1 Process description	259
6.12.2 Processing details	259
6.12.3 Prototype implementation details	259
6.13 Association completion	260
6.13.1 Process description	260
6.13.2 Processing details	261
6.13.3 Prototype implementation details	261
CHAPTER 7	263
TRAJECTORY RECONSTRUCTION PROCESSING BLOCK	263
7.1 Reference trajectory estimation	264
7.1.1 The reference trajectory	264
7.1.2 Reference trajectory estimation from opportunity traffic	265
7.2 Taxonomy of current techniques for reference trajectory reconstruction	267
7.2.1 Introduction	267
7.3 Spline-based techniques	268
7.3.1 Spline history	268
7.3.2 Evolution of interpolation techniques	269
7.3.3 Splines functions	274
7.3.4 B-splines	276
7.3.5 Spline representation	277
7.3.6 Application of spline functions	278
7.3.7 Application of spline curves to the ATM world	280
7.4 Bayesian Estimation Techniques	281
7.4.1 Introduction	281
7.4.2 System classification	282
7.4.3 Dynamical systems	285
7.4.4 The state-space model	289
7.4.5 Stochastic hybrid system models	298
7.4.6 Movement models	300
7.4.7 State estimation	320
7.4.8 Optimal estimation problems	323
7.4.9 Types of smoothing problems	326
7.4.10 Smoothing algorithms	327
7.5 Combined techniques	331
7.6 Examples of trajectory reconstruction techniques based on opportunity traffic	331
7.9.2 MURATREC	331
7.9.3 OTR	333

7.7 Effects due to non-ideal conditions	333
7.7.1 Biases or systematic errors	334
7.7.2 Non-Gaussian random noise	334
7.7.3 Out of sequence or late measurements	334
7.7.4 Outliers	335
7.7.5 Quantified measurements	335
7.7.6 Correlated measurements	336
7.7.7 Manoeuvring targets and not modelled movements	336
7.7.8 High-Energy Manoeuvres	337
7.7.9 Coordinate Transformation	337
7.8 Data smoothing	337
7.8.1 Process description	337
7.8.2 Processing details	338
7.8.3 Prototype implementation details	338
7.9 False target report analysis	350
7.9.1 Process description	350
7.9.2 Processing details	350
7.9.3 Prototype implementation details	351
7.10 Information completion	351
7.10.1 Process description	351
7.10.2 Processing details	351
7.10.3 Prototype implementation details	352
7.11 Trajectory post-processes	352
7.11.1 Process description	352
7.11.2 Processing details	353
7.11.3 Prototype implementation details	354
CHAPTER 8	355
PERFORMANCE EVALUATION PROCESS RESULTS	355
8.1 Introduction	356
8.2 Research Platform Description	356
8.3 Validation approach	357
8.4 Validation framework	358
8.4.1 Real Traffic Scenario: Lisbon Airport	358
8.5 Performance evaluation results	385
CHAPTER 9	387
CONTRIBUTIONS AND FUTURE RESEARCH DIRECTIONS	387
9.1 Current dynamizers of performance evaluation	388
9.1.1 Performance-based operations	388
9.1.2 Gate-to-gate trajectories	388
9.1.3 Evolution of the surveillance chain	389
9.1.4 User expected functionalities	390
9.1.5 Airport capacity optimization	390
9.2 Contributions	390
9.3 Research directions and future work	393

9.4 A Visionary Look at Performance Evaluation Tools	395
ANNEX I	397
FILTERING ALGORITHMS	397
A.1 Introduction	397
A.2 Wiener filter	397
A.2.1 General description	397
A.2.2 Assumptions of the Wiener filter	398
A.2.3 Types of Wiener filter	398
A.2.4 Non causal Wiener filter equations	399
A.2.5 Final considerations	401
A.3 Steady-state filters	401
A.3.1 General description	401
A.3.2 Assumptions of the steady-state filters	401
A.3.3 Types of steady-state filters	402
A.3.4 Steady-state filter equations	402
A.3.5 Final considerations	403
A.4 Kalman filter	404
A.4.1 General description	404
A.4.2 Assumptions of the Kalman filter	404
A.4.3 Types of Kalman filter	405
A.4.4 Kalman filter equations	406
A.4.5 Final considerations	407
A.5 Extended Kalman filter	407
A.5.1 General description	407
A.5.2 Assumptions of the EKF filter	408
A.5.3 Types of EKF filter	408
A.5.4 EKF filter equations	408
A.5.5 Final considerations	410
A.6 Unscented Kalman filter	410
A.6.1 General description	410
A.6.2 Assumptions of the UKF filter	411
A.6.3 Types of UKF filter	411
A.6.4 UKF filter equations	411
A.6.5 Final considerations	414
A.7 Interacting multiple model filter	414
A.7.1 General description	415
A.7.2 Assumptions of the IMM filter	415
A.7.3 Types of IMM filter	415
A.7.4 IMM filter equations	416
A.7.5 Final considerations	418
ANNEX II	419
AIRCRAFT INFORMATION	419
B.1 Target information retrieval	419

ANNEX III	421
ANTENNAE POSITION	421
C.1 A320: ATC and TCAS antenna location	421
C.2 AIRBUS family: GPS antennae location	422
BIBLIOGRAPHY	423
LIST OF ABBREVIATIONS	447

LIST OF FIGURES

Figure 2.1: ASTERIX in the OSI Reference Model	46
Figure 2.2: ASTERIX logical model	48
Figure 2.3: ASTERIX physical model	51
Figure 3.1: MAGS Explorer graphical result presentations	123
Figure 3.2: MAGS Explorer graphical result presentations. PoU per cell	124
Figure 3.3: AccPoDTool configuration windows and graphical display	125
Figure 3.4: AccPoDTool graphical display focused on selected airport area	126
Figure 3.5: AccPoDTool 3D graphical representation	126
Figure 3.6: AccPoDTool system architecture	127
Figure 3.7: PASS service web interface	128
Figure 3.8: SASS-C Supervision	133
Figure 3.9: SASS-C OTR reconstructed trajectories	134
Figure 3.10: SASS-C ECA analysis – Probability of Update (PUA)	134
Figure 3.11: SASS-C Imported data from airport area	135
Figure 3.12: NSV-4 Aerodrome ATC System	139
Figure 3.13: SESAR baseline architecture for the surface surveillance function	140
Figure 3.14: Context diagram of the “performance assessment” FB	143
Figure 3.15: Functional architecture of a “performance assessment” FB	148
Figure 3.16: Internal data flows of a “performance assessment” FB	155
Figure 3.17: External interfaces of a “performance assessment” FB	160
Figure 4.1: Framework of a performance evaluation tool	166
Figure 4.2: Block diagram of a generic performance evaluation tool	167
Figure 4.3: Detailed block diagram of a generic performance evaluation tool	169
Figure 5.1: Block diagram focussed on the “data reading” processing block	179
Figure 5.2: Output of the “data acquisition” process	184
Figure 5.3: Block diagram focussed on the “data acquisition” process	185
Figure 5.4: Block diagram focussed on the “data cleaning” process	187
Figure 5.5: Block diagram focussed on the “data allocation” process	189
Figure 5.6: Block diagram focussed on the “data presentation” process	190
Figure 6.1: Block diagram of the “data preparation” processing block	194
Figure 6.2: Block diagram focussed on the “initial data selection” process	196
Figure 6.3: Block diagram focussed on the “known bias cancellation” process	199
Figure 6.4: Block diagram focussed on the “coordinate transformation” process	201
Figure 6.5: Block diagram focussed on the “final data selection” process	203
Figure 6.6: Block diagram focussed on the “time ordering” process	205
Figure 6.7: Block diagram focussed on the “data association” process	211
Figure 6.8: Block diagram focussed on the “data location in the airport” process	213
Figure 6.9: Issues found at intersections	217

Figure 6.10: Block diagram focussed on the “movement model identification” process	221
Figure 6.11: High level description of the bias estimation	229
Figure 6.12: Block diagram focussed on the “bias estimation and cancellation” process	231
Figure 6.13: Aircraft following centre line at Brussels airport	233
Figure 6.14: Computation of the separation distance	233
Figure 6.15: Antenna position induced bias	238
Figure 6.16: Time offset effect	239
Figure 6.17: Block diagram focussed on the “noise estimation” process	258
Figure 6.18: Block diagram focussed on the “time re-ordering” process	259
Figure 6.19: Block diagram focussed on the “association completion” process	261
Figure 7.1: Example of the lofting technique	269
Figure 7.2: Normalized B-splines of order $k = 1, 2, 3, 4$	276
Figure 7.3: Black box (a) and white box (b) models	282
Figure 7.4: System representation	287
Figure 7.5: First-order Markov process	291
Figure 7.6: Hidden Markov process	292
Figure 7.7: Linear-Gaussian state-space model	294
Figure 7.8: Markovian jump linear system	296
Figure 7.9: Three-wheel model of an aircraft	318
Figure 7.10: Block diagram of state estimation	321
Figure 7.11: Block diagram of a tracking system	321
Figure 7.12: Block diagram of a smoothing system	322
Figure 7.13: Block diagram of a generic PET	322
Figure 7.14: Filtered state estimate	323
Figure 7.15: Predicted state estimate	324
Figure 7.16: Interpolated state estimate	325
Figure 7.17: Smoothed state estimate	325
Figure 7.18: Block diagram focussed on the “data smoothing” process	338
Figure 7.19: Detail of the IMM backward smoothing step for the k -th measurement	349
Figure 7.20: Block diagram focussed on the “false target report analysis” process	350
Figure 7.21: Block diagram focussed on the “information completion” process	352
Figure 7.22: Block diagram focussed on the “trajectory post processes”	354
Figure 8.1: Airport map and sensor location	359
Figure 8.2: SMR local track updates imported by the PET	361
Figure 8.3: Detail of the extrapolations performed by the SMR local tracker	362
Figure 8.4: Detections provided by the SMR sensor	362
Figure 8.5: MLAT sensor local tracking imported by the PET	363
Figure 8.6: Detail of the extrapolations performed by the MLAT local tracker	364
Figure 8.7: ADS-B measurements imported by the PET	365
Figure 8.8: Detail of the ADS-B measurements	365

Figure 8.9: Quantization effect in ADS-B measurements	366
Figure 8.10: SSR local track updates imported by the PET	367
Figure 8.11: Detections provided by the SSR sensor in the proximity of the airport	367
Figure 8.12: Detections provided by the SSR sensor around the airport	368
Figure 8.13: Selected SMR detections	370
Figure 8.14: Selected MLAT track updates	370
Figure 8.15: Selected ADS-B measurements	371
Figure 8.16: Selected SSR detections	371
Figure 8.17: Overall picture of the taxiing trajectory	372
Figure 8.18: Detailed picture of the taxiing trajectory	373
Figure 8.19: Overall picture of a departure from runway 03	373
Figure 8.20: Detailed picture of a departure from runway 03	374
Figure 8.21: Overall picture of an arrival to runway 03	374
Figure 8.22: Detailed picture of an arrival to runway 03	375
Figure 8.23: Continuous path of the taxiing trajectory	376
Figure 8.24: Details of the continuous path followed by the taxiing aircraft	376
Figure 8.25: Continuous path of a departure from runway 03	377
Figure 8.26: Details of the continuous path followed by the departing aircraft	377
Figure 8.27: Continuous path of an arrival to runway 03	378
Figure 8.28: Details of the continuous path of an arrival to runway 03	378
Figure 8.29: Convergence of the estimated range bias of the SMR sensor	379
Figure 8.30: Convergence of the estimated azimuth bias of the SMR sensor	380
Figure 8.31: Detail of the correction applied to the SMR plots	380
Figure 8.32: Detail of the correction applied to the SMR plots	381
Figure 8.33: Convergence of the ATC antenna position bias of the MLAT sensor	381
Figure 8.34: Detail of the reconstructed information	382
Figure 8.35: Detail of the correction applied to the SMR plots	383
Figure 8.36: Comparison of the speed of the IMM forward and the backward smoother	383
Figure 8.37: Comparison of the heading of the IMM forward and the backward smoother	384
Figure 8.38: Comparison of the steering angle of the IMM forward and the backward smoother	384
Figure 8.39: Evolution of the separation distance all along the data procesing	385

LIST OF TABLES

Table 1: Surveillance sensor classification	39
Table 2: Identification information within ASTERIX Cat.010 and 048	60
Table 3: Time information within ASTERIX Cat.010 and 048	61
Table 4: Target kinematic information within ASTERIX Cat.010 and 048	61
Table 5: Track information within ASTERIX Cat.010 and 048	63
Table 6: Quality information within ASTERIX Cat.010 and 048	65
Table 7: Airborne information within ASTERIX Cat.010 and 048	65
Table 8: Identification and classification information within ASTERIX Cat.021	86
Table 9: Time information within ASTERIX Cat.021	87
Table 10: Target information within ASTERIX Cat.021	89
Table 11: Ground referenced velocity information within ASTERIX Cat.021	90
Table 12: Quality information within ASTERIX Cat.021	91
Table 13: Airborne information within ASTERIX Cat.010 and 021.	91
Table 14: Identification and classification information within ASTERIX Cat.020	99
Table 15: Time information within ASTERIX Cat.020	100
Table 16: Target information within ASTERIX Cat.020	100
Table 17: Ground referenced velocity information within ASTERIX Cat.020	102
Table 18: Quality information within ASTERIX Cat.020	103
Table 19: Internal data flows	159
Table 20: Data flows through external interfaces	163
Table 21: Mode-matched filter selection based on context and prior information	345
Table 22: Airport information	358
Table 23: Sensor information	359
Table 24: Radar measurement model parameters	360
Table 25: Multilateration measurement model parameters	360
Table 26: ADS-B measurement model parameters	360
Table 27: aircraft physical information per target type	419
Table 28: airport vehicle physical information	420

CHAPTER 1

INTRODUCTION AND MOTIVATION

“Success is walking from failure to failure with no loss of enthusiasm.”

— *Winston Churchill.*

The new environment based on the SESAR¹ paradigm [SESAR MP] pretends to change the ATC controller (ATCo) decision making, nowadays more based on a reactive behaviour, into a more proactive one. Besides, the ATM Target Concept described in the SESAR Definition Phase [SESAR D3] is developed around the “4D Trajectory” concept.² In this case, the aim is not only to improve the current knowledge about the aircraft location (in time and in space), but also to improve the future one. The latter implies that conflicts can be envisaged with enough margin to propose more efficient, cost-effective, and greener solutions than in the case in which tactical controller interventions are needed.

Other important matter about SESAR and its successive extensions is that its ultimate aim is to achieve the Integrated Airport Operations into the ATM Network [SESAR MP]. This fact is based on the extension of the 4D trajectory management to include the movements around the airport and, as such, reaching the gate-to-gate³ (G2G) concept. The effect of

¹ SESAR stands for Single European Sky ATM Research and represents the technological and operational dimension of the Single European Sky (SES) initiative to meet future capacity and air safety needs [SJU about]. The SESAR programme is one of the most ambitious research and development ever launched by the European Community.

² This concept considers the spatial position of the aircraft adding also the time dimension.

³ “Gate-to-gate is a concept that involves considering and managing a flight as a continuous event, from planning, through execution, to the post-flight activities. Its scope extends from the first interaction of

this extension will be the advanced knowledge about the airport movements that will be reflected in a reduced ATCo intervention over the rest of the mobiles on the airport surface ensuring a degree of strategic de-conflicting while minimizing holding and ground queues.

Within this context, it is expected that all mobiles in the manoeuvring area will have their own assigned route, which will be supervised by the airport Conformance Monitoring function in order to check that they are behaving as expected. In this situation, it is important to point out that the information coming from the Surveillance function is crucial as it provides the current situational awareness picture to be compared with the predicted airport situation.

The same happens with the Safety Nets. In this case, the main problem is that they need very accurate surveillance information to calculate the possible conflicts. A recognized issue for the implementation of the A-SMGCS⁴ “Control” function [ICAO 9830] is the appearance of false alarms that interfere with controller operations. This is also related to the initial definition of the A-SMGCS Implementation Levels [ECTL A-SMGCS], which linked the evolution of the concept to the available procedures and technology performances. Nowadays, this definition has changed the approach on how the A-SMGCS system is decomposed and, as a result, it has become a functional definition focused on functions/services [ECTL A-SMGCS 20].

The aim, in the SESAR paradigm, is to consolidate A-SMGCS Level II (i.e. Safety nets or control function) and then move towards level III (i.e. Planning/Routing function) and, finally, reaching level IV, that is, the Guidance function [SESAR MP]. In order to reach these implementation levels, the contribution from the Surveillance function can come from:

- The introduction of the no longer “new sensors”, such as ADS-B and multilateration systems, and elements of the new sensor family known as “Independent and Non-Cooperative Systems” (INCS).
- The improvement of the current ones (i.e. surface moment radars (SMR), primary surveillance radars (PSR) and secondary surveillance radars (SSR), either conventional ones or Mode-S).
- The improvement of data fusion algorithms.

There are papers that focus on the first option [Soto 11] due to the fact that SESAR considers that ADS-B is the cornerstone for moving towards the following A-SMGCS Levels, as it provides more accurate and frequent surveillance information than conventional sensors, which improves the situational awareness picture presented to the tower ATCo, for example, in the Integrated Tower Working Position (ITWP). Besides, it

the flight with ATM (which for commercial flights may be up to 6 months ahead of the date of the flight), through the execution of the flight on its appointed day, to performance registration and the calculation of charges for the services received after it has taken place” [ECTL 2004].

⁴ A-SMGCS (Advanced Surface Movement Guidance & Control System) is defined by ICAO as “a system providing routing, guidance and surveillance for the control of aircraft and vehicles in order to maintain the declared surface movement rate under all weather conditions within the aerodrome visibility operational level (AVOL) while maintaining the required level of safety” [ICAO 9830].

can easily provide a seamless hand-over between the TMA and the Tower control helping in track continuity. However, this sensor⁵ also introduces some drawbacks, which are mainly caused by the error in the time stamping of the surveillance information (known as the “Uncompensated Latency”), that need to be addressed before fusing its data with the one coming from other surveillance sensors as its high update rate and its generally declared as “quite good” quality measurements will for sure kidnap any system track.

Multilateration systems have extended their presence in the last ten years becoming not only a complementary sensor to the SMR, but also a replacement of it. The main issue with these sensors is the lack of a generally accepted measurement model.

Finally, INCS systems are not yet ready to take over any of the previous sensor types, but they are finding their place in the market as gap fillers or as solution with a very reduced RF consumption. So, for the moment, they are still quite unknown systems.

If the introduction of a new sensor is not possible, another possibility is the improvement of legacy ones, such as the SMR or the different types of SSR systems. This improvement can come from two different directions: one is more challenging because it implies improving the sensor technology, which can be very expensive, and the other tries to improve the statistical characterization of the sensor in order to better define its measurement error model. This model mainly comprises random errors and systematic ones. Both are taken into consideration in the sensor measurement model. The statistical characterization of random errors is exploited by the state estimate algorithms within the data fusion system, while the knowledge about systematic errors can be applied to the design of bias estimation algorithms, which enable first their estimation and then their direct removal from the measurements provided by the sensor.

The last alternative implies improving the data fusion algorithms which are incorporated in any Surveillance Data Processing (SDP) system of a certain prestige. This is not an easy task as there are many variables that are out of the control of the system engineer in charge of designing such a system. However, this field has considerably evolved in the last decades, so now different kinds of data fusion algorithms can be found in commercial products, e.g. Kalman filters, EKF, UKF, Cubature KF, particle filters, pseudo Bayesian algorithms, IMM, etc. Such a vast variety of alternatives opens the door to the evaluation and assessment of each of them, not just when being designed, but also once they are commissioned.

The evolution of these contribution areas is modulated by external factors such as the global economic context, which currently points to a new deceleration, and other world challenges such as the ones coming from health threats (e.g. COVID-19) that might affect the air traffic growth. It seems reasonable assuming that neither much investment is going to be devoted to improving legacy technologies nor many resources are going to be compromised in developing new ones; and so, any improvement in the quality of surveillance information will mainly come from, on one hand, the improvement on sensor measurement models and on the other, from fine tuning data fusion algorithms.

⁵ Purely speaking ADS-B is not a sensor but a surveillance application. However, in the ATC surveillance world is it referred as a sensor and as such it is going to be considered in this thesis.

In this overall framework, the topic of this thesis, that is, performance assessment tools can be of great importance as they can efficiently contribute to the automation of:

- The characterization of the error components of sensor measurement models that impact the filtering of random errors and the estimation and removal of biases.
- The automatic assessment of surveillance sensors and data fusion systems currently in operation.

Everything performed in an efficient way by taking advantage of already available information. Consequently, these activities contribute to the same objective that is optimizing the performance of the Surveillance function for both the air and the ground domains not only to guarantee a safe operation within the performance standard regulation but also to do it in a performant way.

The following sections will present a break down structure of the Air Navigation System, placing the focus on the ATM services and aiming at providing the context where the Performance Evaluation Tools (PET) will carry out their task.

1.1 Air Traffic Management

The Air Navigation System (ANS) can be analysed from three different perspectives [Calvo 07]. Thus, one can focus on:

- Services that are provided.
- Elements that provide these services.
- Domains in which the elements are organized.

The division in such groups permit an analysis of the ANS from different points of view and as such tackling more aspects than if only one is considered. For the sake of completeness all them are going to be presented starting from the main services of an Air Navigation System, which are:

- The Airspace Management (ASM).
- The Air Traffic Flow Management (ATFM).
- The Air Traffic Services (ATS).

All together are also known as the Air Traffic Management (ATM). Additionally, there are other services with less importance, such as: the Aeronautical Information Service (AIS), the Search and Rescue service (SAR) and the Meteorological service (MET).

From the optics of the elements that are in charge of implementing these services, it can be identified:

- The staff.

- The procedures.
- The infrastructure.

Which can be bundled in several domains. The ones identified in the Air Navigation System are:

- Communications (C).
- Navigation (N).
- Surveillance (S).

Which are jointly referred to as CNS.

The focus of this work will be on the provided services, as the final target will be the improvement of the quality of service (QoS) of the “Surface surveillance” function, which is the foundation of the aerodrome ATC service. This is very relevant as this function is an enabler for other A-SMGCS functions, such as the “Control” one (safety nets) or the “Planning/Routing”. So, improving its quality is a key issue in the process of improving the aerodrome ATC service and so, reaching one of the key features of SESAR, i.e. the Airport Integration & Throughput.

Thinking on defining the area of interest of the performance evaluation tools, a brief description of the main services is going to be provided aiming at presenting the environment where the performance evaluation tools work.

1.1.1 Airspace Management

The Airspace Management (ASM)⁶ is the service in charge of designing the airspace, which comprises the definition of its static structure, planning the use of such structure and tactically managing the allocation of its different elements such as: information regions, airways, control areas, restricted areas (RA), temporary restricted areas (TRA) terminal manoeuvring areas (TMA), non-transgression zone (NTZ), etc. to maximize the airspace capacity with as much flexibility as possible.

In EUROCONTROL words [ERNIP P3 12]: "Airspace management (ASM): is a planning function with the primary objective of maximising the utilisation of available airspace by dynamic time-sharing and, at times, the segregation of airspace among various categories of users based on short-term needs. In future systems, airspace management will also have a strategic function associated with infrastructure planning".

The target of this service is to design, review, update and restructure the airspace organization to meet the operational needs of the air traffic demand aiming at achieving the most efficient solution. It is the most strategical service as it settles down the static airspace structure, which is afterward adapted to cope with the changing demand. This structure is optimized considering different metrics, tools and data sources.

⁶ The ICAO definition says that “Airspace management (ASM) is the process by which airspace options are selected and applied to meet the needs of the airspace users”.

One of the most important concepts associated to this service is the Flexible Use of Airspace (FUA), which tries to optimize the airspace in a flexible way taking into account the airspace user demands, that is, both the civil and the military ones [ERNIP P3 12]. This concept has evolved to the Advanced FUA (AFUA) and it was part of the Centralized Service 4 (CS4)⁷ which was called the Advanced Flexible Use of Airspace Support Service (AFUAS).

This service is of reduced importance for the target of this work, whose focus is placed mainly in the airport.

1.1.2 Air Traffic Flow Management

The Air Traffic Flow Management (ATFM)⁸ is the service, which is part of the core of ATM network operations,⁹ in charge of optimizing traffic flows taking into consideration the ANS capacity and offering airspace users the possibility of operating safe and efficient flights. Its main target is, from EUROCONTROL website [Flow management], to:

- Manage the overall ATFCM Network in Europe and assist air navigation service providers (ANSP) to manage local capacity.
- Achieve balance between demand and capacity.

The planning of the operations is started as soon as possible to prepare the staff and the infrastructure to the expected demand. This sometimes reaches several months in advance or even years. The ATFM tries to anticipate both: the airlines demand, and the expected capacity to be declared by ANSPs and airports. Other extreme situations are also taken into consideration such as seasonal peaks (Christmas holidays, sporting events, or international meetings) or unforeseen events, such as, volcano ashes or more recently the impact of the COVID19 pandemic. These tasks are led by the former CFMU (Control Flow Management Unit) which now has been renamed as the Network Manager Operations Centre (NMOC).

The management process is organized in phases. From what is further in time, up to the current moment. The strategic phase is the one that starts the process. It can begin more than a year before the flight takes place until one week before the actual time of the flight. This is a collaborative phase in which the NMOC jointly with the ANSPs estimates the capacity to be provided by the Air Traffic Centres. This fact also affects the ASM service as some routes might be adapted or the use of some restricted areas might be requested to increase capacity. There is a document that gathers the capacity commitments of all the operational stakeholders, which is called the Network Operation Plan (NOP). This

⁷ EUROCONTROL developed the concept of Centralized Services (CS) to speed up the implementation of the Single European Sky initiative as mandated by the European Commission. The concept relied on the belief that there were several ANS support services that could be managed more efficiently at a central or network level. However, this idea was generally not accepted by ANSPs and National Administrations, so its development has been cancelled in most of the cases. For more details about centralized services please visit: https://www.skybrary.aero/index.php/Centralised_Services_in_ATM

⁸ Also referred as Air Traffic Flow and Capacity Management (ATFCM).

⁹ For more details about network operations please visit: <http://www.eurocontrol.int/network-operations>

document is updated all along the process to keep track of the actual capacity of the ATM network.

The next phase is called pre-tactical and goes from the week before the operation to the previous day. During this period, the NMOC staff leads a collaborative decision making (CDM) process, where all the operational stakeholders are involved, in which the target is to coordinate the preparation of a “daily plan” that aims at optimizing the ATM network performance. In case it is not possible to meet the demand with the expected capacity, the ATFM service will define the regulations that will be in force the following day in the European airspace. All this information is published in the daily plan, so all the actors will be informed through the NOP Portal.

The last phase is the tactical one, which covers the day of operations. The target of this phase is to monitor that the daily plan is aligned with the actual demand and capacity situation. If some discrepancies are detected, the plan is updated. The NMOC continues with the capacity optimization even during the day of operations, based on a more accurate knowledge of the actual demand and capacity, by providing alternative routes when regulations are in force.

Based on this process, the ATFM service can offer to the flights taking place on a day, the following services:

- Allocation of departure slots.
- Alternative routes for avoiding regulation or bottlenecks.
- Alternative flight profiles aiming at maximizing flight efficiency.

All these services are provided aiming at improving the global ATM network performance and as part of a CDM process that tries to guarantee that the interest of all operational partners is taken into account.

1.1.3 Air Traffic Services

The term “Air Traffic Services” (ATS) includes several services aiming at controlling, informing, and assisting aircraft in real-time in order to guarantee the safety of operations. To be more precise, the objectives of ATS services [Calvo 07] are to:

- Prevent collisions between aircraft.
- Prevent collision between aircraft and obstacles.
- Speed up and synchronize air traffic to maintain an orderly and expeditious flow.
- Provide advice concerning the safety and efficiency of flights.
- Notify and assist the relevant agencies in relation to aircraft needing help or rescue.

ATS is divided into four categories:

- Air Traffic Control (ATC).
- Flight Information Service (FIS).
- Alerting service (AS).
- Air Traffic Advisory Service.

The focus of this work is the first category and within it, the service in charge of providing the Aerodrome ATC service. Nowadays, performance assessment tools mainly contribute to the tuning and automatic assessment of air surveillance and in a near future their scope will also embrace the ground surveillance. Both surveillance functions are in charge of generating the air traffic picture that is presented to En-route/TMA/Tower controllers.

1.1.3.1 Air Traffic Control

As mentioned before, this service is responsible for, on one hand, preventing collisions between aircraft, and aircraft and obstacles; and, on the other, reaching a safe, orderly and expeditious air traffic flow on both, the ground and in the air. This service is provided by controllers who are responsible for providing separation to the aircraft on the ground or in a controlled airspace. They release ATC instructions to aircraft, so pilots are aware about what to do to optimize air traffic flows and to avoid conflicting situations. Additionally, they can also provide advisories, different types of information and other support to pilots.

The separation provision is guaranteed by means of enforcing traffic separation rules. These rules depend on the type of aircraft, the class of airspace, the airport procedures, the weather, etc. Their target is to keep aircraft far enough from each other so, the collision risk is kept under a minimum value. This is possible thanks to the surveillance function which provides among other information, the 2D position of aircraft, their barometric altitude and, sometimes, their geometric height, and their identification.

Depending on the airspace where the service is provided, the following division is found:

1.1.3.1.1 Aerodrome Air Traffic Control

The "Aerodrome ATC" service is responsible for providing separation to vehicles that move around the manoeuvring area of an airport and aircraft that are in its close vicinity (arrivals and departures). This area encompasses the runways and the taxiways opposite to the platform or the apron area, which is the area where aircraft are parked, refuelled, loaded, unloaded, etc. The whole area is referred to as the movement area and includes the manoeuvring area and all the apron areas of the airport.

The focus of this particular control service is on the runway use. Thus, it assures that there is enough runway separation between aircraft landing and departing, but always pursuing an efficient use of the runway. Additionally, they also take responsibility in the provision of taxiway instructions while aircraft move around the manoeuvring area. In some airports, they also provide a separation service on the apron. However, the latter is normally provided by the apron management service.

The aerodrome ATC service is provided in little airports by visual observation; however, this is not always possible in major airports, where buildings hide sensitive parts of the airport. This is the reason why in such airports, there is a need for a surveillance infrastructure which extend the eyes of the controllers to the entire airport surface.

This service is usually provided from a control tower (TWR), which is located within the airport surface. In this facility, there are several roles which assume the duties that have been allocated to such facility. These are:

- The “Flight data” controller, who is responsible for assisting the other controllers in the tower, relaying IFR¹⁰ departure clearances to the clearance delivery controller and forwarding weather and NOTAM¹¹ information to other controller working positions.
- The “Clearance delivery” controller, who is in charge of managing departure clearances to pilots.
- The “Ground control” role is assumed by a controller who is then responsible for taxiing aircraft.
- The “Local control” role is assumed by the controller who ensures that proper runway separation exist between departing aircraft.

In major airports, there are additional roles which issue clearances and instructions that affect the apron area of the airport. This is the case of the “authorization” role, which is responsible for the “Start-Up” (S/U) clearance, the “Push back” (P/B) clearance, the “taxi” clearance for the apron area, etc. Additionally, it also manages the “ready” message (REA) and the “parking” clearance. Although this apron management service is provided mainly by controllers is not considered as an ATC¹² service.

Depending on the size of the airport, such roles can be assigned to one or several positions, so we can talk about integrated roles which also leads to the concept of integrated working positions. Further details concerning the tower operations in the NAS¹³ can be found in [Nolan 11].

1.1.3.1.2 Approach Air Traffic Control

The "Approach ATC" service is responsible for separation provision in the TMA; and sequencing and merging aircraft with final destination one of the airports within the TMA. This service is provided to airborne aircraft such as arrivals, departures and overflights.

¹⁰ IFR stands for “Instrument Flight Rules”, opposite to “Visual Flight Rules” (VFR).

¹¹ The NOTAM (Notices To Airmen) is a piece of information that is distributed by the country’s aviation authority to inform pilots of new or changed aeronautical facilities, services, procedures, or hazards, temporary or permanent.

¹² In some airports such as Adolfo Suárez Madrid-Barajas, the apron control has been outsourced so now the service is provided by platform operators not controllers.

¹³ The NAS (National Airspace System) groups the airspace, the navigation facilities and the airports of the United States of America.

The coordination¹⁴ transfer from the en-route sector is normally initiated before the Initial Approach Fix (IAF) and ends around 5 to 10 NM to the runway threshold.

This service is provided by air traffic controllers whose objectives are mainly to prevent collisions between aircraft and expedite and maintain an orderly flow of air traffic. There are two separated roles: the planner controller and the tactical¹⁵ one. The planner controller is in charge of all the coordination activities previous to the aircraft entering the sector under control and also the resolution of medium term conflicts aiming at reducing the workload of the tactical controller; while the latter is responsible for short term conflict avoidance by means of issuing tactical instructions.

The approach ATC service is usually provided from Approach Control Centres (APP) which in some situations are integrated either on an Area Control Centre (ACC) or in a tower (TWR), depending on the traffic density of the airspace. Controllers are supported by Controller Working Positions (CWP) where flight information and surveillance tracks are presented to them.

1.1.3.1.3 En-route Air Traffic Control

The "en-route ATC" service is the one in charge of providing separation provision for airborne aircraft which are in their en-route phase also known as cruise phase. This service is provided by ATC controllers once the aircraft has left the TMA or until it is going to enter it. Additionally, controllers also provide other services such as provision of information or advisory. The service provided depends on the kind of separation rules which apply to the aircraft (i.e. IFR or VFR).

The work performed by the controller can be divided into the same roles as in the previous section. Thus, there are planner controllers who deal with strategic actions and coordination matters; and tactical controllers who issue tactical clearances and instructions for airborne aircraft, which must be implemented by pilots.

This service is provided from a facility called Area Control Centre (ACC), which can be located at any place, even out of the Flight Information Region (FIR) under its responsibility. These areas often extend up to millions of square kilometres. To manage such a huge airspace, controllers are supported by systems, such as:

- Flight data processing (FDP) systems.
- Surveillance data processors (SDP).
- Controller working positions (CWP)

These systems allow a division of the airspace, which is under the responsibility of a certain facility, into pieces that are called sectors, which are assigned to a controller team that is composed of a planner and a tactical controller. The ways this division is done is called sectorization and it varies along time due to changes in the air traffic density. This

¹⁴ Coordination is a controller task in charge of guaranteeing a seamless handover of flights between ATC facilities so that the service is provided without disruption. It can also be between sectors within the same ATC facility.

¹⁵ Also called the executive controller.

service exploits the information provided by long range radars (PSR and SSR) and other sensors. such as ADS-B or WAM.

1.1.3.1.4 Oceanic Air Traffic Control

This service is responsible for separation provision in an airspace where there is no surveillance infrastructure; that is, in the middle of the ocean or in vast extensions of continental areas. This control is based on voice reports and more recently on ADS-C messages and CPDLC commands. For the purposes of this thesis, it is not relevant.

1.1.3.2 Flight Information Service

The flight information service is the responsible of updating information and providing advices for safe and efficient flights. Most of this information is initially provided by other services, but once the aircraft is airborne, it is updated by this service using different means. The information that is provided includes aeronautical information, meteorological information, operational procedures, and traffic advisories.

For further information please visit [ICAO Annex 11] chapter 4.

1.1.3.3 Alerting service

The alerting service is the one in charge of giving support to aircraft in danger or which have suffered an accident by means of notifying the appropriate organizations responsible for search and rescue aid, and support them in their activities.

For further information please visit [ICAO Annex 11] chapter 5.

1.1.3.4 Air Traffic Advisory Service

This service provides advisory to aircraft within advisory airspace to ensure separation. It is a kind of transition service between the FIS and the ATC service. It is used in some places for training controllers as it seems that they control the airspace while they do not have any responsibility on the service provision.

For further information please visit [ICAO 4444], Chapter 9, Section 9.1.4.

1.1.4 Other services

Apart from the previously listed global services (ASM, ATFM and ATS), there are several more which are closely related to the ATM concept but not of special interest for the topic of this thesis. These are:

- Aeronautical Information Service (AIS). See [ICAO Annex 15].
- Search and Rescue Service (SAR). See [ICAO Annex 12].
- Meteorological Service (MET). See [ICAO Annex 3].

1.2 Air Traffic Control

This section intends to be a quick summary of the elements that take part in an ATC system focussing on those ones which are of importance for this work.

1.2.1 Aircraft and their aircrew

Aircraft are the main elements to be controlled within the airport surface and aircraft flows are to be optimized in a safe and efficient way. Most of this work depends on how predictable the behaviour of aircraft is and how good the model that is used for estimating their evolution along time is.

But it is important to bear in mind that aircraft are commanded by pilots, who are the ones responsible for their safety. Thus, pilots are the final safety authority, so they are allowed not to follow ATC instructions whether they consider that the aircraft can be jeopardized. Thus, they introduce another variable to be included in the behavioural model of an aircraft as each pilot manages and flies an aircraft in a different way. So, this is an important fact to be considered.

Apart from pilots, there are also stewards and passengers who are not relevant for the target of this work.

1.2.2 Air Traffic Control facilities

This section lists all the facilities where the ATC service is provided.

1.2.2.1 Area Control Centre

The Area Control Centre (ACC) provides air traffic services to en-route flights of a certain region of the airspace, normally called a FIR. It is normally located within the surface of the airspace belonging to the FIR although it is not mandatory. En-route controllers are responsible for separating airborne aircraft one from another and to organize traffic flows in the most efficient way.

1.2.2.2 Approach Control Centre

An "Approach Control Centre" (APP) is a building usually located in the proximities of an airport or close to a group of airports with enough traffic to justify the definition of a TMA. Its main activities are related to separation provision of departures, arrivals, and overflights; and with the organization of traffic flows with final destination or which have just departed from one of the airports in the TMA managed by the APP. They are responsible for sequencing and merging traffic flows in an efficient and expeditious way ensuring that the altitude and the velocity is the appropriate one for each of the flight phases in which the aircraft can be in the airspace under the responsibility of the APP.

1.2.2.3 Tower

The "Tower" (TWR) is the ATC facility that is located in the airport grounds. It is usually a tall windowed building which has visual observation of the whole airport or at least of most of it. The tower hosts the activities of tower controllers who are in charge of

providing separations to airport mobiles and also to guarantee an efficient and safe flow of traffics around the airport. Their main interest is the management of the runway and the taxiways that connect the former with the apron area. The tower also hosts the systems or at least the CWP's that are used by tower controllers for their tasks including those used for performance evaluation activities.¹⁶

Nowadays, there is a trend motivated by the great evolution of data communications and the pressure on the price of the service, which consists in attending low traffic density airports together from a certain facility in order to reduce costs. Thus, several airports can be controlled by the same controller in case the traffic allows it. This is possible thanks to the communications between the airports, where the information is taken, and the location, where the information is presented to the controller. This is known as “remote” or “virtual” towers.

Thus, if the controller is controlling from a facility other than the tower, then it is said to be in a “remote tower”. This concept allows concentrating in just one place the control of several airports increasing the efficiency of the service. On the other hand, a “virtual tower” is a concept slightly different. It is based on the separation of the presentation of the information from the place where it is processed. Thus, a tower can be “virtual”, if the servers that provide the information to be shown to the tower controllers are stored in other facility different from the tower itself.

1.2.2.4 Oceanic Control Centre

This facility could be in a separate place to the others or it can share the same location. It receives all the information concerning oceanic or continental flights that is processed and presented to those controllers providing oceanic control. The systems that are used are slightly different from the ones that can be found in APPs or ACCs.

1.2.3 Air Traffic Control systems

Within each of the previously presented facilities, there are two types of information to be managed by controllers, namely: the one concerning the planned evolution of a flight and the one regarding its actual behaviour. Usually, the first one is managed mainly by the planner controller and the second one by the tactical one.

The first type of information is provided by what is known nowadays as the flight plan (FP). This piece of information contains most of the information concerning the flight, that is, the aircraft type, from which airport the flight is going to depart, in which one it is going to land, what route is going to follow, at what altitude is going to fly, etc. This information is updated during the whole flight. It is foreseen that the flight plan is progressively superseded by a new concept that is called the flight object¹⁷ (FO). This concept aims at increasing the details of the flight plan that the aircraft is going to fly to allow a better prediction of the trajectory that is going to be flown. Additionally, it also

¹⁶ With the advent of the concept of “virtual towers”, this statement might not be true as now it is possible to have the servers which run the applications that are used by tower controller in other facility different from the tower.

¹⁷ The flight object is a concept that has been developed during the last ten to fifteen years to supersede the flight plan and improve coordination aspects. It was foreseen that it will be distributed by SWIM.

aims at extending the trajectory to be G2G, so the ground segment of the route is expected to be included.

The other information refers to what it is really happening at each moment. It is already known that there are circumstances that are out of the control of human beings such as the weather, alert situations, abnormal behaviours, emergencies, etc., which may force an unforeseen change in the initial flight plan. All this information is gathered by what is called a “track” by means of fusing surveillance sensor information. Thus, tracks contain a summary of the current and updated information concerning a certain aircraft, such as: its position and velocity, its flight level, its equipment, its identification, the ages of all these pieces of information, etc. The number and quality of this information depends on the type of surveillance sensors that detect the aircraft within their coverage and that are additionally configured to be processed by the tracking system (i.e. SDP).

It is quite common to find in ATC systems, a subsystem in charge of matching both pieces of information. These subsystems are usually called correlators, as they correlate flight plan information with the one provided by SDP systems (i.e. tracks). This fact is normally reflected in the information included in the track that is then enriched with what is called a “mini” flight plan, which is provided to the tracker by means of an ASTERIX Cat.032 message [ASTERIX Cat.032], devoted for ARTAS system, or by a proprietary one.

1.2.3.1 Flight Data Processing system

The Flight Data Processor (FDP) is in charge of processing, exchanging and exploiting the information contained in flight plans and, when available, flight objects.¹⁸ Depending on the manufacturer, they can include more or less functionalities. The most common implementations are the one based on iTEC (interoperability Through European Collaboration) which is provided by Indra¹⁹ and the one based on Coflight which is a joint venture between Leonardo (former Selex-ES) and Thales.²⁰

These systems extract the information included in the flight plan and by means of trajectory models such as BADA [BADA], predict the path that the aircraft is going to fly and the overfly times for some fixes belonging to the trajectory. This information is then used by many other subsystems to detect conflicts (MTCD)²¹, check conformance to the plan (CMON)²², ease coordination between centres, establish data link communications, show the information to the controller (CWP), etc.

1.2.3.2 Working positions

In an ATC system, there are many Human-Machine Interfaces (HMI) that show different types of information to controllers, each one is normally associated with the output of a certain system or tool. These interfaces are grouped depending on the functionality they contribute to and they are shown in the working position in charge of gathering,

¹⁸ It is expected that the flight plan concept will be superseded by the flight object concept.

¹⁹ For further details, please visit: www.indra.es

²⁰ For further details, please visit: www.selex-es.com or www.thalesgroup.com.

²¹ The Medium-Term Conflict Detection (MTCD) is a controller tool, mainly managed by the planner controller, which aware him about medium term conflicts. It uses planned trajectories to compute the possible conflicts.

²² The conformance monitoring (CMON) is a controller tool, in charge of warning about any deviation of the flight from its expected trajectory. It uses the information about the route that it is included in the flight plan and the one coming from the surveillance chain.

processing, and presenting the information related to such functionality. This is so because it is normally managed by a different controller role. Nowadays, this approach has been relaxed due to the increase in the computing power and flexibility of working positions, not only from a software point of view but also from the hardware one, which can be seen, for example, on the size of screens (2K x 2K) and the fact that most of them are even touchable. So, it is usually possible to integrate the information related to different ATC functionalities in just one working position. This fact has led to what it is called today, for the en-route domain, the "Integrated Controller Working position" or iCWP;²³ and whose equivalent in the ground domain would be the Integrated Tower Working Position (ITWP).

The terms mentioned in this section correspond to the ones defined for the iTEC philosophy.

1.2.3.2.1 Controller Working Position

The Controller Working Position (CWP) is the most important position for the controller as it merges the information concerning flight plans and tracks, apart from integrating many other pieces of information. Thus, the controller has in an integrated way the planned behaviour of the flight and the actual one. The complexity of such a system is found on the fact that there is a lot of information to be shown in a reduced and limited surface, so it is important that it is presented in a modular, adaptable and selectable way. This is normally done by using different screens that are located along the furniture where the controllers are sited in front.

The CWP is fed by many systems, each one with its associated HMI. The main ones are the FDP, which sends flight plans, and the SDP, which sends tracks. Additionally, the controller can be prompted with meteorological information, coordination data, aeronautical information, status of communication means, and with the HMI part of different tools such as internal supervision, MTCD, STCA, TBS,²⁴ etc.

1.2.3.2.2 Flight Data Operator Position

The Flight Data Operator (FDO) position is in charge of processing and sorting out those flight plans which arrive with incomplete, incoherent or with erroneous information. Every time the FDP detects that a flight plan is incorrect or incoherent, it sends it to the FDO where a specialized operator analyses and amends it. Once corrected, the flight plan is sent back to the FDP and which restores it in the database of flight plans.

1.2.3.2.3 Operational Supervision Position

The Operational Supervision Position (OSP) is in charge of configuring the resources and managing the sectorization according to the airspace traffic load and mix.²⁵ This information is generally managed by the operational supervisor, who is a controller with team coordination responsibilities within a facility.

²³ The term iCWP is deprecated as a common acronym for referring to the "Integrated Controller Working Position" because it has been registered by Frequentis AG.

²⁴ TBS stands for Time Based Separation. It is a new way of separating aircraft while in final approach to the airport. Currently, the separation between aircraft is defined in distance, not in time, which leads to a capacity reduction when the airport suffers strong head winds.

²⁵ The "Traffic mix" refers to the different types or categories (concerning wake turbulence) of aircraft that can be flying in a certain air space.

1.2.3.2.4 Technical Supervision Position

The Technical Supervision Position (TSP) is the one which allows to monitor and control the system elements. It normally presents a box per system and subsystem with different colours depending on their technical status (i.e. “red” if it is down or switched off, “green” if it is up and running or “orange” if there is any warning). This position is the one in charge of presenting the problems in any of the systems that are under supervision²⁶ to the technical supervision operator. Thus, only supervised elements are shown.

1.2.3.2.5 Advanced Controller Working Position (ACWP)

Finally, the Advanced Controller Working Position (ACWP) is the evolution of the traditional CWP which integrates most of the information on a huge console of 2K per 2K (2048 x 2048 pixel resolution) so everything is accessed through this console in an efficient and timely manner avoiding facing the controller with too much clutter.

It is a concept that was developed in the framework of SESAR where several prototypes were implemented (projects 06.09.02, 10.10.03 and 12.05.04). The idea of merging all the information in just one screen is improved by the use of touch screens and animated menus as can be seen in products such as Indra iFOCUS, which has been shown in the last editions of the World ATM Congress (WAC)²⁷ in Madrid.

1.2.3.3 Support systems

By the term “support systems” is understood a set of systems that are recommended by international authorities (ICAO, EUROCONTROL, EASA, etc.) in order to analyse problems that have appeared on the normal operation of the ATM system that are not of interest for controlling aircraft.

Thus, in such a set, one can find the following categories of “support systems”:

- Recording systems, which are in charge of storing all the appropriate information concerning the operation of the system. It can include controller communications by voice or by data link, surveillance information, flight plan data, meteorological information, aeronautical data, etc.
- Playback functionalities, which allows to replay the information recorded by the recording system for different purposes such as training controllers, incident investigation or to feed performance evaluation tools.

1.2.3.4 Billing systems

These systems are the ones responsible for charging air space users for the ATM services that they have consumed while operating their flights. These payments come through air navigation charges, which are normally billed and collected by the EUROCONTROL Central Route Charges Office (CRCO) in the European area. The following information comes from the European air navigation context.

²⁶ This supervision is usually called the “external” supervision to be differentiated from the “internal” one, which is the one that monitors and supervise the internal components of the systems or tools. The information reported by the “Internal” supervision is generally used for defining the status of the system to be sent to the “External” supervision.

²⁷ For further information please visit: <http://www.worldatmcongress.org/>

There are three types of air navigation charges:

- Routes charges.
- Terminal navigation charges.
- Communication charges.

The systems responsible for computing these charges take advantage of the information provided by some of the previously described ATC systems.

1.2.3.5 Simulation Systems

It is quite common to find replicas of the infrastructure of an ATC facility in the same building or in other locations, which are fed with synthetic data or with reinjected real data. These systems have different targets, namely:

- For verifying or validating new functionality or new operational procedures and so training the controllers.
- For evaluating the performance of a system.
- For generating and testing abnormal situations.

These systems are usually present in testing platforms aiming at providing controllers with the appropriate environment where they can train their skills, test new functionalities or play with the operational system without running the risk of affecting neither the rest of the ATC infrastructure nor the air space users.

1.2.3.6 Weather Nowcast and Forecast Systems

These systems receive, process, and internally distribute weather information from external providers such as national weather agencies such as AEMET²⁸ in Spain, the Met Office²⁹ in the UK or Météo-France³⁰ in France. Usually, these agencies provide weather forecast with a validity of several hours.

Nowadays, due to the advent of new procedures or separation rules such as TBS, there is a need for having almost real-time weather forecasts. This is what is called weather nowcast. This information is generally gathered from weather sensors that are under the responsibility of the service provider either the ANSP or the airport. This is the case of Lidars, MET radars or wind profilers that detect wake vortex or that measure wind speed in the proximity of the airport. Other systems that provide meteorological information are radars. Primary radars can also provide meteorological information, such as: the position of clouds, rainy areas, fog, etc. While Mode S sensors can download, from aircraft, meteorological information gathered by the FMS through the meteorological radar that aircraft have in their nose.

²⁸ AEMET stands for "Agencia Estatal de Meteorología" which is an agency depending on the "Ministerio para la transición ecológica y el reto demográfico" of the Spanish government. For further information please visit: <http://www.aemet.es>

²⁹ For further information please visit: <http://www.metoffice.gov.uk>

³⁰ For further information please visit: <http://www.meteofrance.com>

1.2.3.7 Supervision Systems

As already mentioned when talking about the “Technical Supervision Position” (TSP), there are two types of supervisions, the one that internally performs each system (i.e. Internal supervision) and the one that covers the whole ATC system (i.e. External supervision). The latter is the target of these systems. Thus, they perform constant checks to detect communication failures, malfunctioning situations, and changes in the status of each supervised system. Once detected, this information is sent to the Technical Supervision Position (ref. 1.2.3.2.4), where a specialized operator will address the problem.

On the other hand, the internal supervision is generally performed by a subsystem inside the system. It is normally a piece of software which is embedded in the system and which checks the status of the main components including the communication interfaces. These modules are the ones that are interrogated by supervision systems, which build a global picture of how the system is currently performing.

1.2.3.8 Communication Networks

The communication networks are a fundamental part not only of the ATC system but also the ATM one as they allow the exchange of information by means of interconnecting all the actors that are involved in such a system (people, systems, vehicles, aircraft, facilities, etc.). Thus, they provide the communication infrastructure for providing the Aeronautical Telecommunication Service, which is composed of:

- Aeronautical Fixed Telecommunication Service (AFTS).
- Aeronautical Mobile Service (AMS).
- Aeronautical Broadcast Service (ABS).

All these services exchange either voice or data or both. This exchange is performed among different types of communication stations which can have a fixed location or a mobile one. These communications include air-air, ground-ground, and even ground-air communications. Another point to bear in mind is the transmission mean, which can be provided by dedicated physical lines, ground networks, radio and satellite communications.

The presence of communication networks is important because it introduces some effects that should be considered. One of the most problematic is the presence of non-deterministic delays, such as latencies, that appear on the transmission lines. This effect and others will be presented and some proposals for their mitigation will be provided in sections 2.1.6 and 6.10.

1.2.4 Air Traffic Control Operators and Ground Personnel

Another important element of the ATC system is the people who work on the ground side.³¹ There are plenty of activities which are led by human beings and so the results of the system operation are heavily affected by them. This is the reason why it is mandatory to provide them with the best information possible, so they can take the right decision in the right moment.

It is important to notice that there are two different types of people within the ATC world, the ones devoted to operational control activities and ones carrying out non-control related tasks. The focus for this work will be the ATC controllers or operators (ATCo)³² as they will be the main consumers of surveillance information, which will be the input for performance assessment tools. The non-control related people, like flow management operators, air space designers, ground personnel, etc. have no special interest for the purpose of this work as they will not interact with surveillance information.

1.3 Surveillance chain

Surveillance sensors are the “eyes” of controllers when aircraft are flying in the sky, or when moving around the airport surface.

The information they generate has to travel to the different ATC facilities where they are consumed if provided in the appropriate data format (e.g. ASTERIX). It is the task of the surveillance infrastructure to transfer this information from the sources to the destination, which usually is the controller working position. As such, the surveillance infrastructure not only include the communication networks and protocols, but also some processing elements such as front-end processors, routers, or SDP systems. The surveillance infrastructure together with the sensors form what it is called the surveillance chain.

Controllers usually have direct vision of aircraft on the ground from the ATC tower; but as airports grow, it is easy to find areas that are out of their vision (shadow areas) or hidden by other targets. For those areas, surveillance sensors are the mean of locating and identifying aircraft and vehicles.

Currently, there are several families of surveillance sensors, which are described here after.

1.3.1 Surveillance sensors

1.3.1.1 Radar systems

This is the family of surveillance sensors that has been longer supporting ATC controllers with aircraft and vehicle information. They appeared in the military scene during World

³¹ Remember that pilots and stewards were introduced when talking about aircraft.

³² Although in this text controller and ATCo are going to be used as synonyms, there are experts which differentiate between them. Thus, a controller issues ATC clearances and instructions while the operator is more dedicated to routine tasks, which are not in direct contact with aircraft, or in case they have direct communication with aircraft, they just provide recommendations or information services. Thus, the controllers assume more responsibilities than the operators.

War II and were incorporated to the civil air traffic control in the latest forties. Since then, they have evolved in different directions to cope with the necessities of the different ATC facilities. Thus, we can find:

- En-route radars (PSR, conventional SSR,³³ Mode-S)³⁴, which feed ACCs with surveillance information.
- Approach radars (PSR, SSR, PAR,³⁵ etc.), which can be found in the proximities of an airport and provide APPs with surveillance information concerning landings and departures to/from an airport or a group of them.
- Surface radars (SMR, ASDE,³⁶ etc.) are the ones in charge of feeding ATC towers with ground surveillance information.

Further details about these sensors are presented in section 2.2.

Nowadays, these sensors are being substituted in some environments by other surveillance sensors that present similar performances at a reduced cost. This is the case of the next two families. However, as long as there exist the need for detecting non-cooperative targets, primary radars will continue being part of the airport domain.

1.3.1.2 ADS systems

Automatic Dependent Surveillance (ADS) is a surveillance mean with two different variants, ADS-B and ADS-C [Taberner 07].

The ADS-B is a surveillance application that exploits the information that an “ADS-B equipped” aircraft³⁷ broadcasts³⁸ while flying through the airspace or moving around the airport. This information includes identification, position and other data that are computed and updated by the Flight Management System (FMS) of the aircraft, and then stored within the BDS registers of its Mode S transponder. The broadcasted information is then received by all the systems that are in the surroundings of the transmitting aircraft and only those with the appropriate equipment can decode and process such information. This includes ADS-B ground stations, which process and forward it to ATC facilities (including ATC towers at airports), and other “ADS-B In” capable aircraft or vehicles. The information that reaches the ATC facilities is then processed by the Surveillance function, which will fuse it with other surveillance information to present the controller with a complete situation awareness picture. In the case of aircraft and vehicles, the “ADS-B In” capability consists in the ability of processing the ADS-B information provided by other aircraft and being able to exploit it for purposes such as increasing the

³³ In this work, the term “conventional” SSR groups both the systems called “Mode A/C” SSR and the ones which are called “Monopulse” SSR (MSSR), opposite to “Mode-S” SSR systems.

³⁴ This term refers to what are called Mode-S SSR opposite to “conventional” SSR.

³⁵ PAR stand for Precision Approach Radar. When used in the civil domain, it is an SSR with reduced coverage but high update rate that ease the surveillance of aircraft in the approach phase to an airport.

³⁶ ASDE stands for Airport Surface Detection Equipment, which is the american equivalent term for referring to an SMR.

³⁷ These aircraft are generally referred as being “ADS-B Out” capable. The equipage that the aircraft needs to carry depends on the data link technology that the aircraft implements to broadcast its internal information. The most extended ones are Mode-S 1090 ES, UAT or VDL Mode 4. The former is the one selected by ICAO as the worldwide ADS-B standard.

³⁸ This is the reason why the “B” is added to ADS.

situation awareness picture of the pilot or improving the separation management performed by the pilot. Thus, this capability allows the development of applications within these aircraft and vehicles, which are referred as Airborne Separation Assurance System (ASAS) that are further commented in section 2.3.2.

On the other hand, ADS-C is based on contracts³⁹ that can be defined on demand, on a periodic basis or when triggered by an event; being its main application providing Air Traffic Services Units (ATSU)⁴⁰ with surveillance information in regions without radar coverage such as oceanic or continental areas. Recently, other applications based on ADS-C have been proposed such as ADS-C EPP⁴¹ which is of interest for i4D⁴² in order to download FMS information concerning next waypoints and predicted times when they will be over flown.

ADS systems are seen as the cornerstone that will ease the way forward in the automation of the ATC system. However, there are availability and security aspects which should be addressed, and some mitigation means included [Soto 11]. For further details about these sensors please visit section 2.3.

1.3.1.3 Multilateration Systems

These sensors take advantage of the information sent by aircraft and vehicle in response to other systems (such as SSR or Mode S replies, ACAS interrogations, etc.) or due to periodic emissions (e.g. ADS-B messages), if they are passive; or by directly enquiring them in case they are actives; for extracting the location of the mobiles by means of hyperbolic triangulation techniques and, if possible, they also extract part of the content of the message that is used for computing the location to extract identification information and other data.

Depending on the size of the coverage area, there are two categories:

- Wide Area Multilateration (WAM) systems.
- MLAT systems, also referred to as Local Area Multilateration (LAM) systems.

The former is thought for huge areas having a similar coverage as an en-route radar, while MLAT systems are design for covering reduced areas such as airports where having a good precision is important.

For further details about these sensors please go to section 2.4.

1.3.1.4 Independent and Non-Cooperative Surveillance Systems

This is a relatively new family of sensors that share the fact that they provide independent measurements of the target, that is, they are computed by the sensor themselves, without

³⁹ This is the reason why the “C” is added to ADS.

⁴⁰ The ATSU is another way of referring to an ATC facility. Please, do not confuse with the system that some aircraft include that is related to CPDLC.

⁴¹ EPP stands for Extended Profile.

⁴² i4D is an operational concept based on the exchange of trajectory information between the aircraft and ground to keep synchronized this information aiming at improving the sequencing and merging of aircraft at certain waypoints.

the collaboration of the target and that they show low RF demands and a high resilience to some interfering elements such as wind farms. They are interesting from contingency and security purposes as there is no need to trust any information externally generated.

Within this family, there are several sensor technologies which are grouped. There are some that are not really new, such as Multi Static Primary Surveillance Radars (MSPSR) that are systems with one or several transmitters that distribute multiple receivers around the area to be covered that can exploit opportunity signal to perform its measurements.

Other detection technology that is progressing considerably is the one based on continuous wave signals. Within this group it can be found primary short-range radars such as SQUIRE from Thales, which is a man-portable ground surveillance radar for detection/tracking and classification of moving ground targets based on FMCW radar technique [SQUIRE].

Finally, there is the Holographic radar concept from Aveillant⁴³ [Aveillant], which is supported by “a static, staring array to continuously and simultaneously monitor all the airspace it can see and the use of advanced signal processing techniques to discriminate targets”. The THEIA 16A model is proposed as a mitigation for windmills and that adds the possibility of target type recognition. This sensor is considered to belong to an already defined class which is called ubiquitous phased array radars (UPAR). An introduction to these systems can be found in [Skolnik 03].

The problem with these systems is that on one side, there is not yet much experience gained and, as such, there are not mature enough measurement models and, on the other side, the availability of recordings for research purposes are not easy to obtain.

1.3.2 Surveillance distribution systems

1.3.2.1 Networks

As already mentioned, when introducing the surveillance chain concept, there is a need for transporting the detection data generated by sensors to the processing systems (i.e. SDP systems). This is the task of communications networks which will communicate sources and destinations with the appropriate link technologies as sensors can be placed in the more varied locations.

This topic is further developed in section 2.1.6.3.

1.3.2.2 Nodes

The information generated by sensors is generally provided following the ASTERIX data format, which is then transported using a communication protocol that can be different depending on how the communication with the sensor is established. This fact introduces the need for systems that provide the conversion from any special communication protocol into the ones expected by the SDP systems that are going to process the surveillance data, that is to LAN⁴⁴ protocols. This distribution function is performed by front-end processors such as the former EUROCONTROL RMCDE [RMCDE] or its

⁴³ Now Aveillant is a company of the Thales Group.

⁴⁴ LAN stands for Local Area Network. It is a term that groups data communication networks with generally a limited coverage area and a local scope. A well-known example is Ethernet (i.e. IEEE 802.3).

evolution the SDDS [SDDS], which also act as routers or communication gateways for surveillance networks, such as SURNET, allowing the injection and the selection of data to or from these surveillance networks.

These systems also provide additional capabilities to manage the surveillance data flow like data validation, conversion, and filtering.

1.3.3 Surveillance data processing system

Surveillance Data Processors or Surveillance Data Processing (SDP) systems⁴⁵ exploit the information provided by surveillance sensors. Their target is to extract as much information as possible from different sources of surveillance information that detect the aircraft and to fuse it together, so the result is expected to be better than the sum of the different bits of information. This process is called sensor data fusion⁴⁶ as it integrates the information concerning a certain target provided by different information sources (sensor data) into a consistent, improved and enlarged piece of information, that is, the “track”.

The SDP system additionally offer the possibility of configuring, for example, the content, the update period and the sources of the information delivered within the track by the definition of a service, which can then be subscribed by any of the track consuming systems. Even more, there can be several of these configurable services, each one tuned for the needs of a certain client system. Thus, the information computed by SDPs is normally forwarded to more systems than just the CWP, among them is the correlation subsystem, which will compare it with the one coming from the FDP and will correlate the corresponding track with the flight plan whenever is appropriate.⁴⁷ Other systems which are fed by the SDP are the MTCD, the safety nets or STCA, the CMON, etc.

1.3.4 Surveillance data presentation

The information generated by SDP systems, i.e. tracks, is then presented to the controllers on the console of their CWP. Usually, the way of presenting this information is configurable at the HMI provided by the CWP to display surveillance data, so the controllers can decide which one, among the one received from the SDP, is more interesting for their work and which one should be kept hidden. This is usually done by configuring the information displayed on the track label that can be found close to the track symbol. The latter represents the target position on the displayed map where all the surveillance information is presented and that gives rise to the controller situation awareness picture.

⁴⁵ SDP can also be referred to as SDPDS, which stands for Surveillance Data Processing and Distribution System. This distinction is based on the inclusion of the capability of serving the tracked data to other consumer systems based on some particular conditions defined in a service agreement. However, both acronyms are considered as synonyms in this thesis.

⁴⁶ This process is usually implemented by a module of the SDP that is known as Multisensor Data Fusion (MSDF).

⁴⁷ There are several criteria when deciding whether a Flight Plan matches a certain track, among them one can find the SSR code or Mode A, the ACID, the AA, the Mode C, the position, etc.

1.3.5 Performance Evaluation Tools

This kind of system is quite common in ATC facilities mainly for En-route and TMA airspaces to assess the quality of the information that is delivered by the surveillance chain. Due to the importance of surveillance information for the correct performance of the ATC service, it is very convenient to analyse whether the information that is used for building the controller situation awareness picture is coherent with the actual air traffic situation. A well-known example is the SASS-C suite provided by Eurocontrol [SASS-C]. However, such a system lacks an airport counterpart which is the main topic of this thesis.

The aim of these systems is to assess the performance of surveillance sensors and SDP systems, by means of comparing their output (target reports and tracks respectively) with a “reference” trajectory that is computed or imported by the performance evaluation tool.

This “reference” can be:

- Obtained from “Opportunity Traffic”.⁴⁸
- Generated by a more precise surveillance measurement system such as a D-GPS receiver onboard the aircraft or ground vehicle.
- Generated by a simulation system.

In the framework of SESAR PJ14.04.01 [PJ14-04-01], the generation of the “reference” is not considered as part of the performance evaluation tool. However, the author of this thesis considers that the generation of the “reference” is the core part of a performance evaluation tool as not considering it within the scope converts the tool in a pure display where the comparison of the “reference” vs. the input is presented in a more or less fancy way, as there should not be any room in the way the performance metrics are computed and as such the performance results should not show any divergence. These metrics and the procedure to compute them are expected to be fully specified in documents or standards such as ESSAPS or GEN SUR SPR.

Finally, airport-based performance evaluation tools are the main topic of this work, so their inputs, building elements, architecture and algorithms will represent the centre of the discussion all along this document.

1.4 The airport environment

As just said, the contributions of this thesis are constraint to the ground domain and more precisely to the airport environment, so the most important terms are presented within this section. It is convenient to clarify that the “airport” is a particularization of the “aerodrome”, which is a wider concept that represent any area where aircraft operations can be carried out. While aircraft adds some requirements to this concept by requesting the presence of several infrastructures such as one or several runways and one or several passenger terminals, in case of a commercial airport.

⁴⁸ By opportunity traffic is understood all the information coming from mobiles (vehicles and aircraft) that are moving around the airport or airspace under analysis while performing their routine tasks.

The term “aerodrome” is introduced because it is the one used in some contexts (e.g. SESAR) for referring to airports but without loss of generality. The following definitions have been obtained from EUROCONTROL ATM Lexicon [Lexicon].

Consequently, the aerodrome is defined as: "A defined area on land or water (including any buildings, installations, and equipment) intended to be used either wholly or in part for arrival, departure and surface movement of aircraft."

1.4.1 The apron or platform

Definition [Lexicon]: "A defined area on a land aerodrome/heliport, intended to accommodate aircraft/helicopters for purposes of loading or unloading passengers, mail or cargo, and for fuelling, parking or maintenance."

1.4.2 The runway

Definition [Lexicon]: "A defined rectangular area on a land aerodrome prepared for the landing and take-off of aircraft."

1.4.3 The taxiway

Definition [Lexicon]: "A defined path at an aerodrome established for the taxiing of aircraft and intended to provide a ground movement link between one part of the aerodrome and another."

1.4.4 The manoeuvring area

Definition [Lexicon]: "That part of an aerodrome to be used for the take-off, landing and taxiing of aircraft, excluding aprons."

1.4.5 The movement area

Definition [Lexicon]: "That part of an aerodrome to be used for the take-off, landing and taxiing of aircraft, consisting of the manoeuvring area and apron(s)."

1.4.6 Holding position and stop bars

There are several types of holding positions, which are prescribed in ICAO Annex 14 [ICAO Annex 14]. Thus, one can find holding bays, runway-holding positions, intermediate-holding positions and road-holding positions. The runway-holding position is also known as a "holding point". All these positions are intended to protect a certain element of the airport, generally the runway, although it can be an intersection or an obstacle limitation surface or a place where radio navigation aids may be affected. These holding points are defined by means of markings on the ground which identify the location where an aircraft or vehicle is required to stop in case it has not already received

the appropriate clearance from the controller in order to proceed onto the runway. Depending on the type, the marking would be different.

Stop bars lights are recommended by ICAO in Annex 14 when talking about low visibility operations, and they are present in all major airports. They appear as a reinforcement of a holding point to avoid as much as possible runway incursions. Thus, stop bars lights and holding points markings are generally collocated.

The FAA defined the Stop Bars in the following sense [FAA AC 96]: "Stop bar lights consist of elevated and in-pavement red fixtures that are installed at the runway holding position or instrument landing system (ILS) critical area holding position marking. Stop bars may be controllable by ATC and will include a system of in-pavement green taxiway centreline/lead-on lights at locations where aircraft will enter or cross a runway."

Airservices Australia provide a functional view of them more linked to safety issues, thus they say that "Stop bars are considered a valuable defence against aircraft and vehicles inadvertently entering a runway without Air Traffic Control (ATC) clearance".

It is important to mention that stop bars or holding points are very useful references for performance evaluation tools as they represent locations in which the targets are likely to stop. So, this information can be exploited while registering sensors or when estimating their statistical features.

1.5 Increasing air traffic capacity, sustainability and safety. The performance-based operation paradigm

As it is stated in [Nolan 11], changes in air traffic have always been driven by public opinion due to the occurrence of mayor aircraft accidents or incidents. The first decade of the XXI century started with the belief that ATC systems were not correctly dimensioned for the expected future demand.⁴⁹

This fact forced ICAO to devise a deep change in the way the air traffic was managed. As such, they defined a paradigm change in the way of managing air traffic from an airspace-based control system to a trajectory-based management system aiming at increasing air traffic capacity. This evolution was first presented in [ICAO 9854] in 2005 and it was called the "Global Air Traffic Management Operational Concept" (GATMOC), which has its roots in the previously defined FANS (Future Air Navigation System) concept.⁵⁰

⁴⁹ Back in 2008, due to the crisis in the United States and Europe, the need for more capacity was slowed down, so there was the belief that there was some extra time to prepare the ATM system to absorb the expected increase in the air traffic demand. In the beginning of 2020, the situation was like the one experimented in the beginning of the century coming back to important delays and congestion but again there is the opportunity provided by the huge strike of the COVID19 crisis, whose impact will last for several years. Let us see, if this time, systems are prepared for the future.

⁵⁰ The FANS concept was launched in 1983 when ICAO established the Special Committee on Future Air Navigation Systems (FANS). The target was to improve Communication, Navigation and Surveillance by means of an increased automation and the introduction of digital communication such as satellite-

Additionally, there is an increased awareness on the environmental sustainability. Matters such as global warming and CO2 emissions have also boosted the shift. Another matter to tackle is the noise reduction as many of the mayor airports are now surrounded by buildings due to the growth of cities.

Finally, safety⁵¹ aspects [ICAO 9735] have also encouraged the paradigm change as they are considered as one of the key performance areas (KPA) to be improved in the ATM environment. Nowadays, safety is considered a core part of the ATM system which must be considered from the very beginning in the system engineering lifecycle every time a change is proposed to the whole system.

This idea of the modernization of the ATM operational concept was then assumed by the different regions taking a different shape although keeping the same flavour, that is the trajectory-based operations (TBO). Thus, the United States created the NextGen programme [Next Gen] by means of the FAA, the same way that the European Union devised the SESAR programme [SESAR].

The "Next Generation Air Transportation System" (NextGen) programme was designed to be the evolution of the U.S. National Airspace System (NAS) from a ground-based system to a satellite-based system. This way the FAA is implementing an important modernization of the entire U.S. air transportation system seeking to exploit the GPS technology in order to improve the efficiency of the system together with its safety margins.

On the other hand, the SESAR programme is the technological pillar of the Single European Sky (SES) initiative and it was a mandate of the European Commission (EC) delegated to the SESAR Joint Undertaking⁵² (SJU) [SESAR]. Consequently, SESAR contributed to both the SES strategic performance objectives, by means of designing the solutions in order to achieve the Key Performance Indicators (KPI) that had been defined for the estimated traffic growth, and to the SES performance-driven approach which is focussed in four Key Performance Areas (KPA), namely: environment, cost-efficiency, safety and capacity/quality of service. These KPAs drove the research and development activities that were managed by the "initial" SESAR and its fist extension (i.e. "Wave 1") and will drive the subsequent one "Wave 2", which was launched in the beginning of 2020.

Both initiatives, NextGen and SESAR, were also replicated by other countries but in a different scale. All these initiatives pursue the same target, which is to make air travel more cost-efficient, safer, and environmentally friendly. This is expected to be achieved by moving from an airspace-based control to TBO,⁵³ which implies the development of

based ones and data links. These improvements allowed the integration of applications such as CPDLC, ADS-C and satellite navigation.

⁵¹ Safety is defined by ICAO as "The state in which the possibility of harm to persons or property damage is reduced to, and maintained at or below, an acceptable level through a continuing process of hazard identification and risk management."

⁵² The SJU is the group of partners which manages the evolution of the SESAR programme and initially included two founding member (EUROCONTROL and the European Commission), three airport operators (ENAI, ENAV, NATS and NORACON), six air navigation service providers (DFS, DSN, ENAI, ENAV, NATS and NORACON), five ground and aerospace manufacturers (Frequentis, Indra, NATMIG, Selex ES and Thales), two aircraft manufacturers (Airbus and Alenia Aermacchi), two airborne equipment manufacturers (Honeywell and Thales) and several associated partners.

⁵³ In SESAR a further step is envisioned, the performance-based operations (PBO).

the appropriate technologies and the tuning of the operational procedures in order to meet the new paradigm requirements.

The new paradigm shift also affects controllers as their work will change from more tactical activities, with many interventions, to more strategic ones, where conflicts are anticipated with enough time to negotiate the best solution among the available ones. This solution is expected to be reached by means of a negotiation process not only with the aircraft within the sector but also with the rest of the affected actors⁵⁴ (other sectors, controllers, aircraft, vehicles, etc.). This fact changes the role of the controller into a trajectory supervisor/manager. This is possible because there is an improved knowledge of the underlying trajectory which depending on the programme is called a negotiated route⁵⁵ or the Reference Business Trajectory (or RBT)⁵⁶. Both terms incorporate the 4D-trajectory concept which describes the trajectory that the aircraft is going to follow in three dimensions (3D) plus time. This trajectory is expected to be gate-to-gate, so both the "taxi in" and the "taxi out" phases will be incorporated to the trajectory. Another important fact is that in order to exploit this trajectory, it is important to synchronize the information that is available on the aircraft with the one computed on the ground, which leads to an improvement in the communication systems.

However, the paradigm change is not stopped at the TBO, as SESAR also foresaw a subsequent step that was focused on achieving the full SES performance requirements. This concept step aims at reaching a situation in which the ATM system adapts its service level based on the performance characteristics not only of the flight but also of the whole ATM infrastructure.

With the availability of this improved trajectory and the focus placed on the performance of the overall system, it is possible to build on top a set of tools that will support the controller on his new tasks based on these premises. This is normally part of what is called the advance automation.⁵⁷ It is important to mention that the efficiency of this concept relies on the fact that the same number of controllers will control more aircraft than they do nowadays, thanks to the support provided by all these tools. This is obviously done keeping or improving the current safety level.

One of these tools will be the performance assessment tool, which is expected to analyse, assess, and inform in an automated way about the performance of the surveillance function and the surveillance sensors that will cope with the actual location of the flight. Additionally, these tools can also classify the aircraft movements, present the results of a performance evaluation, or assess the compliance of the actual flown trajectory with the initially defined one (i.e. RBT). Their work is done in a background mode, as they use already processed surveillance data mainly coming from opportunity traffic, although they can also come from fixed field transponders (FFT), fixed reflectors or simulators.

⁵⁴ This is generally referred as Collaborative Decision Making (CDM).

⁵⁵ This is the term used for the improved trajectories that will be managed in the future in the NextGen domain.

⁵⁶ This term will be the one used within SESAR.

⁵⁷ SESAR included in its long term and innovative research activities managed by WP-E, the founding of several research networks focused on different topics. One of them deals with advances in automation, this network is: HALA! – Higher Automation Levels in ATM. For further information visit: <http://www.hala-sesar.net>

Depending on the context, they can be a pure offline tool or a pseudo online one where the calculations performed by the tool are fed back to the SDP system.

1.6 Problem statement

The problem that performance assessment tools face consists in estimating the actual trajectory that a mobile (aircraft or ground vehicle) has followed around the airport to use it as the reference to be compared with the information provided by surveillance sensors (i.e. plots, target reports or tracks⁵⁸), each one at a time, or by the SDP system, also referred as the tracking system (i.e. tracks or multitracks).

The estimation of the actual trajectory, which is usually called the “reference” trajectory, will be the reference for carrying out the performance assessment evaluation of the quality of the data provided by sensors and trackers. There are several ways of computing the “reference” trajectory as mentioned in section 1.3.5.

In this thesis, the focus is placed on those techniques based on exploiting the opportunity traffic, whose information comes from different sources, although the others are also described in section 3.4.

Moreover, this problem can be considered as a data fusion problem [Llinas 01], where the reference trajectory is obtained from the information provided by several data sources (surveillance sensors), each one with its own features and its own coverage. Taking this fact into account, all the needed processes are presented and conveniently tuned for addressing this problem with the focus placed on the airport environment.

Additionally, there are some constraints and prior knowledge that are worth exploiting in order to improve the quality of the estimation, these can come from the side of the airport, such as: the use of the airport map restrictions, airport procedures and the aircraft ground movement models; or from the side of the aircraft, such as the aircraft dimension, the wheelbase or the location of some of the antennae used for transmitting or receiving information.

1.7 Contribution and Methodology

The contributions of this thesis are oriented to:

- Extend the performance assessment tools to the airport environment and to the “new” sensors.
- Exploit the information provided by the airport context (e.g. layout, procedures, features, etc.).
- Exploit the information concerning the targets themselves (e.g. dimensions, location of antennae, wheelbase, etc.).

⁵⁸ These “tracks” are often called “monotracks” or “monosensor” tracks as they just include information coming from just one sensor. This is done to differentiate them from the tracks provided by SDP systems which are also called “multitracks” as they usually fuse information coming from more than one sensor.

These contributions range from the definition of sensor measurement models to the presentation and adaptation of ad-hoc reference trajectory computation strategies, passing through the definition of some architectural blocks, the design of airport-based bias estimation algorithms and the proposal of some new applications and future uses.

The sensor measurement models are enlarged to cope with time biases either sensor-based or target-based. The focus is placed on the ones that are now reaching the right maturity level that allow their operational use, such as ADS-B and MLAT; and the new ones, such as the ones known as INCS. The latter are still not very consolidated sensors in the airport domain and, as such, their model need still to be agreed, so an initial proposal is presented.

It is important to bear in mind the exploitation of the airport characteristics and layout together with the aircraft features in both, the bias estimation and in the reference trajectory computation. Additionally, the knowledge of the aircraft type is used for fine tuning the reference trajectory.

The contribution also includes several possible applications of the designed performance assessment tool such as aircraft movement classifiers or airport dynamic layout computation enriching the basic layout with kinematic information.

The methodology that has been used consists in building a prototype which is used for testing and evaluating the different approaches that have been investigated to identify the more appropriate ones for the problem under consideration.

The evaluated algorithms have been mainly tested with real data to confirm that they are suitable for this application as the author had access to data from several airports and different service providers. However only one scenario, the one provided by NAV Portugal complied with all the requirements of sensor overlapping in time and in coverage imposed by the proposed solution. So, all the results from real data presented in CHAPTER 8 will come from this scenario.

When synthetic data are injected to the algorithm, then Monte Carlo simulations are used to show the performance of the selected algorithms either for bias estimation or for trajectory reconstruction. This will be done to compare their performance with the one from other algorithms that can be found in the literature.

1.8 Structure of the Document

The structure and the content of this thesis aimed at presenting the reader with a complete document, so it has been decided to try to make all the chapters as self-content as possible. This is the reason why there is a lot of information that is part of the context of the thesis but not the contribution that it represents.

This first chapter introduces the context of performance assessment tools following a top-down approach. Consequently, the ATM service is presented describing its main actors, the services provided, and the systems in charge of supporting them. This is done aiming at providing the general framework where the contributions are confined. Then, the current driving lines are pointed out mainly focused on the European version of the ATM evolution concept, which is represented by SESAR 1 and its subsequent extensions.

Afterwards the problem to be addressed by this thesis is briefly described and, finally, the contributions are outlined, and the methodology is presented.

Chapter 2 provides essential background information on surveillance sensors. Thus, some notions are provided concerning coordinate systems and projections, surveillance data exchange, and transmission means, aiming at gathering the appropriate knowledge for understanding the particularities of the different sensor families. Next, each of these families is presented following the same layout per member of the family, firstly its features are described, secondly its particularities are outlined, then its proposed measurement model is presented, which will be used throughout this thesis, and finally, the data format used to send the information is shown. This schema is followed not only for radar systems, ADS applications and multilateration systems, which are the most common surveillance sensor in the aerodrome domain, but also for the new family of sensors known as INCS. The chapter ends with a summary of the already known problems and some ideas provided by the author regarding the future of surveillance sensors.

Chapter 3 introduces the state of the art of the Performance Evaluation Tools starting from the beginning of this kind of tools dated in the mid 80's to the current moment. The different methods for implementing performance evaluation tests are enumerated and a summary of them is presented. Afterwards a classification of the variety of performance evaluation tools is listed taking into account the application domain, the reference trajectory computation techniques, but the main division is the one done into those devoted to evaluating surveillance sensors and the one focused on evaluating tracking systems. Some examples of both types are presented. Then, the SESAR functional decomposition for the aerodrome ATC domain is shown. The focus is then placed on the surveillance function, whose functional architecture from the SESAR project 12.03.01 [SESAR P12.03.01 D02] is taken from [Soto 11]. Moreover, the proposed architecture for Performance Evaluation Tools is detailed providing first the context of application, second the functional decomposition, then the logical division and finally, the envisaged external interfaces. Finally, some requirements to be expected from the SESAR paradigm are proposed and some conclusions are presented.

Chapter 4 changes the focus of the description of the tool from the functional view presented in the previous chapter to a process view where all the tasks are presented in the sequential order in which they are executed. This chapter provides the overall view per processing block while the following ones will focus on those which are considered of more importance for this thesis.

Chapter 5 provides the details of the data reading processing block. It starts describing how the data are incorporated to the system. Next, the filtering criteria for selecting the data that will be subject to the performance evaluation is presented and then, the way the remaining data is stored within the system. Finally, there is a presentation capability that offers the possibility of checking whether all this acquisition process has been correctly performed by visually showing the imported data.

Chapter 6 is devoted to the previous adjustments that must be performed to the incoming data to exploit them without altering the rest of the information. Performance assessment tools are data fusion systems and so it is a must to pre-process data in order to perform a further processing with enough guarantee, even more when they come from different sources that use different measurement technologies, provide their output in different coordinates and with disparate quality. In this chapter, the general pre-processing

processes are addressed, which are: the coordinate transformation, the bias estimation and the association. Concerning the bias estimation process, there are several strategies which are presented depending on the features of the environment and the sources of the information; there are also different methods for computing the biases; and finally, the proposed architecture is presented. On the other hand, the target report to trajectory association proposal is detailed and some conclusions are outlined.

Chapter 7 is the core part of the thesis. It focuses on presenting, analysing, designing, and evaluating several reference trajectory computation strategies to select the best one for the problem under discussion. It starts providing a definition of a key concept, the “reference” trajectory, and continues providing the statement of the problem of “reference” trajectory estimation computed from opportunity traffic. Two families of techniques are presented, the Bayesian ones and the Non-Bayesian ones. The focus is going to be placed on the former. Consequently, optimal estimation techniques require a thorough presentation as they involve more than just algorithms. As such, systems are classified depending on the equations that describe their behaviour. Then the focus is on a particular group, the dynamical systems. Afterwards, the state-space model representation is described as it is used for modelling the aircraft behaviour, and in this case, it is the representation that has been chosen for modelling our problem. Some aircraft ground movement models are incorporated from the literature (e.g. the three-wheel model). Once the model is defined, the search of the optimal estimation of the reference trajectory is started. This implies presenting the different types of optimal estimation problems (filtering, smoothing and prediction) and then placing the attention on the smoothing problem which is the one that better suits our problem. Due to the way in which smoothing can be performed, there is a need for introducing some filtering algorithms as they are part of some of the smoothing algorithms under analysis. Additionally, some suboptimal strategies are presented together with other techniques such as particle filters and wavelets. Next, the airport context is considered as it can potentially improve the estimation of the reference trajectory by constraining the possible solutions. As targets do not always follow the model that is assumed, there are some effects that may appear and whose impact must be taken into consideration. The most common ones are presented and the way of dealing with them is defined focusing in their effects in the “reference” trajectory estimation technique that have been presented. Finally, some conclusions are outlined.

Chapter 8 introduces the research platform that has been used for testing the different technologies and solutions that have been analysed and evaluated as part of the performance assessment prototype. To evaluate the different algorithms to be implemented in the prototype, some synthetic scenarios have been defined which try to represent some particular situations⁵⁹ that can be found in an airport. These synthetic scenarios are used in a Monte Carlo simulation to obtain some performance indicators of the different algorithms that are selected as possible candidates for bias estimation or trajectory reconstruction. Additionally, some real traffic scenario are employed in order to check how well the system deals with real data. Thus, the target of the platform is to evaluate the different alternatives to be implemented in the prototype to choose the best one. Then, the chosen ones are assessed and the results are presented at the end of this chapter.

⁵⁹ By situation, it is meant a combination of possible aircraft ground movements and airport taxiway layouts.

Lastly, chapter 9 presents the conclusions and provides recommendations for ongoing research based on the results obtained all along this work and considering an analysis of what can be the performance evaluation tools of the future.

Some annexes are presented for the sake of completeness and clarity.

CHAPTER 2

SURVEILLANCE SENSORS AND THEIR DATA MODELS

“Remember that all models are wrong; the practical question is how wrong do they have to be to not be useful.”

— George E.P. Box

Every data processing has a starting point, that is, a source of the information that later on is going to be elaborated before being provided to its consumer. Surveillance sensors represent this initial source of the information within the surveillance chain.

Once generated, the information provided by surveillance sensor needs to reach the data fusion centre where it will be merge with the information provide by other sensors. This happens thanks to the use of communication protocols, which allow the data to reach its intended destination traversing one or several communication networks. Then the content of these information needs to be understood. Thanks to the ASTERIX data format, this task has been eased as now almost all sensors exchange data using this data format.

Due to the importance of the provided information and the risk that it may mislead us, it must be properly modelled. This means identifying the main sources of their errors that can be either random, usually represented by noises, or systematic. The latter are commonly known as biases and their estimation and subsequent cancellation deserves a relevant part of the research effort in any data fusion process, which is presented in CHAPTER 6. Moreover, each sensor family has its own features and particularities, which are shown in the following sections together with the coordinate conversions that are needed for representing all the information in the same reference framework.

2.1 Surveillance sensors

Surveillance⁶⁰ is one of the key domains of the current ATM concept. It informs about the location and identification of the mobiles (e.g. aircraft and ground vehicles) that take part in the situational awareness picture that is shown to the controller at any time.

Sensors are the senses of the surveillance chain as they provide the inputs for detecting, identifying, classifying, and tracking all the elements that can be found in a certain airspace or airport. They are the foundation layer for all ATCo activities, which are very sensitive since many human lives depend on their correct performance.

Considering that most of the information provided by a sensor is fused with the one coming from other sensors to generate the common situational awareness picture, there is a need for correctly modelling it. As in any data fusion process, coordinate conversion, sensor registration, and noise characterization will be a prerequisite for the good performance not only of the surveillance chain but also of the whole ATC service.

2.1.1 General features

A surveillance sensor is a measurement equipment which principally estimates the target location; and depending on the data received from it, the sensor can also extract some other information, such as its identification, or even provide a classification of the actual nature of the measurement. In the surveillance domain, sensors can be grouped into several families⁶¹ depending on the technique that they use for determining the position of the target.

Thus, radars measure the time elapsed between the transmission of a signal and the reception of its echo or its response. Multilateration surveillance systems compute the aircraft position by analysing the time difference of arrival (TDoA) of the same signal to several ground stations that listen to it. ADS-B relies on the information broadcasted by the target itself.⁶² Lastly, the newcomers grouped under the umbrella of the INCS incorporate different radar-based technologies for determining the position of the target ranging from mono-static continuous wave systems to multi-static location techniques based on exploiting opportunity signals.

As any measurement equipment, sensors are affected by errors. Besides, different sensors provide measurements that include different error terms as they use different location technologies. These error terms can be roughly divided into systematic errors or biases and random errors or noises.⁶³ In the next sections, all these terms are presented in a per sensor basis in order to define the measurement model of the corresponding sensor that

⁶⁰ Surveillance is “the observation of an area or space for the purpose of determining the position and movements of aircraft or vehicles in that area or space” [EATM 2004].

⁶¹ The main families are radars (PSR, SSR, SMR and Mode S), ADS applications (ADS-B and ADS-C), multilateration systems (LAM/MLAT and WAM) and recently the number has been enlarged with independent and non-cooperative systems (INCS).

⁶² Nowadays, this information has to be handled with care as it is susceptible to malicious modifications so some mitigation means have to be included. These mitigations take normally the form of integrity checks.

⁶³ There is a third group which are the correlated errors but they are not going to be considered within this document.

will be used all along this work. The rationale behind this presentation is that the performance of the methods and the algorithms in charge of their mitigation, reduction or cancellation relies on the quality of these models. These methods and algorithms will be presented in section 6.10.4 for biases, and CHAPTER 7 for noise reduction while reconstructing the aircraft trajectory.

Apart from measurement errors, there is a further problem which is that each sensor provides its measurements in its corresponding reference frame which can be local to the sensor or global. Consequently, we can find cases in which the same information is provided by several sensors but each one referred to its local reference frame. This is the case of position information, which depending on the sensor, can be provided in polar or spherical coordinates, in geodetic coordinates or in Cartesian ones. So, one of the first steps is to put all this common information in a common reference frame. This process is normally referred within the literature as registering.⁶⁴ Normally, the common reference frame is a Cartesian coordinate system centred on the fusion centre position which depending on the controlled airspace can be an ACC, APP or TWR. In the case of an airport, this common reference frame is centred in the Airport Reference Point⁶⁵ (ARP).

Furthermore, there is an additional trouble. Each sensor does not measure the distance to the same part of the target body to compute the reported position. Meaning that there is no common reference point to which refer this measurement. Hence, there are sensors, such as the SMR, which reports the position of the centroid of the target, while others report the position of the antenna that is used for either computing the target position, i.e. the GNSS antenna in case of the ADS-B, or broadcasting the signal that is used for calculating the target position, as it is the case for MLT systems when processing SSR replies, Mode-S communications or ADS-B messages. ICAO recommends establishing a common reference point on the aircraft and vehicles in order to be used in A-SMGCS and provides several alternatives, see [ICAO 9830]. The reference point that is going to be adopted in this thesis is the centroid or the geometric centre of the target, that is, the mean position of all the points in the shape. This is in line with the reference position that utilizes the SMR.

2.1.2 Surveillance sensor types

There are several criteria for classifying surveillance sensors, namely:

- **Cooperative/non-cooperative** surveillance sensors depending on the collaboration of the target in the detection and identification process.

A clear example of a cooperative sensor is the conventional SSR as it relies on the Mode A/C transponder on board the aircraft for estimating the position and obtaining the identification of the target. Other cooperative sensors are Mode S radars, ADS-B and multilateration systems (MLAT and WAM).

⁶⁴ Note that this term also includes bias estimation and correction.

⁶⁵ The “Aerodrome Reference Point” is the designated geographical location of an aerodrome, which is always defined in the aeronautical chart of the airport and in its entry in the Aeronautical Information Publication (AIP). It usually represents the geometric centre of the airport taking into consideration the position of the runways.

On the other hand, the SMR is an example of a non-cooperative sensor as it does not need any collaboration from the target because it extracts its position from the echoes of a transmitted signal. Another example is the Primary Radar (PSR), which works in a similar way but for the en-route or TMA domains. Finally, there is the new family of sensor known as INCS that also fits in this classification.

- **Active/passive** surveillance sensors depending on the interaction with the target. A sensor is active if it asks directly for information from the target, while it is passive if it just takes advantage of spontaneous emissions or reflections from it.

ATC Radar sensors are all active ones as they transmit a signal (pulse or interrogation) which is expected to be reflected⁶⁶ or answered⁶⁷ by the target. However, there are passive radars that take advantage of signals which are broadcasted by other systems, such as TV or radio emissions, in order to estimate the target position. These systems are commonly known as MSPSR (Multi Static PSR). Currently, they are not frequently found in the ATM domain, although an increase of their presence is expected in the incoming years.

ADS-B (1090 ES) is an example of a passive system, from the point of view of the ground station, as the latter listens to the spontaneous emissions that the Mode S transponder of the target broadcasts when it is "ADS-B out" capable and enabled.

Multilateration systems (LAM/MLAT and WAM) can be both active and passive depending on the availability of an interrogating ground station. Based on the environment, there can be the need for having an active multilateration system. This is normally the case in those ATC environments in which no signals are expected unless they are explicitly requested or when there is a need for guaranteeing a periodic update of the surveillance information. These systems guarantee that a periodical update is going to be available to the surveillance chain due to forced responses. Nowadays, due to the great number of spontaneous emissions that aircraft broadcast,⁶⁸ the availability of an interrogating ground station is not mandatory.

- **Independent/dependent** surveillance sensors based on where the location information is computed.

As its own acronym states, ADS-B is the only dependent surveillance sensor as it is the target itself the one which broadcasts its own internally calculated position⁶⁹ that is then received by ADS-B ground stations.

The rest of the sensors compute the target location using different techniques. As said before, radars measure the time of flight (ToF) of a signal (echo or reply) and

⁶⁶ SMR and Primary radars wait for echoes of the transmitted signal.

⁶⁷ Secondary (SSR) and Mode S radars wait for a reply from the target.

⁶⁸ ADS-B sends location information every 0.5 s and identification information every 0.5s, if the aircraft is moving, and every 5 seconds, if it is stopped.

⁶⁹ Generally, the aircraft navigation system uses a built-in GNSS receiver to compute its position, which is usually complemented by the inertial navigation system.

its direction, and multilateration systems use time difference of arrival (TDoA) of the same signal at several ground stations.

In the next table, the sensors of interest for this thesis are going to be classified using the previous division:

Table 1: Surveillance sensor classification

	Cooperative/Non-cooperative		Active/passive		Independent/dependent	
	Cooperative	Non-cooperative	Active	Passive	Independent	Dependent
PSR		✓	✓		✓	
SMR		✓	✓		✓	
SSR	✓		✓		✓	
Mode S	✓		✓		✓	
ADS-B	✓			✓		✓
MLAT	✓		✓	✓	✓	
WAM	✓		✓	✓	✓	
INCS		✓	✓	✓	✓	

There are other divisions such as automatic/on demand or primary/secondary sensors, but they are not as important as the ones previously described for the purposes of this work.

2.1.3 Sensor measurement model

As explained in the introduction to this section, surveillance information needs to be modelled in order to be used within the surveillance chain because the different methods and algorithms that are implemented exploit this information in a particular manner. Since their behaviour is quite predictable, estimation techniques based on Bayesian algorithms are the most extended ones. This way the prior knowledge can be exploited. There are three main reasons to obtain these measurement models:

1. They are the base for implementing bias estimation algorithms.
2. They are used for tuning tracking filters.
3. They are needed for simulation purposes to generate synthetic data with a similar behaviour to the one shown by the real sensor to be simulated.

Hence, sensor measurements can be characterized by a series of common error components, which affect surveillance tracking systems in different manners. Thus, the correct statistical modelling of these error components allows designing the appropriate countermeasures that enable the mitigation and/or correction of great part of them.

Especially remarkable for surveillance tracking systems are systematic errors, which introduce biases into the processing. This fact is crucial since these systems are a particular case of a data fusion system, which means that for enabling the alignment of all sensor measurements in a common reference frame, biases must be first estimated and then removed. The same situation happens with Performance Evaluation Tools as they perform a quite similar process. This is the reason why this matter is relevant for this work.

Each type of surveillance sensor has a different nature which implies that it has different error components due to different physical effects. Later in this chapter, some of these components are going to be presented and modelled.

2.1.4 Coordinate systems and transformations

2.1.4.1 Reference frames

The location of a mobile can be obtained as the separation measured by means of either a distance or an angle between the point to be located⁷⁰ and some predetermined references, which can be points or axis, aiming at having a common framework where to exchange information without ambiguity. This implicitly needs the definition of a reference frame that determines the points and the axis to be used as the common framework. This reference frame is defined by what is called a datum,⁷¹ which is a set of parameters that serve as a reference or basis.

Consequently, the information concerning the location of the target can be provided in any of the following reference frames:

- Local.
- Global.
- Common.

Depending on whether the source of the information is a sensor, first two cases, or a data fusion centre, last two ones. All these reference frames are presented hereafter.

2.1.4.1.1 Local reference frames

Each surveillance sensor reports the position and, in some cases, the velocity of the target by means of a measurement that is expressed in a certain coordinate system, which is generally associated to a local reference frame. The origin of such frame is defined on the centre of the sensor body or on the previous target position. The axes point in different directions generally complying with some predetermined rule (such as the right-hand rule).

⁷⁰ This point is a “reference” point within the mobile to be tracked. In the case of an aircraft, there are several points which can be used, namely: the nose of the aircraft, the centroid, the middle point of its longitudinal axis, etc.

⁷¹ The datum as specified by ISO in [ISO 19111] defines the position of the origin, the scale and the orientation of the axes of a coordinate system.

One extended example is the Local-Level Frame (LLF), which is also known as local geodetic or navigation frame. Depending on where the axes point, there are two extended versions:

- The East-North-Up (ENU) frame, where the y -axis points to the true north, the x -axis to the east and the z -axis points up, perpendicular to the reference ellipsoid.
- The North-East-Down (NED) frame is similar to the ENU frame except for the fact the x -axis and y -axis are interchanged and so, the z -axis points down (also perpendicular to the reference ellipsoid) instead of pointing up as in the ENU frame.

2.1.4.1.2 Global reference frames

On the other hand, surveillance sensors can also determine⁷² the location of a target on a terrestrial or global reference frame. This is the case when reference frames related to the Earth are taken into consideration. These frames have their origin fixed in the centre of mass of the Earth and their axes point in different directions depending on the chosen frame. This kind of reference frames allows a seamless exchange of surveillance information in the case that all the affected actors use the same global frame.

Examples of such a reference frames are the Earth-Centred Inertial (ECI) frame, the Earth-Centred Earth-Fixed (ECEF) frame and any of the ellipsoidal reference frames such as the ED50, the ETRS89, the WGS72 or WGS84, among many others.

Therefore, the Earth-Centred Inertial frame defines the z -axis pointing up towards the Conventional Terrestrial Pole (CTP) along the Earth rotational axis, the x -axis is defined over the equatorial plane pointing towards the vernal equinox, and finally, the y -axis is defined following the right-hand rule.

Another example of global reference frame is the Earth-Centred Earth-Fixed (ECEF) frame or Conventional Terrestrial System (CTS). In this case, the z -axis also points in the direction of the CTP, the x -axis is aligned to the intersection of the equatorial plane with the plane defined by the reference meridian (Greenwich meridian) and the y -axis complies with the right-hand rule.

Finally, the most common one is the geodetic reference frame based on defining the position of a point by means of providing its latitude, longitude, and height with respect to a reference ellipsoid.⁷³ These frames are defined by the zero meridian and the equator plane. The former is any arbitrary reference plane that contains the rotational axis; usually fixed as the Greenwich meridian. The latter is a plane perpendicular to the rotational axis of the Earth, which passes through the Earth's centre of mass. Lastly, the height is defined as the normal to the surface of the reference ellipsoid.

2.1.4.1.3 Common reference frames

The last set of reference frames is the one intended for fusing all the information coming from different sensors and so, representing the common framework where all the

⁷² The location can also be obtained by means of a transformation from a local reference frame to a global one.

⁷³ Also called ellipsoidal reference frame.

information is put together in order to be processed in a homogeneous way. Depending on the kind of Earth model that has been assumed, it might imply a projection or not. So, the common reference frame is defined as a Cartesian coordinate system centred at the data fusion centre location.

2.1.4.2 Coordinate systems

A coordinate system allows specifying the position of a point of interest in a planar, spherical, or multidimensional surface by means of set of numbers called coordinates. These coordinates, which are represented as components of a vector (each one referring to one dimension of the coordinate system), define the position of the point uniquely once the reference points and axes are defined by a reference frame.

The coordinate system is usually chosen trying to simplify some calculations or to ease the representation of the location information. Hence, one can find the Cartesian, the polar, the spherical or the ellipsoidal coordinate systems. Each of them has its own coordinates for representing the position of a point in the plane or in the space. Thus, one can talk, respectively, about Cartesian coordinates (x, y, z) , polar coordinates (ρ, θ) , spherical coordinates (ρ, θ, φ) , or geodetic coordinates (ϕ, λ, h) , among others.

Consequently, the sensor measurement is then a set of coordinates, which are related to the chosen coordinate system. Furthermore, the local reference frame admits the possibility of using one or several coordinate systems. However, the fundamental matter is that there is the possibility to convert the coordinates of one system into the coordinates of the others⁷⁴ and vice versa. Depending on the transformation type, it is possible to introduce some errors, so this issue must be handled with care.

2.1.4.3 Coordinate transformations

The measurements that are sent by the sensors and the internal data that are used by the data fusion centre do not usually share neither the coordinate system nor the reference frame, so some kind of transformation is needed in order to have compatible data. There are two main types of transformations, namely:

- The coordinate conversions, i.e. translating the coordinates of a point of interest, on a certain coordinate system, into the coordinates on another coordinate system within the same reference frame or between different reference frames.
- The coordinate projections, which represent coordinates that correspond to points on a geographic coordinate system that is based on a sphere or spheroid, in a 2D plane.

When moving from the local or global reference frame, set by the sensor, to the common one, set by the data fusion centre; one or several coordinate conversions and one or several projections might be performed. These transformations are relevant for performance evaluation tools as in certain situations they represent a potential source of error, which can be included in the sensor measurement models. This fact will be shown in a per sensor family in the following sections.

⁷⁴ This is what is called “coordinate transformation”.

2.1.4.3.1 The stereographic projection

The most common projection is the stereographic one, although there have been others which have been used in the ATC domain such as the Mercator projection or the Lambert conical one. The stereographic projection is the preferred one due to its properties, although its heavier computational load, compared to other projections, has prevented it from being more extensively used in the past. However, nowadays this is not a constraint anymore, so its use has extended considerably.

Going into a bit of detail, the stereographic projection is an azimuthal projection as it uses a plane as the projection surface where the projection point is located opposite to the projection centre. It is considered the only known true perspective projection of any kind that is also conformal in its spherical form that is when the Earth is assumed as a sphere [Snyder 1987].

Some of its features are that directions from the projection point are true, scale increases away from the projection centre and the projection point cannot be plotted as it is located at the infinity. All these features make this projection very suitable for its use in the ATC domain [Paradowski 1998].

Accordingly, the stereographic projection in its oblique aspect⁷⁵ is currently the standard projection for ATC applications. Normally, the positions of the ATC centre or the Airport Reference Point (ARP), in the case of the Airport, are the ones chosen as the projection centres.

2.1.4.3.2 Considerations on coordinate transformations

When using a common reference frame, there are several implications that need to be considered:

- The first one is that as it is well known that the Earth is not a perfect sphere, so it has to be approximated by an ellipsoid.⁷⁶ Thus, there are several ellipsoids of reference that approximate the shape of the Earth (some examples: Struve 1860, Hayford 1924, or the ones included in WGS72 or WGS84). The one that is currently in use in the aeronautical domain is the one defined in WGS84,⁷⁷ which is the same one as that used by the GPS system.
- The second one is that movement equations are normally simplified if expressed in Cartesian coordinates, so almost all fusion systems transform geodetic coordinates (latitude, longitude and height) into Cartesian ones using different projection algorithms before making any calculation.

As it will be presented, transforming from one coordinate system and reference frame to another set could be a step wise process, which makes use of intermediate coordinate transformations and reference frames to achieve the final coordinates and the final reference frame.

⁷⁵ Because the center of the projection is neither any of the poles nor any point over the Equator.

⁷⁶ An ellipsoid is defined by ISO as the surface formed by the rotation of an ellipse about a main axis.

⁷⁷ The WGS84 is a terrestrial reference frame with the following features: it is a geocentric Cartesian coordinate system whose origin is in the centre of mass of the Earth; its z-axis point north towards the mean pole as defined in Conventional International Origin (CIO)

2.1.4.4 Earth models

Another significant aspect to be considered is that the Earth can be approximated by different types of models. An Earth model is needed to convert measurements made on the curved Earth to maps or data storage means, such as files or databases, which are common when using performance evaluation tools. The election of such a model could ease or make more difficult the coordinate transformation process.

In the first bullet of section 2.1.4.3.2, it is commented that the Earth is usually modelled as an ellipsoid, and so there are different reference ellipsoids that can be used in order to provide the geodetic coordinates. These ellipsoids are characterized by several parameters such as their major semi-axis (a), their minor semi-axis (b), their eccentricity (e), their flattening factor (f), etc.

When the precision of the computations allows it, the Earth model can be relaxed and instead of using a reference ellipsoid, the model is simplified to a reference sphere with a radius whose value depends on the latitude (ϕ) where the model is to be used. This radius is taken to be the local best radius, whose expression is:

$$R_T = a \left(1 - \frac{1}{2} e^2 \cos 2\phi \right) \quad (2.1)$$

Where:

- a : semi-major axis.
- e : first eccentricity.
- ϕ : the latitude.

The values of these parameters are taken from the reference ellipsoid, which for the ATM domain is the WGS84. Thus, the values would be $a = 6378137.00m$ and $e = 0.0818191908426$.

Finally, it is also possible to use the flat Earth model, which assumes that the Earth shape is a plane. This model is used when the projection surface is small enough so, in the extremes, the Earth curvature barely has any effect on the measured magnitudes. This model can be used in the airport environment as the surface of the airport and its surrounding area do not suffer from the earth's curvature.

2.1.5 Surveillance data exchange

Performance evaluation systems need to be fed with data to estimate reference trajectories and thereby assess the quality of both sensors and tracking systems. These data are, in principle, the same that SDP systems receive from surveillance sensors. These data are requested to share a common data format as they are exchanged not only inside a certain air space but also between different air navigation service providers (ANSPs), which are the organizations in charge of managing other air spaces. One example of this harmonization is Europe, where the European Organization for the Safety of Air Navigation (EUROCONTROL) developed the ASTERIX data format, which is

nowadays a worldwide accepted standard for surveillance data transmission. This standard is going to be outlined in next section (ref. 2.1.5.1).

The rationale why the characteristics of this standard are presented in this document relies on the fact that it defines the precision of the measurements and the precision of the times when they were taken, apart from other data that can be of interest such as Aircraft Derived Data (ADD) also known as Downlinked Airborne Data (DAP), which may provide more advanced kinematics, the intentions of the aircraft, or flight plan information. All these data can be exploited by reconstruction algorithms to obtain the best estimate of the ideal trajectory,⁷⁸ so depending on the coding features of the ASTERIX data format, the quality of the estimate will be different.

2.1.5.1 The ASTERIX data format

The ASTERIX data format (initially: All Purpose STructured EUROCONTROL Radar Information eXchange) was conceived for surveillance data exchange among air traffic control systems during the 80's aiming at facilitating the exchange of radar data between ANSP of neighbouring countries, which was found to be a complicated issue. As a result, the need for a common European standard for radar data exchange across borders became apparent [ASTERIX history].

The problem that appeared in the 80's was that each national administration⁷⁹ had developed its own internal surveillance data format and its own transmission protocols for sending radar data to its air traffic control centres. Therefore, it was difficult to exchange this information between neighbouring countries and, as such, preventing any surveillance infrastructure rationalization and cost reduction, which is a target that it is nowadays still ongoing.

For the development of this common standard, the AUG (ASTERIX User Group) was created by the RSSP (Radar Systems Specialist Panel) and its work led to the ASTERIX's official approval in the 15th RSSP Meeting in July 1986 [ASTERIX history].

Since then, there has been a huge evolution of the standard. The first milestone was the appearance of the “ASTERIX manual” that described the initial structure of ASTERIX and defined an initial set of "data items" related to monoradar and weather data.

It was in 1991, when the ASTERIX documentation took the current layout. Thereby the “ASTERIX manual” turned into the “ASTERIX Structure Document”, which is divided in several parts. Additionally, the initial structure was improved to provide more flexibility.

From 1994 onwards, it is the SURT (Surveillance Team) who has taken over the responsibilities of the former RSSP leading the development of the ASTERIX standard. The SURT created the Surveillance Task Force for Radar Data Exchange (STFRDE) to continue the work of the ASTERIX Users Group (now renamed Surveillance Data Exchange Task Force). Nowadays, the AUG meets twice or three times a year aiming at evolving the current document in order to cover the needs of the surveillance chain.

⁷⁸ These ideal trajectories are also called “reference” trajectories all along this document.

⁷⁹ Each country has a designated agency for managing the ATM related aspects. They are usually depending on the ministry of transportation.

2.1.5.1.1 General features of the ASTERIX data format

The ASTERIX data format was known for its efficiency, flexibility and versatility while managing radar data. These features fostered the transition from the local or proprietary data formats, used by the different ATC service providers, into a common format without losing information.

From a technical point of view "ASTERIX is an application⁸⁰/presentation⁸¹ protocol responsible for data definition and data assembly developed to support surveillance data transmission and exchanges" as defined in [ASTERIX Part 1]. This definition refers to the application and presentation layers of the International Standards Organization (ISO)/Open Systems Interconnection (OSI) Reference Model [OSI model].

The ASTERIX data format does not specify any of the lower telecommunication support layers from the session layer down to the physical one. This task is left to the partners of the data exchange. The standard just recommends that it is advisable to apply standard telecommunication protocols such as X-25 or TCP/IP [ASTERIX Part 1].

The "EUROCONTROL Standard Document for Surveillance Data Exchange Part 1" [ASTERIX Part 1] states that: "The ASTERIX Standard, as a Presentation protocol, defines the structure of the data to be exchanged over the communication medium, from the encoding of every bit of information up to the organisation of the data within a block of data".

Figure 2.1: ASTERIX in the OSI Reference Model. Source: own elaboration

⁸⁰ As an application protocol, it defines the core of the commonly used surveillance related data easing the communication between heterogeneous applications.

⁸¹ The presentation layer is mainly in charge of providing a common representation of the data transferred between application-entities and preserving them during the transfer [OSI model].

2.1.5.2 The ASTERIX logical model

2.1.5.2.1 The data categories

Due to the widespread use of the ASTERIX standard in the surveillance domain, there is a need for classifying the information it carries. This task is assumed by the different “categories” that have been defined in ASTERIX. This classification is standardized in order to keep informed all users about the exchanged information.

The main purposes of the “categories”, as stated in [ASTERIX Part 1], are:

- Identification of the data exchanged.
- Appropriate delivery to the target application.
- Establishment of a certain hierarchy in the transmitted data.

There are 256 categories, which are divided into three groups. The first one (i.e. categories from 000 to 127) is devoted to standard civil and military applications, the second one (i.e. categories from 128 to 240) is reserved for special civil and military applications, and the last group (categories from 241 to 255) corresponds to both civil and military non-standard applications. The categories of the first group are under the responsibility of the SuRveillance Data Exchange - Task Force (RDE-TF), with endorsement by the Surveillance Team (SURT), which is in charge of assigning them. For more information about the way in which these categories are managed visit [ASTERIX Part 1] Section 7.

2.1.5.2.2 The data items

Focusing on the information sent by each category, there are several concepts that need to be explained:

- The “**Data Item**”, which is the smallest piece of information that can be transmitted by each category. Some examples are the Mode 3A, the Mode C, the longitude, the latitude, or the height of an aircraft. Each data item has its own and unique reference, which unambiguously identifies it, among others. It is of the form Innn/AAA, where "nnn" specifies the data category and "AAA" a three-digit decimal number that identifies the data item, e.g. I010/060, which is a data item of category 010 that contains the Mode-3/A Code in octal representation.
- The “**Catalogue of Data Items**”, which is a list of all the data items that can be part of a certain category. For each category, there is just one standardized catalogue of data items which allows the applications involved in the exchange of information to use the correct data items while communicating. The catalogue also includes the characteristics of each data item such as their reference, definition, format, structure, size, and units (where applicable).

Next figure describes how the ASTERIX surveillance standard is logically organized:

Figure 2.2: ASTERIX logical model. Source: [ASTERIX Part 1]

The next step aims at converting this logical model into a physical one that will be the base for the communication exchange.

2.1.5.2.3 The user application profile

Once the logical model is set, there is a need for a kind of translation table that provides where the logical information is placed in the physical structure that is going to be sent through the communication channel. This element is the User Application Profile (UAP), which is defined as "The mechanism for assigning Data Items to Data Fields, and containing all necessary information which needs to be standardized for the successful encoding and decoding of the messages" [ASTERIX Part 1].

The UAP needs to be known by those programs which attempt to assemble/disassemble the exchanged messages as it informs where the data items are mapped into the data fields that have been transmitted. The standard says that the UAP must be unique per category.

The existence of this mapping table allows optimizing the transmission efficiency without program modifications as it permits to change the physical representation of a certain Data Item if more precision is required or to adapt the delivery frequency of a Data Item depending on its update rate.

2.1.5.3 The ASTERIX physical model

In this section, the physical model that is used for transmitting ASTERIX data is going to be outlined. The objective is to analyse how the data is organized for its transmission through the communication medium as it is important for the following processing. This organization is structured in several levels which go from the minimum piece of transmitted information to the minimum structure for sending the application data.

2.1.5.3.1 The data field

Data fields are the physical implementation of data items. They are allocated to the latter following the indications given by the UAP, which define the correspondence between them for a given application. Thus, each data field is associated to one and only one data item, as defined by the UAP.

The data field is then the smallest piece of transmitted information although it cannot be sent in isolation but as part of a biggest entity, the “Data Block”. Data fields are referenced by a “Field Reference Number” (FRN) and are composed of an integer number of octets. It is important to note that depending on the characteristics of the corresponding data field the following processing can vary, e.g. the quantization noise can be different.

There are several types of data fields whose length can be fixed or variable, namely:

- Fixed length data fields.
- Extended length data fields.
- Explicit length data fields.
- Repetitive data fields.
- Compound data fields.

The particularities of each type of data field can be found in [ASTERIX Part 1]. Additionally, there are two special types of data field:

- **Non-Standard data fields** are a mean for exchanging non-standard information without affecting non-interested users. It is announced by a Special Purpose indicator (SP-bit) in the FSPEC field of a record (see next section).
- **Reserved Expansion data fields**, like the previous ones, are meant to introduce intermediate changes to a given category without affecting other users. In each FSPEC field, there is at least one Reserved Expansion Indicator (RE-bit). When this need appears, this kind of data fields is described in a separate document.

Thus, this variety of data fields provides the standard with the appropriate flexibility and versatility while trying to accommodate all the different kinds of surveillance data. This fact has been the key to its expansion as a world-wide standard.

2.1.5.3.2 The record

A record groups a collection of data fields corresponding to the same category, all of them preceded by an additional field, the Field Specification (FSPEC) field, which states the presence or absence of the various data fields that are feasible taking into account the information in the corresponding UAP.

The FSPEC is a kind of table of contents where each bit represents a certain data field. In the case a bit is set to one, the data field is present, otherwise absent. Thus, the number of data fields is variable depending on the content of the FSPEC field and it is always a multiple of an octet.

The record provides information to a certain application and it is mandatory that it declares the source of the information to be processed.

2.1.5.3.3 The data blocks

Attending to the “General Message Structure” defined in [ASTERIX Part 1]: "The application data to be transmitted over the communication medium shall consist of either one or a concatenation of consecutive Data Blocks". So, it is the minimum structure to be sent through the communication medium that is expected by the target application.

A data block consists of the following elements:

- One “Data Category” (CAT) field, which is a one-octet field that indicates the data category of the transmitted data.
- One “Length Indicator” (LEN) field, which is a two-octet field that provides the total length in octets of the data block including CAT and LEN fields.
- A set of records with at least one.

Since the length of all records is variable and a multiple of an octet, the data block length is also variable and a multiple of an octet. The standard states that this length should be mutually agreed between both data sources and users [ASTERIX Part 1].

2.1.5.3.4 Graphical representation of the physical model

Figure 2.3: ASTERIX physical model. Source: own elaboration

2.1.5.3.5 Known limitations

The ASTERIX data format is a worldwide de-facto standard. However, it has some known limitations, such as:

- The absence of the category version among the transmitted data generates some issues while interpreting the information that is transmitted. This happens when the source of the information is not under the direct control of the data consumer.
- There are some data items in which there is some uncertainty concerning the meaning of a 0 value, that it, is it 0 or it is not available.
- The policy based on minimizing the transmitted information prevents sending empty extensions in extended length data items. As such, there might be downstream systems that may not receive certain information because it is located in an extension after the one that becomes all zeros. This is the case of the third extension of the I062/080 (Track Status) when the track has full sensor support and consequently it is not sent.

2.1.5.4 Other surveillance data formats

There are other surveillance data formats which are still on use, but their presence is being progressively substituted by ASTERIX categories, so they are not going to be described in this document.

Some examples are the Alenia format, CD2, DDE, Eurocontrol, PDS, FPS117, RAT31DL or RAT31SL, PR800, RDIF, TPR1000, RSRP, TRS22, Aircat, Madrec, etc.

2.1.6 Surveillance data transmission

2.1.6.1 Introduction

Communication is one of the domains that currently builds the CNS/ATM concept [ICAO 9750]. It is composed of stations that communicate among each other, a transmission mean and a communication protocol. The information exchanged include both data and voice, although nowadays the trend is to just exchange data converting voice into data and then using VoIP protocols.

The stations can communicate with the others through cables and radio communication such as mobile communications or broadcasting. These stations can be located over the Earth, or in the air in case of satellites or airborne stations. There are currently several initiatives that try to incorporate satellites into the ATM communications, such as the already well known ADS-C, when using satellite-based communications,⁸² the one offered by Aireon⁸³ which is a global space-based ADS-B, or the more recent one “Space Based CNS” to be developed by Indra, which covers the full CNS range.

The communication between stations is possible because there are communication networks and protocols which allow the transmission and correct understanding of a

⁸² ADS-C can also be provided based on VHF communications.

⁸³ For further details, please visit: <http://www.aireon.com/Home>

message sent from one station to others. The most common ones are presented hereafter as they are the ones in charge of delivering usable data.

2.1.6.2 Communication protocols

Surveillance data need to reach the ATC facilities to be exploited by SDP systems, CWP or others. This implies that these data need to be routed from the generating source to the final destination, which is the ATC facility, where the information will be received, decoded and processed before being forwarded to other system, now converted into tracks. The communication protocols [Comer 13] guarantee that this information is received in the destination in proper conditions.

Currently, the former communication protocols (e.g. X-25), dedicated or proprietary ones, have been substituted by internet protocols such as TCP/IP, UDP multicast, UDP unicast, UDP broadcast or PDC. This matter introduces the same drawbacks that can be found in the communications over internet (e.g. message drop, variable latency or data corruption) in the ATM world.

Other protocols which are also involved in the communications and management of the involved systems are the Network Time Protocol (NTP) and the Simple Network Management Protocol (SNMP). The former is in charge of keeping all the systems in the network synchronized by means of sending a time reference. The latter is the most extended protocol for remotely managing devices on IP networks.

2.1.6.3 Communication networks

Surveillance information is sent from sensors, which are deployed all around the area to be controlled, to the ATC facilities. This is done through communication networks [Tanenbaum 11] usually referred as Wide Area Networks (WAN) until they reach the ATC facilities. The local distribution of surveillance data is then performed by means of Local Area Networks (LAN) which could be, for example, Ethernet⁸⁴ or FDDI.⁸⁵

Although FDDI networks are considered as a LAN, due to their capacity they are also used as the backbone of a WAN. Depending on the location of the sensors, their connection to the ATC facilities can also be through dedicated lines which usually connect them to nodes that belong to a WAN. An example of such a node is the EUROCONTROL product RMCDE (Surveillance Message Conversion and Distribution Equipment) which is currently being replaced by the SDDS (Surveillance Data Distribution System) product. On the other hand, an example of a WAN is the Surveillance Data Network which was called RADNET (now renamed as SURNET) or the AFTN (Aeronautical Fixed Telecommunication Network), which is being superseded by the ATN (Aeronautical Telecommunication Network) that integrates different technologies and subnetworks.

Nowadays, the main problem that communication networks introduce is the variable latency. This is a very common issue when surveillance information crosses packet-oriented communication networks. This issue will impact the timestamp of the received information in case it is not done either on the target or by the surveillance sensor.

⁸⁴ Ethernet (i.e. IEEE 802.3) is a well-known example of a physical layer protocol for a LAN network.

⁸⁵ FDDI stands for Fiber Distributed Data Interface. It is data transmission standard intended to LAN networks that use fiber optics as the physical medium. Nowadays, it is a deprecated standard.

2.1.6.3.1 Network delay model

Communication delays⁸⁶ are random in nature, and so, it is important to model them. This is always a complex task as there is no one model which applies to all network delays⁸⁷ as there are plenty of networks with different components, protocols, and topologies. Therefore, one can find a few statistical models for characterizing them.

In our case, it is significant to be aware of the effect of a highly variable network delay as it might affect the surveillance data received at the processing centres. This is a relevant drawback if the incoming data, which come from distinct sensors and so cross different networks, are going to be used by an online performance assessment tool or by a tracking system.

The problem of a disparate delay, among data sources connected through different data networks, impacts on the overall data processing time that is needed for obtaining the output of the online processing system. Many of such systems estimate this delay in order to define the time to be waited before the data processing can start. This is usually done by implementing buffers where the data is stored until it is considered that most of the data should have arrived and only then the processing starts. This implies a trade-off between the percentage of the discarded “late” data and the delay that is introduced in the overall data processing. Consequently, the time that the data is buffered should be optimized to maximize the amount of data that is processed while minimizing the impact on the overall data processing. These buffers can be static, which means that all the data will suffer the same delay, or they can be dynamic in case they consider the network delay as part of the time the data is going to be buffered. The latter is done by taken into consideration the “time of applicability” of the data and not the reception time at the processing system. Linked to this, there is the need for processing time-ordered data that some systems, such as multisensor data fusion ones, impose. This is to avoid the appearance of delayed data also known as “late plots” [ARTAS SRS], as they implies either retracing all the processed data after the delayed one, incorporating their impact in the current track state, or throwing these data away. Depending on the situation, one option can be preferred compared to the others.

There are cases in which the transmission delay may affect the data processing. This happens when the surveillance data message does not incorporate the time of applicability of the information it provides, and so, it has to be the receiving ground station the one that timestamps them.⁸⁸ However, the delay introduced by the transmission mean is far from being as important as if there were a network between the source and the destination. If this is the case, the strategy will be to try to estimate the time bias associated to the measurement assuming that there is a random component over it, which takes into account the variability on the following components: the processing previous to the signal

⁸⁶ The term “communication delays” groups several delays which appear in the process that follow the data since they are generated until they are consumed at the intended destination. This delay includes the delay in generating the data in the source, the time to prepare the data to be sent through the communication mean, the transmission delay, the network delay and the time needed for processing the data at the final destination.

⁸⁷ The network delay is the component among the ones that form the global “communication delay” that introduces more variance to the final value. This is the reason why it is the main focus of this section.

⁸⁸ This fact implies that the time which is available is not the “time of applicability” of such information but the “reception time” at the ground station, which might be considerably different from the former. This action is also a consequence of the requirement that is imposed by the ASTERIX data format concerning the need for having a time reference associated to every ASTERIX message.

transmission, the transmission delay, the network delay, and the data processing at the receiving station.

Hence, communication delays but specially network delays are an important drawback for online systems, such as SDP systems or online performance evaluation tools. However, for offline performance evaluation tools, it is not that critical as they generally use the time of applicability of the data for their processing no matter what has been the reception time, as the data are already stored in a file or a database and so the system does not have to wait for them. As it will be mentioned later, this statement is valid for all sensors but those ADS-B stations that provide the reception time of the ADS-B message and not the time of applicability of its information.

2.1.6.4 Other ways of exchanging surveillance data

Apart from the data directly coming from communication networks, there are other ways of exchanging such information. This fact is important to be mentioned here as the main source of surveillance information that is processed by performance evaluation tools is through the exchange of surveillance data files. Thus, the surveillance data is then imported into such systems, which will process them afterwards.

These files with surveillance data use specific file format such as the one provided by sniffer programs such as Wireshark⁸⁹ or TCPdump; or more surveillance-oriented formats such as the FINAL format.

2.2 Radar systems

2.2.1 General considerations

2.2.1.1 Overall features

The term RADAR (Radio Detection And Ranging) covers a family of sensors which can be divided into two: Primary Radars (PSR) and Secondary Surveillance Radars (SSR). The former includes a special version for the airport domain which is called the Surface Movement Radar (SMR) in Europe or the Airport Surface Detection Equipment (ASDE) in the EEUU. The SSR includes the Conventional SSR radars but also the Mode-S ones. The main difference between them is the ability of the latter for performing selective interrogations and offering a data link for communication purposes.

In this section, a generic measurement model is going to be presented for each radar system that is of importance for performance evaluation tools focussed on the airport domain.

2.2.1.2 Generic measurement model

There is a commonly agreed generic measurement model for radar systems that includes scale and offset biases, and random errors. These error components are usually presented in slant range (ρ) and azimuth (θ) coordinates (i.e. polar coordinates or spherical ones in

⁸⁹ Wireshark is the evolution of the former network protocol analyser Ethereal.

case the elevation angle (φ) is available) due to the measurement process performed by a radar sensor, which is what determines the radar local reference frame.

Concerning biases, this kind of sensors suffer from both sensor-based and aircraft-based biases. The latter is only valid for cooperative radars (i.e. SSR) as they rely on the transponder that is onboard the aircraft. This airborne equipment has a tolerance on the response delay that induces a time bias, which can be further converted into a positional error; see section 2.2.4.2 for further details. Another aircraft-based bias is the one caused by the different location of the ATC antenna with respect to the aircraft reference point.

If the focus is placed on sensor-based biases, there is the offset bias caused by a bad calibration of the measurement process, a scale bias due to the varying propagation conditions faced by the signal all along its path,⁹⁰ and the radar eccentricity due to a difference in the plane where the radar is located and the theoretical plane where it provides its measurements due to a loss of verticality.

This generic model, which is presented in, for example, [Lin 05], corresponds to a multisensor environment mainly focussed on radar systems. Therefore, the range⁹¹ (ρ^m) and the azimuth (θ^m) measurements performed by the i -th radar at time k can be modelled as:

$$\rho_i^m(k) = [1 + \epsilon_i^\rho(k)]\rho_i(k) + b_i^\rho(k) + w_i^\rho(k) \quad (2.2)$$

$$\theta_i^m(k) = [1 + \epsilon_i^\theta(k)]\theta_i(k) + b_i^\theta(k) + w_i^\theta(k) \quad (2.3)$$

Where:

- $\rho_i(k)$, $\theta_i(k)$: true range (ρ) and azimuth (θ) values corresponding to radar i at time k . They are usually called the ideal range and azimuth, respectively.
- $\epsilon_i^\rho(k)$, $\epsilon_i^\theta(k)$: scale biases for the range and azimuth corresponding to radar i at time k . In other references, the scale bias associated to the range (i.e. $\epsilon_i^\rho(k)$) is called the radar gain.
- $b_i^\rho(k)$, $b_i^\theta(k)$: offset biases for the range and the azimuth corresponding to radar i at time k . They are generally known as the range and azimuth biases, respectively.
- $w_i^\rho(k)$, $w_i^\theta(k)$: measurement noises for the range and the azimuth corresponding to radar i at time k . Usually, they are referred to as the range and azimuth noises, respectively.

⁹⁰ The propagation through the atmosphere is not performed following a straight line but describing a curve due to the not constant speed of the electromagnetic waves while traversing all the different layers of the atmosphere.

⁹¹ Truly speaking, the measurement provided by a radar sensor, unless corrected with the height of the target compared with the height of the radar, is the slant range. For the sake of simplicity, it has been preferred to talk about just "range" because taking into consideration the difference of heights between the radar and the target, and the distances that can be found in an airport, its effect should be negligible except when taking off or landing. In such case, it must be analysed.

Unless other thing is said, $w_i^\rho(k)$ and $w_i^\theta(k)$ are modelled as zero-mean white Gaussian random sequences with corresponding variances σ_ρ^2 and σ_θ^2 , which are assumed mutually independent of each other. The covariance matrix of the measurement error vector formed by both components looks like:

$$R = \begin{bmatrix} \sigma_\rho^2 & 0 \\ 0 & \sigma_\theta^2 \end{bmatrix} \quad (2.4)$$

Normally, biases are considered constant along time, so the dependence with time can be removed (i.e. ϵ_i^ρ , ϵ_i^θ , b_i^ρ and b_i^θ). However, it has to be noted that some of them are surely changing in time due to their dependence on RF propagation conditions (i.e. the scale bias), the degradation process of some of the components, or as a result of a periodical self-contained calibration mean.

A more general model can also include some of the aircraft-based biases such as the response delay or the antenna position; or more specific sensor-based biases such as the radar's eccentricity.⁹² Thus, the previous model can be extended to cope with these three new sources of error:

$$\rho_i^m(k) = [1 + \epsilon_i^\rho(k)]\rho_i(k) + b_i^\rho(k) + \Delta R_j + \Delta l_j^\rho + w_i^\rho(k) \quad (2.5)$$

$$\theta_i^m(k) = [1 + \epsilon_i^\theta(k)]\theta_i(k) + b_i^\theta(k) + \Delta\theta_i^{e1}\sin\theta_i(k) + \Delta\theta_i^{e2}\cos\theta_i(k) + \Delta l_j^\theta + w_i^\theta(k) \quad (2.6)$$

Where:

- ΔR_j : transponder response delay induced bias for the j -th aircraft, which is an offset bias consisting in the conversion into range of the tolerance in the transponder response delay.
- $\Delta l_j^\rho, \Delta l_j^\theta$: offset biases that depend on the ATC antenna position for aircraft j .
- $\Delta\theta_i^{e1}, \Delta\theta_i^{e2}$: values that characterize the radar's azimuth eccentricity,⁹³ related to maximum eccentricity and angle of maximum eccentricity.

It is crucial to bear in mind that the model has to be as accurate as possible because the objective to achieve, when tracking aircraft and vehicles in an airport, is quite demanding, that is, less than 7.5 m of error in horizontal position for more than the 95% of time [ICAO 9830].

Like other sensors, radars can suffer a potential time bias due to a different time reference at the fusion centre and at the sensors. If this is the case, the plots, which are timestamped at the sensor, will correspond to a time in the future or in the past with respect to the actual time, which is assumed to be defined by the clock at the data fusion centre. Nowadays, this synchronization problem is highly mitigated due to the use of GNSS clocks.

⁹² The radar eccentricity will be described in section 2.2.4.2 when taking about the particularities of secondary radars.

⁹³ For the range in which this performance evaluation system is going to work, this effect is negligible.

On the other hand, a similar situation appears when the information coming from the sensor arrives without timestamp. In such situation, it is the ground station the one in charge of timestamping the data. Consequently, the time bias that is introduced is related to the time spent previous to sending the data through the transmission mean, the transmission delay, and the time needed to process the received data at the ground station.

In case this information crosses a data network, then a network delay will appear such as the one described in section 2.1.6.3.1. This misalignment in the time reference leads to an equivalent position bias aligned with the target velocity. As a consequence, the more convenient local reference frame for dealing with this error is the one based on Cartesian coordinates. Thus, (X, Y) projected measurements⁹⁴ will suffer an additional bias of the form:

$$X_i^m(k) = X_i(k) - V_j^x(k)\Delta t_i + w_i^x(k) \quad (2.7)$$

$$Y_i^m(k) = Y_i(k) - V_j^y(k)\Delta t_i + w_i^y(k) \quad (2.8)$$

Where:

- $X_i(k), Y_i(k)$: true stereographic projection of the target location provided by radar i at time k .
- $V_j^x(k), V_j^y(k)$: velocity vector components of the j -th target at time k .
- Δt_i : time bias for the i -th radar, which is considered constant along time.
- $w_i^x(k), w_i^y(k)$: measurement noise errors for the x and y components corresponding to radar i at time k .

The time bias (Δt_i) for the i -th radar can be defined as:

$$\Delta t_i = k - k_i \quad (2.9)$$

Where:

- k : time of the target report that would have been measured using the time reference at the data fusion centre.
- k_i : time measured using the time reference at the i -th radar sensor.

If $\Delta t_i > 0$, it implies that the target is located forward in time and $\Delta t_i < 0$ means that the target is placed backwards in time.

As already pointed out in the beginning of this section, in the following subsections the general model is going to be tuned for each of the radar sensors which can be available at the airport and its approach zone.

⁹⁴ Currently, stereographic projections are frequently found the ATC environment using a point in the airport surface as the center of the projection. This point is normally known as the airport reference point (ARP).

2.2.1.3 Data format

On one hand, PSR, conventional SSR, MSSR and Mode-S radars mainly provide their output and their status in ASTERIX Cat.048 (e.g. plots, target reports or local track updates) and ASTERIX Cat.034 (i.e. sensor status and service messages), respectively. Additionally, there are other previous ASTERIX categories which have also been used for such purpose like ASTERIX Cat.001 and Cat.002, which have not been considered in this text as they have been superseded by the former ones.

On the other hand, SMR or ASDE radars provide target reports and service messages in ASTERIX Cat.010, like other sensors such as ADS-B, multilateration systems or magnetic loops.

In order to evaluate any radar system, it is interesting to know the features of the information that is going to be processed, which includes the format in which it will be provided, its frequency and any previous data processing performed at the sensor site or previous to reaching the performance evaluation tool. It is relevant to point out that the focus is placed on non-processed data because any previous processing is usually unknown and so will be the statistical behaviour of such data. Therefore, unless this processing is well stated, the strategy is to discard this information, although it can be exploited for other purposes as it is commented in section 6.10, and just keep the raw one, that is, the one directly measured by the sensor.

Thus, the most important data items are the ones related to identification (for association purposes), the time of applicability of the surveillance data or their reception time,⁹⁵ and the ones related to position and velocity (kinematic information). Some other bits of information are also of interest, such as the quality of the information provided, which usually comes in the form of covariance matrices, standard deviations, or variances; and so, they will be presented here after.

In the process of reconstructing the reference trajectory, one of the most important steps, as it will be commented, is the data association (see section 6.7). This process groups all the target reports, from all possible sources, that are believed to come from the same target. Such process is eased if the target reports provide the identification of the reported target. Consequently, the Mode A, the ICAO 24-bit address, any local track number,⁹⁶ or the callsign/flight ID are pieces of information that conveniently used help while identifying which plots, target reports or monosensor track updates belong to which specific trajectory (i.e. target). Obviously, all this information must be handled with care as the sources of information may not be working properly. This is the reason why the process is divided into several steps and why some additional coherency checks, mainly based on time and position, are implemented.

⁹⁵ In case no time of applicability is provided by the sensor.

⁹⁶ It can be generated by the surveillance sensor itself or any other data tracking system.

Coming back to the surveillance data provided using ASTERIX Cat.010 and 048, the identification information and the data items of interest are summarized below:

Table 2: Identification information within ASTERIX Cat.010 and 048

Identification information	Data Item	Description
Mode 3A code	I010/060	<i>Mode-3/A Code in Octal Representation</i> Mode 3A code converted into octal representation.
	I048/070	<i>Mode-3/A Code in Octal Representation</i> Mode 3A code converted into octal representation.
ICAO 24-bit address	I010/220	<i>Target Address</i> Target address (24-bits address) assigned uniquely to each target.
	I048/220	<i>Aircraft Address</i> Aircraft address (24-bits Mode S address) assigned uniquely to each aircraft.
Local track number	I010/161	<i>Track Number</i> An integer value representing a unique reference to a track record within a particular track file.
	I048/161	<i>Track Number</i> An integer value representing a unique reference to a track record within a particular track file.
Callsign	I010/245	<i>Target Identification</i> Target (aircraft or vehicle) identification in 8 characters.
	I048/240	<i>Aircraft Identification</i> Aircraft identification (in 8 characters) obtained from an aircraft equipped with a Mode S transponder when flight plan is available or the registration marking when no flight plan is available.

Concerning the timestamp, it must be consistent with the data information provided by the target reports, mainly regarding to position. The ASTERIX data format requires that each target report has its corresponding timestamp and the time reference to be used is the co-ordinated Universal Time (UTC).⁹⁷ Thus, the data items concerning time that are going to be taken into consideration and their units are:

⁹⁷ The use of UTC is required by ICAO in its Annex 5.

Table 3: Time information within ASTERIX Cat.010 and 048

Time information	Data Item	Description
Time of applicability	I010/140	<i>Time of Day</i> Absolute time stamping expressed as Co-ordinated Universal Time (UTC). LSB: 1/128s
	I048/140	<i>Time of Day</i> Absolute time stamping expressed as UTC. LSB: 1/128s

This value represents the time instant when the target has been detected by the radar. It provides the number of 1/128 of a second elapsed since last midnight. Every midnight the counter is set to 0.

Finally, kinematic information is provided in several data items. It includes position information and sometimes velocity. This information is normally provided in the radar local reference plane, so it implies that a certain coordinate transformation needs to be applied when fusing it with the one provided by other sensors. The position information and the data items, where it is provided, are presented in the following table:

Table 4: Target kinematic information within ASTERIX Cat.010 and 048

Target information	Data Item	Description
2D position		
Position in polar coordinates	I010/040	<i>Measured Position in Polar Co-ordinates</i> Measured position of a target in local polar coordinates. LSB Range (RHO): 1m LSB Azimuth (THETA): $360/2^{16}$ degrees
	I048/040	<i>Measured Position in Polar Co-ordinates</i> Measured position of an aircraft in local polar coordinates. LSB Range (RHO): 1/256 NM LSB Azimuth (THETA): $360/2^{16}$ degrees
Position in geodetic coordinates	I010/041	<i>Position in WGS-84 Co-ordinates</i> Position of a target in WGS84 coordinates. LSB Latitude: $180/2^{31}$ degrees LSB Longitude: $180/2^{31}$ degrees

Target information	Data Item	Description
Position in Cartesian coordinates	I010/042	<i>Position in Cartesian Co-ordinates</i> Position of a target in Cartesian coordinates, in two's complement form. LSB X-component: 1m LSB Y-component: 1m
	I048/042	<i>Calculated Position in Cartesian Co-ordinates</i> Calculated position of an aircraft in Cartesian coordinates. LSB X-component: 1/128 NM LSB Y-component: 1/128 NM
Height		
Mode C code	I010/090	<i>Flight Level in Binary Representation</i> Flight Level (Mode C / Mode S Altitude) converted into binary two's complement representation. LSB: ¼ FL, i.e. 25ft
	I048/090	<i>Flight Level in Binary Representation</i> Flight Level converted into binary representation. LSB: ¼ FL, i.e. 25ft
Height	I010/091	<i>Measured Height</i> Height above local 2D coordinate reference system (two's complement) based on direct measurements not related to barometric pressure. LSB: 6.25ft
	I048/110	<i>Height Measured by a 3D Radar</i> Height of a target as measured by a 3D radar. The height shall use mean sea level as the zero-reference level LSB: 25ft
Size and orientation		
Size and heading	I010/270	<i>Target Size & Orientation</i> Target size defined as length and width of the detected target, and orientation. The orientation gives the direction which the aircraft nose is pointing, relative to the Geographical North. LSB Length: 1m LSB Orientation: 360/128 degrees LSB Width: 1m

The position provided by means of I010/040 and I048/040 is the measured position, as contained in target reports or tracks. So, it comes in polar coordinates with the radar location acting as the centre of the local reference system, the range as the slant range, and the azimuth measured with respect to the local geographical north. This fact implies that there is a need to convert this information into the system coordinates.

On the other hand, the height of the target is only measured if the radar is 3D. This height would be provided taking into consideration that the zero-reference level is defined by the MSL (Mean Sea Level).

Concerning the orientation of the target in I010/270, the computed heading takes as its reference the geographical north at the target position.

As it is mentioned in the ASTERIX standard, it is fundamental to keep the coherence among all the values that are sent to the ATC centre. This fact is crucial when there are local tracker derived information elements. Hence, to have consistent sets of values, they all should be expressed in the same coordinate reference system. This is the case, for example, of state vector components and their associated standard deviations or covariance values. Additionally, data items I010/170 and I048/170 (i.e. track status data items) point out the type of coordinate reference system in use in the local tracker elements for the cases in which they might be used.

Table 5: Track information within ASTERIX Cat.010 and 048

Track information	Data Item	Description
Velocity		
Velocity in polar coordinates	I010/200	<i>Calculated Track Velocity in Polar Co-ordinates</i> Calculated track velocity expressed in polar coordinates. LSB Ground speed: 2^{-14} NM/s LSB Track angle: $360^\circ / 2^{16}$
	I048/200	<i>Calculated Track Velocity in Polar Co-ordinates</i> Calculated track velocity expressed in polar coordinates. LSB Calculated ground speed: 2^{-14} NM/s LSB Calculated heading: $360^\circ / 2^{16}$
Velocity in Cartesian coordinates	I010/202	<i>Calculated Track Velocity in Cartesian Co-ordinates</i> Calculated track velocity expressed in Cartesian co-ordinates, in two's complement representation. LSB V_x : 0.25m/s LSB V_y : 0.25m/s

Track information	Data Item	Description
Acceleration		
Acceleration in Cartesian coordinates	I010/210	<p><i>Calculated Acceleration</i></p> <p>Calculated Acceleration of the target, in two's complement form.</p> <p>LSB A_x: 0.25m/s²</p> <p>LSB A_y: 0.25m/s²</p>
Projection		
Plot coordinate transformation mechanism	I010/170	<p><i>Track Status</i></p> <p>Status of track. Bit-3 (TCC) from first part:</p> <p>If “0”, tracking performed in 'Sensor Plane', i.e. neither slant range correction nor projection was applied.</p> <p>If “1”, slant range correction and a suitable projection technique are used to track in a 2D reference plane, tangential to the earth model at the Sensor Site coordinates.</p>
	I048/170	<p><i>Track Status</i></p> <p>Status of monoradar track (PSR and/or SSR updated). Bit-5 from first extend, (TCC) Type of plot coordinate transformation mechanism:</p> <p>If “0”, tracking performed in so called 'Radar Plane', i.e. neither slant range correction nor stereographical projection was applied.</p> <p>If “1”, slant range correction and a suitable projection technique are used to track in a 2D reference plane, tangential to the earth model at the Radar Site coordinates</p>

In some situations, radars also provide information about the quality of the estimation they provide, as the sensor itself might compute an internal local track and provide its output to the rest of the surveillance chain,⁹⁸ which includes the quality estimation. As an example, Cat.048 offers the possibility of sending information that is related to the local track computed at the sensor site. However, in Cat.010, this information is related to the uncertainty of the measurement itself mainly when provided by a MLAT system. These data items are:

⁹⁸ In surveillance, this is normally referred as a monotrack opposite to multitrack, which is computed in the ATC centre by means of fusing data coming from several data associated to the same target.

Table 6: Quality information within ASTERIX Cat.010 and 048

Quality information	Data Item	Description
Standard deviation	I010/500	<i>Standard Deviation of Position.</i> Standard deviation of position. LSB Standard deviation of X (σ_x): 0.25m LSB Standard deviation of Y (σ_y): 0.25m LSB Covariance of XY (σ_{xy}): 0.25m ²
	I048/210	<i>Track Quality</i> Track quality in the form of a vector of standard deviations. LSB Standard deviation of X (σ_x): 1/128NM LSB Standard deviation of Y (σ_y): 1/128NM LSB Standard deviation of ground speed (σ_v): 2 ⁻¹⁴ NM/s LSB Standard deviation of heading (σ_H): 360°/ 2 ¹²

Nowadays, there are sensors that can get information from onboard systems of the aircraft. This is the case not only of Mode-S radars and ADS-B, but also Multilateration systems, if they use the Mode-S information for their calculations. Such sensors are able to extract or receive extended information from Mode-S capable targets. These pieces of information are commonly known as Downlink Airborne Parameter (DAP) or Aircraft Derived Data (ADD) and are stored in the BDS registers of the Mode-S transponders. This information, once received at the sensors, is sent to the ATC centre by means of the following data items:

Table 7: Airborne information within ASTERIX Cat.010 and 048

Airborne information	Data Item	Description
Downlink Airborne Parameters	I010/250	<i>Mode S MB Data</i> Mode S Comm B data as extracted from the aircraft transponder
	I048/250	<i>Mode S MB Data</i> Mode S Comm B data as extracted from the aircraft transponder

Further information concerning DAPs can be found in ICAO document 9871 “Technical Provisions for Mode S Services and Extended Squitter” [ICAO 9871].

2.2.1.4 Coordinate transformations

2.2.1.4.1 Local coordinate system

The position information provided by radars is presented in a slant polar coordinate system. This means that the raw measurement is composed of the range measurement (ρ), which is the slant range to the target, and the azimuth (θ), which is the angle with respect to a reference, usually the geographical north at the sensor site location. These measurements are taken with respect to the radar local plane and are generally expected without any slant range correction or projection.

Nowadays, almost all new radars also provide the possibility to apply a stereographic projection when obtaining the Cartesian coordinates. These coordinates are used to track in a 2D reference plane which is tangential to the chosen earth model (e.g. WGS84 Ellipsoid) at the radar location. A slant range correction can also be applied in some of the situations if the height of the target is known or it can be approximated.

2.2.1.4.2 Transformations to system coordinates

All sensor measurements need coordinate conversions as they are not directly delivered in the system coordinates. This conversion might have some impact on the quality of the measurements. For example, going from Range-Azimuth to Cartesian coordinates using the same plane and without any type of projection has no negative effect on the quality of the measurements. However, it is not always the case, as some of them need numerical algorithms to be applied to get an approximated value. The latter happens when transforming geocentric Cartesian coordinates into geodetic coordinates as the latitude must be computed by iteration.

The process for transforming radar local polar coordinates into system ones is [ASTERIX Cat.017 App.A]:

1. From local polar coordinates plus height to local spherical coordinates:

$$[\rho, \theta] + H \rightarrow [\rho, \theta, \varphi] \quad (2.10)$$

The height over the ellipsoid (H) is usually provided by the sensor. The elevation (φ) is computed using the following formulae:

$$\sin \varphi = \frac{2R_R(H - h_R) + H^2 - h_R^2 - \rho^2}{2\rho(R_R - h_R)} \quad (2.11)$$

Where:

- R_R : local Earth radius at the latitude (ϕ_R) of the radar:

$$R_R = \frac{a(1 - e^2)}{\sqrt{(1 - e^2 \sin^2 \phi_R)^3}} \quad (2.12)$$

- a : semi-major axis of the Earth reference ellipsoid (i.e WGS84).
- e : eccentricity of the Earth reference ellipsoid.

- h_R : height of the radar over the reference ellipsoid.

2. From local spherical coordinates to local Cartesian coordinates:

$$[\rho, \theta, \varphi] \rightarrow [X_L, Y_L, Z_L] \quad (2.13)$$

The equations are:

$$X_L = \rho \cos \varphi \sin \theta \quad (2.14)$$

$$Y_L = \rho \cos \varphi \cos \theta \quad (2.15)$$

$$Z_L = \rho \sin \varphi \quad (2.16)$$

3. From local Cartesian coordinates to geocentric Cartesian coordinates:

$$[X_L, Y_L, Z_L] \rightarrow [X_G, Y_G, Z_G] \quad (2.17)$$

This transformation means a change in the origin of the coordinate system and also a rotation of the Cartesian axis. This is done by the following equations:

$$\begin{bmatrix} X_G \\ Y_G \\ Z_G \end{bmatrix} = R_{GR}^T \begin{bmatrix} X_L \\ Y_L \\ Z_L \end{bmatrix} + T_G = R_{RG} \begin{bmatrix} X_L \\ Y_L \\ Z_L \end{bmatrix} + T_G \quad (2.18)$$

Where:

- $[\phi_R, \lambda_R, h_R]$: geodetic coordinates of the radar (i.e. latitude, longitude and height).
- R_{GR} : rotation matrix, which converts vectors in the geocentric coordinate system into vectors in the radar local system assuming both coordinate systems share the same centre. It has the following expression:

$$R_{GR} = \begin{bmatrix} -\sin \lambda_R & \cos \lambda_R & 0 \\ -\sin \phi_R \cos \lambda_R & -\sin \phi_R \sin \lambda_R & \cos \phi_R \\ \cos \phi_R \cos \lambda_R & \cos \phi_R \sin \lambda_R & \sin \phi_R \end{bmatrix} = R_{RG}^T \quad (2.19)$$

Its transpose (R_{GR}^T) does the opposite.

- T_G : position of the radar system in the geocentric coordinate system.

$$T_G = \begin{bmatrix} (\eta_R + h_R) \cos \phi_R \cos \lambda_R \\ (\eta_R + h_R) \cos \phi_R \sin \lambda_R \\ (\eta_R(1 - e^2) + h_R) \sin \phi_R \end{bmatrix} \quad (2.20)$$

- η_R : normalized Earth radius at the latitude of the radar.

$$\eta_R = \frac{a}{\sqrt{1 - e^2 \sin^2 \phi_R}} \quad (2.21)$$

4. From geocentric Cartesian coordinates to system Cartesian coordinates:

$$[X_G, Y_G, Z_G] \rightarrow [X_S, Y_S, Z_S] \quad (2.22)$$

Like the previous one, this transformation means a change in the origin of the coordinate system and also a rotation of the Cartesian axis. This is done by the following equations:

$$\begin{bmatrix} X_S \\ Y_S \\ Z_S \end{bmatrix} = R_{GS} \left(\begin{bmatrix} X_G \\ Y_G \\ Z_G \end{bmatrix} - T_S \right) \quad (2.23)$$

Where:

- $[\phi_S, \lambda_S, h_S]$: geodetic coordinates of the origin of the system (i.e. latitude, longitude and height).
- R_{GS} : rotation matrix, which converts vectors in the geocentric coordinate system into vectors in the system coordinates assuming both coordinate systems share the same centre. It has the following expression:

$$R_{GS} = \begin{bmatrix} -\sin \lambda_S & \cos \lambda_S & 0 \\ -\sin \phi_S \cos \lambda_S & -\sin \phi_S \sin \lambda_S & \cos \phi_S \\ \cos \phi_S \cos \lambda_S & \cos \phi_S \sin \lambda_S & \sin \phi_S \end{bmatrix} \quad (2.24)$$

- T_S : position of the origin of the system coordinates in the geocentric coordinate system.

$$T_S = \begin{bmatrix} (\eta_S + h_S) \cos \phi_R \cos \lambda_R \\ (\eta_S + h_S) \cos \phi_R \sin \lambda_R \\ (\eta_S(1 - e^2) + h_S) \sin \phi_R \end{bmatrix} \quad (2.25)$$

- η_S : normalized Earth radius at the latitude of the origin of the system coordinates.

$$\eta_S = \frac{a}{\sqrt{1 - e^2 \sin^2 \phi_S}} \quad (2.26)$$

Note that it could be possible to go directly from radar local coordinates to system coordinates by means of applying the following equations:

$$[X_L, Y_L, Z_L] \rightarrow [X_S, Y_S, Z_S] \quad (2.27)$$

$$\begin{bmatrix} X_S \\ Y_S \\ Z_S \end{bmatrix} = R_{RS} \begin{bmatrix} X_L \\ Y_L \\ Z_L \end{bmatrix} + T_{R-S} \quad (2.28)$$

Where:

- R_{RS} : rotation matrix, which converts vectors in the radar local coordinate system into vectors in the system coordinates assuming both coordinate systems share the same centre. It can be obtained as a composition of the other two rotation matrices:

$$R_{RS} = R_{GS} R_{RG} = R_{GS} R_{GR}^T \quad (2.29)$$

- T_{R-S} : translation vector, which can be computed as:

$$T_{R-S} = R_{GS}(T_R - T_S) \quad (2.30)$$

5. From system Cartesian coordinates to system stereographic coordinates:

$$[X_S, Y_S, Z_S] \rightarrow [U, V, H] \quad (2.31)$$

The projection is described by the following expression:

$$[U, V, H] = \left[kX_S, kY_S, \frac{H_1(R_S + H_2) + d_{xy}^2}{R_S \sqrt{H_2^2 + d_{xy}^2}} \right] \quad (2.32)$$

Where:

- R_S : local Earth radius at the latitude (ϕ_S) of the system centre.
- k : constant with the following value:

$$k = \frac{1}{1 + \frac{1}{2} \frac{h_S + Z_S + H}{R_S}} \quad (2.33)$$

- H_1 : constant with the following value:

$$H_1 = Z_S + h_S \quad (2.34)$$

- H_2 : constant with the following value:

$$H_2 = R_S + H_1 \quad (2.35)$$

- d_{xy}^2 : constant with the following value:

$$d_{xy}^2 = X_S^2 + Y_S^2 \quad (2.36)$$

All the previous coordinate transformations and projection allow to place the target in a common reference frame where all the information coming from different sources can be compared and, if wanted, fused.

2.2.1.4.3 Airport simplifications

The airport environment covers an area of several squared kilometres. This area covers the runway(s), part of the glidepath and taking off segment, the taxiways and the apron. Assuming that the system centre, which is usually collocated with the ARP, is more or less in the middle of the airport area, the approach of the flat Earth model holds. Even more, the height of the aircraft is also very close to the height of the system centre. This means that the difference between using the system stereographic projection instead of the system local one is negligible, so the last step can be avoided.

2.2.2 Primary surveillance radars

2.2.2.1 Features

Primary Surveillance Radars (PSR) are active non-cooperative independent surveillance sensors which provide the capability of detecting non-cooperative or non-equipped targets. This is the reason why it is expected that they continue in the airport domain for the short future. PSR are usually covering the terminal manoeuvring area (TMA) and the approach zone (APP) of an airport, and are commonly collocated with a SSR, so they help to provide reinforced plots (also called combined plots) of the targets on the landing and taking off phases. Sometimes these sensors are also called Airport Surveillance Radar (ASR), mainly in the USA.

A PSR for terminal areas has an update period between 4 and 5 seconds, opposite to the 10 to 12 seconds for an en-route PSR.

2.2.2.2 Particularities

Since PSR systems are non-cooperative sensors, all aircraft-based biases can be removed as they do not affect the sensor measurement. There is an additional source of error which is the slant range. This error can be corrected if height information is available; if not, it can be considered within the noise error which should be conveniently increased in order to cope with this issue.

2.2.2.3 Measurement model

The generic radar measurement model (eqns. 2.02 and 2.03) can be simplified, and its final expression is:

$$\rho_i^m(k) = [1 + \epsilon_i^\rho(k)]\rho_i(k) + b_i^\rho(k) + w_i^\rho(k) \quad (2.37)$$

$$\theta_i^m(k) = \theta_i(k) + b_i^\theta(k) + w_i^\theta(k) \quad (2.38)$$

Where:

- $\rho_i(k), \theta_i(k)$: true range (ρ) and azimuth (θ) values corresponding to the i -th PSR radar at time k .
- $\epsilon_i^\rho(k)$: scale bias for the range corresponding to the i -th PSR radar at time k .
- $b_i^\rho(k), b_i^\theta(k)$: offset biases for the range and azimuth corresponding to the i -th PSR radar at time k .
- $w_i^\rho(k), w_i^\theta(k)$: are the measurement noises for the range and azimuth corresponding to the i -th PSR radar at time k .

Thus, $\epsilon_i^\rho(k)$ includes the effect of the RF propagation of the signal as it goes from the radar to the target and the other way round. The general assumption is that this propagation goes in a straight line and at a constant speed, which is not always true. Consequently, depending on the characteristics of the atmosphere and the layers that the signal traverses, the direction and the speed of the signal might be affected causing this value to be more or less important. Usually, it is measured in parts per million (ppm).

$b_i^{\rho}(k)$ is an offset bias that models any calibration error in the processing chain while measuring the time elapsed between the transmitted signals and the received echoes. It represents an error along a radial direction from the radar with a value that is normally around the tens of meters. It is expected to be corrected by any self-calibrating process within the radar. However, the latter might not be always available.

$b_i^{\theta}(k)$ is an offset bias that incorporates the misalignment between the actual azimuth of the antenna and the value reported by the radar. This value can be of hundredths of degrees.

Finally, $w_i^{\rho}(k)$ and $w_i^{\theta}(k)$ are statistically characterized as described in the generic measurement model, see section 2.2.1.2.

$w_i^{\rho}(k)$ includes the thermal noise and the jitter within the RF signal detector. Furthermore, it also includes any other source of error which cannot be modelled as a systematic error such as the quantization error. A current value for its standard deviation is $\sigma_r = 100\text{m}$.

On the other hand, $w_i^{\theta}(k)$ includes the measurement noise in the estimation of the antenna azimuth. A current value for its standard deviation is $\sigma_{\theta} = 0.09^{\circ}$.

2.2.2.4 Data Format

This sensor is expected to provide its output and its status in ASTERIX Cat.048 and Cat.034, respectively. For further information, see 2.2.1.3.

2.2.3 Surface movement radars

2.2.3.1 Features

The Surface Movement Radar (SMR) is a kind of primary radar with special features to be used in the airport domain. Thus, it is an active non-cooperative independent surveillance sensor that provides the capability of detecting non-cooperative or non-equipped targets the same way as a primary radar does. Due to its high resolution and high update rate, it is an almost mandatory sensor for any airport of medium/high capacity. The inconvenience is that it does not provide identification, so it has to be complemented either by the controller who can identify the target by other means, usually visual ones, or by other surveillance sensors with the capability of identifying the target such as ADS-B or MLAT. In the USA, this sensor is called Airport Surface Detection Equipment (ASDE).

The SMR usually covers the runway and taxiways and in some cases also the apron, which means a 5 to 10 km of coverage. It does not usually reach more than 100m in height, so its coverage is mainly focussed on the airport surface.

The SMR has an update rate of a second that is in line with the update rate required by the airport surface surveillance function in order to support the other functions that are included in an A-SMGCS system such as Safety Nets,⁹⁹ Routing or Guidance.

⁹⁹ It is also known as the Control function.

2.2.3.2 Particularities

The main particularities of an SMR are the ones related to the features that make it an interesting sensor for the airport surface surveillance.

Firstly, the SMR provides radar images¹⁰⁰ of the targets opposite to a PSR that detects the target just as a point. This feature has its advantages, as more information about the target is available, but also its drawbacks. Radar images are not continuous, which means that a further processing of the radar image must be done. This processing is focussed on the analysis of each of the different continuous parts, which compose the radar image, that are called blobs. Therefore, blob processing includes both, the association of each blob to a target and the tracking of the grouped blobs. This topic was analysed and a tracking algorithm proposed in [García 03].

Secondly, a SMR extracts a lot of information about the target analysing the blobs that compose it. Thus, the length and width of the target can be estimated and then used for target classification (e.g. an aircraft, a vehicle, etc.). Also, the orientation of the target can be extracted. In fact, as it has been shown in subsection 2.2.1.3, the ASTERIX Cat.010 defines a data item for sending the size and the orientation of the target (i.e. I010/270). On the other hand, this information can also be misleading in case there are bad associations of blobs to targets or holes in the radar images due normally to occlusions caused by other parts of the target, other targets, or buildings. These issues might cause the position estimation algorithm the loss of some precision while computing the position and orientation of the target.

Finally, the measured position provided in the target report is quite dependant on the correct association of the blobs to the actual target. The surveillance system is fed with the measured position, which implies that the blob processing is assumed to be within the SMR business logic and so, out of the target of this work.

2.2.3.3 Measurement model

Considering the features of this sensor, its measurement model can be reduced to:

$$\rho_i^m(k) = \rho_i(k) + b_i^\rho(k) + w_i^\rho(k) \quad (2.39)$$

$$\theta_i^m(k) = \theta_i(k) + b_i^\theta(k) + w_i^\theta(k) \quad (2.40)$$

Where:

- $\rho_i(k), \theta_i(k)$: true range (ρ) and azimuth (θ) values corresponding to the i -th SMR at time k .
- $b_i^\rho(k), b_i^\theta(k)$: are the offset biases for the range and azimuth corresponding to the i -th SMR at time k .

¹⁰⁰ Due to its high resolution, a target is represented not just as a detection, but as a group of detections which represents an image. Thus, they are referred as “extended” targets, opposite to a point target, which is the case for the rest of the radars as the dimension of a target is small compared to the angular and range resolution of such radar, that is, its resolution cell.

- $w_i^{\rho}(k)$, $w_i^{\theta}(k)$ are the measurement noises for the range and azimuth corresponding to the i -th SMR at time k .

Similar to the previous sensor, $b_i^{\rho}(k)$ is an offset bias that models any calibration error in the processing chain while measuring the time elapsed between the transmitted signal and the received echo. This represents an error along a radial direction from the radar with a value that is normally around several meters. This issue might be corrected by a self-calibrating process within the radar, so it can be estimated and then cancelled from the measurements.

$b_i^{\theta}(k)$ is also an offset bias that incorporates the misalignment between the actual azimuth of the antenna and the value reported by the radar. This value can be of hundredths of degrees.

As for the PSR, $w_i^{\rho}(k)$ and $w_i^{\theta}(k)$ are statistically characterized as described in the generic measurement model, see section 2.2.1.2.

$w_i^{\rho}(k)$ includes the thermal noise and the jitter within the RF signal detector. Additionally, it also includes the correlated error associated to the extraction of the position of the centroid, when it is not a bias, that is provided to the tracking system and the quantization error. A current value for its standard deviation is $\sigma_{\rho}=5\text{m}$.

Finally, $w_i^{\theta}(k)$ considers the measurement noise in the estimation of the antenna azimuth. A current value for its standard deviation is $\sigma_{\theta}=0.05^{\circ}$.

2.2.3.4 Data format

This sensor is expected to provide its output and its status in ASTERIX Cat.010. For further information, see 2.2.1.3.

2.2.4 Secondary surveillance radars

2.2.4.1 Features

Conventional Secondary Surveillance Radars (SSR) and their next generations, Monopulse Secondary Surveillance Radars (MSSR) and Mode S Radars are the sensors in which relies today's en-route and TMA surveillance. They are active cooperative independent surveillance sensors which provide the capability of interrogating cooperative targets to obtain more than just location or classification. In the USA, these sensors are usually referred as ATCRBS (SSR) and DABS (Mode-S Radar).

Conventional SSR systems interrogate aircraft about their Mode 3A (identification code) and Mode C (barometric height), so they allow the identification of aircraft and/or vehicles and the estimation of their height. They transmit the interrogations in 1030MHz and wait the responses in 1090MHz. This way the power needed is much reduced

compared to a PSR. They do not suffer of clutter¹⁰¹ but on the other hand, they have other drawbacks such as reflections, garbling and FRUIT.¹⁰²

The location technique is based on estimating the range and the azimuth of the target. Thus, the technique used for range estimation is similar to the one used by PSR, that is, measuring the elapsed time between the moment in which the interrogation is sent and the moment in which the corresponding response is received. In this case, there is a difference from PSRs, which is that the SSR transponder of the aircraft must wait a certain time before sending the response. The tolerance associated to the response delay precludes the use of these sensors for surface surveillance (see next subsection), apart from the fact that in some airports, local procedures mandate that the SSR transponder has to be switch off while the aircraft is on the ground. Nowadays with the advent of Mode S MLT systems, some of these procedures have been removed or modified to allow the use of Mode S interrogations in the airport surface. Concerning the azimuth, these systems estimate it using the sliding window technique, which is currently being substituted by the monopulse technique.

Monopulse Secondary Surveillance Radars are an upgrade of SSR systems. The main improvement is the employment of the monopulse technique for obtaining a more accurate estimation of the target azimuth. This technique triples the accuracy provided by a conventional SSR and improves the decoding performance as it reduces the appearance of garbling or FRUIT by about 90%.

Finally, Mode S Radars are an evolutionary replacement of SSR, which apart from including the monopulse technique for the azimuth estimation, they also present the ability of performing selective interrogations an also offering a data link for communication purposes. They have been introduced gradually since the 90's. However, they needed a change in the airborne transponders, so until the ICAO mandate for carrying Mode S transponders, their commissioning has been much reduced.

Additionally, there is a family of SSR systems that is devoted to cover the approach phase to airports with dependant parallel runways. They are called Precision Runway Monitoring (PRM).

The coverage ranges from the tens of kilometres for TMA to several hundred for the en-route airspace. As already mentioned, these sensors are mainly in charge of the en-route and TMA air surveillance, so for the purpose of this work, they are just interesting for the approach and departure phases of flight of those aircraft which land at the airport or take off from it. When devoted to airport surveillance, they are usually complemented with a PSR in order to cover both cooperative and non-cooperative targets.

These systems have a very wide interrogation period that ranges from the 1 or 2 seconds for a PRM to 10 to 12 seconds for en-route SSR, MSSR or Mode-S radars. If they are devoted to TMA surveillance, the interrogation period is usually configured between 4 and 5 seconds.

¹⁰¹ Clutter groups all the echoes that arrive to the radar sensor that do not belong to an actual target for the radar sensor. These echoes complicate the detection process if not properly removed.

¹⁰² FRUIT stands for "False Replies Unsynchronized in Time" and appears when responses to a radar are received and processed by another radar which is not the interrogating one.

2.2.4.2 Particularities

SSR, MSSR and Mode S radars share the same drawbacks while being used in the airport surface. Since they are cooperative sensors, they rely on equipment that is onboard the aircraft, so they also have to deal with the issues related to these equipments, i.e. the transponder.

The transponder response delay has a value of $3\mu\text{s}$ for SSR responses and $128\mu\text{s}$ for Mode S ones with a tolerance of $0.5\mu\text{s}$ and $0.25\mu\text{s}$, respectively. This tolerance can cause a potential transponder induced bias, which is different for each aircraft. If this tolerance is translated into position error, it can reach up to 75m in the conventional SSR case. In fact, this error is currently modelled as a uniform random variable which goes from -75m to +75m. This fact prevents them from being used in the airport domain where an error less than 7.5m the 95 % of the time is expected, if A-SMGCS level II is requested [ICAO 9830]. Although, this bias can be estimated and then cancelled as it is described in [Soto 04], the measurement noise that this sensor presents makes it difficult to be used in the airport surface. Additionally, this is a systematic error that adds to any other range bias (sensor-based) affecting the radar. This fact implies that if both are present with just the measurements coming from a target, they are not observable, and so they have to be estimated together. However, as they have different origins (i.e. one is sensor-based and the other target-based), it is possible to estimate them separately, if enough data is available.

There is a further complication which is that the transponder antenna¹⁰³ position is not close to the centroid of the aircraft (see ANNEX III). In this manner, there is a difference between the antenna position, which is the position measured by the radar, and the position to be reported to the surface surveillance system. The issue is that such surveillance system, as a data fusion application, requires all the information coming from the surveillance sensors to be referenced to the same point within the target, before merging it.¹⁰⁴ This is the only way to guarantee that the result of the fusion process is better than the one obtained by each of the information sources (surveillance sensors) alone. The reported position to the rest of the processing chain should be the aircraft position reference point which is defined in [RTCA DO-242B] depending on the dimensions of the aircraft. Thus, the position estimate shall be corrected by considering the actual position of the transponder antenna and the orientation of the target. The distance between these two points can be estimated and further corrected by applying the appropriate bias estimation algorithms [Besada 04b][Besada 05]. This issue is fundamental as future surface movement and runway incursion applications will request improved surveillance information in order to satisfy their operational needs.

Finally, there is a trouble that takes more importance in long range radar sensors which is the antenna eccentricity. This effect is reflected in an error in the estimation of the azimuth of the target due to the loss of verticality of the antenna rotating axis, which depends on the elevation angle where the target is located. However, for the problem that is addressed in this thesis, i.e. state smoothing of targets moving around an airport, this trouble is considered negligible.

¹⁰³ The transponder antenna is also referred to as the ATC antenna.

¹⁰⁴ For air surveillance, this requirement is not such demanding as the random errors (noises) that affect these measurements are quite above the magnitude of the antenna location induced bias.

2.2.4.3 Measurement model

The measurement model of SSR systems includes more error terms than in the previous cases as there are aircraft-based biases. Furthermore, the antenna eccentricity is also included for the sake of completeness. Thus, the expressions for the range (ρ^m) and azimuth (θ^m) measurements are:

$$\rho_i^m(k) = [1 + \epsilon_i^\rho(k)]\rho_i(k) + b_i^\rho(k) + \Delta R_j + \Delta l_j^\rho + w_i^\rho(k) \quad (2.41)$$

$$\theta_i^m(k) = \theta_i(k) + b_i^\theta(k) + \Delta\theta_i^{e1}\sin\theta_i(k) + \Delta\theta_i^{e2}\cos\theta_i(k) + \Delta l_j^\theta + w_i^\theta(k) \quad (2.42)$$

Where:

- $\rho_i(k)$, $\theta_i(k)$: true range (ρ) and azimuth (θ) values corresponding to radar i at time k .
- $\epsilon_i^\rho(k)$: sensor-based scale bias for the range corresponding to radar i at time k .
- $b_i^\rho(k)$, $b_i^\theta(k)$: sensor-based offset biases for the range and azimuth corresponding to radar i at time k .
- ΔR_j : transponder response delay induced error for aircraft j , so it is an aircraft-based offset bias.
- Δl_j^ρ , Δl_j^θ : offset biases depending on the ATC antenna position and the orientation (α) of the j -th aircraft.
- $\Delta\theta_i^{e1}$, $\Delta\theta_i^{e2}$: values which characterize the radar's antenna eccentricity, related to maximum eccentricity and angle of maximum eccentricity.
- $w_i^\rho(k)$, $w_i^\theta(k)$: measurement noises for the range and azimuth corresponding to radar i at time k .

Like PSR systems (see section 2.2.2.3), $\epsilon_i^\rho(k)$ models the effect of the RF propagation of the transmitted signal as it goes from the radar to the target and the inverse path performed by the reply. For secondary radars, this error term can be of importance due to the large coverage of these sensors.

$b_i^\rho(k)$ has the same physical meaning as in the previous radar sensors, but instead of talking about echoes, here it refers to responses or replies. So, it is an offset bias that considers any calibration error in the processing chain while measuring the time elapsed between the transmitted signals and the received responses. Its value can reach few hundreds of meters.

$b_i^\theta(k)$ is an offset bias that considers the misalignment between the actual azimuth of the antenna and the value reported by the radar. This value can reach tenths of degree.

ΔR_j models any error due to the tolerance in the calibration of the transponder response delay for the j -th aircraft. It is coupled with any other range bias (i.e. $b_i^\rho(k)$) that the

sensor may introduce. This aircraft-based bias is modelled as a uniform random variable between $-75m$ and $75m$.

Δl_j^ρ and Δl_j^θ represent the projection into both coordinates (ρ, θ) of the offset bias due to the discrepancy between the ATC antenna location and the aircraft position reference point or centroid (Δl_j). Δl_j can also be modelled as a uniform random variable between $-40m$ and $40m$. However, for some aircraft manufacturers, this bias can become a known value ($\overline{\Delta l_{ATC}}$ as presented in section 6.10.7.5) that can be obtained from a database if the surveillance system is able to obtain the "Aircraft Type" of the target. Their values are:

$$\begin{bmatrix} \Delta l_j^\rho \\ \Delta l_j^\theta \end{bmatrix} = \Delta l_j \begin{bmatrix} \cos \theta & \sin \theta \\ -\rho \sin \theta & \rho \cos \theta \end{bmatrix} \begin{bmatrix} \sin \alpha \\ \cos \alpha \end{bmatrix} \quad (2.43)$$

$\Delta \theta_i^{e1}$ and $\Delta \theta_i^{e2}$ are the terms that characterize the antenna eccentricity of a radar system. These terms are usually considered for long range radars. For the case of the airport, their value is assumed negligible as the elevation (ϕ) of the targets is close to 0 and so $\tan \phi$ is also 0, see [Guiar 86] for further details.

As for the previous radars, $w_i^\rho(k)$ and $w_i^\theta(k)$ are statistically characterized as described in the generic measurement model, see section 2.2.1.2.

$w_i^\rho(k)$ includes the thermal noise and the jitter within the RF signal detector. Additionally, it also includes any other source of error which cannot be modelled as systematic error such as the quantization error. A current value for its standard deviation is $\sigma_\rho = 100m$ for conventional SSR and MSSR, and $\sigma_\rho = 70m$ for Mode S radars.

On the other hand, $w_i^\theta(k)$ includes the measurement noise in the estimation of the antenna azimuth. A current value for its standard deviation is $\sigma_\theta = 0.09^\circ$ for those radar using the sliding window technique (conventional SSR) and $\sigma_\theta = 0.03^\circ$ for the ones using the monopulse one (MSSR or Mode S radars).

For a more detailed SSR systematic error model please refer to [Jarama 17].

2.2.4.4 Data format

These sensors provide their output and their status in the following ASTERIX Categories:

- Target plots/reports/tracks: ASTERIX Cat.001 (into abeyance) and Cat.048.
- Sensor status and service messages: ASTERIX Cat.002 (into abeyance) and Cat.034.

For more details about these data formats see section 2.2.1.3. or visit the ASTERIX website (<https://www.eurocontrol.int/asterix>).

2.3 Automatic dependent surveillance

2.3.1 General considerations

The term “Automatic Dependent Surveillance” (ADS) groups a family of surveillance applications that automatically provide dependent information to the ground control segment without requiring any intervention of the pilot. This information is self-calculated by the aircraft on-board systems, so its availability depends on them.

This information includes:

- Aircraft identification.
- Aircraft position in three dimensions plus time (i.e. 4D).
- Other information related to the aircraft such as flight plan related data.

AENA defines ADS as "A surveillance technique by which an aircraft transmits, via data link, a series of parameters extracted from the on-board navigation and positioning systems." [AENA ADS]. From this definition, there are some requirements on the airborne side which consist of including within the avionics both, a navigation system and also a data link. Moreover, the ground control segment must be also able to receive, process and send ADS information to the corresponding SDP system. This is done by means of ground stations which are spaced all along the surveillance area.

Depending on how the transmission of the ADS information is performed, it can be divided into:

- ADS-B (Broadcast): it consists in frequently and periodically broadcasting a set of aircraft internally calculated parameters (i.e. ADD) by means of a data link.
- ADS-C (Contract): it implies a point to point connection between the aircraft and a certain ground station for transmitting some already agreed aircraft information.

Both alternatives are going to be analysed in the following subsections with the focus mainly placed on ADS-B as it is the one that is more related to surveillance aspects on the airport domain as ADS-C has been also proposed to contribute to some trajectory management topics like its contribution to the EPP concept (see section 2.3.3) for certain types of airport approach procedures.

2.3.2 Automatic dependent surveillance - Broadcast

2.3.2.1 Features

Automatic Dependent Surveillance - Broadcast (ADS-B) is a surveillance application, as defined by ICAO,¹⁰⁵ that is considered nowadays as the cornerstone for the short term evolution of Air Traffic Control both for en-route/TMA airspaces and for the airport surface. Thus, ADS-B is envisioned as a key enabler for the future ATM, which will contribute to the performance objectives that have been proposed worldwide by ICAO in [ICAO 9854] and more in detail in the Single European Sky (SES) policy for the European airspace and in the NEXTGEN programme for the U.S. one. These objectives have been translated into key performance indicators (KPI) which apply to safety, capacity, efficiency, and environmental sustainability.¹⁰⁶ Consequently, it is expected that ADS-B will progressively substitute radar sensors due to its improved performances and low cost. However, this substitution will not happen in the short term, therefore it is envisaged a considerably long transition.

As an application, ADS-B can currently be sent by means of different data link technologies such as:

- 1090ES [EUROCAE ED102A][RTCA DO-260B].
- VDL-Mode 4 [EUROCAE ED108A].
- UAT [RTCA DO-282B].

In this work, the main focus is going to be placed on the 1090ES data link, as it has been the one chosen by ICAO as the worldwide standard. However, in the future, it could be sent by means of any of the currently under development data links. This is the case of the future data link for the airport (AeroMACS¹⁰⁷ [EUROCAE ED223]), which is being developed under the umbrella of both, SESAR and NEXTGEN programmes.

ADS-B is based on periodic non-requested broadcast transmissions performed by the aircraft that are listened by other surrounding aircraft and by a network of ground stations that compile the information and send it to control centres.¹⁰⁸ This information includes position, velocity, identification, etc.

From the point of view of air surveillance, nothing prevents the rest of aircraft that surround the target from exploiting this information by means of a series of airborne applications. These applications are known as Airborne Separation Assurance System (ASAS)¹⁰⁹ and they are categorized depending on the level of responsibility that is

¹⁰⁵ ADS-B is defined by ICAO as "A surveillance application transmitting parameters, such as position, track and ground speed, via a broadcast mode data link, at specified intervals, for utilisation by any air and/or ground users requiring it".

¹⁰⁶ These four concepts are also known as the Key Performance Areas of the SESAR programme.

¹⁰⁷ AeroMACS stands for Aeronautical Mobile Airport Communications System. It is a broadband wireless mobile communications technology in the protected AMS spectrum (5091-5150MHz) for airport surface applications. It is based on WiMAX (IEEE 802.16).

¹⁰⁸ Depending on the airspace, it can be an Area Control Centre (ACC), Approach Control Center or Tower (TWR) or to all of them at the same time.

¹⁰⁹ For more information visit "The ASAS Thematic Network": <http://www.asas-tn.org/>

delegated to the pilot based on the capabilities of the aircraft while processing “ADS-B In” information. Thus, a first classification can divide them into:

- Airborne Traffic Situation Awareness applications (ATSA or ATSAW).
- Airborne Spacing applications (ASPA).
- Airborne Separation applications (ASEP).
- Self-Separation applications (SSEP).

These applications range from the ones in which no separation is delegated to the pilot (ATSA) to the one in which all separation responsibilities concerning the rest of the aircraft/vehicles are delegated to the pilot (SSEP). The latter will be the foundation for a free flight environment.

Focussing on the ground side, an ADS-B ground station is a passive cooperative dependent surveillance sensor as it waits for the reception (passive) of any of the periodically broadcasted ADS-B messages that are sent by the aircraft (cooperative). It is dependent because the position is not computed by the ground station but by the aircraft navigation system. Consequently, ground stations receive different ADS-B messages from an ADS-B equipped aircraft that are decoded, filtered, processed and compiled in order to build an ADS-B report that is then sent to one or several SDP systems, which might be located in different control centres that can be managing different airspaces (e.g. en-route, TMA and/or tower).

The coverage of ADS-B ground stations is up to 250NM (around 400Km) so they can be used for both en-route/TMA surveillance and airport surface surveillance. The only requirement is that Mode S transponders must be enabled while aircraft are moving around the airport. Due to its features, ADS-B should be supported by a non-cooperative sensor, such as an SMR, or at least by an independent sensor such as a MLAT system.

There are countries which have entrusted their complete air surveillance to ADS-B. This was the case of Australia¹¹⁰ but also the intention of the United States. However, there is still a lot of reluctance on this technology; so many countries are still testing it. Even in the case of Australia, they have come back to Mode-S radar coverage in some areas mainly close to the East coast.

Finally, a new concept has recently appeared: the space-based ADS-B. This initiative, which is known by the name of Aireon,¹¹¹ aims at providing a global aviation surveillance system based on a satellite communication network.

For further information on ADS-B, please visit the EUROCONTROL ADS-B website (<https://www.eurocontrol.int/service/automatic-dependent-surveillance-broadcast>).

¹¹⁰ Their initial idea was just to depend on ADS-B, however Australia has invested in several Mode S radars for covering the coast areas, so their surveillance has evolved into a mix of ADS-B and radar.

¹¹¹ Aireon was proposed by a group of companies (Iridium, Nav Canada, ENAV, IAA, NAVIAIR, HARRIS, EXELIS and BOEING) and it aimed at delivering “the first-ever space-based global navigation system” by 2017. Aireon is a reality since April 2019 when it became fully operational and in trial use over the North Atlantic. For further information, please visit: <http://www.aireon.com>.

2.3.2.2 Particularities

ADS-B, as a “quite” modern surveillance application that implies some modification on the airborne equipment, has some troubles since not all the aircraft are conveniently equipped. Even though there are airspaces where almost the one hundred percent of the aircraft are equipped, there are others in which this level is far from being reached. This fact prevents from achieving the full benefit that ADS-B can bring to the surveillance domain. However, some countries have released mandates concerning the ADS-B carriage which will speed up the adoption of this new technology as it is presented later.

The main issue with ADS-B is that the data link technology that has been selected for the worldwide standard by ICAO, i.e. 1090ES, implies a trade-off between the cost and easiness of the deployment and the capabilities that it offers. Consequently, based on the availability of ICAO SARPs¹¹² for Mode S transponders, on the mandates for carrying them and on the fact that 1090ES is just a software update for a Mode S transponder, the deployment has been eased. Furthermore, there are also special mandates for carriage of ADS-B at least in Europe (June 2020) and the U.S. (January 2020).

However, this data link has several drawbacks, namely:

- Low throughput.
- Absence of a time reference.
- Security and integrity concerns.
- Complications associated to the frequency and modulation used by Mode-S transponders.

This is the price that has been paid to be backward compatible.

Focussing, from now on, on 1090ES ADS-B, one of the issues that has been pointed out is that ADS-B messages are not timestamped, so a potential time bias may appear. Nowadays, many aircraft are UTC synchronized, which means that they know the time that will be associated to the information that they send to the ground and so, what they do is to extrapolate this information to this time. The latter is estimated and stored by the ground station in the “Time of applicability for position” data items of Cat.021 (I021/071). In the case of no synchronization, it is the ground station the one that timestamps the message when it arrives, which implies a high degree of uncertainty concerning the generation time of such information. This time is then stored in the “Time of message reception for position” data items of Cat.021 (I021/073), if it is an information related to position, and in the “Time of message reception for velocity” data items of Cat.021 (I021/075), otherwise.

Moreover, ADS-B provides target location in geodetic coordinates, that is, latitude, longitude, and height over the ellipsoid, i.e. (ϕ, λ, H) . These coordinates must be converted into Cartesian coordinates (X, Y, Z) by means of a stereographic projection based on the data fusion centre, which in the case of the airport is normally the ARP. This

¹¹² SARP stands for Standards and Recommend Practices.

conversion may imply the inclusion of some error that has to be considered in the sensor model.

Like the case of SSR systems, the position provided in ADS-B messages, unless corrected by the navigation system, corresponds to the location of the GNSS antenna of the aircraft and not to its centroid. This fact introduces a systematic error that needs to be corrected before allowing the ADS-B information entering in any data fusion process that the surveillance system might perform. As stated in [Garcia 10], the feedback from various airport movement trials has shown the importance of removing this bias because of its effect on other systems such as airport safety nets. In the past, the correction of the reported position, when performed, was stated in a bit called the Position Offset Applied (POA) [EUROCAE ED102]. However, in the latest version of the MOPS¹¹³ which define ADS-B [EUROCAE ED102A], a new field ("GPS Antenna Offset") has been introduced in order to inform about the position of the GPS antenna with 2m of resolution.

The 1090ES data link uses a very congested frequency that is also quite close to the one used by other systems within the airport. This is the case of nav aids such as DME or VOR, or systems such as ACAS and SSR radars. This fact sometimes explains the disappearance of ADS-B data, as was realized by the FAA while testing ADS-B at Atlanta Airport [NASA 1999]. Furthermore, reflections and occlusions due to the buildings and other aircraft may affect the detection of the signal which can be lost during several seconds.

ADS-B, as a cooperative surveillance application, requires the cooperation of the target, which implies that it must rely on the onboard Mode S Transponder. Moreover, it is also a dependant application, so it also relies on the aircraft navigation system for the position and velocity information, which means another failure point. The problem with the latter is not that the aircraft may not send its position or its velocity but, even worse, as it can send erroneous values which cannot be detected by the sensor because they satisfy the CRC (Cyclic Redundancy Check) code that protects from transmission errors. These wrong values might not only affect the "pure" position, but also its uncertainty estimate as provided by position quality indicators (PQI) such as the NIC, NAC, NUC, SIL, etc.

Nowadays, there is a further problem as navigation systems mainly rely on GPS, which means a common failure point. The advent of GALILEO will sort out this problem. However, this is not going to happen in the short term. If this problem is merged with the one related to the security concerns that affect ADS-B, there is a clear need for coherency checks on the information provided by ADS-B.

Finally, the security concerns are mainly due to the fact that the ADS-B signal is broadcast and not encrypted, so it is accessible to anyone with the correct receiver; and even worse, it can easily suffer from spoofing attacks, allowing the insertion of false targets into the air traffic awareness picture that is used by controllers for providing the ATC service.

2.3.2.3 Measurement model

Considering that the navigation systems currently deployed in modern aircraft are precise enough,¹¹⁴ ADS-B position reports suffer mainly from a time-stamping error and from a

¹¹³ MOPS stands for Minimum Operation Performance Standard.

¹¹⁴ Nowadays, navigation systems combine both GNSS based and inertial navigation systems, so they put together the good features of both systems.

bias due to the GNSS antenna position. In the case of ADS-B velocity reports, the time-stamping error is the main one. All these errors are aircraft dependent.

Besides, ADS-B position information is provided in geodetic coordinates, which are not easily manageable for tracking purposes. This is the reason why they are converted into Cartesian coordinates by means of a stereographic projection centred on the fusion centre location, in this case the ARP. For further details, go to section 2.3.2.5.

Thus, the measurement model for ADS-B position measures in Cartesian coordinates (X^m, Y^m) for the j -th aircraft at time k is:

$$X_j^m(k) = X_j(k) - V_j^x(k)\Delta t_j - \Delta l_j^x + w_j^x(k) \quad (2.44)$$

$$Y_j^m(k) = Y_j(k) - V_j^y(k)\Delta t_j - \Delta l_j^y + w_j^y(k) \quad (2.45)$$

Where:

- $X_j(k), Y_j(k)$: x and y components of the stereographic projection of the j -th target true position provided by ADS-B at time k .
- $V_j^x(k), V_j^y(k)$: velocity vector of the j -th target at time k .
- Δt_j : time bias for the j -th target.
- $\Delta l_j^x, \Delta l_j^y$: offset bias components that depend on the GNSS antenna position and the orientation (α) of the j -th aircraft.
- $w_j^x(k), w_j^y(k)$: measurement noise errors for the x and y components corresponding to the position measurement for the j -th target at time k .

Analysing the characteristics of each of the elements of the previous expressions, $V_j^x(k)$ and $V_j^y(k)$ are the velocity components of the state vector for the j -th target expressed in Cartesian coordinates. They are extracted from the track information and may not be totally coincident with the orientation of the target, for example, for a towed aircraft.

Δt_j is the time bias which is computed as the difference between the real time of the message and the timestamp provided by the ADS-B ground station in the ASTERIX message. This can also be stated as:

$$\Delta t_j = k - k_i \quad (2.46)$$

Where:

- k is the time of the ADS-B message in case it would had been timestamped taking into consideration the time reference for the data fusion process. This is usually the clock at the data fusion centre.
- k_i is the time stamp of the ADS-B message as it is provided by the ADS-B ground station, which depending on the features of the aircraft, it can be the time of applicability or the time of reception.

This new bias term mainly accounts for the part of the latency that is known as the “Uncompensated Latency” [GEN-SUR SPR Annex J], which appears between the moment in which the measurement is generated and the time in which it is stored in the BDS registers of the Mode S transponder, that is not properly estimated by the onboard system and as such it is “not compensated”. If the avionics are not able to properly estimate the latency, which heavily depends on the communication chain that each manufacturer implements in its aircraft, then an along trajectory error appears whose magnitude depends on the MOPS version of the Mode S transponder that stores the measurement in its BDS registers. The navigation system of the aircraft compensates the “estimated” latency by extrapolating the initial position measurements. The main difference between the transponders with different MOPS versions is that Version 2 (V2) ones have a generation time which is defined, so the new measurement can only be generated at some time instants regardless the other transponders (MOPS Version 0 and 1), which does not have such reference. The latter is translated in a bigger position error as the ADS-B ground station does not have any time reference for computing the applicable time better than the “reception time”, opposite to the case of MOPS V2 transponder where it has a time reference and its guess (i.e. time of applicability) is usually better than the one from the other transponders (i.e. time of reception). This issue is a consequence of the lack of a timestamp in the ADS-B message received at the ground station, which is one of the main issues with ADS-B information from a surveillance processing point of view.

Δl_j^x and Δl_j^y represent the projection into both coordinates x and y of the offset bias due to the discrepancy between the GNSS antenna location and the aircraft position reference point or centroid (Δl_j). Similar to the SSR case, Δl_j can also be modelled as a uniform random variable between $-40m$ and $40m$.

However, there are means for obtaining an estimation of this offset in case not corrected by the navigation system of the aircraft, as stated in the previous section. Moreover, for some aircraft manufacturers, this bias can become a known value ($\overline{\Delta l_{GNSS}}$ as presented in section 6.10.7.4) that can be obtained from a database, if the surveillance system is able to obtain the "Aircraft Type" of the target. The projected values are:

$$\Delta l_j^x = \Delta l_j \sin \alpha \quad (2.47)$$

$$\Delta l_j^y = \Delta l_j \cos \alpha \quad (2.48)$$

Finally, $w_j^x(k)$ and $w_j^y(k)$ account for the random measurement errors which are generally modelled as zero-mean white Gaussian random sequences with corresponding variances σ_x^2 and σ_y^2 , and covariance σ_{xy} . They can be estimated by means of the quality indicators that are included in ADS-B measurements (e.g. NIC, NAC, SIL or NUCp/NUCv) [ICAO 9871].

Furthermore, ADS-B ground stations can also provide ADS-B velocity reports when ADS-B velocity messages are broadcast by the aircraft. The velocity information is provided in polar coordinates, that is ground speed (GS) and heading (HDG) components. So, there is a need for a conversion before feeding the surveillance system with it.

$$V_j^{m,x}(k) = GS \sin(HDG) \quad (2.49)$$

$$V_j^{m,y}(k) = GS \cos(HDG) \quad (2.50)$$

The measurement model for ADS-B velocity measures in Cartesian coordinates $(V_j^{m,x}(k), V_j^{m,y}(k))$ for the j -th aircraft is modelled as:

$$V_j^{m,x}(k) = V_j^x(k) + w_j^{Vx}(k) \quad (2.51)$$

$$V_j^{m,y}(k) = V_j^y(k) + w_j^{Vy}(k) \quad (2.52)$$

Where:

- $V_j^x(k), V_j^y(k)$: stereographic projection of the j -th aircraft true velocity vector provided by the ADS-B ground station at time k .
- $w_j^{Vx}(k), w_j^{Vy}(k)$: measurement noise errors for the x and y components corresponding to the velocity measurement for the j -th aircraft at time k .

The terms $w_j^{Vx}(k)$ and $w_j^{Vy}(k)$ account for the random measurement errors and for not modelled accelerations. As in the previous cases, they are going to be modelled as zero-mean white Gaussian random sequences with corresponding variances σ_{Vx}^2 and σ_{Vy}^2 , and covariances $\sigma_{Vx \cdot Vy}$.

2.3.2.4 Data format

ADS-B ground stations provide their output (target reports or local track) in ASTERIX Cat.021 and Cat.010, while their sensor status and service messages are currently delivered in ASTERIX Cat.023, although they might also be received in Cat.025.

Similar to radar information, it is relevant to know the features of the information that is going to be processed, which includes the format in which it will be provided, its frequency and any previous data processing performed at the sensor site or previous to reaching the performance evaluation tool. It is pertinent to point out that the focus will be placed on non-processed data because the details of any previous data processing is usually unknown and so will be the statistical behaviour of such data. So, unless this processing is well stated, the strategy will be not to process this information, although it can be exploited for other purposes, such as coherency checks, and just keep the raw data, that is, the one directly measured by the sensor.

Following the same strategy presented in section 2.2.1.3, the focus is placed in identification, time and kinematic information. Hence, identification information will mainly contribute to the data association process, while time and kinematic information will be exploited during the trajectory reconstruction process. All this information is presented in the next tables. In order to avoid duplications, the information already presented in section 2.2.1.3 will be omitted:

Table 8: Identification and classification information within ASTERIX Cat.021

Identification information	Data Item	Description
Identification		
Mode A code	I021/070	<i>Mode 3/A Code in Octal Representation</i> Mode 3A code converted into octal representation.
ICAO 24-bit address	I021/080	<i>Target Address</i> Target address (emitter identifier) assigned uniquely to each target.
Local track number	I021/161	<i>Track Number</i> An integer value representing a unique reference to a track record within a particular track file.
Callsign	I021/170	<i>Target Identification</i> Target (aircraft or vehicle) identification in 8 characters, as reported by the target.
Classification		
Emitter Category	I021/020	<i>Emitter Category</i> Characteristics of the originating ADS-B unit.

The table above collects the data items from ASTERIX Cat.021, which deals with identification and classification information. The ones for Cat.010 were already reported.

This information is used for association purposes. Consequently, the Mode 3A is the piece of information used for association when tracking in a conventional radar environment. The ICAO 24-bit address is the evolution due to the shortage of Mode 3A codes,¹¹⁵ however the association of target reports to tracks is still heavily based on them. Based on this 24-bit address it is possible to reach the aircraft type, so some assumptions can be performed about the way the aircraft is going to behave or even some specific information can be obtained from the corresponding databases, such as the ATC or GNSS antenna location. Since many sensors perform a local tracking, sometimes it is useful to take advantage of such information to ease the further multisensor processing. This is something done by using the local track number. Moreover, there is the callsign that is mainly used for correlation¹¹⁶ purposes. The latter is useful if the flight plan information can be exploited for performance evaluation, which is often the case, as it provides

¹¹⁵ The number of Mode 3A codes is 4096 from which there are several related to special purposes (e.g. emergencies, radio malfunctioning or kidnapping) and others which are assigned to groups of aircraft such as general aviation (i.e. 7000) or aircraft which are associated using Mode S information and so they all are assigned the Mode S conspicuity code (i.e. 1000). However, this number is not enough for several air spaces where it is possible to find duplicated Mode 3A codes.

¹¹⁶ Correlation is the process of merging the information coming from the flight plan with the one coming from the surveillance chain. The main information that is used for such purpose is the Mode 3A and the callsign/flight ID.

information that cannot be obtained from the surveillance chain. Finally, the emitter category provides a classification based on the target type.

Regarding the timestamp, it is essential to be consistent with the data information provided by the target reports, mainly regarding to position. The ASTERIX data format requires that each target report has its corresponding timestamp and the time reference to be used is the co-ordinated Universal Time (UTC).¹¹⁷ Consequently, the data items concerning time to be considered are:

Table 9: Time information within ASTERIX Cat.021

Time information	Data Item	Description
Time of applicability	I021/071	<i>Time of Applicability for Position</i> Time of applicability of the reported position, in the form of elapsed time since last midnight, expressed as UTC. LSB: 1/128s
	I021/072	<i>Time of Applicability for Velocity</i> Time of applicability (measurement) of the reported velocity, in the form of elapsed time since last midnight, expressed as UTC. LSB: 1/128s
Time of reception	I021/073	<i>Time of Message Reception for Position</i> Time of reception of the latest position squitter in the Ground Station, in the form of elapsed time since last midnight, expressed as UTC. LSB: 1/128s
	I021/074	<i>Time of Message Reception of Position–High Precision</i> Time at which the latest ADS-B position information was received by the ground station, expressed as fraction of the second of the UTC Time. LSB: 2 ⁻³⁰ s.
	I021/075	<i>Time of Message Reception for Velocity</i> Time of reception of the latest velocity squitter in the Ground Station, in the form of elapsed time since last midnight, expressed as UTC. LSB: 1/128s

¹¹⁷ The use of UTC is required by ICAO in its Annex 5.

Time information	Data Item	Description
	I021/076	<i>Time of Message Reception of Velocity–High Precision</i> Time at which the latest ADS-B velocity information was received by the ground station, expressed as fraction of the second of the UTC Time. LSB: 2^{-30} s.
Time of transmission	I021/077	<i>Time of ASTERIX Report Transmission</i> Time of the transmission of the ASTERIX category 021 report in the form of elapsed time since last midnight, expressed as UTC. LSB: 1/128s

In the case of ADS-B, there are more time measurements than in other sensors because:

- ADS-B messages are not time stamped.
- Targets present different capabilities.
- Measurement precision requirements.

Concerning the target equipment, there are some aircraft that are UTC synchronized and others not. If the targets are UTC synchronized, usually through a GPS source, then it is possible for the ground station to know exactly the time corresponding to the measurement (position, velocity or both) to be processed and so, what the “time of applicability” is for the received information. This is the case for the aircraft carrying a MOPS Version 2 transponder. For non-synchronized targets, there is no time synchronization to stick to, so the target report is timestamped with the time of reception of the latest squitter (sending position or velocity) at the ground station. This fact obviously represents a loss in the quality of the measurement and, as such, an increase in its uncertainty.

However, the “time of reception” measurement opens the door to other possible calculations, such as the one based on multilateration, if this measurement is provided with enough precision. These calculations provide a coherency check for the position measurement provided by the target transponder. This is the reason why data items I021/074 and I021/076 increase the precision of the “time of reception” measurement until 2^{-30} s, opposite to the rest of measurement which represent the number of 1/128s elapsed since last midnight. It is important to note that they must be transmitted together with data items I021/073 and I021/075 respectively.

Another feature related to ADS-B is that also the “time of transmission” from the ground station is sent as it may take some time filling the information to be sent in the ADS-B target report from the different squitters that provide the information to be compiled (position, velocity, identification, etc.).

All these time measurements are reset to 0 every midnight. Moreover, some of these measurements (i.e. I021/072) might not be available in some ADS-B technologies and it is possible to find several of them in the same ADS-B target report (e.g. I021/071 and I021/073).

Like Mode S radars working in EHS, kinematic information is also provided in several data items. It includes position information and sometimes velocity. The position information is provided in geodetic coordinates and the velocity in polar ones, that is, in ground speed and heading components. This fact implies that a certain coordinate transformation needs to be applied.

The position information and the data items where it is provided are presented in the following table:

Table 10: Target information within ASTERIX Cat.021

Target information	Data Item	Description
2D position		
Position in geodetic coordinates	I021/130	<i>Position in WGS-84 Co-ordinates</i> Position in WGS-84 coordinates. LSB Latitude: 180/2 ²³ degrees LSB Longitude: 180/2 ²³ degrees
	I021/131	<i>High-Resolution Position in WGS-84 Co-ordinates</i> Position in WGS-84 coordinates in high resolution. LSB Latitude: 180/2 ³⁰ degrees LSB Longitude: 180/2 ³⁰ degrees
Height		
Mode C code	I021/145	<i>Flight Level</i> Flight Level from barometric measurements, not QNH ¹¹⁸ corrected, in two's complement form. LSB: ¼ FL, i.e. 25ft
Height	I021/140	<i>Geometric Height</i> Minimum height from a plane tangent to the earth's ellipsoid, defined by WGS-84, in two's complement form. LSB: 6.25ft

¹¹⁸ The QNH is an aeronautical Q code whose common mnemonic "Query: Natural Height" that represents the atmospheric pressure adjusted to the mean sea level. This value is provided on a regional base. When set for an airport, it makes the altimeter provide the airport altitude usually referred to the ARP or the altitude of the centre point of the main runway above sea level on landing, irrespective of the temperature. In the world, QNH is given in hectopascals or millibars, while in North America, QNH is given in hundredths of inches of mercury [Wikipedia].

Target information	Data Item	Description
Size and orientation		
Size and heading		Not available in Cat.021. Only in Cat.010.

Regarding position, as already mentioned, it is provided in geodetic coordinates, usually referred to the WGS84 ellipsoid. This fact implies that there is a need for transforming them into the coordinate system of the system centre. This process will be detailed in the next section.

Concerning height, two values are provided: the barometric altitude, in the form of Mode C values, that is, flight level not QNH corrected; and the geometric height, which is computed as the minimum height from a plane tangent to the reference Earth's ellipsoid.

Apart from position, the ADS-B messages also provide velocity information. The velocity information can be air or ground referenced. As it is said in [EUROCAE ED102A], air referenced velocity information (i.e. airspeed and magnetic heading) is sent in the “very rare event that no ground velocity data such as GPS velocity vector is available from airborne avionics”. So, in general, the one available is the ground reference information. This information is provided in the following data items:

Table 11: Ground referenced velocity information within ASTERIX Cat.021

Download information	Data Item	Description
Velocity		
Velocity in polar coordinates	I021/160	<i>Airborne Ground Vector</i> Ground Speed and Track Angle elements of Airborne Ground Vector. LSB Ground speed: 2^{-14} NM/s (referenced to WGS-84). LSB Track angle: $360/2^{16}$ degrees (clockwise reference to “True North”).

ASTERIX Cat.010 messages can be used for sending data coming from several sensors. Depending on the sensor, velocity information may be computed by the sensor, provided by the aircraft or calculated by a local tracker. In the case of the SMR, velocity information is generated by a local tracker and not provided by a sensor or the target itself. The latter is the case of ADS-B. This is the reason why this information has been removed from the previous table as they are expected to be provided by a local tracker.

ADS-B as a dependent technology also provides information about the quality of its position and velocity estimations. The indicators to be provided depend on the link technology protocol version usually reported as the “MOPS version” within I021/210. These data items are:

Table 12: Quality information within ASTERIX Cat.021

Quality information	Data Item	Description
Position Quality Indicators	I021/090	<i>Quality Indicators</i> ADS-B quality indicators transmitted by a/c according to MOPS version. <ul style="list-style-type: none"> • NUC_r, NUC_p. • NAC_v, NAC_p, NIC, SIL • NIC_{BARO}, SILS, SDA, GVA, PIC

ADS-B quality indicators sent by the aircraft are transmitted according to the corresponding MOPS version. Thus, there is a primary subfield which guarantees the backward compatibility and then three extensions. The first one is related to Navigation Accuracy Category for Position and the other two for Position Quality Indicators.

The PET will consider all these values in order to generate the corresponding approximate standard deviation values or covariance values to be introduced in the data fusion process in charge of generating the reconstructed trajectory. This step will be detailed as part of the data acquisition which will be described section 5.2.

Another feature of ADS-B is that it allows the extraction of onboard information from the BDS registers of the Mode S transponder (e.g. BDS 040,050,070, etc.) which are then broadcast by means of the 1090ES data link. This information, once received at the ground stations, is sent to the ATC centres by means of the following data items:

Table 13: Airborne information within ASTERIX Cat.010 and 021.

Airborne information	Data Item	Description
Downlink Airborne Parameters	I021/250	<i>Mode S MB Data</i> Mode S Comm B data as extracted from the aircraft transponder.

Further information concerning ADD can be found in [ICAO 9871].

For more details about these data formats, please go to ASTERIX website (<https://www.eurocontrol.int/asterix>)

2.3.2.5 Coordinate transformations

2.3.2.5.1 Global coordinate system

The position information provided by ADS-B systems is presented in geodetic coordinates; that is, in latitude (ϕ), longitude (λ) and height (H) over the ellipsoid defined by the WGS84 reference frame. This means that this information is already provided in a global coordinate system, so merging it with others should be eased. However, since movement equations are not easily expressed in such coordinates, there is a need to transform them into Cartesian coordinates, where the calculations are more manageable.

2.3.2.5.2 Transformations to system coordinates

In this case, the task is to transform the geodetic coordinates provided by the ADS-B ground stations into the system centre Cartesian coordinates where all the surveillance information will be fused. This process is like one undergone with radar measurements. Therefore, it follows a stepwise approach that takes advantage of several intermediate transformations that aims at simplifying the whole process. The steps are [ASTERIX Cat.017 App.A]:

1. From geodetic coordinates to geocentric Cartesian coordinates:

$$[\phi, \lambda, H] \rightarrow [X_G, Y_G, Z_G] \quad (2.53)$$

The equations are:

$$\begin{bmatrix} X_G \\ Y_G \\ Z_G \end{bmatrix} = \frac{a}{\sqrt{1 - e^2 \sin^2 \phi}} \begin{bmatrix} \cos \phi \cos \lambda \\ \cos \phi \sin \lambda \\ (1 - e^2) \sin \phi \end{bmatrix} + H \begin{bmatrix} \cos \phi \cos \lambda \\ \cos \phi \sin \lambda \\ \sin \phi \end{bmatrix} \quad (2.54)$$

Where:

- a : semi-major axis of the Earth reference ellipsoid (i.e. WGS84).
- e : eccentricity of the Earth reference ellipsoid.

2. From geocentric Cartesian coordinates to system Cartesian coordinates:

$$[X_G, Y_G, Z_G] \rightarrow [X_S, Y_S, Z_S] \quad (2.55)$$

This transformation means a change in the origin of the coordinate system and a rotation of the Cartesian axis. This is done by the following equations:

$$\begin{bmatrix} X_S \\ Y_S \\ Z_S \end{bmatrix} = R_{GS} \left(\begin{bmatrix} X_G \\ Y_G \\ Z_G \end{bmatrix} - T_S \right) \quad (2.56)$$

Where:

- $[\phi_S, \lambda_S, h_S]$: geodetic coordinates of the system origin (i.e. latitude, longitude and height).

- R_{GS} : rotation matrix, which converts vectors in the geocentric coordinate system into vectors in the system coordinates assuming both coordinate systems share the same centre. It has the following expression:

$$R_{GS} = \begin{bmatrix} -\sin \lambda_S & \cos \lambda_S & 0 \\ -\sin \phi_S \cos \lambda_S & -\sin \phi_S \sin \lambda_S & \cos \phi_S \\ \cos \phi_S \cos \lambda_S & \cos \phi_S \sin \lambda_S & \sin \phi_S \end{bmatrix} \quad (2.57)$$

- T_S : position of the system origin in the geocentric coordinate system.

$$T_S = \begin{bmatrix} (\eta_S + h_S) \cos \phi_R \cos \lambda_R \\ (\eta_S + h_S) \cos \phi_R \sin \lambda_R \\ (\eta_S(1 - e^2) + h_S) \sin \phi_R \end{bmatrix} \quad (2.58)$$

- η_S : normalized Earth radius at the latitude of the origin of the system coordinates.

$$\eta_S = \frac{a}{\sqrt{1 - e^2 \sin^2 \phi_S}} \quad (2.59)$$

3. From system Cartesian coordinates to system stereographic coordinates:

$$[X_S, Y_S, Z_S] \rightarrow [U, V, H] \quad (2.60)$$

The projection is described by the following expression:

$$[U, V, H] = \left[kX_S, kY_S, \frac{H_1(R_S + H_2) + d_{xy}^2}{R_S \sqrt{H_2^2 + d_{xy}^2}} \right] \quad (2.61)$$

Where:

- R_S : local radius of the Earth at the latitude (ϕ_S) of the system centre.

$$R_S = \frac{a(1 - e^2)}{\sqrt{(1 - e^2 \sin^2 \phi_S)^3}} \quad (2.62)$$

- k : constant with the following value:

$$k = \frac{1}{1 + \frac{1}{2} \frac{h_S + Z_S + H}{R_S}} \quad (2.63)$$

- H_1 : constant with the following value:

$$H_1 = Z_S + h_S \quad (2.64)$$

- H_2 : constant with the following value:

$$H_2 = R_S + H_1 \quad (2.65)$$

- d_{xy}^2 : constant with the following value:

$$d_{xy}^2 = X_S^2 + Y_S^2 \quad (2.66)$$

All the previous coordinate transformations and projections allow placing the target in a common reference frame where all the information coming from different sources can be compared and, if wanted, fused.

2.3.3 Automatic dependent surveillance - Contract

2.3.3.1 Features

As previously mentioned, while presenting ADS surveillance applications, ADS-C implies a point to point connection between the aircraft and a certain ground station usually referred to as ATSU for transmitting some already agreed aircraft information.

In this case, it is the ground control who decides about the information that is automatically downloaded from the aircraft by means of one or several contracts,¹¹⁹ which are established before any information exchange. In fact, these contracts can only be initiated by the ground control and the pilot is not allowed to modify them. These contracts also specify under what specific circumstances (except in emergencies) the aircraft must provide its on-board data. Thus, there are the following types of contracts:

- Periodical.
- On-demand.
- By event.
- Emergency.

This application is of special interest in oceanic regions or in those areas in which there is little or no radar coverage as a mean of complementing current surveillance information. Consequently, it is not very relevant in the airport domain. However, there has been noticed an increase in the expected use of ADS-C in this domain as there are some ASAS applications (e.g. ASPA-IM-S&M) in which it is considered that ADS-C is the appropriate way of sending some “Trajectory Management” information such as the “Aircraft Intent” which is provided in an ADS-C Extended Projected Profile (EPP) report.

ADS-C has to comply with "RTCA DO-212: Minimum Operational Performance Standard (MOPS) for Airborne ADS equipment" [RTCA DO-212].

2.3.3.2 Particularities

ADS-C is normally considered more a communication mean than a surveillance application, so most of its particularities are related to this fact.

Once the aircraft has logged on the application, it is required that a contract be established between the aircraft and the ground control before the latter starts receiving onboard data. Normally, the ground control establishes several contracts to monitor and control

¹¹⁹ A “contract”, in this context, is the result of a dynamic negotiation between the aircraft and the ground control previous to any onboard information exchange.

different aspects of the flight such as position, aircraft intent, identification, emergencies, etc. Additionally, the ground control also defines the frequency of the transmissions and their parameters.

There are currently two technologies which support ADS-C [Tabernero 07], namely:

- FANS-1/A, based on the ACARS data link.
- ADS-C ATN system.

The use of satellite communications makes ADS-C a quite expensive application. This fact has prevented ADS-C from being a popular communication link, and so not reaching scale economies. Nowadays with the advent of satellite-based ADS-B systems, like Aireon, the interest in this application might decay.

Furthermore, the use of a satellite link to send information to the ground control centre implies that there can be a not negligible latency in the communication.

2.3.3.3 Measurement model

Since ADS-C is not expected to be used for Airport Surface surveillance. This subsection has been omitted. However, the measurement model for ADS-B should be also valid for ADS-C.

2.3.3.4 Data format

The data format that is used for sending ADS-C reports to the ATSU depends on the communication technology that is implemented, which can be ACARS or one of the subnetworks of the ATN.

Additionally, EUROCONTROL has reserved a future ASTERIX category for sending ADS-C report, i.e. Cat.024. Currently, it is under definition and development.

2.4 Multilateration Systems

2.4.1 General considerations

2.4.1.1 Overall Features

Multilateration is a location technique based on measuring the difference in distance to two already known reference points (focus). In our case, these points are represented by the location of two “Ground stations” and the measurements represent the distance of a target to each of these ground stations. However, this will only provide a surface where the target is expected to be found, this is the reason why multilateration surveillance systems use at least three ground stations to compute the location of the target. These ground stations are placed in strategic locations within the airport, in the airport surrounding area and/or in wider areas that could cover a whole country as it is presented in [Langhans 13]. This fact will depend on the coverage of the multilateration (MLT) system.

These ground stations are listening to the signals which are sent by aircraft, generally in the 1090MHz band (SSR and Mode S replies, Mode S ES messages, ACAS transmissions, etc.). Every time the same signal is received at three or more ground stations, the position of the aircraft or airport vehicle can be extracted by applying quite advance computation techniques. This fact may happen several times per second, opposite to radar systems or ADS-B whose update rates do not reach such a high frequency. However, this may become a difficulty for data processing systems if not managed correctly.

The difference in distance, which is the key for computing the location of the target, is measured through the measurement of the time difference of arrival (TDOA) that is then converted into distance multiplying by the speed of electromagnetic signals (c). This is shown in the following expression:

$$\Delta d = c\Delta t = c(ToA_i - ToA_j) = c \cdot TDoA_{i,j} \quad \forall i,j \quad (2.67)$$

This process is usually performed at a Central Processing Station (CPS) that receives the time of arrival (TOA) of the same signal to the ground stations that have received it. This CPS processes these TOA measurements and obtains the TDOA values. The relevant fact is that the time when the signal was sent is unknown to the system. This is the reason why a multilateration technique is used, instead of other techniques such as trilateration¹²⁰ or triangulation.¹²¹

Each TDOA value defines a locus of points which comply with the premise of having the same time difference (or equivalently the same distance difference) to the two ground stations which provided the TOA measurements that were subtracted for obtaining the TDOA value. This locus of points is a well-known geometric figure, the hyperbola.¹²² The complication is that this geometric figure has two branches and infinite points that comply with the condition of presenting the same difference of distance. The first ambiguity can be solved just checking which TOA is bigger, and then choosing the branch that is closer to the other ground station, which provided the smaller TOA. However, to resolve the location of the point of interest, there is a need for introducing a new constraint in the system of equations to be resolved. This is usually done by introducing a new TDOA, which will define another hyperbola. If the position of the different ground stations is favourable, then the two hyperbolas will intersect defining the location of the point of interest. It could be possible that the geometry of these ground stations would be not favourable, causing a not "clean" intersection between the two hyperbolas. In such a situation, instead of having just a point as the location of the target, the system obtains a region where the target could be. This is not desirable as it introduces some uncertainty to the location.

The 2D problem just described, can be extended to 3D by just changing the geometric figure into a hyperboloid, which is a geometric surface in a 3D space. Consequently,

¹²⁰ Trilateration is another location technique based on the measurement of distances from the unknown position to several reference points. The implied calculations exploit the properties of circumferences, spheres, and triangles. This is the technique used by GPS.

¹²¹ Triangulation is another location technique that measures the angles of the unknown position with respect to several known points whose position and, which is more important, the distance between them is already known. Using the properties of triangles, the position to be located can be obtained.

¹²² A hyperbola is defined as the locus of points where the absolute value of the difference of the distances to the two foci is a constant equal to two times the distance between its two vertices.

obtaining the location of the point of interest in 3D requires a third TDOA (this fact implies a fourth TOA measurement) meaning that, at least, four ground stations must receive the signal transmitted by the aircraft or ground vehicle. In this case, the location of the target, aircraft or ground vehicle, will also include height.¹²³ This location will be the intersection of at least three hyperboloids. As each hyperboloid describes a geometric surface, then the intersection of two of them will define at least a curve, which will intersect the remaining hyperboloid in several points. Applying some additional constraints, this ambiguity can be solved, and a position obtained. Obviously, this description assumes that the location of the ground stations is good enough for not generating an ambiguous solution (e.g. an area or volume, instead of a point). As this is not always true, there are proposals for improving the location of the target of interest by introducing new constraints into the system of equations. An example is presented in [Miyazaki 14], where a measurement of the distance to the target is incorporated to remove the ambiguity. In this case, it is one of the ground stations the one that interrogates the target obtaining a measurement of the distance to it.

The first uses of MLT systems as a surveillance sensor were in the airport domain in the 1980's in the United States [Wood 00]. Although, it was initially developed by the military, the idea was rapidly extended to the civilian environment. Thus, by 1999 the FAA contracted a first system for the Dallas/Ft Worth (DFW) Airport. On the other hand, multilateration was used in Europe for height monitoring while analysing the possibility of reducing the vertical separation minima in the upper airspace (from FL290 to FL410) within the EUROCONTROL RVSM¹²⁴ project [RVSM]. The status of multilateration systems for extensive areas in Europe in 2014 was described in [Rekkas 14]. Since then, the number of multilateration systems has not stopped growing.

Depending on the area that is covered by the MLT system, there are two ways of referring to these systems. In case of limited areas, such as airports, the terms that are used are LAM (Local Area Multilateration) or MLAT; while in the case of large areas, the usual term is WAM (Wide Area Multilateration). There are several differences between both systems that are going to be outlined in the following sections. WAM systems are the natural substitute of SSR systems due to their performance and coverage. For the purpose of this thesis, the focus is going to be placed on the MLAT systems, which are the ones of interest in the airport environment.

2.4.1.2 Generic Measurement Model

These sensors have a complex measurement model and so, it is usually simplified when it is used in data fusion systems.

As for any other sensor, the main error sources are:

- Systematic errors.
- Some correlated errors.
- Random errors.

¹²³ This height is usually referred to as "geometric" height to distinguish it from the "barometric" height, which is the one provided by the transponder of the aircraft.

¹²⁴ RVSM stands for Reduced Vertical Separation Minima.

A detailed description can be found in [Trofimova 07], [Pourvoyeur 11] or [Abbud 15]. The systematic ones are like the ones identified in other sensors (such as the antenna position offset in ADS-B) or the synchronization errors. The correlated ones are closely related to propagations conditions (e.g. the speed of electromagnetic waves is less than the speed of light and depends on the traversed layers of the atmosphere). Both are either constant or slowly variant, so they can be estimated and then removed. However, it is assumed that any MLT system implements a self-calibration build-in logic which cancels synchronization and propagation errors.

Additionally, there are also random errors that cope with the jitter of the leading edge of the pulse, the quantization error of the processing station and a random component due to the selected time synchronization technology ("common time" or "distributed time"). All these terms are going to be treated all together. Thus, the measurement model for MLT position measures in Cartesian coordinates (X_j^m, Y_j^m) for the j -th aircraft can be assumed as:

$$X_j^m(k) = X_j(k) + \Delta l_j^x + w_j^x(k) \quad (2.68)$$

$$Y_j^m(k) = Y_j(k) + \Delta l_j^y + w_j^y(k) \quad (2.69)$$

Where:

- $X_j(k), Y_j(k)$: true position of the j -th target in Cartesian coordinates at time k .
- $\Delta l_j^x, \Delta l_j^y$: offset bias components that depend on the ATC antenna position and the orientation (α) of the j -th aircraft.
- $w_j^x(k), w_j^y(k)$: measurement noise errors for the x and y components corresponding to the position measurement for the j -th target at time k .

Δl_j^x and Δl_j^y represent the projection into both coordinates x and y of the offset bias due to the discrepancy between the ATC antenna location and the aircraft position reference point or centroid (Δl_j). Similar to the SSR and ADS-B cases, Δl_j can also be modelled as a uniform random variable between $-40m$ and $40m$ and its projected values in Cartesian coordinates are:

$$\Delta l_j^x = \Delta l_j \sin \alpha \quad (2.70)$$

$$\Delta l_j^y = \Delta l_j \cos \alpha \quad (2.71)$$

However, there are means for obtaining an estimation of this offset directly from the MLT target reports as already commented in previous sections. Moreover, for some aircraft manufacturers, this bias can become a known value ($\overline{\Delta l_{ATC}}$ as presented in section 6.10.7.3) that can be obtained from a database, if the surveillance system is able to obtain the "Aircraft Type" of the target. Both situations are addressed in section 6.10.7.3.

At last, $w_j^x(k)$ and $w_j^y(k)$ account for the random measurement errors which are generally modelled as zero-mean white Gaussian random sequences with corresponding variances σ_x^2 and σ_y^2 , and covariances σ_{xy} . These values are assumed constant (i.e. not

position dependent) within the area inside the perimeter defined by the farthest ground stations. Due to the complexity of the measurement model of such a sensor, these components group all the effects of the random errors that were identified in the beginning of this section. There are some error models that taking into consideration the location of the ground stations allow the computation of the "Dilution of Precision" (DOP), which provides a position dependent estimation of the magnitude of the measurement error. Some examples can be found in [Torrieri 84] and [Levanon 00]. Furthermore, the measurement errors can also be computed directly from the measurement as it is proposed in [Pourvoyeur 08].

As already said, there are some bias terms which are surely changing in time depending on, for example, propagation conditions that are assumed to be cancelled by built-in sensor calibration means, such as station synchronization procedures for multilateration (e.g. with reference transponders).

2.4.1.3 Data format

MLT ground stations provide their output (target reports or local tracks) in ASTERIX Cat.010 and Cat.020, while their sensor status and service messages are currently delivered in ASTERIX Cat.019, although they might also be received in Cat.025.

Following the same approach as for the previous sensor families, the raw data, i.e. the one directly measured by the sensor, is the one to be exploited. The concept of sensor in this case is broader as it refers to a set on ground stations that are connected to a Central Processing Station (CPS).

Like in section 2.2.1.3, the focus is going to be placed in identification, time and kinematic information. As before, all this information will be presented in the next tables:

Table 14: Identification and classification information within ASTERIX Cat.020

Identification information	Data Item	Description
Identification		
Mode A code	I020/070	<i>Mode 3/A Code in Octal Representation</i> Mode 3A code converted into octal representation.
ICAO 24-bit address	I020/220	<i>Target Address</i> Target address (ICAO 24-bit address) assigned uniquely to each Target.
Local track number	I020/161	<i>Track Number</i> An integer value representing a unique reference to a track record within a particular track file.
Callsign	I020/245	<i>Target Identification</i> Target (aircraft or vehicle) identification in 8 characters.

Identification information	Data Item	Description
Classification		
Mobile Category	I020/300	<i>Vehicle Fleet Identification</i> Vehicle fleet identification number.

The table above collects the data items from ASTERIX Cat.020, which deals with identification and classification information. This information is used for data association purposes.

The data items concerning time that are going to be taken into consideration are:

Table 15: Time information within ASTERIX Cat.020

Time information	Data Item	Description
Time of applicability	I020/140	<i>Time of Day</i> Absolute time stamping expressed as UTC. LSB: 1/128s

All these time measurements are reset to 0 every midnight.

Like Mode S radars working in EHS and ADS-B, kinematic information is also provided in several data items. It includes position information, although some velocity information can be provided in case of having a local tracker. Additionally, the Reserved Expansion Field also offers the possibility of sending velocity information by means of the “Ground Velocity Vector” and its associated “Ground Velocity Accuracy” data field [ASTERIX Cat.020].

The position information can be provided either in geodetic coordinates or in local Cartesian ones. This fact implies that a certain coordinate transformation needs to be applied as it is outlined in section 2.4.1.4. The position information and the data items where it is provided are presented in the following table:

Table 16: Target information within ASTERIX Cat.020

Target information	Data Item	Description
2D position		
Position in geodetic coordinates	I020/041	<i>Position in WGS-84 Coordinates</i> Position of a target in WGS-84 Coordinates. LSB Latitude: 180/2 ²⁵ degrees LSB Longitude: 180/2 ²⁵ degrees

Target information	Data Item	Description
Position in Cartesian coordinates	I020/042	<i>Position in Cartesian Coordinates</i> Calculated position in Cartesian Coordinates, in two's complement representation. LSB X-component: 0.5m LSB Y-component: 0.5m
	Height	
Barometric height	I020/090	<i>Flight Level</i> Flight Level (Mode S Altitude) converted into binary two's complement representation. LSB: ¼ FL, i.e. 25ft
	I020/100	<i>Mode-C Code</i> Mode-C height in Gray notation as received from the transponder together with the confidence level for each reply bit as provided by a MSSR/Mode-S station.
Geometric height	I020/105	<i>Geometric Height (WGS-84)</i> Vertical distance between the target and the projection of its position on the earth's ellipsoid, as defined by WGS84, in two's complement form. LSB: 6.25ft
	I020/110	<i>Measured Height (Local Cartesian Coordinates)</i> Height above local 2D co-ordinate system in reference to the MLT System Reference Point as defined in item I019/610, in two's complement form, based on a direct measurement not related to barometric pressure. LSB: 6.25ft

It is relevant to point out that in the airport domain, MLT target reports might provide either item I020/041 or item I020/042. This fact implies that there is a need for transforming them into the coordinate system of the data fusion centre. This process will be detailed in the next section.

Concerning height, two values are provided:

- The barometric altitude, in the form of Mode C/Flight Level values.
- The geometric height, which is computed as the minimum height from a plane tangent to the reference Earth's ellipsoid or to the local coordinate system.

Similar to radars, MLT systems also offer the possibility of sending local tracking information to the ATC centre where the data is going to be fused. This information can be used for initiating tracks, integrity checks and confirmation purposes.

Table 17: Ground referenced velocity information within ASTERIX Cat.020

Download information	Data Item	Description
Velocity		
Velocity in polar coordinates	I020/REF (GVV)	<p><i>Ground Velocity Vector</i></p> <p>Ground Speed and Track Angle elements of Ground Velocity Vector.</p> <p>LSB Ground speed: 2^{-14} NM/s (referenced to WGS-84).</p> <p>LSB Track angle: $360/2^{16}$ degrees (clockwise reference to “True North”).</p>

The MLT system also provides information about the quality of its measurements or calculations. This is stated by means of three indicators, namely:

- The Dilution of Precision of position (DOP).
- The Standard Deviation of Position (SDP).
- The Standard Deviation of Geometric Height (SDGH).

As their names anticipate, the first two ones refer to the quality of the position and the latter to the one of the height. Additionally, the Cat.021 standard also offers another quality indicator within the Reserved Expansion Field (REF) which is considered more complete as it provides the Standard Deviation of Position in WGS84. Currently, in systems like ARTAS, the values within the I020/REF are privileged over the ones coming in I020/500 as they have empirically shown a more consistent value.

The PET will consider all these values to generate the corresponding approximated standard deviation values or covariance values to be introduced in the data fusion process in charge of generating the reconstructed trajectory. This information complements position and velocity estimates with the inherent uncertainty of the measurement process, which usually takes the form of a covariance matrix. This step will be detailed as part of the data acquisition which will be described in the beginning of CHAPTER 5.

These data items are:

Table 18: Quality information within ASTERIX Cat.020

Quality information	Data Item	Description
Position Quality Indicators	I020/500 (#1, #2)	<p><i>Position Accuracy</i></p> <p>Subfield #1: DOP of Position</p> <p>LSB DOP along X axis (DOP_x): 0.25</p> <p>LSB DOP along Y axis (DOP_y): 0.25</p> <p>LSB Correlation coefficient of XY (DOP_{xy}): 0.25</p> <p>Subfield #2: Standard Deviation of Position</p> <p>LSB Standard deviation of X (σ_x): 0.25m</p> <p>LSB Standard deviation of Y (σ_y): 0.25m</p> <p>LSB Correlation coefficient of XY ($\rho(X, Y)$): 0.25m²</p>
	I020/REF (PA)	<p><i>Position Accuracy</i></p> <p>Subfield #1: DOP of Position</p> <p>LSB DOP along X axis (DOP_x): 0.25</p> <p>LSB DOP along Y axis (DOP_y): 0.25</p> <p>LSB Covariance Component (DOP_{xy}): 0.25</p> <p>Subfield #2: Standard Deviation of Position (Cartesian)</p> <p>LSB Standard deviation of X (SDC_x): 0.25m</p> <p>LSB Standard deviation of Y (SDC_y): 0.25m</p> <p>LSB Covariance Component ($COV-XY$): 0.25m</p> <p>Subfield #4: Standard Deviation of Position (WGS-84)</p> <p>LSB Standard deviation of X (SDW_x): 180/ 2²⁵ degrees</p> <p>LSB Standard deviation of Y (SDW_y): 180/ 2²⁵ degrees</p> <p>LSB Covariance Component ($COV-WGS$): 180/ 2²⁵ degrees</p>
Height Quality Indicators	I020/500 (#3)	<p><i>Position Accuracy</i></p> <p>Subfield #3: Standard Deviation of Geometric Height (WGS 84)</p> <p>LSB Standard deviation of Geometric Height (σ_{GH}): 0.5m</p>

Quality information	Data Item	Description
	I020/REF (PA)	<i>Position Accuracy</i> Subfield #3: Standard Deviation of Geometric Height (WGS 84) LSB Standard deviation of Geometric Height (<i>SDH</i>): 1ft
Velocity Quality Indicators	I020/REF (GVA)	<i>Ground Velocity Accuracy</i> Accuracy of the Ground Speed and Track Angle elements of Ground Velocity Vector. LSB Standard deviation of the Ground Speed: 2^{-14} NM/s LSB Standard deviation of the Track Angle: $360/2^{16}$ degrees

For more details about these data formats please go to ASTERIX website (<https://www.eurocontrol.int/asterix>).

2.4.1.4 Coordinate transformations

Two coordinates systems are supported: a local Cartesian and a geodetic coordinate system. For the sake of clarity, the coordinate conversion equations are not going to be repeated in this section, as they can be checked in the corresponding section for Radar systems (ref. 2.2.1.4.2) and for ADS-B systems (ref. 2.3.2.5.2). Only a quick summary of the steps to be followed is presented.

When the position information is provided in the local coordinate system, the steps to follow are the computation of the rotation matrix and the translation vector in order to be able to transform the coordinates in the local system into coordinates into the coordinate system of the data fusion centre. An example of the equations to be used can be found in point 4 “From geocentric Cartesian coordinates to system Cartesian coordinates” of section 2.2.1.4.2. If there is a need for performing a stereographic projection, the equations are in point 5 of the same section.

On the other hand, when the information is provided in geodetic coordinates, the process coincides with the one described for ADS-B position information, which can be seen in section 2.3.2.5.2.

2.4.2 LAM or MLAT

2.4.2.1 Features

As already mentioned, the coverage area of a LAM or MLAT system is the airport and sometimes its surrounding area [Galati 05][Mantilla 15]. It usually consists of 10 to 15 ground stations and a CPS that sends the surveillance information to the surveillance function of an A-SMGCS system [EUROCAE ED117], to a tracking system (in case no A-SMGCS is present) or to an SDP in case that the information is also exploited by the air surveillance.

These systems rely on the reception of spurious emissions sent by aircraft, while approaching, landing, departing or taxiing around the airport, to several ground stations located all around the airport terrains. These emissions are usually received in the 1090MHz band (SSR or Mode S replies, Mode S ES messages, ACAS¹²⁵ transmissions, etc.), which is heavily congested. In case that MLAT ground stations do not receive enough signals for computing the location of the target, there is a possibility of including interrogating ground stations that will force the aircraft to generate answers. In this case, the MLAT sensor would be active instead of passive.

Although the system can provide 3D measurements, computing the height by itself, for the purposes of the surveillance system of the airport it is not mandatory. Thus, it can just provide 2D measurements, which eases the calculations. When the measurements provided by the MLAT system are intended also to the air surveillance, then the 3D measurements are desired.

2.4.2.2 Particularities

Due to the fact that MLAT systems are mainly passive sensors, the number of spurious signals that are received at the ground stations has to be taken into account as it could be affected by local procedures, which can establish that some airborne equipment has to be switched off once the aircraft has landed. In such a situation, it is always possible to force the airborne system to respond to some direct interrogations.

Usually, MLAT systems limit the number of position updates that are sent to the surveillance tracking system in order not to overload it. Tracking systems do not deal very well with high data rates as their tracking algorithms tend to diverge if not conveniently protected. So, in such a case, it is more than recommended that the CPS limits the position updates to not more than one or two per second, as the SMR or the ADS-B.

Like SSR and ADS-B, MLAT systems also suffer from target-dependent biases. Therefore, the position reported by the MLAT sensor corresponds to the position of the antenna whose signal is processed by the ground stations to obtain the location of the target. This is usually the ATC antenna, which is the one used for sending SSR and Mode S replies, but there can be others since there are several multilateration types as the ASTERIX data format already foresees within I020/020 Target Report Descriptor.

Another problematic issue is that the MLAT sensor computes the local geometric height¹²⁶ although it has access to the barometric one, as it can decode the information contained in the messages (e.g. the Mode A/C or Mode S replies) that are used for calculating the position of the target. It is fundamental to handle with care these two different pieces of information as a bias in the aircraft location might be introduced. This mainly happens while approaching to or departing from the airport. The difference between the two heights is usually referred to as the QNH correction (ref. 2.6.4) and, if possible, it should be estimated and corrected from the barometric altitude measurements.

¹²⁵ In medium/high density airports, ACAS is not usually activated in the ground due to the great number of aircraft that can be close together and that can generate many false alarms, i.e. traffic advisories (TA) or resolution advisories (RA).

¹²⁶ In the ASTERIX Cat.020, the sensor can specify both, the local geometric height considering its local reference frame, and the geometric height in WGS84.

The position calculated by MLAT systems is generally provided in Cartesian coordinates referred to a reference frame fixed in the ARP, although it can be converted in the CPS into geodetic ones.

2.4.2.3 Measurement Model

There are no particularities to be added to the generic measurement model presented in section 2.4.1.2. So, the same equations are valid for modelling LAM or MLAT systems.

2.4.2.4 Data Format

These sensors provide their output and their status in the following ASTERIX categories:

- Target reports: ASTERIX Cat.010 and Cat.020.
- Sensor status and service messages: ASTERIX Cat. 019.

For more details about these data formats please go to section 2.1.5.2.

2.4.3 Wide Area Multilateration

2.4.3.1 Features

The features of a WAM system are roughly the same as for a MLAT system. The infrastructure is the same, that is, several ground stations and a CPS, although the range is much bigger [EUROCAE ED142]. As already mentioned, they are foreseen as the natural substitutes of current SSR systems. The number of ground stations depends on the area to be covered. Therefore, it can reach several tens of ground stations, if the aim is to cover a whole country as in [Langhans 13] and [Auer 14].

The signals to be used for obtaining the TOA measurements are the same ones (i.e. SSR and Mode S replies, Mode S ES messages, ACAS transmissions, etc.) and the position is provided in geodetic coordinate although it can also be provided in Cartesian ones.

2.4.3.2 Particularities

The particularities are also shared with an MLAT system. Consequently, it also suffers the problem of having a big number of signals to be used to compute the location of the target. However, the difficulty of not having enough signals is by far not as critical as it can be in the airport environment due to the mandate of carrying ADS-B capable transponders and the continuous use of ACAS, which might not be the case at airports.

2.4.3.3 Measurement Model

The measurement model is affected by the same troubles that were outlined when talking about MLAT sensors. In WAM systems, the most relevant errors are the ones caused by tropospheric refraction, changes in the signal speed, and synchronization issues. These effects are addressed in works like [Abbud 15], while the challenge of integrating WAM data in an ATM system architecture is shown in [Auer 14]. The antenna position offset is also present in this sensor, but it is not as critical as in its airport counterpart.

For these systems, it is quite usual to obtain maps of the DOP to be used when fusing WAM data with the rest of the information coming from other sensors.

2.4.3.4 Data Format

This sensor provides its output and its status in the following ASTERIX categories:

- Target reports: ASTERIX Cat.020 although it can also be provided in Cat.010.
- Sensor status and service messages: ASTERIX Cat.019.

For more details about these data formats, please go to section 2.1.5.2.

2.5 Independent and Non-Cooperative Systems

2.5.1 General considerations

2.5.1.1 Overall Features

The main features of these systems are the ones highlighted in the acronym that groups them. As such, they are independent and so, they compute, by their own means, the location of the target. On the other side, they are non-cooperatives, meaning that they do not expect any collaboration from the target. These two features make them very interesting sensors for the airport environment as they guarantee a detection of the target even in the event of losing its communication capability or under other adverse conditions, such as the lack of visibility. Some examples were already presented in section 1.3.1.4.

2.5.1.2 Generic Measurement Model

As already mentioned in CHAPTER 1, the difficulty with these systems is that on one side, there is not yet much empirical experience gained with them and, as such, there are not mature enough measurement models and, on the other side, the availability of recordings for research purposes are not easy to obtain, which makes even more difficult reaching the needed experience. So, the proposal is to try to adapt one of the previously presented models to each of these sensors. Most of them are based on radar technologies, so the generic model presented in section 2.2.1.2 could be the starting point for modelling their behaviour. If this is not appropriate, the MLT measurement model is another possibility to try by just removing the target-based bias, see eqns. 2.68 and 2.69.

2.5.1.3 Data format

This sensor provides its output and its status in the following ASTERIX categories:

- Target reports: ASTERIX Cat.015.
- Sensor status and service messages: ASTERIX Cat.016.

For more details about these data formats, please go to section 2.1.5.2.

2.5.1.4 Coordinate transformations

As in the case of the previous sensor, the coordinate transformations that might be needed will fall in any of the already presented ones as the information is expected either in local Cartesian coordinates or in geodetic coordinates.

2.6 Already known issues with surveillance systems

2.6.1 Time offset

The lack of a common time reference and the presence of different timestamping techniques introduce a variable time offset between the time of applicability of measurements coming from different sensors, which requires some type of alignment. This time offset is expected to be present in all measurement coming from any sensor except in the one whose time reference is the one assumed as the common reference. Its value is assumed to be almost constant as it is justified by a lack of synchronization between relatively accurate time references. The estimation of this time bias will be one of the objectives when registering the contributing sensor to the data fusion process.

The impact of this time bias is translated into an along trajectory error as it was presented for radar positions in section 2.2.1.2, and as such it can be seen as a timestamp induced bias.

2.6.2 Data processing issues

The surveillance function is fed with data coming from several different sensors, each one with its own characteristics as it has been presented in the previous sections. These data are affected by issues such as outliers, blunders, low probability of detection, false alarms, gaps, missing values, out of sequence messages, etc.

Outliers are measurements which do not follow the general measurement error model of the sensor which has generated them. These measurements fall outside of the probability density function (pdf) that characterizes the regular behaviour of the sensor. They usually mean two things: either a trouble while measuring or that the pdf chosen for the sensor does not represents reasonably the observations meaning that a more heavily tailed distribution should be chosen.

Blunders represents the result of performing a mistake in any of the actions that form part of the measurement process and, as such, it cannot be modelled in magnitude, although its frequency might be estimated. The objective with these issues is to detect and remove them from the subsequent processing.

Surveillance sensors are also characterized by two quality indicators which are the "Probability of Detection" (PD) and the "Probability of False Alarm" (PFA). The former gives an idea of how often the sensor detects the target when it should, and the latter the other way round, the number of times that the sensor detects a target when it should not. Ideally, the value of the PD should be 100% and the PFA the 0%, however this barely happens. So, it is quite common that sensors provide data relative to a certain target which have gaps, that is, some not detections among the detections. This means that the

consumer system (in our case, the surveillance tracker or the performance evaluation tool) does not receive a piece of data when it should and, more important, when it expects it. Gaps are frequent, as there are many issues that can cause them, and so, the system in charge of processing surveillance data should know how to cope with them. When gaps are very long, they can cause the loss of tracks, which is something to be avoided. This is not always the case as targets are usually detected by several sensors (under multisensor coverage) and so the effect of gaps can be minimized. When gaps appear, the sensor can follow three different behaviours:

- Doing nothing, so the consumer will realize that there has been a loss in the data.
- Alerting the consumer about non-detections.
- Extrapolating the previous detection to the position where it supposes the target would be (like a coasting track in a tracking system).

Another issue that tracking systems must deal with is missing values. This happens when a target report sent by the sensor does not contain all the values. In this case, the consumer system receives just part of the expected information. Thus, it may not receive height in a position update, it may receive velocity information without position, or may receive the position information but without identification (e.g. Mode 3A or 24-bit address), etc. If the tracking system is able to associate this information with the one available in any of the already initiated tracks, it can exploit it. Otherwise, it can either disregard it or use it for creating a new track. This will depend on the configuration of the tracking system.

Finally, the amount of data to be processed could also be a trouble if it causes the system to overload. This issue has a huge impact in the algorithms and hardware to be used in the calculations. However, nowadays most of the systems include protections that basically throw away data when a certain computer load is reached. On the sensor side, there are now less data driven modes and more periodic ones, which regulates the amount of data that is sent and prevent the saturation of the receiving system.

2.6.3 Different reference points

This matter has been introduced in the measurement models of the different sensors that have been presented. It refers to the fact each sensor provides the position of the target taking into consideration a different reference point within it. Thus, the position calculated by an SSR system is the position of the antenna used for sending the replies. In the case of the SMR, it provides “in mean” the centroid of the target, as it depends on the algorithms for estimating the position based on the different blobs that form the radar image of the target. ADS-B provides, unless corrected by the navigation system, the position of the GPS antenna, which is used for calculating the global position of the target. Finally, the position computed by a MLT system coincides with the position of the antenna sending the message that has been used for calculating the position.

As one can see, each reference point is in a different location of the target which can be several tens of meters apart from each other. Although this is not a big issue for air surveillance, for surface surveillance it is not the same; as an error of less than 7.5m is requested for the 95% of the time [ICAO 9830]. So, referring all the measurements to the same point is almost mandatory, even more if there are functions, such as the safety nets,

that use them as an input. A poor accuracy will cause the appearance of false alarms which are totally undesirable as they will prevent the controller to rely on the supporting tool.

This problem represents a target-dependent bias, which should be removed before feeding the tracking algorithms or the reconstruction ones with such data. As any data fusion system, the trajectory reconstruction process should be fed with unbiased data so a bias estimation and further cancellation step must be introduced in order to comply with this requirement. This step is part of what is called data pre-processing. These activities are described and some bias estimation algorithms presented in CHAPTER 6.

2.6.4 Dealing with different types of height measurements

When talking about height, there is also the question of which one to use, as different surveillance sensors provide different types of heights. Thus, the SSR provides a barometric altitude (Mode C) which is measured by the onboard sensors of the aircraft and it refers to the Mean Sea Level (MSL). These sensors measure the barometric pressure which is then converted into height assuming the standard pressure (1013,25hPa) with a resolution of hundreds of feet and presented as Flight Levels (FL340 $\equiv$ 34000ft). When flying above the transition altitude,¹²⁷ aircraft use the Mode C for vertical separation. Otherwise, the pilot needs the QNH or QFE¹²⁸ for obtaining an absolute measure of the height to separate from other aircraft but mainly from the terrain.

The issue comes when there are other surveillance sensors, like WAM, that provide the geometric height,¹²⁹ which can be with respect to an ellipsoid (usually the WGS84) or to the local stereographic plane. Fusing both types of heights, barometric and geometric, will cause the system to diverge unless both measurements could be aligned by means of estimating and then removing the offset between them, which is usually refer to as the QNH correction. This is another bias cancellation processing.

2.6.5 ADS-B known issues

As already mentioned, ADS-B is considered by many people the cornerstone in the change process that the ATM world is undergoing nowadays. However, it has several known issues [Soto 11], which are important to bear in mind in order to conveniently tackle them.

Thus, ADS-B is currently mandatory in parts of the Australian airspace, in Europe there is a mandate to equip some types of aircraft since June 2020 and, in the U.S., it is required that some types of aircraft to be equipped since January 2020. This fact implies a transition period in which the full benefit of pure ADS-B cannot be realized.

¹²⁷ The transition altitude is the altitude above the MSL at which the aircraft change from the use of altitude to the use of flight levels.

¹²⁸ The QFE is an aeronautical Q code, whose common mnemonic “Query: Field Elevation”, that represents the altimeter setting that will cause the altimeter to read the height above a specific aerodrome or ground level, and therefore read zero on landing, irrespective of the temperature. In the world, QFE is given in hectopascals or millibars, while in North America, QFE is given in hundredths of inches of mercury [Wikipedia].

¹²⁹ Although they can also provide the barometric altitude because it is in the payload of the messages that they use for computing the position of the target.

ADS-B messages are not time stamped, so there is some uncertainty concerning the time of applicability of the information they carry, which comes from the onboard flight systems of the aircraft. This fact must be considered when processing this information. Nowadays, there are some aircraft in which this information is UTC synchronized, which is stated by setting the T-bit to 1 within the position squitter, see [ICAO 9871]. However, depending on the delay introduced by the transponder, this issue can remain,¹³⁰ although it would be slightly mitigated.

The error introduced due to coordinate transformation is not relevant as currently there are algorithms that keep it almost negligible. However, the main bias that could appear is the antenna position offset, which can be automatically cancelled by the navigation system of some aircraft, or more recently, it is possible to provide its approximated value as one of the ADD that ADS-B can broadcast, i.e. the GPS Antenna Offset [ICAO 9871]. If this is not the case, it should be estimated and then cancelled. This issue will be presented in CHAPTER 6 and some estimation strategies will be proposed.

Another trouble that affects ADS-B is the one related to the transmission problems that causes the disappearance of ADS-B messages. This means that the PD is less than the unity. The congestion in the transmitting frequency, the presence of buildings and other vehicles, and the transmission equipment explain the loss of ADS-B messages. This trouble forces the data processing systems to implement countermeasures to deal with the absence of these messages, such as track coasting.

However, the most worrying issues are the ones related to integrity and security. Thus, due to the cooperative dependent nature of the ADS-B application, nothing prevents the transponder or the navigation system from providing inappropriate data to the ground and, what is more dangerous, this information can pass the integrity checks as it will comply with the CRC code that it includes.

Additionally, there is also an important security concern as the ADS-B signal is not protected by any encryption code or secured protocol. Therefore, it is not difficult to introduce non-existent aircraft into the situation awareness picture, which is used by the controllers, by means of spoofing ADS-B messages. Even more, the information is available to anyone with the appropriate receiver, so there is no restriction to the access to these data, which represents a high risk if it is gathered by the wrong people.

These issues explain why there is still some reluctance, as is expressed in [Pourvoyeur 11], to use ADS-B for operational purposes and the appearance of many proposals to cope with them in the next versions of the ATC systems both for the air [De Miguel 08] and for the ground [Mathias 08] or [Soto 11].

2.7 The future of surveillance

Concerning ground-based surveillance, it is fundamental to make the distinction between air surveillance and surface surveillance as they share several aspects, but they

¹³⁰ This is the case when the delay introduced by the systems in charge of transmitting this information is bigger than 0.2s as there is no way to know whether the information was obtained in the last update period of in any of the previous one. This issue introduces a requirement into the transmitting chain.

differentiate in many others. The former covers the en-route and terminal areas, while the latter copes with airports (runways, taxiways and aprons) and their surrounding areas.

When focussing on air surveillance, it looks like quite feasible that in the next ten to fifteen years, it will continue being dominated by radars, which are evolving from conventional SSRs to MSSR and Mode S radars. In some difficult access areas, such as mountains, they might be substituted or complemented by WAM systems. It is expected that after a transition period, which would last around ten years, (ground-based or space-based) 1090 ES ADS-B will take the lead in combination with independent surveillance sources, such as the provided by MSSR, Mode S radars and WAM systems. The author's vision foresees that although the independent surveillance will always exist, either as we currently know it or may be provided by other means, like MSPSR or other members of the INCS family; in the future, the dependent surveillance will be the primary means of surveillance in a medium-term horizon. It is expected that actual data links (1090ES, VDL mode 2 and 4, UAT, etc.) will be substituted by more powerful and resilient ones that will provide bigger throughputs, reduced delays, and more immunity to transmission problems. These features will allow the application of current security measures in the communications between aircraft and ATC centres. The L-band Digital Aeronautical Communication System (LDACS) and its future improved versions are expected to lead this paradigm change as it is presented in [Schnell 14].

Conversely, surface surveillance will undergo a much quicker transformation as the installation of the new systems is easier, has a limited impact, and supposes a clear reduction in maintenance and commissioning costs; and, what is more relevant, they are the foundation layer for improved A-SMGCS systems that will boost the integrated airport operations. Thus, ADS-B and MLAT systems will extend in the short term as a locally optimised mix together with at least one non-cooperative sensor, such as the SMR or maybe a system from the new INCS family. It is believed that non-cooperative sensors will remain as a support mean for those cases in which no collaborative mean of location and identification is available. The high pressure, which is expected over the surveillance function to support the evolution of the A-SMGCS functions toward level III and IV, i.e. consolidating safety nets use and introducing routing and guidance, will force a change in the data link technology to be used for sending ADS-B messages within the airport. Like in the previous case, the author believes that there will be a transition to a new data link which will cover the airport and that will solve all the drawbacks that are known today. Such a data link is called AeroMACS, already presented in CHAPTER 1, which has been defined within SESAR and NextGen programmes. The availability of a data link with high throughput, reduced latency and good propagation features will allow the incorporation of many already available Internet techniques and protocols to airport communications. Furthermore, it would allow sharing a common situation awareness view with all the mobiles that can be found around the airport. This way all of them could have a consolidated view of such information in a suitable display that could take, for example, the form of a moving map with all the surrounding vehicles and aircraft. The possibility of sending ADS-B information through a channel with security, integrity and continuity measures is more than appealing, if the target is reaching A-SMGCS level IV.

After pointing out the importance of the surface surveillance in the evolution of the ATC in both domains, i.e. air and surface, it seems that it is more than necessary to count on a mean for evaluating the performance of such a crucial function. Even more if we consider the current performance-based environment in which the ATM world wants to be

converted. This is the reason why performance evaluation tools are of paramount importance as external and independent evaluators of the behaviour of ground-based surveillance.

CHAPTER 3

PERFORMANCE EVALUATION TOOLS FOR SENSORS AND TRACKING SYSTEMS.

“Giants are not what we think they are. The same qualities that appear to give them strength are often the sources of great weakness.”

— **Malcolm Gladwell, David and Goliath: Underdogs, Misfits, and the Art of Battling Giants.**

Due to the criticality and importance of the surveillance data in current ATC environments, a continuous performance assessment is more than needed. Not only of the surveillance sensor performance, whose output will propagate all through the surveillance chain, but also of any other processing system that manipulates it, like Surveillance Data Processors (SDP) systems, as their outcome will finally end on the console of the ATC controller.

Consequently, a low-quality input or a degraded outcome not only imply a degradation of the situational awareness picture to be presented to the controller, but also affects other systems which take the surveillance information as one of their inputs. Some examples of such systems are: the Trajectory Prediction (TP) module, the Conformance Monitoring (CMON) tool, the Short Term Collision Avoidance (STCA) tool, any A-SMGCS routing function, airport safety nets (i.e. A-SMGCS control function) or the Medium Term Conflict Detection (MTCD) system. Therefore, sensor and tracker performance analysis are a must.

3.1 Introduction to performance evaluation

3.1.1 A little bit of history. From mid-80's to nowadays

In the late 80's, the Radar Surveillance Specialist Panel (RSSP) detected the need for standard assessment tools to evaluate the performance of the surveillance infrastructure. This was a moment in which the economic integration of several countries boosted the exchange of information between different ANSPs. Due to the dissimilar quality of surveillance information, there was a need for assessing the data quality before the incorporation of such data into the surveillance chain of the neighbour ANSP.

The target that was pursued was to complete a set of radar performance evaluation tools, as in that moment the available ones were mainly ad-hoc tools which did not provide comparable results (performance indicators or figures of merit) and did not use commonly accepted methods. This was the seed of the current SASS-C toolbox [SASS-C].

3.1.2 Definition of performance evaluation tools

“Performance Evaluation Tools” (PET) is a generic term to refer to all those systems that, applying a determined evaluation criterion, are able to obtain a number that gives an idea of how well the system is performing. This number is usually called a performance indicator (PI) or a figure of merit (FOM). Usually, these tools provide more than one of these numbers.

In, [Besada 13] there is a list of different aspects that are likely to need the definition of metrics in order to extract some performance indicators. The elements of such a list are:

- Sensor probability of detection.
- Sensor measuring error.
- Monosensor and multisensor track initiation.
- Data association.
- Tracking accuracy.
- Tracking robustness.
- Mode of flight (MoF) detection.
- Timeliness in the kinematic status information.

It is crucial to bear in mind that it is not easy to find a way for defining such a metric as some of these aspects are very complex to evaluate. Although the list is tuned to en-route and TMA environments, it can be easily extended to cover the airport just considering "surface movement" detection instead of just modes of flight.

This thesis concentrates its efforts in a subset of performance evaluation tools that have in common the need for building a reference to be compared to the actual information directly measured by sensors or computed by trackers to obtain the appropriate performance indicators when focussed on the airport domain. Even more, the scope does not cover all such tools but those which obtain the reference from what is called “opportunity traffic”.

3.1.3 Objectives of performance evaluation tools

The main objective of performance evaluation tools is to perform an automatic periodical assessment of the output of some critical elements of the surveillance chain to avoid the propagation of the effects of any malfunctioning system in an economic, efficient, and complete way. More in detail, this evaluation is intended to monitor, assess, and control the performances of any operationally running sensor or tracking system to detect and alert about any performance degradation that might affect any downstream system.

The cornerstone to achieve this objective is the computation of the actual mobile trajectory with a high level of accuracy not only concerning position but also time. This information is also called the mobile “reference trajectory”, the “ideal trajectory” or the “ground truth” as it will be the reference to which the actual output of sensors and tracking systems will be compared to. The result of this comparison will help in the monitoring and assessment of the performance of the surveillance chain by means of the calculation of several figures of merit.

Besides, these tools are also used for:

- Checking the performances before commissioning a new surveillance sensor or tracking system to certify their compliance to nominal performance.
- Supporting air incident investigation.
- Contributing to research and investment activities concerning sensors and tracking systems.

Moreover, there are other possible uses such as the comparison of surface planned trajectories obtained from movement plans to the reference trajectories calculated by processing real data aiming at evaluating how close they are. This way some figures of merit can be obtained about the predictability and reliability of airport planned routes.

Other application based on the reference trajectory processing is to classify the different aircraft ground movements depending on the aircraft type aiming at obtaining statistics about the most common surface movements performed by a certain type of aircraft and their features (e.g. mean/maximum velocity, mean turn rate, etc.).

3.2 Performance evaluation standards

The evolution of the performance evaluation standards is linked to the progressive adoption of performance-based operations whose objective is reaching a certain desired result by means of setting performance objectives keeping a systematic approach to safety

aspects. Besides, it leads to informed decision making based on the desired results and relies mainly on facts and data which need to be periodically assessed by performance measurements. This approach is technology agnostic although it should be kept interoperable, should allow a flexible architecture adaptable to different needs and should lead to the optimization of performance and the reduction of costs [ICAO 9883].

This strategy is applied to the three pillar the compose the ATM, that is communications, navigation, and surveillance (CNS) and it is closely linked to the Required Performance concepts defined by ICAO, (i.e. RCP, RNP and RSP). It aims at achieving strategic objectives on key areas such as safety, capacity, efficiency, and environment. Like the objectives set for SESAR.

Focussing on performance-based surveillance (PBS), it defines performance requirements for air traffic service provision to be allocated to surveillance system components as it is the intention with GEN-SUR SPR, which is led by the EUROCAE WG-102, and that it is currently in consultation phase. It is worth noting that there is a tendency to address PBS together with PBC, leading to the concept PBCS. The latter is the subject of ICAO Performance-Based Communication and Surveillance (PBCS) Manual [ICAO 9869].

The European way towards PBS, started back in the late 90's when the EUROCONTROL Standard Document for Radar Surveillance in En-Route Airspace and Major Terminal Areas was published [Blue book]. It defined performance requirements for the surveillance infrastructure based upon the capabilities of the sensors at that time, which were mainly radars. This document provided the criteria on which performance values should be requested to surveillance sensors and SDP systems However, it was not focussed on safety but on guaranteeing continuity and reliability.

Next, in 2012 (v1.0) and later in 2015 (v1.1), EUROCONTROL released ESASSP [ESASSP 12][ESASSP 15], which is the EUROCONTROL Specification for ATM Surveillance System Performance. This specification was devoted to the definition of performance requirements for ATM surveillance systems and how the associated conformity assessment must be performed. This way the document is independent of technologies, it is application driven (3 and 5 NM horizontal separation) and it does not impose specific architectures. It accommodates new surveillance sensor like Mode S, ADS-B, WAM, etc.

Finally, the already mentioned GEN-SUR SPR [GEN-SUR SPR] is expected to provide a top-down methodology to reach the PBS concept. So, from separation requirements like in ESASSP, it derives generic surveillance requirements which are allocated to specific surveillance components. The specification also derives the evaluation methods and guides towards the implementation of the assessment tools. All in all, for reaching surveillance standards which should be introduced into the corresponding ICAO documents.

3.3 Taxonomy of performance evaluation tools. Application domains. State of the art

3.3.1 Types of performance evaluation tools

There are different ways of classifying performance evaluation tools for sensor/tracker analysis and performance verification.

There are tools, which are sensor dependent, that are focussed on a precise type of sensor or system. Thus, there has been plenty of such tools for radar performance evaluation, several ones for tracker evaluation, while there are not as many for more recent sensors such as ADS-B or multilateration systems, although its number is increasing in the recent years.

Some of these tools are just focussed on predicting or evaluating a certain feature of the sensor/system (e.g. probability of detection, probability of false alarm, coverage map, screening, etc.) while others are suites which evaluate some more.

Regarding the target user, there are general purpose tools not linked to any manufacturer, while there are others which are ad-hoc tools intended to evaluate a system of a manufacturer and which are not, in general, available to the general public.

Concerning the finality of the performance evaluation tools, there are ones for anticipating the performances of a sensor; there are others which perform a comprehensive modelling for predicting the sensor behaviour in a variety of scenarios and sensor parameter options; others perform an on-line operational supervision by means of capturing live data which they use for understanding what the sensor is currently doing; and finally, there are tools which are focussed on performing an analysis and a performance evaluation generally using offline processing methods such as the one based on trajectory reconstruction which will be presented in depth in the following chapters.

In [ECTL 97] there is a further division taking into consideration the evaluation phases that needs to be passed when commissioning a sensor or system. This division is not directly presented as such but taking into consideration a multilevel approach for the performance analysis. However, for our interest this approach is directly translated into tools which are intended for factory acceptance tests (FAT), site acceptance tests (SAT), commissioning, radar track quality assessment (RTQA)/maintenance, problem investigation and post modifications. Also in [ECTL 97], there is a proposal for the kind of target reports that can be used for each of the evaluation phases. This proposal includes opportunity traffic, special test flight and simulated data.

3.3.2 Performance evaluation tools for surveillance sensors

In the past, most of the performance evaluation tools that appeared were related to radar as it has been the only sensor providing surveillance information in the ATC domain for many years. Currently, the situation has changed as now there are more options than just radar systems while selecting a surveillance sensor. This trend has also reached the world

of performance evaluation tools, so now it is possible to find tools devoted to ADS-B and MLT systems as it will be presented in the coming subsections.

Thus, focussing on radar systems, in [Studer 98], there is a list of legacy radar analysis tools. Some of them provided a comprehensive modelling for radar detection and coverage analysis in complex environments, such as MRSAS [Barton 92] or CARPET 1.0 [Huizing 91]. Check section 3.3.2.3, for more details about the current situation of CARPET.

There were others more focussed on particular aspects of radar performance modelling such as RGCAL [Fielding 87], which computes the detection range in clear environment; DETPROB [Skillman 91], which calculated the probability of detection, and track formation; SIGCLUT [Skillman 87], which was intended to signal and clutter profile estimation; VCCALC [Fielding 88], which computed the vertical coverage and multipath effects; SBDSA [Macfadzean 92], which was focussed on surface-to-air-missile (SAM) radar analysis; RWS 2.1 [Studer 98], which provided several performance indicators such as the radar detection range, vertical coverage diagrams, different types of plots, the propagation factor and detection probability diagrams for radar performance prediction; GRM-SPW [Lilly 97], which provided a scheme for generic modelling of radar, radar environment and scenario using a highly flexible object-oriented approach.

RADAC (Radar Data Acquisition Converter) was a universal recording system which was used by the "Royal Air Force Signals Engineering Establishment" (RAFSEE) for radar flight trials. It also provided a real plan position indicator (PPI) display, a tracking facility and simple analysis tools [Merrington 98].

PRISMA (Portable Recording Interactive System for Maintenance and Analysis) and its associated toolbox (PRISMA Analysis Tool) were software packages intended to radar performance evaluation by means of recording, displaying and analysing data produced by radar systems. PRISMA includes functions for data recording, data manipulation, data display and data analysis. Further details can be found in [Studer 98]. Please do not confuse with ATC automation product manufactured by COMSOFT.

RAP (Radar Analysis Program) is a Radar-based analysis tool for flight path reconstruction. It was developed by Accident Investigation and Research Inc. (AIR) as an interactive computer program intended to help investigators/analysts during an air accident investigation or while analysing any other aspect of a flight. It was designed to run in a Silicon Graphics computer workstation. More information can be found in [Roberts 94].

The RSS (Radar Support System) is an example of an ad-hoc tool designed initially by Technology Service Corporation and MITRE for the FAA and the USAF Rome Laboratory [O'Hern 97]. This tool, as the previous ones, was initially conceived for siting radar sensors and predicting their performance. However, its upgraded versions, as can be seen in [FAA 07], also covered ADS-B and Multilateration deployments.

The CS-RAST (CAE Soft Radar Analysis/Simulation Tool) and the CS-IRSS (CAE Soft Interactive Radar Simulation System) were two tools provided by CAE Soft Corporation that simulated different aspects of radar systems. Thus, the former focussed on radar functional simulation (Multi-radar, multi-sensor) and electronic countermeasure (ECM)

scenario analysis. The latter simulated the Tx/Rx channel and signal processing. The operation of such tools is described in [Horrell 93].

Lastly, some examples of state-of-the-art performance evaluation tools devoted not only to radar but also to ADS-B and MLT systems, are presented.

3.3.2.1 RASS-S and RASS-R

Intersoft electronics¹³¹ is a company which has two toolboxes, i.e. the RASS-S (Radar Analysis Support Systems for Sites) and the RASS-R (Radar Analysis Support Systems Real Time Tools), for radar subsystem evaluation, radar target generation, radar data analysis and display systems. These suites include both software and hardware elements which support radar engineers while performing quality assessment of primary and secondary surveillance radars, and also ADS-B equipment.

RASS-S is focussed on FAT and SAT phases as it performs very thorough evaluations of the performances of the radar [RASS-S]. This suite also includes tools that permit the simulation of targets aiming at avoiding expensive flight trials or as an alternative for the lack of opportunity traffic, see [IE 09] for an example.

On the other hand, RASS-R addresses daily monitoring and real time quality control [RASS-R]. These quality assessments are performed either on-site or in the ATC centre while the radar is running in operational conditions. The subsection of RASS-R which is of more interest for this thesis is the SSPAT (Surveillance Sensor Performance Assessment Toolset) which is composed of the RCM/D (Radar Comparator Mono/Dual), the TRACKAN (Track Analysis) and the CMC (Coverage Map Calculator). The RCM/D is intended to estimate random errors and biases by means of exploiting opportunity traffic from one sensor (Mono-M) or two sensors (Dual-D). The term “sensors” also includes ADS-B. This analysis is based on trajectory reconstruction. If a multi-sensor analysis has to be performed, it will be done per pairs. TRACKAN monitors in real-time different aspects of the behaviour of sensors (PSR, SSR, Mode-S and ADS-B) such as non-compliant transponders, detection problems, low probability of detection targets, abnormal trajectories, etc.

3.3.2.2 RMD, SMD and RAPS 3

Another company which also offers surveillance performance monitoring tools is FREQUENTIS-COMSOFT.¹³² Thus, it has a surveillance monitoring product family. One of its elements is the RMD (Radar Monitoring Display) which is a real-time monitoring, analysis, and reporting tool for surveillance data to support technical services, investigations, and training [RMD-V2]. It handles both real-time and recorded data from different sources including system tracks, weather information, radar plots, ADS-B and multilateration reports, audio, and screen recordings. It looks like an ATC working position which includes most features of a modern display system. The way in which it can be integrated into the surveillance chain was presented in [CSO 03]. But apart from providing access to several functions related to data processing and presentation, it also offers the support for online and offline data analysis.

¹³¹ For further information please visit: <https://intersoft-electronics.com/>

¹³² For further information please visit: <https://www.frequentis-comsoft.com>

Another product of the family is the SMD (Sensor Monitoring Display) which monitors and analyses ATC surveillance data [SMD]. It provides technical monitoring for operation, or test and validation environments, which is supported by a real-time air situation picture of raw unprocessed target data, a sensor status monitoring functionality with user notification and alarming, and data source diagnostics and event logging. Furthermore, it includes background data recording and what it is most interesting for this work, an offline analysis of recorded data which includes the injection of simulated data.

Another relevant tool is RAPS 3, which is a powerful portable toolset for testing, assessment, validation, and generation of surveillance data [RAPS3]. This toolset is very tuned to the ASTERIX data format, being the first system officially qualified by EUROCONTROL. This company also has an automation product called PRISMA, which should not be confounded with PRISMA (Portable Recording Interactive System for Maintenance and Analysis) mentioned above.

3.3.2.3 CARPET

CARPET (Computer Aided Radar Performance Evaluation Tool) is a software that appeared in 1993 and has survived along time until nowadays [Huizing 93]. Therefore, the initial CARPET 1.0 was developed for running on IBM personal computers and compatibles under versions of DOS 3.0 and higher. It was then migrate to the Windows environment with the upgrade to CARPET 2.0 (i.e. Windows 95 and higher), whose latest known version was CARPET v2.13, which was released the 10th of April 2007. Finally, in the beginning of 2017, CARPET 3.0 was launched after solving some funding issues. CARPET has been developed at the TNO Physics and Electronics Laboratory (TNO-FEL) in The Netherlands.

CARPET is defined as “a comprehensive computer program which alleviates the difficult task of designing and evaluating surface-based radar systems” [Huizing 91][Huizing 91]. Opposite to other tools CARPET considers the entire radar system and its environment, which means taking into consideration emitter and receiver characteristics, clutter, jamming, and propagation phenomena. CARPET generates diagrams which are particularly useful in assessing the detection performance of a radar system. The features, capabilities, and the underlying models of CARPET are discussed in [Huizing 93]. Some graphical examples of the output of CARPET 3.0 can be found in [Theil 17].

3.3.2.4 MAGS Explorer and AGATE

The MAGS (Multilateration and ADS-B Ground Surveillance System) Explorer is the Thales software tool for performance assessment of ADS-B and multilateration systems.

The tool offers features to obtain, analyse and visualize the system performance of the sensors in all operational environments, which can cover the airport domain, TMA airspace and en-route areas, and to further put the measured performance in relation to the required performance values.

The MAGS Explorer imports real sensor data in ASTERIX format and assesses it with respect to an externally provided reference such as the one coming from a DGPS source, or any other system providing ASTERIX data such as a tracker or another sensor (e.g. Radar, MLAT, ADS-B). It is capable of configuring and executing different performance evaluations scenarios. The tool supports the definition of operational service volumes (OSV), to limit the performance analysis to certain airspaces of interest, and other useful

filters. It also permits the configuration of different operational environments in line with the new EUROCAE standards supporting the performance-based surveillance (PBS) approach. The calculation of performance values is conducted per volume, per trajectory and per cell grid. The performance results are provided in numerical format and are additionally graphically visualised to provide a complete view on the obtained performance as can be seen in the pictures below.

The latest version of MAGS Explorer is aligned with newest standards within the SESAR frame for both WAM and ADS-B. It is forecast that the airport extension is accomplished in the current SESAR phase.

MAGS Explorer supports the following standard analyses:

- Probability of update (PoU).
- Probability of long gaps (PLG).
- Horizontal position accuracy (HPA).
- Probability of false & ghost detection (target report and track evaluation).
- Latency.

Finally, some of the graphical results that can be obtained are shown in the following figures:

Figure 3.1: MAGS Explorer graphical result presentations. Source: Thales.

Figure 3.2: MAGS Explorer graphical result presentations. Probability of update per cell. Source: Thales.

In line with the agreements taken in PJ14.04.01 that were commented in CHAPTER 1, MAGS Explorer uses external references. Up to now, for Thales general use-case, no trajectory reconstruction was deemed necessary due to the unavailability of more than one sensor data that could be taken as reference. As such, no trajectory reconstruction is implemented as part of its processing although it implements mechanisms to avoid using “faulty reference data”. The lack of a built-in reference opens the door to feed MAGS Explorer with the output of the trajectory reconstruction that is performed within this thesis.

In addition to an offline mode of operation, Thales validation platform supports the quasi-real time monitoring, which is covered by a newly introduced feature in MAGS Explorer.

Moreover, Thales has not only tools devoted to the performance evaluation of sensors but also others devoted to tracking systems. This is the case of AGATE which provides “xml” reports based on ESASSP KPIs. As such it does not provide any specific tool display, but it offers some web interfaces where the results can be accessible.

For radar systems, Thales relies on SASS-C (ref. 3.3.3.1) as the main mean of evaluating their performances.

3.3.2.5 AccPoDTool and PASS service

Indra, as the other European big player together with Thales and Leonardo, also incorporates in its portfolio several SPM tools.¹³³ As an example, in the framework of the PJ14.04.01 solution, Indra has developed AccPoDTool, which is a SPM tool designed following the PBS approach to determine the KPI of surveillance systems and, as such, paving the path to Performance-Based Operations.

¹³³ As stated in [Ródenas 20], “Surveillance Performance Monitoring (SPM) tools are seen as enablers of a harmonized performance monitoring of surveillance systems. SPM tools seek to identify degradation trends early, using both off-line and in continuous quasi real-time processes, demonstrating the correct functioning of the ATM surveillance function at the individual sensor level or at ATC end-to-end level”.

The tool allows the configuration of the sensor features, geographical areas, and the service volume together with the performance analysis to be carried out and the parameters and thresholds of such analysis.

Figure 3.3 shows some configuration windows and an example of a configured geographical area for a surface scenario located at Lisbon airport (Portugal).

Figure 3.3: AccPoDTool configuration windows and graphical display. Source: Indra.

Once the scenario is configured, the exercise is launched, and the tool calculates performance indicators and statical results. This information is the output of the performance assessment which is then shown to the user by means of graphical representations and reports.

AccPoDTool analyses the performance of cooperative surveillance sensors, like ADS-B and WAM/MLAT, or non-cooperative, like the SMR, provided in ASTERIX data format against an externally provided reference trajectory, calculating performance metrics such as:

- Probability of detection (PoD).
- Probability of update (PoU).
- Probability of long gaps (PLG).
- Root mean square position error (RMS).
- Probability of false & ghost targets (PFT/PGT).
- Etc.

The tool provides 2D and 3D graphical representation of the results, which can be provided per OSV, cell and individual tracks. This graphical representation includes geographical displays and figures where the results are plotted offering a quick visualization of the overall result of the analysis. This analysis can be performed for different environments such as en-route, TMA or surface. In en-route and TMA the analysis is performed per trajectory and per cell, while for surface, the analysis is performed per area (e.g. set of taxiways, runway, or apron) or per trajectory. Figure 3.4 shows a performance analysis executed in a specific area of the airport. In fact, it is the one shown before in Figure 3.3.

Figure 3.4: AccPoDTool graphical display focused on selected airport area. Source: Indra.

Graphical displays enable inspection of the results from intermediate processing stages up to the final ones and integrate a mechanism to investigate unexpected values that allows the identification of the cause that adversely affect the result of the performance metrics. One of the key features that the tool offers is a powerful 3D graphical representation, which allows moving through the scenario visualizing the trajectories from different view angles with the possibility to exclude some erratic ones or to get further information, just by clicking over them.

Figure 3.5: AccPoDTool 3D graphical representation. Source: Indra.

AccPoDTool also generates reports containing all the results that present the list of computed performance metrics in a standardised way. These reports also show whether the calculated values are within the accepted limits or not. These limits are defined by the corresponding EUROCAE standard document although they are also user customizable. Thus, for surface, the applicable standards are:

- ED-116 for SMR.
- ED-117 for MLAT.
- ED-129B for ADS-B.

While for en-route and TMA, the applicable ones are:

- ED-142 for WAM.
- ED-129B for ADS-B.

Additionally, EUROCONTROL ESASSP [ESASSP 15] is also considered for the “end-to-end” evaluation.

In Figure 3.6, it is possible to see the complete architecture of the AccPoDTool. It is remarkable that it integrates an ASTERIX input interface that imports data from surveillance sensors and tracking systems through two ways, one from TCP/IP networks and the other from pre-recorded data files. The tool expects that all data are received from either locally or remotely connected sensors through a unique system LAN and delivered in ASTERIX data format. Therefore, this interface implements standard LAN protocols to directly acquire the input from the systems to be assessed.

Figure 3.6: AccPoDTool system architecture. Source: Indra.

The tool is configured following a two-layer configuration approach, which is stored within the “Data Base” module. The first layer allows to configure the static adaptation and the geographical information of the scenario that will define the overall context where the subsequent evaluation campaigns are going to take place. This information is stored under “Environment” data while the detailed information of each of the exercises that are executed are stored under “Campaign” data. The information concerning a measurement campaign includes the external reference trajectory that is used for the assessment together with the list of evaluations that have been performed. Next, the “Calculation engine” computes the performance evaluation and generates the corresponding statistics. This information is then sent to different modules in order to prepare it to be displayed to the user. Part of the final result is the one shown in the figures above.

All in all, AccPoDTool focus on providing a nice visual interface where showing the calculated performance indicators and required statistics computed taking as the reference an externally provided input. Consequently, it does not currently perform any trajectory reconstruction as it also happened with MAGS Explorer (ref. 3.3.2.4). However, Indra foresees integrating a trajectory reconstruction module in the future. Both tools completely instantiate the processing blocks that are named in this work as “data reading” (ref. 4.3) and “performance assess” (ref. 4.6), while they partially implement the “data preparation” processing block (ref. 4.4).

Moreover, Indra also offers to their clients a surveillance performance monitoring service, i.e. the Performance Assessment of Surveillance Systems (PASS) service, which provides the possibility of performing remote periodical evaluations of radar sensors aiming at early detecting degradations that could allow the identification of performance trends that can be checked during predictive maintenance or even, depending on their criticality, triggering corrective actions. In many cases, the results of these periodic system performance reports permit an initial identification of the cause of the degradations. Furthermore, they are compared towards standardized performance indicators and then presented in a web interface (see below). This allows having technical meeting with the users where some actions and decisions can be taken about the operation of the sensor. This way it is possible to offer a highly specialized and continuous technical assistance without the cost of sending the technicians to the sensor without a justified reason.

Figure 3.7: PASS service web interface. Source: Indra.

3.3.2.6 AWP

The AWP (Analysis Working Position) is the DFS (Deutsche Flugsicherung GmbH) solution for their online performance analysis needs that were not initially covered by SASS-C [Heidger 07], when the latter was designed. Opposite to the previously described tools, which were mainly focussed on offline performance analysis, the AWP is an online tool that is, in fact, embedded in their operational tracking system PHOENIX [Heidger 04], which is also developed by the DFS System House.

The AWP is an implementation of the related quality assurance and service level monitoring requirements that were stated on EUROCONTROL surveillance standard in 1997 [Blue book], nowadays tuned to the current standard values such as the one defined in EUROCONTROL ESASSP v1.0 [ESASSP 12] and ESASSP v1.1 [ESASSP 15].

The capabilities that the AWP provides as stated in [Heidger 07] are:

- Online collocated display of tracks, together with correlated and uncorrelated plots of all radars with several filtering and identification features for analysis.
- Online analysis of radar measurement deviations.
- Evaluation of travel time effects.
- Online analysis of message statistics per radar.
- In-depth ASTERIX message inspection.
- Recording and replay for scene inspection.

As mentioned in [Heidger 07], the tool misses a feature provided by SASS-C, i.e. track accuracy analysis (TAA), that can only be computed offline, when all the information is available. This is not the case when processing online data as it is the one reaching the tracker and, as such, the AWP. However, the tool, based on the ideas provided by other ANSPs, designed a semi-online processing for computing TAA per batches. So, once a batch is closed, a TAA analysis can be launched. However, it is also pointed out in the same reference that a combination of both tools AWP and SASS-C is a convenient solution for covering the needs of the DFS.

The AWP represents an example of another axis of the performance evaluation tools in which the access to online performance evaluations is the objective. This type of issues can also be addressed with solutions similar to the one presented in CHAPTER 7 by just changing the future time horizon to a fixed time interval. In other words, instead of trying to solve a fixed-interval smoothing problem (ref. 7.4.9.1), as it is the case in that chapter, the approach can be changed into one addressing a fixed-lag smoothing problem (ref. 7.4.9.3). This way the results are available in a delayed but pseudo-online manner.

3.3.2.7 Passive radar performance analysis tool

This last tool that is presented in this section represents an example of a performance prediction tool which is developed and implemented to evaluate and enhance the expected performances of a passive radar system developed by the former EADS Cassidian, nowadays integrated in Airbus Space and Defense [Meyer 13]. The Cassidian passive

radar demonstrator handled opportunity signals emitted by FM, DAB and DVB-T transmitters.

As it was mentioned in the previous chapter, INCS sensors and, in particular, MSPSR systems do not account for a commonly accepted measurement model to represent the particularities of their measurement process. So, each company designs the one tuned for their system and based on it they implement the corresponding performance analysis tool. When dealing with passive radar systems it is crucial to bear in mind that, as it is pointed out in [Meyer 13], "the system is highly dependent on the properties of the chosen illuminators and the multistatic geometries which involve illuminators, receiver(s), targets and the surrounding terrain". Taking this into consideration, there is a clear need to specify and build an ad-hoc tool, which combining the information, location, and models of all the involved actors, estimates the predicted performance behaviour at the based on a certain geometry.

The knowledge of this performance behaviour allows a better tuning of the system by better locating the receivers or managing the number of them that are used for estimating the position of the target.

A figure with an overview of the passive radar performance analysis tool can be found in [Meyer 13], where it is described the two operating modes of the tools (i.e. coverage and trajectory). The coverage mode provides the result in a map-based view which shows the track probability depending on geographical coordinates. On the other hand, the trajectory mode focusses the results on a particular reference trajectory. Furthermore, the division of the model in three parts is presented, which are summarized hereafter:

- Wave propagation model: it covers the wave propagation and the analogue part of the receiver.
- Sensor performance model: it exploits the information provided by the previous step in order to analyse signal processing methods and detection.
- Tracker performance model/standard tracking: depending on the operating mode, the tool will perform what is called a tracker performance model in the case of selecting the coverage mode or, in case of selecting the trajectory mode, a standard tracking where the output of the tool is compared with the actual plot data generated by the sensor. It is worth to mention that this tracking refers to the 3D target location obtained from the multilateration calculations, based on the intersection of the corresponding ellipsoids, when several illuminators are used.

As can be seen, the progress on the sensor technology also fosters the appearance of tools that are focussed on the analysis of the predicted performance instead on the evaluation of the performance of the sensors. This axis of performance analysis tools for performance prediction is out of the scope of this thesis. However, they are part of the spectrum of performance tools that are used for sensor design and tuning which justifies their inclusion as part of this state of the art.

3.3.3 Performance evaluation systems for centres

The almost unique paradigm of performance evaluation tool for centres¹³⁴ is SASS-C (Surveillance Analysis Support System for ATC-Centres) developed by EUROCONTROL. This software suite has evolved along time adapting its capabilities to the market or to user needs. Hence, from initial versions just based on PSR and SSR radar systems, it has been upgraded to progressively cope with Mode-S radars, MLT systems and ADS-B data. Additionally, it has also evolved to follow software and hardware technology changes going from workstation running proprietary operating systems to PC platforms running under Linux operating system (e.g. Oracle Enterprise Linux v6.x or higher in the latest versions).

Nowadays, ANSPs consider that SASS-C provides the reference to which all the operational systems have to be compared to in order to evaluate their correct operation and performances, and the tool to be used when weird behaviours appear. The latest versions include the capability to automatically run the test to check whether the ESASSP requirements are achieved or not in the daily operation.

A similar tool was offered by the company “Si ATM AB” (<http://www.siatm.com>), which was called TRAS (Si ATM Test tool for analysis and evaluation of radar infrastructures and tracker systems). Its functional description is quite close to SASS-C, as it relies on the same performance evaluation principle, which is to obtain a reference trajectory by means of exploiting opportunity traffic data. So, it does not provide any new idea to the evaluation of sensors and trackers. Currently, it seems to be discontinued.

3.3.3.1 SASS-C: The paradigm of performance evaluation suites for ATC-centres

As already mentioned, SASS-C is nowadays “the performance evaluation tool” for centres not only within Europe or the ECAC area, where EUROCONTROL has its influence, but also outside these boundaries. This suite was conceived to provide standardized methods and tools for assessing the performance of the surveillance chain. It started as an initial attempt to build a common performance evaluation tool within Europe for radar and tracker assessment. This attempt was led by EUROCONTROL in cooperation with NLR,¹³⁵ who designed its first version back in the late 80’s.

The evaluation of the incoming data is done by means of comparing them with a reference that is computed based on these incoming data, which is usually referred to as “Opportunity Traffic”. One of the advantages of this approach is that its cost is much reduced than others, like test flight, as the data is already available. This concept of “Opportunity Traffic” will be presented in the next section.

The evolution of SASS-C is not only a matter of EUROCONTROL, but also of European national administrations outsourcing most of the life cycle activities to ground industries

¹³⁴ The term “centre” is used for meaning that the evaluation is performed for more than just a unique sensor at a time. Thus, the idea of centre is related to a place where all the information is available as everything is received there. This way either surveillance sensors or tracking systems can be evaluated.

¹³⁵ NLR stands for National Aerospace Laboratory and it is defined as the independent knowledge enterprise in the Netherlands on aerospace. For further information: www.nlr.nl

(Thomson,¹³⁶ COMSOFT,¹³⁷ Espelsa,¹³⁸ CapGemini,¹³⁹ etc.) and Research & Development organizations such as the NLR or the UPM.¹⁴⁰

The latest versions have been:

- Version 5 (v5): it just addressed PSR and SSR radars, and trackers. It was a version running on both, HP-UX (workstations) and Linux operating system (PC platform).
- Version 6 (v6): it added Mode-S radars. This version ran just in Linux operating systems, leaving the HP-UX running systems. The selected hardware platform was based on a PC platform.
- Version 7 (v7): it changed the software architecture of the suite adapting it to new software construction principles and incorporating ADS-B and multilateration systems as new types of sensors to be evaluated. The operating system continued to be a Linux-based one. The introduction of the “multisource acquisition suite” (MAS) and the design of the “trajectory reconstruction and evaluation suite” (TRES) were a big change compared to the previous versions. These two suites were complemented with two more, one devoted to simulation (SMART) and the other to sensor coverage prediction (SCAS). The use of a database for storing the data was generalized.
- Version 8 (v8): it is the latest available version. It adapted the system to perform an automatic analysis based on the new standard for surveillance performance defined by EUROCONTROL in ESASSP. This version is divided into two main functions:
 - Verification (VERIF).
 - Prediction (PREDICT).

VERIF is focussed on the performance verification of the operational surveillance sensor infrastructure (e.g. radar, ADS-B and MLT systems) and SDP systems (SDPS) while working in true operational environments with real traffic. Moreover, PREDICT concentrates on anticipating the behavior of different surveillance sensors.

¹³⁶ Thomson is the previous name of Thales Group (www.thalesgroup.com).

¹³⁷ For further information about COMSOFT, visit: www.comsoft.aero, although now it is part for the FREQUENTIS group.

¹³⁸ ESPELSA was a company of FCC that in 2010 was fused with other companies of the FCC group in order to create FCC Industrial, which is its current denomination. Further details in: fccindustrial.com.

¹³⁹ For further information about CapGemini, visit: www.fr.capgemini.com.

¹⁴⁰ The UPM is the Polytechnic University of Madrid (www.upm.es).

The latest available version of VERIF is 8.4, which was released on the 9th November 2020. Some pictures of this version are presented hereafter:

Figure 3.8: SASS-C Supervision. Source: EUROCONTROL.

The SASS-C architecture has slightly evolved along time. The main modules that build or have built the suite are listed hereafter:

- The system control module gives access to the different tools/programs that the suite provides. In v5, it was called the “cockpit” and, nowadays, it has been renamed to “supervision”, whose look like is shown in Figure 3.8.
- The input reconstruction module computes the reference trajectories from the input data. In v5 and v6, its name was MURATREC (in fact there were two versions, I and II), and from v7 onwards it is called OTR which stands for “opportunity traffic reconstruction”. Both modules are going to be presented in CHAPTER 7, sections 7.6.1 and 7.6.2, respectively. Moreover, Figure 3.9 presents the output of an OTR reconstruction.
- The evaluation module allows the user to carry out performance evaluation tests. In v5 and v6, it was called Evaluation while from v7 onwards the name has changed into “comparator” (CMP) and “ESASSP conformity assessment” (ECA), for more concrete evaluations. An example of its output is provided in Figure 3.10.
- The simulation module generated realistic traffic data in order to test certain features of a system. This module first created the trajectory to be followed by the simulated aircraft and then sampled it to obtain realistic plot data. Its initial name was SMART, which stands for “simulator for multi-radar analysis for realistic traffic”. In v8, this module was initially called SIMULation (SIMUL). However, in the latest updates of v8, it has been removed.
- The sensor coverage module calculates the theoretical coverage of a sensor. This was initially implemented by both, the “radar sharing calculation” (RASCAL) and the “coverage and planning tool for ADS-B and TIS-B” (CAPT) that took

advantage of another tool, the “screening angle and line of sight analysis from digital terrain data” (SALADT), which was focussed on obtaining the screening file of the radar making use of digital terrain data. In v7, all of them were put together under the name of “surveillance coverage analysis suite” (SCAS). Nowadays, this functionality is performed by the PREDICT software suite of SASS-C v8.

Next figure provides an example of the output of the OTR:

Figure 3.9: SASS-C OTR reconstructed trajectories. Source: EUROCONTROL.

Another interesting feature of SASS-C is its built-in ECA evaluation. An example of one of its outputs can be seen in the figure below:

Figure 3.10: SASS-C ECA analysis – Probability of Update (PUA). Source: EUROCONTROL.

As can be seen, the current focus of SASS-C is the TMA and en-route domains. However, due to the current trend related to Performance-Based Operations, which considers the whole trajectory that is Gate-to-Gate, it is expected that the current scope of SASS-C is to be extended to cover the airport environment and surface movements. Some imported data from an airport area are presented in the following picture:

Figure 3.11: SASS-C Imported data from airport area. Source: EUROCONTROL.

This is in line with the future evolution of ARTAS, the paradigm of SDP systems, which will also cover the airport domain [Van Hintum 13] in a not far future. Some initial steps have already been taken in this direction during the execution of the SESAR PJ14.04.01 solution, in which EUROCONTROL participated with the SASS-C tool. However, the tool is far from being ready to carry out performance evaluations on the surface domain as the OTR is not prepared yet for reconstructing airport trajectories.

So, the focus of this work is aligned with the expected evolution of the market, which means that possibly parts of this work can be transferred to commercial products.

3.4 Different methods for performance evaluation based on reference trajectories

The performance evaluation of surveillance sensors and tracking systems based on reference trajectories can be carried out in three different ways, depending on how the reference trajectory has been obtained. Thus, the most common methods to obtain references trajectories are the ones based on:

- Opportunity traffic.
- Special test vehicles with improved measuring equipment and flight trials (e.g. D-GPS test flights).
- Simulations based methods.

Each of these methods has its pros and its cons which are going to be analysed in the following subsections. As it has been stated, the focus of this thesis is placed on those techniques based on opportunity traffic.

3.4.1 Opportunity traffic

These methods are directly fed with information generated by surveillance sensors, while in operational use, concerning the surrounding traffic within their coverage (i.e. opportunity traffic). This information includes the position and sometimes other pieces of information such as identity, velocity, intentions, etc. about the aircraft that are inside their coverage area or domain of interest.

This information can be divided into two:

- Raw information, which is the one directly measured or received by the sensors
- Processed information, in case some data processing is applied to the data, as surveillance sensors may include a built-in tracking algorithm.

Raw surveillance information is affected by different errors such as measurement noise, quantization noise, time misalignment, systematic errors, which admit a statistical characterization, opposite to processed information that is more difficult to characterize as it is quite common that the data processing, which usually include nonlinear calculations, distorted any previous statistical characterization of the input data.

So, the aim of these methods is to estimate the actual position along time of a mobile (ground vehicle or aircraft) based on all the surveillance information gathered during a certain period of time, which should be enough for the internal algorithms to converge. All this information is used for building what is usually called the Reconstructed Reference Trajectory (RRT) of the target, in SASS-C terminology, or the Multisensor Reconstructed Trajectory (MRT) in this text. This is the reason why this method is considered a “batch processing” as it does not start until all data has been gathered. Furthermore, the fact that the reference trajectories are built by exploiting all the input data from opportunity traffic explains why they are also referred to as “reconstructed” trajectories.

Consequently, RRTs are computed from noisy multi-sensor data, which suggests that the tool must implement a data fusion smoothing algorithm to filter out the error terms. Before attempting it, systematic errors on surveillance data must be estimated and then cancelled from the data. This step is a prerequisite of any data fusion algorithm to guarantee that its results are better than the ones provided by each sensor independently. The details of such process are presented in section 6.10.

The main advantage of this approach is that the surveillance data is already available when the data processing algorithm starts, so there is no extra cost apart from storing them. Another advantage is that as an offline processing, there is no time constraints so it can be scheduled at any moment of the day. Besides, it will take into consideration most of the real situations that can happen in the airport or in any airspace under analysis. However, very special cases may be checked or forced with another method. The main disadvantage is that the tool must manage huge amounts of data, which sometimes may

overload it. Furthermore, it is important to consider that dealing with several sensors with dissimilar features from different providers is usually a very complicated matter and so, the data processing algorithms are usually quite complex.

3.4.2 Flight trials

The second method consists on equipping a ground vehicle or an aircraft with a high-precision measurement equipment (e.g. a D-GPS receiver, accelerometers, gyroscopes, magnetron, etc.) which records the dynamical data (position, velocity, acceleration, orientation, etc.) and then after finishing the live flight test, the recorded data are compared with those provided by surveillance sensors and also by the tracking system obtaining the corresponding performance indicators. The systems, which exploit GPS data, assume that it has no error compared to the one coming from surveillance sensors and so, the reference trajectory just consists on the interpolation of the GPS data (ref. 7.4.8.3) at the time of the actual measurements and, as such, it can be considered as a “measured” ideal trajectory.

The main drawback of this method is that it is quite expensive. A cost estimation of a test-aircraft flight per hour was £5k in 1998 as can be found in [Merrington 98], and very constrained to a certain test scenario, so their use is very limited. Furthermore, due to the features of these tests, results are not repeatable. Moreover, to provide a statistically significant numbers of runs would be cost prohibitive. Only in situations where the aim is to assess a certain aspect of a surveillance sensor, which is usually in operation, this approach is recommended.

3.4.3 Simulations

Finally, simulation-based methods have the advantage of having the ideal trajectory followed by the target as it is used for simulating the surveillance data generated by the simulated sensors. This fact also guaranties the repeatability of the performance tests even with the possibility of changing the error components and their distributions on demand by the test engineer.

In this case, the reference trajectory is computed based on the movement equations of the mobile to be simulated. Another advantage based on the same matter is that almost any trajectory around the airport or in a certain airspace can be generated by simulation, so very strange behaviours can be defined and tested. This approach is mainly intended for “in factory” performance tests in which the system, normally a tracker, is fed with simulated surveillance data to measure its performances. This can be checked in [ECTL 97] where simulation is used for “Detailed Technical Performance” evaluation. Although, not only, as it has been shown when presenting for example RASS-S (ref. 3.3.2.1).

An additional issue with this approach is that simulations are based on models and these models are not always as close as desired to the reality. So, in some cases, the detected problems might not be caused by the system itself, but by the simulation model. This fact implies that care must be taken when extracting conclusions from this kind of approaches.

3.5 SESAR framework for a generic performance evaluation tool

This section provides the operational framework where a performance evaluation tool for the airport domain can be located taken into consideration the general ATM picture defined by SESAR. It is important to mention that the architecture to be presented is functional and logical, not physical. This way any manufacturer can implement it in the more appropriate fashion considering its global system implementation. To keep the functional and logical architectures separated from the physical one, the term “Performance Assessment” Functional Block (FB) is introduced. So, any reference to a Performance Evaluation Tool means a physical instance of the “Performance Assessment” FB. The same way, the use of the term component has been left just for referring to physical grouping of functions. Following these principles, the target has been to separate the theoretical concept from any possible implementation of it.

There are several options when including the functionality provided by this tool. One is to consider that it is part of the Technical Supervision and the other is to include it as an additional FB in the Aerodrome ATC Domain System. In this case, the preferred option has been the second one. For the sake of clarity, the Performance Assessment function is only focused on Surveillance Performance Evaluation.¹⁴¹

The diagram in the next page (Figure 3.12) presents the first version of the functional decomposition of the Aerodrome ATC Domain System which has been agreed in the framework of SESAR. As can be seen, this diagram lacks of a Functional block in charge of Performance Assessment which seems to be necessary in order to cope with the need for guaranteeing Quality Control, even more in the SESAR paradigm where the performance-based operations is the final aim. Thus, SESAR 1 took on board the airport segments when defining the 4D trajectory concept extending the evaluation of the performance to the whole route, that is, gate-to-gate. More recently, SESAR 2020 wave 1 has recognized the importance of surveillance performance monitoring by defining PJ14.04.01 solution which focus on new test and evaluation procedures for new surveillance systems.

Nowadays, many controller decisions are based on the information provided by the ground support tools provided by automation systems, so it is clear that a performance assessment is needed; even more if we think in the evolution suffered by the surveillance chain in the recent past. Hence, its complexity has been increased by the integration of new sensor types (e.g. Mode S radar, ADS-B and MLT systems), data link applications (e.g. ADS-C and CPDLC) and networking technologies. Likewise, this fact triggered the need for an upgrade in the tools that carried out this performance evaluation.

¹⁴¹ Generally speaking, the Performance Assessment function can include the evaluation of other functions such as A-G Voice Communications, G-G Voice Communications or Air-Ground Data link Communications, not just the Aerodrome Surveillance and the Ground Correlation Manager.

3.5.1 Functional decomposition for aerodrome ATC domain system

Figure 3.12: NSV-4 Aerodrome ATC System. Source: [SESAR B4.3 D09]

3.5.2 Surveillance function architecture

The baseline architecture of the surface surveillance function that has been defined in the SESAR framework is presented hereafter. This architecture is described in detail in [SESAR P12.03.01 D02]. However, for the sake of completeness, an overview of the baseline functions, which make up the surface surveillance function, has been provided in order to present those elements that can be of interest or affected by the introduction of a performance evaluation tool. The logical system architecture view is provided in the figure below:

Figure 3.13: SESAR baseline architecture for the surface surveillance function. Source: [Soto 11]

The previous figure presents the proposal for the architecture of the surveillance function agreed on the framework of project P12.03.01.

The main internal functions identified as part of the surface surveillance function are [Soto 11]:

- **External Surveillance Tracks Pre-processing:** it is the function capable of acquiring and pre-processing external surveillance tracks which are the ones coming from other collaborating air or surface surveillance functions.
- **Acquisition and Data Pre-processing:** it is an intermediary in the reception of data coming from all sensors (e.g. radar, MLAT or ADS-B). It performs the conversion both of media and format, when necessary, and could include coordinate transformations, sensor systematic error (bias) correction, measurement completion, etc.
- **Sensor Status Data Processing:** this function acquires and processes sensor status data and maintains a sensor status list with the current status information for each sensor feeding the surveillance function.
- **Airport Layout Processing:** this function interprets the airport layout adaptation information, merges it with potential operational supervision information related to airport layout (dynamic airport configuration information), and sends these data to the “acquisition and data pre-processing”, “external surveillance tracks processing” and “multisensor data fusion” functions.
- **Airport Flight Plan Processing:** this function exploits the information coming from correlated flight plans to improve target identification and to adapt “acquisition and data pre-processing” and “multisensor data fusion” functions to flight plans.
- **Surveillance Environment Assessment:** this function processes the sensor status and quality (summarized in the sensor status list described in previous “sensor status data processing” function) and potentially derives some additional quality assessments of the measurement process for each sensor (e.g. covariance matrix, probabilities of detection, bad measurement areas, etc.) which can be used to improve its associated sensor model.
- **Surveillance Environment Adaptation:** this function adapts “acquisition and data pre-processing” and “multisensor data fusion” functions to surveillance sensors status and quality. The different sources of information are:
 - Static data from the adaptation system.
 - Dynamic data provided by technical supervision systems.
 - Surveillance environment assessment output.

This adaptation is performed by precluding the use of a given malfunctioning data source and adapting measurement assumed quality to current situation.

- **Multisensor Data Fusion:** this function receives measures (e.g. plots, target reports and monosensor track updates) and tracks from the “surface surveillance acquisition and data processing” coming from all available sensors, monosensor trackers, and potentially from cooperating surveillance functions (pre-processed by the “external surveillance tracks pre-processing” function) in order to generate

surface surveillance tracks, which include an assessment of their quality, describing the whole surface situational awareness picture. It can also be divided into three subfunctions:

- Multitarget Multisensor Tracker.
- Sensor Calibration.
- Classification/identification.
- **Output Data Server:** this function is in charge of the distribution of the surface situational awareness picture, providing system track data to internal A-SMGCS modules (control, routing or guidance), to the CWP for air situational awareness display elaboration, or to external systems (i.e. other external surveillance functions).
- **Data Recording:** this function, which has access to the internal status of the surveillance function and makes use of the services provided by data recording and playback infrastructure, is in charge of recording surveillance related information.
- **Internal Supervision:** this function is responsible for:
 - Monitoring and displaying the status of the surveillance function.
 - Performing the start-up, shutdown and restart of the surveillance function and its components.
 - Allowing the operator to modify variable system parameters (VSP) that customize the system's technical characteristics.
 - Generating alerts and displaying diagnostic information in the event of faults in the system.
 - Exporting status, alerts, and diagnostic data to External Supervision.
 - Allowing the operator to re-configure the systems to maintain agreed levels of service provision as appropriate.
 - Managing changes in the airport layout due to operational supervisor actions.

Although not explicitly mentioned in any of the previous descriptions, the “internal supervision” controls the behaviour of all of them.

The focus will be place on the functions that are more relevant for a performance evaluation tool. These functions are mainly the “acquisition and data pre-processing”, the “external surveillance tracks pre-processing”, the “airport layout processing”, the “multisensor data fusion”, the “surveillance environment assessment and adaptation”.

3.6 Architecture proposal for a generic performance evaluation tool

The functionality provided by a Performance Evaluation Tool (PET) is considered in this document as another “functional block” of the aerodrome ATC domain system which has been named “performance assessment”. Consequently, the next sections are going to present a proposal for the logical and functional decomposition of such a tool, leaving the physical break down to the next chapters.

First, the general context is presented. The actors and external systems that interact with the performance assessment FB are shown aiming at identifying those elements of the aerodrome ATC domain system that are involved in the process of evaluating the quality of sensors and trackers. Next, the internal functions are presented. Their design follows the principle of maximum cohesion and minimum interdependencies between them. Afterwards, the internal data flows between the internal functions identified in the previous section are described. Finally, the external interfaces are detailed.

This way a complete view of the architecture of a generic performance evaluation tool is defined to ease its integration in any future architecture of an aerodrome ATC domain system.

3.6.1 Context

Figure 3.14: Context diagram of the “performance assessment” FB. Source: own elaboration

3.6.1.1 External actors description

This section provides a description of each of the external actors that have been presented in the context diagram. The objective is to present how the external actors interact with the performance assessment FB. The details of the information exchanged are summarized in section 3.6.4.

3.6.1.1.1 Ground correlation manager

"This function has the responsibility of logically associating surveillance data with an aircraft or vehicle. This association is called ground correlation" [SESAR B4.3 D09].

This is one of the external actors that is under the analysis of the performance assessment FB. In this case, the aim is to check how the ground correlations are performed. As the performance assessment FB must carry out this correlation, this fact can be exploited to assess the ground correlations performed online by the ground correlation manager.

3.6.1.1.2 Aerodrome surveillance

"This function provides controllers with airport situational awareness on the SID(s), STAR(s), apron(s), taxiway(s), runway(s) and landing/take-off paths by providing position and identification of all surface traffic (aircraft and vehicles). It merges the surveillance information provided by the different surveillance sources providing a unique picture of the actual traffic situation. For each aircraft, vehicle and obstacle a system track is generated" [SESAR B4.3 D09].

Consequently, this external actor together with the sensors are the main target for the performance assessment FB as they will be the most important systems to be evaluated.

The objective is to assess the performance of the surveillance function of an A-SMGCS taking into account different points in its processing. Initially the assessment is focused on the output of this system, but if it implements the appropriate logic, the internal data can also be assessed by a PET. Some points which can be of interest are the output of "acquisition and data pre-processing", "multisensor data fusion",¹⁴² "airport layout processing" and "surveillance environment assessment".

On the other hand, the calculations done by the performance assessment FB can be imported in the aerodrome surveillance by means of the "surveillance assessment adaptation" function.

3.6.1.1.3 Surveillance sensors

Under this title, all the surface surveillance sensors characterized in CHAPTER 2 have been considered, namely:

- SMR.
- Conventional SSR.
- Mode S.
- MLAT.
- ADS-B.
- INCS.

¹⁴² For this function, the main focus is on the bias calculation subfunction as its output can be compared with the one provided by the PET.

These systems represent the other important external actor for the performance assessment FB as they play a dual role. On one side, they provide the inputs for the further processing performed by the PET for obtaining the reference trajectories; and from the other, they are also assessed by the tool in order to generate the corresponding figures of merit or performance indicators concerning how well they are operating. This is also known in other systems as Service Level Indicators.¹⁴³

Consequently, the PET will receive from them their plots, target reports, measurements and status information. These inputs would be expected in the common surveillance data format ASTERIX (ref. 2.1.5.1).

3.6.1.1.4 Surface routing

"This function is aimed to select the most appropriate taxi plan for each departing/arriving flight on the basis of the aerodrome flight information, current operational airport resources conditions (i.e. availability and restrictions for surface traffic on runways, taxiways and aircraft stands) and potential interactions with other mobiles" [SESAR B4.3 D09].

The reason why it is included as an external actor is because it is expected to have a trajectory generator which can be used as an external provider of reference trajectories. So, the performance assessment FB will expect its trajectories that would be calculated based on the movement plans of the targets and the airport layout. This information can be send either using a proprietary format or using ASTERIX Cat.062.

3.6.1.1.5 Aerodrome flight data processing

"This function manages the creation, update and modification of system flight plans up to/from the moment the aircraft takes-off/lands. In addition, it calculates the predicted trajectory by taking into account applicable constraints and relevant data (e.g. aircraft performance, weather data, airport configuration data, wake vortex characteristics associated to each aircraft). Aerodrome Flight Plans should be updated at the occurrence of identified significant events such as, for example: request and approval clearance for Start-Up, Push Back, Taxi, Line-Up, Take-Off and vice-versa for landing aircraft" [SESAR B4.3 D09].

As in the previous section, this is another FB that can provide reference trajectories to the performance assessment FB, in fact every flight plan processing function usually implements a Trajectory Predictor (TP).¹⁴⁴ Once the SESAR concept is implemented, the trajectory managed by the ground control should be very similar to the one flown by the aircraft. Consequently, this information can be exploited by a PET for its calculation.

3.6.1.1.6 Adaptation

"This function is responsible for the creation, identification, modification, deletion and management of all environment and configuration data needed by the aerodrome ATC

¹⁴³ "Service level indicators are the performance metrics to be used to measure the agreed-upon levels of service as documented in SLAs for reimbursable services or service declarations for non-reimbursable services". From: [Definedterm]

¹⁴⁴ The TP is a module in charge of generating the trajectory to be flown or to be followed by the aircraft from the information specified in its flight plan.

system (with the exception of those dynamic operational environment configuration data for which the supervision function is responsible). Its responsibility includes:

- The aerodrome topology data management and their distribution.
- The sensor static data (sensor location, default quality level, etc.) management.
- Input measurement spatial filters.
- Other context data."

The previous definition is obtained from [SESAR P12.03.01 D02].

The performance assessment FB receives from this functional block all the initial configuration and static information needed for starting up the system which can be summarized as the aerodrome topology and the sensor static information.

3.6.1.1.7 Support functions

The support functions do not directly affect the provision of operational ATM services. This actor contains at least the following functionalities:

- **Recording:** it is responsible for performing the recording of the ATM system data related to the aerodrome ATC and buffering those data on a persistent database.
- **Playback:** it provides support for display and voice recording, display and voice playback, other data recording and reduction, etc.
- **Data analysis:** it provides support for maintenance, investigation, etc.
- **Automatic safety data gathering tool:** it offers support for safety aspects.

Initially, the data processing carried out by the performance assessment FB is an offline processing which means that in certain occasions, it can work with recorded data from support functions. In fact, it can also provide information to these functions for data analysis purposes, so it is envisaged an important exchange of information between these FBs.

3.6.1.1.8 Technical supervision

"This function is in charge of:

- Monitoring and displaying the status of all the services provided by the Aerodrome ATC system.
- Performing the start-up, shutdown and restart of the entire system or part of it; modifying VSPs (Variable System Parameters) that customize the system's technical characteristics.
- Monitoring and control capabilities for the elements of the system including sub-systems, LANs, external links, etc.

- Automatic re-configuration of systems to maintain agreed levels of service provision as appropriate; displays supervision information and exports data to external clients (through a dedicated HMI)."

The previous definition is obtained from [SESAR B4.3 D09].

The performance assessment FB, as another functional block of the aerodrome ATC domain system, is monitored by the technical supervision. This fact implies that there is an exchange of information between them to guarantee that the latter has updated information about the status and performance of the performance assessment FB.

On the other hand, the performance assessment FB provides the technical supervision with the results of its performance assessments concerning the sensors that feed the aerodrome surveillance and the function itself.

The modification of its VSP is expected to be physically performed by a dedicated HMI which is provided by the PET itself.

3.6.1.1.9 Operational supervision

"This function allows the supervisor to manage the most appropriate operational configuration, according to traffic demand and aerodrome needs, and react in case of system fault, re-assigning and distributing available resources in order to maintain adequate safety levels and quality of service" [SESAR B4.3 D09].

The performance assessment FB uses the information provided by the operational supervision in order to update its internal information. Hence, changes in the airport layout or in the sensor configuration are of interest to the performance assessment FB which should take them into account while performing its evaluations.

3.6.2 Functional decomposition. The performance evaluation tool as a data fusion system

Figure 3.15: Functional architecture of a "performance assessment" FB. Source: own elaboration

3.6.2.1 Surveillance data acquisition and pre-processing

This function consists of two main processes:

- Data acquisition.
- Data pre-processing.

The “data acquisition” consists of the reception and storage of the data coming from: sensors (e.g. radar, MLAT, ADS-B, etc.), any external source of reference trajectories (such as the surface routing or the aerodrome flight data processing) or tracks from the aerodrome surveillance function. When required, it also performs the conversion of both media and format,¹⁴⁵ and additionally, it also includes a playback functionality.

Moreover, the “data pre-processing” performs all the computations needed to prepare the incoming data for the further data processing. This may mean all the needed coordinate transformations to a common fusion coordinate frame, any data filtering, sensor parameter estimations, any bias cancellation,¹⁴⁶ measurement completion,¹⁴⁷ etc.

The incoming data depends on the system that sends the information. Hence, the system may receive raw measurements (plots or target reports), monosensor tracks, multisensor tracks, Reference Business Trajectories (RBT),¹⁴⁸ aircraft flight plans or a mixture of them.

Additionally, it receives bias estimates from the “sensor model update” function which are used for cancelling them from the incoming surveillance data. Furthermore, it also provides updated noise figures which can be exploited by the rest of the processing. Flight plan information is sent to the “aircraft flight plan processing” in order to extract the relevant information for the “reference trajectory computation” such as the aircraft type,¹⁴⁹ the RBT or the surface route.

In addition, this function must select the output data of interest for carrying out the performance assessment by using spatial and temporal measurement selection, specific sensor-based data filtering, and removing targets of little interest for the current evaluation. Thus, it provides converted plots to the next function in the processing scheme that is the “plot to chain association” function, which will generate tentative monosensor and multisensor associated tracks.

Once the data is additionally unbiased, apart from converted into the common fusion coordinate frame, this function returns it to the “plot to chain association” function, which aims at refining the association.

¹⁴⁵ Working as a kind of input data server.

¹⁴⁶ It includes not only sensor-based biases but also target-based ones.

¹⁴⁷ It is already known that for the sake of compatibility, the information coming from several sensors is often converted into another ASTERIX Category which normally implies that some of the specific items related to this sensor are not available.

¹⁴⁸ The RBT is a concept widely spread in the SESAR terminology which defines the trajectory gate-to-gate that the aircraft agrees to fly, and the ground control agrees to facilitate.

¹⁴⁹ This information is of interest while removing the bias concerning antenna position in ADS-B or MLAT target reports or for improving the planned surface trajectory generation.

Finally, there is information that is directly stored in the “internal data storage” to keep it there for further analysis. Several examples are the external reference trajectories, multisensor tracks or the raw plots which can be used by the “performance evaluation” function.

This function together with “plot to chain association”, “aircraft flight plan processing” and “sensor model update” form what is called the “data preparation” processing block. For more details about it, please refer to CHAPTER 6.

3.6.2.2 Plot to trajectory association

This function performs a monosensor and multisensor association in which all the information coming from the same target, which can take the form of measurements, plots, target reports or monosensor tracks updates,¹⁵⁰ is bundled together in a structure called “trajectory”.

It performs the necessary spatial, attribute-based and temporal checks for providing the best plot to trajectory association. This process may imply several iterations depending on the biases of the sensor providing the plot. Thus, there might be an initial tentative association that is further enhanced once the estimated biases are removed from the incoming data. This is necessary due to the close relation between the “sensor model update” function and the “plot to trajectory association” one, when there are, for example, target-based biases.

The inputs to this function are the converted plots or reports from the “surveillance data acquisition and pre-processing” function. Depending on the implemented processing, these plots or reports might also be unbiased. The output is composed of several lists, one for the associated plots, one for the un-associated plots and another one for those plots whose association is not clear so further processing is needed. The associated plots are grouped in multisensor trajectories that can be split into monosensor trajectories if the further processing requires it.

As in the previous case, this function is also part of what is called the “data preparation” processing block. For more details about its local implementation in this work, please refer to sections 6.7 and 6.13.

3.6.2.3 Sensor model update

The “sensor model update” function performs the appropriate calculations for estimating both sensor bias term and noise figures. Bias terms include either target-based biases or sensor-based ones, depending on the kind of sensors that can be found in the airport.

Based on the sensor mix, this function applies different estimation architectures and implements several estimation algorithms that provide estimates of the systematic error terms and noise values included in the plots provided by the surveillance sensors.

Therefore, depending on the bias estimation algorithm, raw plots or converted ones are received from the “surveillance data acquisition and pre-processing” function. For target-

¹⁵⁰ There are sensors which provide track instead of plots or reports. This is the reason why the information provided by these sensors is called monosensor tracks and each bit of them track update.

based bias estimation, information concerning association is needed which can be requested to the “plot to chain association” function.

Moreover, the noise estimation takes advantage of the airport layout to refine the initial values provided by the user as default values. Straight segments might be exploited in order to carry on such estimation.

The obtained bias estimates and noise values are then delivered to the “surveillance data acquisition and pre-processing” function in order to: on one hand, remove biases from the received plots or target reports, and on the other, forward noises through the “plot to trajectory association” function to the “reference trajectory computation” function.

This way, biases can be cancelled from the received plots and so, time-ordered unbiased plots or target reports are delivered to the further processing steps (e.g. “association completion” process). Furthermore, noise values can also be exploited in the next processing steps improving, together with the bias cancellation, the results of the “trajectory reconstruction” processing block.

As in the previous case, this function is considered as part of what is called the "Data Preparation" processing block. For more details about it, please refer to CHAPTER 6.

3.6.2.4 Aircraft flight plan processing

This internal function exploits the information coming from the “aerodrome flight data processing” external function to improve the further computations.

The inputs to this function are the “aerodrome flight plans”, which should provide information such as the “aircraft type” and the “surface route” that can be extracted to be exploited by the “reference trajectory computation” function. The output is composed of the pieces of information which are of interest for the “reference trajectory computation” function.

As before, this function is considered as part of what is called the "data preparation" processing block. For more details about it, please refer to CHAPTER 6.

3.6.2.5 Aerodrome layout processing

This function exploits the information concerning the “aerodrome” map, so it can be used by the other internal functions. Thus, the aerodrome layout is decomposed in segments whose axes might be of vital importance for the “sensor model update” and the “reference trajectory computation” functions. As their implementations might exploit this information for different purposes, the first one for bias estimation and the second one for trajectory reconstruction.

The input to this function is mainly the “aerodrome layout”, which is composed of static parameters, such as the airport map, and by dynamic ones, which depend on the current operational configuration of the airport. Due to the big dependency with weather, this configuration can change several times along the day. Therefore, the static information is expected to be received by the “adaptation” external function and the dynamic one from the “operational supervision”. Within the dynamic information, it is included the allowed movement direction of the taxiways and use of the runways.

The output of this function is the information concerning the geometry of the different segments that are part of the aerodrome, their operational status and some statistics such as the mean speed or mean sojourn time of aircraft in each segment. The internal functions which can be interested in this information are the “sensor model update” and the “reference trajectory computation”.

The tasks performed by this function can be divided into several processes. The ones related to the static information are addressed by the “initial configuration” processing block, the ones concerning the geometry and statistics of airport segments might be addressed by the “data preparation” processing block, and the ones related to the dynamic information might be handled by the “airport map update” processing block. For more details about them, please refer to CHAPTER 4 and CHAPTER 6.

3.6.2.6 Reference trajectory computation

This function is the one which deals with the reconstruction of the reference trajectories from the unbiased and time-ordered version of the measurements initially provided by sensors, any flight plan related information, any information extracted from the airport layout and the updated sensor measurement models.

The reconstruction of the reference trajectories exploits as much as possible the information concerning the aerodrome layout and the one from flight plans. For the reconstruction algorithms, it is crucial the knowledge of the aerodrome map and its division into segments as they impose constraints on the aircraft movement. The “aircraft type” also provides information about the target, such as its approximate dimensions, some design values, and motion features.

It is assumed that the main inputs, which are the sensor measurements, are provided without biases and time aligned. This is a prerequisite for the reconstruction algorithms as they are another type of data fusion algorithms. These algorithms are fed with the actual sensor measurement model which was computed by the “sensor model update” function. Apart from the previous inputs, the “reference trajectory computation” function receives the airport related information from the “aerodrome layout processing” function and the flight plan information from the “aircraft flight plan processing” function.

Once computed, reference trajectories are available to the “performance evaluation” function to calculate the appropriate Service Level Indicator (SLI) either for a particular sensor or for the “aerodrome surveillance” function. These reference trajectories are of vital importance as they are the base for further evaluations, so they have to be as precise as possible. Therefore, several techniques are going to be proposed in this work and some of them even assessed.

This function is the core part of the so-called “trajectory reconstruction” processing block. For more details about it, please refer to section 4.5 or CHAPTER 7.

3.6.2.7 Performance evaluation

The “performance evaluation” function is the internal function which computes and provides the results of the assessment performed on sensors and/or tracking systems.¹⁵¹

¹⁵¹ The term tracker or tracking systems is used as a synonym of the Surveillance Function.

This assessment is mainly done by comparing the output of sensors or tracking systems with the already calculated “reference trajectories”.

The main inputs to this function are the outputs of the systems under test (sensors and trackers) and the reference trajectories computed by the “reference trajectory computation function. The result of the comparison between them is a series of different SLIs or performances indicators, which depend on the system under analysis. This information is then sent to the "human machine interface" function to be shown in a human readable format to the operator.

3.6.2.8 Output server

The “output server” function delivers the evaluation results in the appropriate form and timely manner to the different consumers that requested them. This may imply a certain tuning of the output data depending on the consumer needs, such as data format, update frequency or specified contents.

The input is provided by the “performance evaluation” function and it consists on the results of the assessment performed on sensors or trackers requested on the service definition, which should include the SLIs or performance indicators that are needed.

The output is tuned to the specific consumer by means of the requested service features. Consequently, there are different services expected such as online assessment, offline evaluation or periodical one (i.e. RQM in SASS-C terminology) each one with their different parameter configuration.

3.6.2.9 Human machine interface

This “human machine interface” (HMI) function provides the means for interacting with the PET, so the operator can manage it through this interface.

Within the functionality associated to this function, there are more than just analysis facilities for viewing the results of the analysis; thereby, there can also be found management utilities¹⁵² and configuration menus.

The inputs to this function are the results computed by the "performance evaluation" function and formatted by the “output server”, the data recovered from the "internal data storage" and the information provided by the "internal supervision" function.

Moreover, the operator is provided with the access to the set of analysis tools for carrying out the appropriate assessments, some configuration menus for adapting the tool to its necessities and some management utilities.

3.6.2.10 Internal data storage

This function is in charge of storing all the information that can be of interest for further processing such as raw¹⁵³ and processed plots, target reports, tracks, association data,

¹⁵² Some examples can be the status of the tool and the network, the monitoring of the system load, and the measurement of the duration of the current performance evaluation.

¹⁵³ The ones directly receive as an input to the system from the surveillance sensors.

sensor model values, airport information, reference trajectories, calculated performance values, etc.

It is designed as an element that provides support to the rest of the functions by not only offering a place to store data, but also proving the possibility of retrieving any of the stored data in different fashions.

3.6.2.11 Internal Supervision

This function performs the internal tasks related to “supervision” activities that directly affect all the other functions.¹⁵⁴

It is the interface with the “technical” and “operational supervision”, and with the “adaptation” external function. It performs all the tasks which are a consequence of changes in the airport operational situation that are informed by the “operational supervision”.¹⁵⁵ Changes in the airport map, in the configuration of the airport, or in the airport procedures¹⁵⁶ are notified through this interface. Additionally, it processes the technical commands, service requests and configuration changes that are sent by the “technical supervision”. It also receives the adaptation data to check their coherence and then distribute them to the rest of the internal functions.

Furthermore, this function monitors and displays the status of all the services that it provides, manages the start-up and shutdown processes, and finally monitors and controls the capabilities of the internal functions that intervene in the whole process.

Moreover, the output of this function is sent to the “technical supervision” which will integrate it with the rest of the information coming from the other systems which are supervised. Additionally, part of this information is sent to the HMI in order to keep updated the operator about how the system is internally behaving.

¹⁵⁴ For the sake of clarity, all the links between the “internal supervision” and the rest of the functions have been omitted.

¹⁵⁵ Assuming a semi-online processing, this can mean a change in the operational configuration due to weather effects, any emergency, or a re-distribution of the available resources.

¹⁵⁶ There are procedures such as the Brake to Vacate (BTV) that could be of interest to the PET.

3.6.3 Logical decomposition. Internal data flows

Figure 3.16: Internal data flows of a "performance assessment" FB. Source: own elaboration

Origin	Destination	Data Flow	Description
Surveillance Data Acquisition and Pre-processing			
Surveillance Data Acquisition and Pre-processing	Aircraft Flight Plan Processing	Flight plans	Information describing the features of the aircraft and its planned behaviour (e.g. Aircraft Type, RBT and Surface Route).
Surveillance Data Acquisition and Pre-processing	Sensor Model Update	Converted plots	Plots transformed into the common fusion coordinate frame (i.e. in stereographic coordinates centred on the fusion centre).
Surveillance Data Acquisition and Pre-processing	Plot to Trajectory Association	Converted plots and unbiased plots	Converted Plots both with and without cancelled biases.
Surveillance Data Acquisition and Pre-processing	Internal Data Storage	Plots, converted plots, unbiased plots, sensor parameters, flight plans and tracks	Information that needs to be store for further processing or for replaying needs. It includes all the inputs to the performance assessment FB and the processed data.
Surveillance Data Acquisition and Pre-processing	Performance Evaluation	Plots, converted plots, unbiased plots and tracks	Needed input information in case online processing is requested. It includes all the inputs to the performance assessment FB and the processed data.
Surveillance Data Acquisition and Pre-processing	Internal Supervision	Internal status and errors	Internal status of the function and any anomaly detected while performing its tasks.
Aircraft Flight Plan Processing			
Aircraft Flight Plan Processing	Reference Trajectory Computation	Aircraft types, RBTs and surface routes	Information concerning the aircraft features and its planned actions. This information reflects the intentions of the mobiles.
Aircraft Flight Plan Processing	Internal Supervision	Internal status and errors	Internal status of the function and any anomaly detected while performing its tasks.
Sensor Model Update			
Sensor Model Update	Surveillance Data Acquisition and Pre-processing	Bias and noise estimates	Sensor-based and target-based systematic errors estimation and estimated noise values.

Origin	Destination	Data Flow	Description
Sensor Model Update	Internal Supervision	Internal status and errors	Internal status of the function and any anomaly detected while performing its tasks.
Plot to Trajectory Association			
Plot to Trajectory Association	Sensor Model Update	Tentative association	Initial tentative association for bias estimation purposes.
Plot to Trajectory Association	Reference Trajectory Computation	Enhanced association	Lists of unbiased converted plots which have been associated to a certain trajectory (i.e. target).
Plot to Trajectory Association	Internal Data Storage	Enhanced association	Lists of unbiased converted plots which have been associated to a certain trajectory (i.e. target).
Plot to Trajectory Association	Internal Supervision	Internal status and errors	Internal status of the function and any anomaly detected while performing its tasks.
Aerodrome Layout Processing			
Aerodrome Layout Processing	Sensor Model Update	Airport segments	List of airport segments to be use for bias estimation purposes.
Aerodrome Layout Processing	Reference Trajectory Computation	Airport information	Airport map, taxiways and runways configuration, airport procedures, etc. which can be exploited to improve the mobile trajectory reconstruction.
Aerodrome Layout Processing	Internal Supervision	Internal status and errors	Internal status of the function and any anomaly detected while performing its tasks.
Internal Data Storage			
Internal Data Storage	Surveillance Data Acquisition and Pre-processing	Plots, converted plots, unbiased plots, sensor parameters, flight plans and tracks	Surveillance information needed for replaying a certain scenario.
Internal Data Storage	Reference Trajectory Computation	Unbiased plots and sensor parameters	Surveillance information required by the trajectory reconstruction algorithms.
Internal Data Storage	Performance Evaluation	Plots, converted plots, unbiased plots, association, sensor parameters, flight plans and tracks	Surveillance information needed for performing the evaluation assessments and obtaining the evaluation results.

Origin	Destination	Data Flow	Description
Internal Data Storage	Human Machine Interface	Plots, converted plots, unbiased plots, association, reference trajectories, airport information, sensor parameters, flight plans and tracks	Surveillance information needed for building the whole picture of a certain scenario.
Internal Data Storage	Internal Supervision	Internal status and errors	Internal status of the function and any anomaly detected while performing its tasks.
Reference Trajectory Computation			
Reference Trajectory Computation	Performance Evaluation	Reference trajectories and enhanced association	Trajectories reconstructed from the input data (i.e. opportunity traffic plots and surface routes) to be used as reference trajectories and the association information in case of online processing.
Reference Trajectory Computation	Internal Data Storage	Reference trajectories and airport information	Trajectories reconstructed from the input data (i.e. opportunity traffic plots and surface routes) to be used as reference trajectories and the airport information (e.g. airport map, taxiways and runways configuration, airport procedures, etc.).
Reference Trajectory Computation	Internal Supervision	Internal status and errors	Internal status of the function and any anomaly detected while performing its tasks.
Internal Supervision			
Internal Supervision	All	Commands	Commands sent to all the functions in order to start up, pause, resume or stop.
Internal Supervision	All	Service request	Information about the services requested by the user.
Internal Supervision	All	Adaptation data	Initial configuration values that allow each of the functions to start up properly. In some sense, they are the “by default” initial parameters. They include all the static information such as the airport map, the sensor models or some specific parameters; but also initial configuration parameters which can be changed afterwards through the technical supervision or directly by the user through the HMI (e.g. runway and taxiways initial status, initial reconstruction parameters, initial meteorological conditions, presentation settings, etc.).

Origin	Destination	Data Flow	Description
Internal Supervision	All	Traffic context data	All dynamic data, except traffic information, related to the airport environment (e.g. aerodrome current configuration, runways and taxiways current status, traffic mix, actual meteorological conditions).
Internal Supervision	All	Variable system parameters	Parameters that customize the system technical characteristics (input formats, bias estimation algorithms parameters, reconstruction parameters, presentation settings, etc.) which are expected to be managed either by the technical supervisor or the ATC operator.
Internal Supervision	All	Sensor information	Sensor information from the database (sensor ID, sensor parameters, coverage map, user defined biases, etc.).
Internal Supervision	All	Status	Supervision information (e.g. operational status, equipment status) about all the elements of the architecture (sensors, communication infrastructure, surveillance data processors, etc.).
Performance Evaluation			
Performance Evaluation	Output Server	Evaluation results	Results of the performance assessment FB to be shown to the operator or to be sent to other destinations (e.g. the aerodrome surveillance FB or the ground correlation manager).
Performance Evaluation	Internal Data Storage	Evaluation results	Results of the performance assessment FB to be stored for further use.
Performance Evaluation	Internal Supervision	Internal status and errors	Internal status of the function and any anomaly detected while performing its tasks.
Output Server			
Output Server	Human Machine Interface	Evaluation results	Results sent by the performance assessment FB to be shown to the operator.
Output Server	Internal Supervision	Internal status and errors	Internal status of the function and any anomaly detected while performing its tasks.
Human Machine Interface			
Human Machine Interface	Internal Supervision	Internal status and errors	Internal Status of the function and any anomaly detected while performing its tasks

Table 19: Internal data flows

3.6.4 External Interfaces

Figure 3.17: External interfaces of a “performance assessment” FB. Source: own elaboration

Origin	Destination	Data Flow	Description	Protocol
Ground Correlation Manager				
Output Server	Ground Correlation Manager	Evaluation results	Results of the performance assessment FB to be exploited by the ground correlation manager FB.	Proprietary format
Aerodrome Surveillance				
Aerodrome Surveillance	Surveillance Data Acquisition and Pre-processing	Tracks	Multisensor tracks computed by the aerodrome surveillance FB.	ASTERIX Cat.062 or Cat.011
Aerodrome Surveillance	Surveillance Data Acquisition and Pre-processing	Sensor status	Sensor status reported by the Aerodrome Surveillance function.	ASTERIX Cat.063
Output Server	Aerodrome Surveillance	Evaluation results	Results of the performance assessment FB to be exploited by the aerodrome surveillance FB.	Proprietary format
Surveillance Sensors				
Surveillance Sensors	Surveillance Data Acquisition and Pre-processing	Plots, monosensor tracks	Surveillance information coming from the sensors which belong to the scenario under analysis.	ASTERIX Cat.001, 010, 015, 020, 021, 034, 048
Surveillance Sensors	Surveillance Data Acquisition and Pre-processing	Sensor status	Information regarding the status reported by the own sensors.	ASTERIX Cat.002, 019, 023
Surface Routing				
Surface Routing	Surveillance Data Acquisition and Pre-processing	Reference trajectories	Reference trajectories computed by the surface routing FB or any other external actor.	Proprietary format
Aerodrome Flight Plan Processing				
Aerodrome Flight Plan Processing	Surveillance Data Acquisition and Pre-processing	Flight plans	Reference trajectories computed by the surface routing FB or any other external actor.	ADEXP, ICAO
Support Functions				
Support Functions	Surveillance Data Acquisition and Pre-processing	Stored surveillance data	Data for replaying scenarios for analysis.	Depending on the stored information
Air Traffic Control Operator (ATCo)				
ATCo	Human Machine Interface	VSP modifications	Online changes to the VSP.	N/A

Origin	Destination	Data Flow	Description	Protocol
ATCo	Human Machine Interface	Commands	Management and technical commands.	N/A
ATCo	Human Machine Interface	Scenario configuration adjustment	Modification of the parameters which define a certain scenario for analysis purposes.	N/A
Human Machine Interface	ATCo	Scenario configuration parameters	Information concerning the parameters which define the currently analysed scenario.	N/A
Human Machine Interface	ATCo	Evaluation results	Results of the performance assessment FB to be presented to the operator.	N/A
Human Machine Interface	ATCo	Component status	Information regarding the status of the functions which build the performance assessment FB.	N/A
Adaptation				
Adaptation	Internal Supervision	Adaptation data	Distribution of all static and initial configuration defined parameters for setting the internal functions.	Proprietary format
Technical Supervision				
Technical Supervision	Internal Supervision	Command	Management and technical commands.	Proprietary format, SNMP
Technical Supervision	Internal Supervision	Service request	Request for a certain service provided by the Performance Assessment FB.	Proprietary format, SNMP
Technical Supervision	Internal Supervision	VSP modifications	Online changes to the VSP.	Proprietary format, SNMP
Technical Supervision	Internal Supervision	Sensor information	Information regarding the status and configuration parameters of the sensors whose data are part of the analysed scenario.	Proprietary format, SNMP
Internal Supervision	Technical Supervision	Component status	Information regarding the status of the functions which build the performance assessment FB.	Proprietary format, SNMP
Internal Supervision	Technical Supervision	Evaluation results	Results of the performance assessment FB to be used for decision taking concerning the sensors which should take part in the data fusion process and for evaluating the performance of the different systems under analysis.	Proprietary format, SNMP

Origin	Destination	Data Flow	Description	Protocol
Internal Supervision	Technical Supervision	Service status	Information concerning the status of the services provided by the performance assessment FB.	Proprietary format, SNMP
Operational Supervision				
Operational Supervision	Internal Supervision	Operational information	Information concerning the current operational situation of the aerodrome to be taken into account by the performance assessment FB.	Proprietary format

Table 20: Data flows through external interfaces

3.7 Requirements imposed by the new paradigm based on SESAR

The evolution towards the Performance-Based Operations (PBO) will imply changes at all levels as it has already been anticipated by SESAR. Operationally, the trajectories will have to be managed as a whole opposite as how it is done today where the airspace boundaries limit the service provided by the different ANSP to the aircraft. This means a big transition that has been designed to be performed little by little, which is a more than questionable result for the moment. From the point of view of the author, we are still not only far from operating in a performance-based fashion but also not really close to even the TBO. However, a performance-based environment is much more than just managing trajectories as a whole. It means being capable of offering different ATC services to aircraft depending on their performances, which can refer to communication, navigation or surveillance (CNS).

From a “Business level” point of view, the change in the paradigm will tend to force ANSPs to a concentration. Something has already been observed with the creation, although still not working, of the Functional Area Blocks (FAB). This is an initial attempt to unify the already fragmented European airspace aiming at reaching a similar model to the one managing the National Air Space (NAS) in the US, where there is just one entity in charge of managing the whole ATM service, that is, the Federal Aviation Administration (FAA). This evolution has been slowed down by ANSPs as there is no clear advantage for them. Even more, the process might force some of them to disappear.

Probably, the “system level” is the one that has advanced more. Nowadays one can find many systems which are said to be trajectory-based, although it seems that they are more focussed on easily managing an airspace-based region than on managing a real trajectory-based airspace. However, these systems, mainly FDPs, are seen as the seed that will booster the evolution of the whole ATM environment. This, together with enhanced communications such as future data links (e.g. LDACS or AeroMACS) or new surveillance technologies such as ADS-B, MLAT or Mode-S radars will pave the way towards TBO first and to PBO later.

It might be also important to request a change in the mentality of many of the current actors, as they will have to change their mind in order to accept a different way of

working. This is of special importance in the operational side, which is quite sensitive to changes in their way of carrying out their tasks. Once this resistance to the change is overcome, we will assist to the desired improvement in ATM.

The main impact on the airport domain will come from the extension of the planned trajectory to also cover the airport segment. Hence, the 4D trajectory will be gate-to-gate incorporating not only time but also the taxi-in and taxi-out phases of the flights. Additionally, the evolution of the ATM will also put some pressure on the progression toward the consolidation of the following A-SMGCS levels, that is level III and level IV. This fact means that the routing and guidance functions will need to reach a maturity level enough to be considered operationally and also that the surveillance and control functions are completely trustworthy.

It is in the last point where performance evaluation tools will play an important role as on one hand they will assess the correct behaviour of the surveillance chain, which is the foundation layer for the other A-SMGCS functions, and on the other hand, they will generate service level indicators related to the quality of the information provided to the rest of the functions. Consequently, only an independent and external evaluator can provide both things in an objective, neutral and reliable way. So, it is expected that the evolution towards the SESAR paradigm will push forward the incorporation of performance evaluation tools, not only in the offline assessments but also on pseudo-online ones. This evolution has been fostered by the creation of some Wave 1 SESAR 2020 solutions such as PJ14.04.01, which is expected to be continued in Wave 2.

3.8 Conclusions

In this chapter, the concept of a PET is introduced and the different uses that it can provide are highlighted. The PET state of the art for sensors and centres has been presented. An exhaustive review of the most important evaluation tools is shown with the focus placed on those which are currently still in operation in some place. Next, the different ways of computing a reference trajectory are described and their pros and cons are analysed and presented. The most important for this work is the one based on reconstructing the reference trajectory from opportunity traffic. The latter will be the technique which will be described and analysed in the forthcoming chapters.

An overview of the framework where the PET will have to be integrated is shown. The surveillance function of an A-SMGCS is the natural environment for a PET but not the only one. Therefore, there are other functional blocks of the aerodrome ATC domain which are subject to an external evaluation, although for the target of this thesis, the focus is just going to be placed on the surveillance function.

The following section showed a generic architecture for a PET, providing the context view with a description of the external actors that can interact with it, a functional view that details the internal functions, a logical view that represents the internal data flows and finally, an external view with the data flows to be exchanged with the external actors.

Finally, some considerations about the expected evolution of such tools taking into consideration the SESAR paradigm are presented.

CHAPTER 4

PERFORMANCE EVALUATION OVERALL PROCESS

“When performance is measured, performance improves. When performance is measured and reported, the rate of improvement accelerates”

— Thomas S. Monson

The performance evaluation as any data fusion process can be compared with the elaboration of any good plate like those ones prepared in any of the great restaurants that can be found anywhere in Spain. In this chapter, the recipe to obtain a successful performance evaluation analysis is presented trying to describe all the steps that need to be accomplished before the objective of our cooking is reached, that is, the display of the results obtained all along the performance evaluation process. These steps can then be translated into the processes, subprocesses and tasks whose correct and timely implementation allows to go from the theory to the actual construction of a performance evaluation tool.

As in any recipe, the implemented process admits variations and, as such, it can provide slightly different nuances although keeping a similar flavour, which allows its comparison with the one implemented by other cookers. The process cannot be very strict as the inputs rarely perfectly match the full assumptions of the process, like when cooking, when the available ingredients do not fully comply with what it is required in the recipe or you discover that there is a missing ingredient which has to be substituted or removed. This problem will also be faced, as the information coming from the sensors is not always the same, meaning that it does not fully meet the initial assumptions or that it may lack some of the expected information.

Thereby, the performance evaluation process is now described at high level trying to link it, on one side, to the previously described architecture and, on the other, to the implementation done of such process for testing purposes within this thesis which also introduces the contributions of this work.

4.1 Performance evaluation process overview

The performance evaluation tool (PET) physically implements the whole process of assessing the surveillance chain through its generated data. After presenting the generic architecture for a PET in the previous chapter, it is now the turn for the description of the performance evaluation process that the surveillance data will undergo, that is, from the generating source to the consumers of these data.

This description covers all the necessary tasks, namely: preparing the scenario, importing the surveillance data, performing the convenient tasks to leave the data ready for further processing, carrying out a data fusion process in order to reconstruct the trajectories followed by aircraft, and finally, presenting the result of the surveillance data quality evaluation to the ATC operator in charge of surveillance performance assessment in a convenient and handy manner. It is assumed that all this process is based on exploiting opportunity traffic and, as such, it is mainly an offline processing.

The aim of this section is to present a high-level view of the performance evaluation process to define the framework where it will take place. Opposite to the information presented in the previous chapter, where the functionality took the lead, in this chapter, the time-ordered sequence of tasks or processes to be performed will guide the line of argument. In the following figure, a general overview of the framework, where the performance evaluation tool will work, is presented.

Figure 4.1: Framework of a performance evaluation tool. Source: own elaboration.

The data to be assessed are generated by the surveillance sensors when they detect targets in their surrounding environment that are within their coverage area. These targets will be detected through different detection techniques as it has been presented in CHAPTER 2. Thus, all surveillance sensors detecting the target will process the detection information (i.e. electromagnetic signals) and generate plots, target reports or monosensor track

updates that will be the input not only to tracking systems but also to performance evaluation tools. The main difference between them is that tracking systems (i.e. SDP) are online operational systems, while performance evaluation tools are off-line or, at most, quasi-online non-operational tools, so it is normal that they are prepared to work with surveillance data recordings.

Consequently, performances evaluation tools exploit their inputs (together with some other information computed while processing them) by means of different techniques, like the ones presented in CHAPTER 3, in order to generate their performance evaluation reports. These reports support the operators while taking decisions concerning the surveillance chain.

As already mentioned, the scope of this work is the performance evaluation tools which rely on opportunity traffic to estimate the “reference” trajectory followed by the target. This technique is composed of several sequential steps, which at some points may overlap, that cover from the reception, selection and storage of the incoming data to the elaboration of the output report with the result of the performance assessment done over the surveillance data, passing through all the intermediate steps of a data fusion process, such as: coordinate conversion, bias estimation and cancellation, and final trajectory reconstruction. The aggregation of all these tasks into several high-level processing blocks obeys to a strategy based on grouping tasks that collaborate in a common and clear objective.

Figure 4.2 presents the main processing blocks that are implemented by any performance evaluation tool. These are:

- Data reading.
- Data preparation.
- Trajectory reconstruction.
- Performance assessment.

All them are described in the following sections.

Figure 4.2: Block diagram of a generic performance evaluation tool. Source: own elaboration.

Apart from the aforementioned processing blocks, there are others which contribute to the main processing stream, i.e. the one previously described in Figure 4.2, namely:

- The “initial configuration”, which is the one in charge of providing all the static configuration data needed for starting the tool operation. This information includes the airport layout, the sensor models, and the default parameters.
- The “sensor measurement model”, which is focussed on dynamically updating the measurement models of surveillance sensors. This includes both biases values and noise figures.
- The “airport statistics”, which is a parallel process that aims at gathering all the available statistics concerning the use of the airport segments. The target is, for example, to estimate the mean sojourn time in the segments, the turning radii, or the mean speed of aircraft to estimate a mean speed value for the segments.

This division follows a similar approach to the one presented in [Besada 13]. The differences are mainly due to the additional information coming from the airport layout that does not need to be computed. For example, the “constant velocity segmentation” only requires knowing whether the target is travelling at a constant speed as the constant course can be assumed if located in the middle of a straight segment, which is an information already computed by the “data location in the airport” process. Consequently, the need for identifying constant velocity segments is converted into the need for finding aircraft within straight airport segments that traverse them at a constant speed. Additionally, in order to take advantage of the possibilities that having the airport layout opens, the mode of surface movement (MoSM) segmentation, which is the equivalent to the mode of flight (MoF) segmentation in [Besada 13], is then carried out earlier in the processing by the “movement model identification” process in order to help in the bias estimation. This is the reason why in this document such a task is considered as part of the “data preparation” processing block and not in the “trajectory reconstruction”,¹⁵⁷ where it would probably better match considering the topic it addresses. It is relevant to point out that this work is the evolution of one of the future research lines that were identified in [Besada 13]; that is, the one focussed on the airport domain.

Figure 4.3 presents the whole path that the surveillance data must go through, from its generation in the surveillance sensor to its presentation in the PET display. For the sake of clarity, the diagram does not represent some support activities that interact with the “performance evaluation” process. This is the case of the “internal data storage” function, which interfaces almost all the processes providing the needed storage capacity for either the information that have been imported or the one generated all along the data processing. This includes raw or projected measurements, association or location information, bias estimations, computed noises, corrected measurements, reconstructed monosensor or multisensor trajectories, or any other intermediate data.

Finally, it worth noting that “data preparation” and “trajectory reconstruction” processing blocks are the core of this work. So, it is in their design and implementation where the main contributions of this thesis are focussed.

¹⁵⁷ This processing block is called “reconstruction” in [Besada 13] and it groups: MoF identification, RRT reconstruction and postprocessing.

Figure 4.3: Detailed block diagram of a generic performance evaluation tool. Source: own elaboration.

The process that is described in this chapter has been almost completely prototyped in MATLAB in order not only to check that the designed processes are properly placed in the right order but also to be sure that, technologically speaking, they are affordable with some evolutions of the currently used techniques. This prototype has been designed based on object-oriented programming as this approach has shown a much more efficient way of managing the internal data structures, where the data is stored, than when a structured programming approach was followed.

In the next sections, all the processing blocks are going to be presented. Being the “data reading” the subject of the next chapter, the “data preparation” the topic of the following one and the “trajectory reconstruction” the target of the one after.

4.2 Initial configuration

Previous to any data processing, there is a need for performing several initial tasks such as starting up the “internal supervision” function, which loads the default configuration checking its coherency and opens the connection to both the “operational” and “technical” supervisions and, on the other hand, it also launches the “internal data storage” function, which builds the internal structures that store not only the surveillance data but also the information concerning the scenario where the assessment is going to take place, such as the sensor measurement models, the airport layout, the airport configuration, weather conditions, etc. and later on, all the generated information.

Part of this configuration information is static and part dynamic. This behaviour depends on the update rate of such information. The static one is expected not to change all along the performance evaluation while the dynamic one might change at some point. Thus, as previously mentioned, the static information will be loaded by the “internal supervision” at the beginning of the processing from the “adaptation” external function, while the dynamic one will be managed by other internal functions, depending on the subject, as it will be shown in the next sections.

In the MATLAB prototype, this configuration information is distributed in several configuration files that are read by the constructor of the “performance evaluation tool” object, which is invoked when the performance evaluation process is launched. These are text files which contain the information concerning the scenario to be executed, that is, the recording to be read, the sensors to be configured, the airport information to be loaded, the selected trajectories, etc.

4.3 Data reading

The “data reading” processing block groups all processes and tasks which contribute to the reception and distribution of the incoming data, and the population of the internal data structures. It represents the interface of the PET with the systems to be analysed. Therefore, this implies performing some conversions from the external data formats used by those systems to the ones internally used. Additionally, it also performs some data cleaning before the incoming data is allocated to the corresponding internal data structure from where the subsequent data processing will start. To be sure that the data acquisition has been accomplished properly, it provides the means to display the imported data. The details associated to this processing block will be presented in the following chapter.

4.4 Data preparation

The “data preparation” processing block is one of the core processing blocks of the “performance evaluation” process. It amalgamates all the processes and tasks which contribute to the selection, pre-processing and association of the data that will finally undergo the evaluation process.

Depending on the objective of the PET operator, the initial dataset can be better defined by applying different selection criteria, which can be activated when processing the data. This way incoming data can be filtered out based on, for example: the data source; the data type; some identification criteria such as, the Mode 3A, the aircraft address, the aircraft identifier or callsign, etc.; some geographical criteria such as, data being closer than a certain distance to the ARP or below a certain altitude or height.

By pre-processing, it is understood all the actions intended to present time-ordered unbiased data in a common and homogeneous framework where to continue the further processing. Therefore, this pre-processing includes any needed coordinate transformation or projection, and any known bias cancellation or registering step.

Additionally, this block implements the “plot to trajectory association” function by splitting it in two processes, i.e. the “data association” first and the “association completion” second. The former is followed by two ancillary subprocess focussed on allocating each associated datum to an airport segment (i.e. “data location in the airport” process) and on identifying the different movement models that lie below the associated data (i.e. “movement model identification” process). The “data association” gives rise to the concept of “associated trajectory”, which is instantiated in a “trajectory” object, that can be monosensor or multisensor depending on the sensors that take part in this initial association. The system can decide to maintain a set of “monosensor” trajectories per sensor, a set of “multisensor” trajectories for the tool or both, depending on the needs of the subsequent processing. During this process, the information of a multisensor “trajectory” object is completed with the pieces of information coming from the different monosensor “trajectory” objects which are associated to it.

Afterwards, the bias and noise estimations are carried out; then, the biases are cancelled, which might trigger a data re-ordering in case the times are corrected (i.e. “time re-ordering” subprocess); and lastly, there is a final process (i.e. “association completion”) that refines the initial association taking into account the information that has been generated all along the data processing (i.e. the initial association, the bias correction and the noise estimation).

The output of this block is the list of time-ordered plots, target records or monosensor track updates, which are grouped into monosensor or multisensor associated trajectories, expressed in common coordinates and with the systematic errors corrected.

4.5 Trajectory reconstruction

This processing block can be seen, in a generic way, as the one providing the reference towards which the information to be assessed will be compared to. As such, the “trajectory reconstruction” processing block is the other core processing block of the “performance evaluation” process. It performs all the data fusion tasks that are intended

to reconstruct the trajectory that has been followed by the aircraft taking as input the list of bias-corrected plots, target reports and monosensor track updates that have been associated to such target in the common coordinate framework.

This processing block starts by implementing the “false target report analysis” process, which applies certain algorithms in order to identify and classify the possible false plot, target report or monosensor tracks to remove them from the subsequent processing.

After this, the set of multisensor trajectories are smoothed to obtain the multisensor reconstructed trajectories. This subprocess focus on improving the kinematic information of the multisensor trajectories, which is usually composed of position, velocity and some angular magnitude (e.g. the turn rate or the steering angle), so they can become the actual reference for the performance comparison. This improved kinematic information is based on the integration not only of the past position information but also the future one. This reconstruction is usually performed by smoothing algorithms. Following the same philosophy as in [Besada 08b] or [Besada 13], this reconstruction is a 3-dimensional (3D) one from a spatial point of view, which is divided into two components: the horizontal reconstruction (2D) and the vertical one. Although, the focus of this work is on the horizontal plane, something will also be mentioned about the vertical component. Furthermore, the time dimension is also reconstructed which means that the reconstruction is in fact a 4-dimensional one (4D), that is, 3D plus time.

Next, the “information completion” process completes the information of the samples of these multisensor reconstructed trajectories by enriching them with additional information. Thus, the Mode 3A, Mode C and Mode S information¹⁵⁸, the aircraft identifier (ACID) or the SPI bit are then incorporated to the generated samples of the reconstructed trajectory. The result is what is known as surface reconstructed reference trajectories (SRRT) as they will be the reference to which the incoming data will be compared to.

Lastly, the “trajectory post-processes” process performs some trajectory post-processing activities focussed on preparing the data for its further exploitation by the next processing block. These subprocesses are [Besada 13]:

- Classification of the trajectories into aircraft categories.
- Gap analysis.
- Surface track correlation with flight plan information.
- External surface track enrichment.

Some of these post-processing activities can be seen as part of the “trajectory reconstruction” processing block or as part of the initial activities of the “performance evaluation” one. Due to the importance of trajectory reconstruction within the overall process, the whole CHAPTER 7 is going to be devoted to it. Thus, all the algorithms for trajectory reconstruction are described there.

¹⁵⁸ This information includes not only the codes themselves but also the management of their features as received from the sensor, that is, whether the code has been validated, or if it is garbled or smoothed.

4.6 Performance assessment

The “performance assessment” processing block is the last processing block that has been identified. It is the one that carries out the performance evaluation of the surveillance sensors and tracking systems (i.e. SDP). This assessment is performed by comparison between the SRRT, which is considered as the “reference” or the “real” trajectory that the target has followed, and either the plots, target reports or monosensor track updates coming from surveillance sensors; or, the multisensor track updates provided by surface surveillance tracking systems.

This comparison means the generation of all the ancillary data that is needed for the computation of the SLI, performance figures or performance quality indicators, such as: the sensor measurement accuracy (RMS), the probability of update (PoU), the probability of detection (PD or PoD), the probability of false alarm (PFA), etc. So, depending on the accuracy while generating the reconstructed trajectory, the performance evaluation figures will be more or less reliable and representative of the quality of the sensors and tracking systems under analysis.

4.6.1 Data assessment

This process computes all the background results which allow to estimate the different figures that are used for assessing the performance of a sensor or a tracker. These background results are comparisons between the data to be assessed (plots, target reports or monosensor track updates from sensors, and system track updates from surveillance tracking systems), and the SRRT. This usually means either generating the samples from the SRRT at the same time instant or extrapolating them to the time of the plot, target report, monosensor track update or system track update. With these background results, different types of performance analyses can be carried out. Some examples are:

- Accuracy analysis.
- Gap analysis.
- Detection analysis.
- Update period.

The “accuracy analysis” estimates the noise value that is associated to the sensor measurement process. Thus, it tries to approximate the sensor measurement model assuming there are no bias terms. The “gap analysis” is based on identifying periods of time in which no input is received from the sensor while the target is in the sensor coverage and there is no prior reason for not detecting it; this might be caused by coverage problems, RF congestion, transponder saturation, or occlusions. The “detection analysis” tries to estimate the PD or the PFA of the sensors under analysis taking into consideration its theoretical coverage area. Finally, the “update period” analysis estimates the scan/update period of the sensor that feeds the PET.

4.6.2 Result presentation

Once all the ancillary data is generated and the result of the performance evaluation is obtained, the next step is to deliver it to the corresponding consumers in the proper format and in a timely manner so they can be used for taking decisions about the surveillance chain.

The consumers can be internal to the tool or external, and as such, they might require the performance evaluation analysis in different formats and graphical representations, so some preparation must be done. This task is carried out by the “output server” that is the one aware of the features of the service to be provided to the different consumers. It is defined by the “technical supervision” through the “internal supervision” function. These features include the expected results, their format, and the delivery frequency.

One of these consumers is the HMI function (ref. 3.6.2.9) where the information is presented in a human readable graphical format to the final consumer (i.e. ATC operator). It usually offers several graphical ways of showing the results of the analysis such as: histograms, bar charts, timeline charts or maps. Other performance evaluation output data consumers have been presented in section 3.6.1.

4.7 Ancillary processing blocks

There are two important pieces of information that can impact the “performance assessment” process, but their management is not mandatory as part of the process itself. These are the “sensor measurement model” and the “airport statistics”. Consequently, two auxiliary processes, which run in parallel with the “performance evaluation” process, are introduced to support the continuous update of such information. They are presented in the next two sections.

4.7.1 Sensor measurement model

Sensor measurement models, which are initiated with static values provided by the “adaptation” external function during the “initial configuration” of the tool, might be further improved while processing the incoming data. The initial noise features (standard deviations, variances or covariance matrices) and bias terms (mean values and their uncertainty), which affect the measurements, can be updated all along the data processing allowing a faster convergence of the estimators into consistent values in successive runs of the scenario.

This ancillary processing block is in charge of managing the dynamically estimated values of both noises and biases introduced by all contributing sensors and, therefore, they can be fed back to the data processing. This way the estimated values of the measurement models might get closer to the real ones. Additionally, it can also provide a quicker access to this information if requested by the ATC operator.

The processes in charge of the dynamic estimation are the “noise estimation” and the “bias estimation and cancellation”, which are described in sections 6.11 and 6.10 respectively, while the function in charge of such estimation is the “sensor model update”.

4.7.2 Airport statistics

Airport information can also be updated. Similar to the previous case, part of the airport information is static and so provided by the “adaptation” function during the “initial configuration”, and part is dynamic and depends on the current operational situation of the airport being managed by the “aerodrome layout processing” function through the “operational supervision”. The former mainly refers to the airport layout and the latter to the things related to the operational state such as the weather conditions, the configuration of the different segments and the statistical information concerning the use of the segments, that is, the number of aircraft that crosses them, the mean sojourn time, the mean speed, etc. Most of this information is gathered during the “movement model identification” process as it might locate the plots in the airport segments as an intermediate step in its processing.

CHAPTER 5

DATA READING PROCESSING BLOCK

“You can have data without information, but you cannot have information without data.”

— Daniel Keys Moran

The process of “cooking” the data for its use in a PET starts with the acquisition of such data. This action allows incoming data to reach the following processing stages. But before this happens, the tool must ensure that these data will contribute to the objective of the tool and that it will not lead to saturation, introduce instability, or degenerate the results.

Depending on how data are acquired, the precautions might be different. Therefore, if a pseudo-online performance evaluation is desired, then data are received and processed directly from the source. This can lead to saturation if the source floods the system with data. There might also be issues in the communication channel and due to this, data might be received intermittently or even corrupted. There might be encoding/decoding misalignments between the source and the consumers. Consequently, the PET, as well as the SDP system, must incorporate protection means to degrade or deactivate sensors, and to check the integrity, quality, and frequency of the incoming data. Offline processing is also sensitive to most of these issues so similar protection measures are expected.

After acquiring the incoming data, there is a need to keep the ones that are relevant for the subsequent assessment. This means discarding all the data with no relevance like the one from certain sensor types, from a particular sensor, from a certain target, outside the domain of interest, etc. so only the remaining data are stored for a delayed assessment. Moreover, they can be even presented to check their correct acquisition and selection.

This chapter is devoted to the understanding of the tasks and processes which are needed to correctly import the data to be used in the performance analysis.

5.1 Data reading processing block overview

The “data reading” processing block implements the part of the “surveillance data acquisition and pre-processing” function devoted to data acquisition supported by the “internal data storage” function, which provides the appropriate storage capacity. Furthermore, this processing block also performs some other tasks like:

- Selection of the inputs that comply with certain conditions.
- Initial allocation to the internal data structures.
- Presentation for checking purposes.

The data can reach the PET following different paths. The most common ones are the direct capture from a data network or its reading from a data file. There are different data networks, as already mentioned in section 2.1.6.3, which can be used for sending surveillance data, like dedicated point to point lines or packet oriented networks such as, wide or local area networks (e.g. WAM, LAN or Ethernet).

Besides, surveillance information can reach the performance evaluation tool coded in different data formats (see sections 2.1.5.1 and 2.1.5.3.5) and using different protocols (see section 2.1.6.2). Moreover, recording systems can save this information in different file formats. So, the first step is to receive these data and decode them, trying to express them in a more convenient (usually internal and proprietary) data format.

Once the data are acquired, there are two options:

- Directly process them.
- Record and store them for a later processing.

Whenever a performance analysis is resumed, the “data reading” processing block will request the appropriate data to the “internal data storage” function, which will retrieve them. By such, these data will re-enter the processing flow in the same conditions as if they were just acquired.

“Data reading” is a processing block that has no special research interest as such, as it just consists in gathering data either from a network or a file. However, for the PET, it is crucial as it is the door that allows the data to enter the tool. Thus, this process must be carefully implemented as it will let the data in, so only appropriate and selected data should be accepted. Consequently, it is relevant that all the decoders and data readers are aligned with the protocol versions and data formats that have been used by the encoding side, i.e. the source of the information. This way any possible incompatibility can be avoided.

In the next sections, all the different processes are going to be detailed paying attention to their input and output data, and how they have been implemented in the prototype.

This process can be graphically represented as:

Figure 5.1: Block diagram focussed on the “data reading” processing block. Source: own elaboration.

5.2 Data acquisition

5.2.1 Process description

The “data acquisition” process carries out the acquisition of input data generated by a sensor (i.e. plots, target reports or monosensor track updates) or a tracking system (i.e. system or multisensor track updates) from a communication network or a file into the PET. These data arrive using a certain communication protocol and coded in a certain data format, so there is a need for correctly decoding them before any further processing. Hence, this process implements the interface of the PET with the systems to be analysed.

Nowadays, the more extended data formats to exchange surveillance data are the so-called ASTERIX Final format, pure ASTERIX and a binary format based on the output of network sniffing tools like Wireshark or Ethereal. This process also manages any problem that might appear when decoding the data, such as, a mismatch between the declared ASTERIX category version and the one received, or a problem with the recording format.

Depending on the performance evaluation that will take place, there are two possibilities: the incoming data is directly forwarded to the next process, i.e. the “data cleaning” or it can be recorded and stored for a subsequent assessment. Focussing on the second possibility, which is the most usual one, the “data acquisition” process is responsible for populating a storage mean, which can be a data base or a file in a hard drive, to keep the imported data until being retrieved. All the relevant information for the performance evaluation must be then stored correspondingly to be properly exploited. This storage is usually done in an intermediate format, such as Madrec, ASTERIX Final or a proprietary one, which is managed by the “internal data storage” function.

Furthermore, the “data acquisition” process is also responsible for retrieving the data from the storage mean in case it has been previously stored for a subsequent analysis. Similar to what happens when the data is directly forwarded to the following process, the “data acquisition” process loads those data into a data structure in the RAM memory allocated to the PET, which is referred as the “exchange” data structure. This piece of information is the one passed to the next process.

5.2.2 Processing details

5.2.2.1 Input data

Each sensor that is included in the set of sensors ($S_1, \dots, S_i, \dots, S_N$), which defines the data sources of the surveillance chain, generates plots, target reports or monosensor track updates, which are usually delivered in an ASTERIX record (ref. 2.1.5.3.2), on a regular basis.

The set of target reports coming from the generic i -th sensor (S_i) can be mathematically represented as:

$$\{TR^{S_i}(t_1^{S_i}), \dots, TR^{S_i}(t_k^{S_i}), \dots, TR^{S_i}(t_{n_i}^{S_i})\} \quad (5.1)$$

Hence, a target report (TR) provided by sensor S_i when detecting a target can be represented by the following data structure:

$$TR^{S_i}(t_k^{S_i}) = \{Meas(ToA_k^{S_i}), Id_{data}, Assoc_{info}, Time, Others\} \quad (5.2)$$

Being:

- $t_k^{S_i}$: the time associated to the plot, target report or monosensor track update as generated by the sensor itself. Depending on the sensor (S_i), it can be the time of applicability (ToA), the time of reception (ToR) or the recording time although it is usually refer to as the time of day (ToD). So:

$$t_k^{S_i} = \begin{cases} ToA \\ ToR \\ Rectime \end{cases}, \quad k \in \{1, 2, \dots, n_i\} \quad (5.3)$$

Where:

- k : target report¹⁵⁹ number for sensor S_i .
- n_i : total number of acquired target reports delivered by sensor S_i .
- $Meas(ToA_k^{S_i})$: the “raw” measurement information generated by the sensor, when detecting a target, at time $ToA_k^{S_i}$, ideally the time of applicability. The measurement information is composed of the measurement vector and a quality indicator. The former mainly refers to the position provided either in Cartesian coordinates (x, y, z) , in Polar coordinates (ρ, Az, el) or in Geodetic ones (Lat, Lon, H) . However, it can also provide information concerning velocity, usually in Cartesian coordinates (V_x, V_y, V_z) or Polar ones (GS, Hdg) ; and accelerations, which can be delivered in Cartesian coordinates (a_x, a_y, a_z) or in longitudinal or transversal components (a_L, a_T) ; usually coming from a local tracking performed by the sensor. Furthermore, the quality indicator is provided by means of a standard deviation value, a variance value, a correlation coefficient, or a covariance matrix, depending on the dimensions and information of the measurement vector. This figure of merit gives an idea of the uncertainty associated to the measurement vector, which in some situations is also seen as the opposite, that is, as a measure of its accuracy.

This can be summarized by the following schema:

$$Meas(ToA_k^{S_i}): \begin{cases} \text{Measurement vector} \begin{cases} (x, y, z) \text{ and/or } (V_x, V_y, V_z) \\ (\rho, Az, el) \text{ and/or } (GS, Hdg) \\ (Lat, Lon, H) \end{cases} \\ \text{Quality indicator} \begin{cases} \text{Variance, Std, Correlation coef.} \\ \text{Covariance matrix} \end{cases} \end{cases} \quad (5.4)$$

¹⁵⁹ For the sake of clarity, it is assumed that when referring to a target report, it is meant either a plot, a target report or a monosensor track update.

Moreover, the measurement information can then be decomposed in:

$$Meas(ToA_k^{S_i}) = \{\underline{x}_M(ToA_k^{S_i}), Cov_M[\underline{x}(ToA_k^{S_i})]\} \quad (5.5)$$

Where:

- $\underline{x}_M(ToA_k^{S_i})$: measurement vector of a certain target at time $ToA_k^{S_i}$.
- $Cov_M[\underline{x}(ToA_k^{S_i})]$: covariance matrix of the measurement vector at time $ToA_k^{S_i}$.
- Id_{data} : identification data such as the Mode 3A code, the Aircraft Address (AA),¹⁶⁰ the flight Id or the ACID.
- $Assoc_{info}$: in case the generating sensor performs a local tracking, this information might be available and could be used for improving the “data association” process.
- $Others$: any other information provided by sensor S_i concerning the target.

On the other hand, there can be one or several trackers $(T_1, \dots, T_j, \dots, T_M)$ which generate system or multisensor track updates:

$$\{TU^{T_j}(t_1^{T_j}), \dots, TU^{T_j}(t_k^{T_j}), \dots, TU^{T_j}(t_{m_j}^{T_j})\} \quad (5.6)$$

These track updates are obtained by exploiting all the data provided by sensors or, at least, part of it. So, the same information that reaches the surface surveillance tracker (or trackers) should be the same reaching the PET.

The track updates also include kinematic information, time, identification data, association information and other pieces of information added by the tracking process. Thus, a track update (TU) provided by the j -th tracker (T_j) concerning a certain target can be represented by the following data structure:

$$TU^{T_j}(t_k^{T_j}) = \{State_{info}(t_k^{T_j}), Id_{data}, Assoc_{info}, Time, Others\} \quad (5.7)$$

Being:

- $t_k^{T_j}$: time associated to the multisensor track update. Where:
 - k : track update number for tracker T_j which ranges from 1 to m_j .
 - m_j : total number of acquired track updates delivered by tracker T_j .
- $State_{info}(t_k^{T_j})$: information concerning the state of the target. It bundles together kinematic information (e.g. position, velocity, acceleration, etc.) and its

¹⁶⁰ The Aircraft Address is also known as the Mode S code, the ACAD or the ICAO 24-bit address.

uncertainty as computed by the tracker. The latter is expressed by a covariance matrix as it can be seen in the expression hereafter:

$$State_{info}(t_k^{Tj}) = \{\underline{x}(t_k^{Tj}), Cov[\underline{x}(t_k^{Tj})]\} \quad (5.8)$$

Where:

- $\underline{x}(t_k^{Tj})$: state vector associated to the multisensor track update, TU^{Tj} , at time t_k^{Tj} . The state vector contains all the state variables that allow representing the target kinematic state. Depending on the target motion model (ref. 7.4.6.3), which is selected by the tracker, these state variables can vary. Thus, it can be expressed as:

$$\underline{x}(t_k^{Tj}) = \begin{cases} \text{Position: } (x, y, z) \\ \text{Velocity: } (V_x, V_y, V_z) \text{ or } (GS, Hdg) \\ \text{Acceleration: } (a_x, a_y, a_z) \text{ or } (a_L, a_T) \\ \text{Angular velocity: } \omega \\ \text{Steering angle: } \delta \end{cases} \quad (5.9)$$

- $Cov[\underline{x}(t_k^{Tj})]$: the covariance matrix associated to the state vector, $\underline{x}(t_k^{Tj})$, of the multisensor track update, TU^{Tj} , provided at time t_k^{Tj} . The covariance matrix estimates the uncertainty of the target kinematic state vector. Its dimension depends on the size of the state vector.

Consequently, the “data acquisition” process will have to deal with both target reports and track updates. Due to their different nature, each of them must be stored taking into consideration its particular features.

5.2.2.2 Output data

Up to now, the possible inputs have been presented. The output of the “data acquisition” process is a list with all the input data placed one after the other in the same order in which they have been received or read from a file. This structure is known as the “exchange” data structure and its possible look like is presented in Figure 5.2.

The list assumes no order concerning the data sources, so target reports and track updates can be received mixed up. Even the elements of the list, which are presented with their *ToA*, cannot be assumed as time-ordered as each of them may have suffered a different network delay and, as such, their reception time at the recording system or directly at the PET might show that some previously generated data have overtaken newer data. This issue points to the need for time-ordering the data based on the *ToA*. It is worth noting that sensors or trackers with different update periods are not explicitly represented as it has no especial effect at this point.

Logically speaking the surveillance data flow arriving at the “data reading” processing block is converted by the “data acquisition” process from several parallel data flows (one per contributing source) into a serial data flow, which can be either saved in the storage mean or directly forwarded to the next process (i.e. the “data cleaning”).

$$\left\{ \begin{array}{l} \left(TR^{S_1}(t_1^{S_1}), \dots, TR^{S_1}(t_{n_1}^{S_1}) \right) \\ \left(TR^{S_2}(t_1^{S_2}), \dots, TR^{S_2}(t_{n_2}^{S_2}) \right) \\ \dots \\ \left(TR^{S_N}(t_1^{S_N}), \dots, TR^{S_N}(t_{n_N}^{S_N}) \right) \\ \left(TU^{T_1}(t_1^{T_1}), \dots, TU^{T_1}(t_{m_1}^{T_1}) \right) \\ \left(TU^{T_2}(t_1^{T_2}), \dots, TU^{T_2}(t_{m_2}^{T_2}) \right) \\ \dots \\ \left(TU^{T_M}(t_1^{T_M}), \dots, TU^{T_M}(t_{m_Q}^{T_M}) \right) \end{array} \right\} \rightarrow \boxed{\begin{array}{c} \textit{Data} \\ \textit{acquisition} \end{array}} \rightarrow \{TR^{S_1}(t_1^{S_1}), TR^{S_1}(t_2^{S_1}), TR^{S_2}(t_1^{S_2}), \dots, TR^{S_i}(t_1^{S_i}), \dots, TU^{T_2}(t_1^{T_2}), TR^{S_N}(t_1^{S_N}), TU^{T_1}(t_1^{T_1}), \dots\}$$

(5.10)

Figure 5.2: Output of the “data acquisition” process. Source: own elaboration.

It is said logically, as particular implementations of this process may consider that keeping the parallel data flows per sensor or by groups of sensors is effective for the following processing steps. A tool, like the current SASS-C v8, imports these data and stores them in different tables of a database per sensor family. Hence, there is a table for radar data (sd_radar), one for ADS-B data (sd_ads), other for Multilateration data (sd_mlat) and finally, one for track updates (sd_track). Conversely, previous versions of SASS-C, such as v5 or v6, used a file for storing the data. This means that there is no unique way to store the incoming data.

5.2.2.3 Block diagram

This process can be graphically represented as:

Figure 5.3: Block diagram focussed on the “data acquisition” process. Source: own elaboration.

5.2.3 Prototype implementation details

In the prototype implemented in this thesis, the expected input consists of files (one or several) with pure ASTERIX data that are then converted one-by-one into text files, which represent the chosen storage mean. For the sake of simplicity, each input is provided in a pure ASTERIX file per category and per sensor and, as such, there is a text file per category and per sensor. However, the prototype is ready to process files with a combination of data types from different sensors.

Depending on the sensor generating the data (e.g. SSR, SMR, MLAT or ADS-B), the category of the ASTERIX data blocks will be different (Cat.001 or Cat.048, Cat.010, Cat.020 and Cat.021, respectively). As such, the prototype implements an ASTERIX decoder for the following categories to process their data:

- Cat.001: for PSR and SSR radars.
- Cat.010: for ASMGCS sensors such as SMR or MLAT.
- Cat.015: for INCS systems.
- Cat.020: for MLAT or WAM systems.
- Cat.021: for ADS-B systems.
- Cat.048: for PSR, SSR and Mode S radars.

- Cat.062: for tracking systems (i.e. SDP).

The data items in the ASTERIX records (i.e. target reports) are then converted into an internal format keeping the same resolution as in the original information, but in units of the International System of Units (SI). Per ASTERIX data item or bit of information of interest, the system fulfils a line in the already mentioned text files. The number of lines per ASTERIX record is kept constant regardless the ASTERIX category of the data block that encapsulates it (i.e. 33 in our case). So, the final number of lines of the text file is always a multiple of the constant number of lines that are generated per ASTERIX record.

This means that there might be pieces of information in the internal format which are not delivered by the processed ASTERIX record. This is handled in the prototype by defining these pieces of information as “NaN”, which is the MATLAB value for the scalar representation of “not a number”.¹⁶¹ The reason behind is because linked to this special value, MATLAB implements some built-in functions that allow easily filtering those values (i.e. `isnan(x)`).

These text files are generated only once unless there are modifications on the ASTERIX decoders or there is a new piece of information that needs to be added to the internal format. The tool retrieves the information from these text files and loads it in a MATLAB variable whenever there is a new performance analysis to be executed. This MATLAB variable, which is going to be referred as the “exchange” variable, stores the target reports and the track updates as columns of a matrix. This “exchange” variable is then passed to the following process.

5.3 Data cleaning

5.3.1 Process description

The data that reaches the PET is not error free and, as such, there is a need for removing those converted ASTERIX records that are not correctly decoded, present values out of the expected range, or come without the minimum information to be accepted in the subsequent processing.

This is the reason why there is a process in charge of cleaning those data by removing plots, target reports or track updates that present decoding issues or do not comply with a minimum set of ASTERIX data items.

Therefore, depending on the sensor, this set is defined. In general, the minimum information is composed of:

- Data type: it declares what kind of data is and determines how to process it.
- Time information: usually the time of day (ToD).
- Identity information.

¹⁶¹ Operations return NaN when they have undefined numeric results, such as 0/0 or 0*Inf [MATLAB].

- Position information.
- Height information.

In certain cases, some information can be reconstructed from the incoming data itself. If this is the case, this is done in the “coordinate transformation” process, if the missing information is a position or a height, or in the “time ordering”, when the ToD is missing.

This cleaning process does not mean any type of data selection, which will be done later in the data processing. It is just focussed on keeping the right data to be processed and discarding the wrong or not usable one. The cleaned data are then passed to the following process.

5.3.2 Processing details

5.3.2.1 Input data

The input data are the set of plots, target reports or track updates coming from sensors, or tracking systems in a layout like the one presented in Figure 5.2, that is, with the incoming target reports (*TR*) or track updates (*TU*) listed one after another without any special order but the recording one. This information is received by means of the “exchange” data structure.

5.3.2.2 Output data

The output data share the same layout as the input one, but in this case, the number of elements (target reports or track updates) might be reduced in case there are some elements that do not meet the minimum set of ASTERIX data items or they present some malformed or corrupted data items. The remaining information is kept in the “exchange” data structure, which is then forwarded to the next process.

5.3.2.3 Block diagram

This process can be graphically represented as:

Figure 5.4: Block diagram focussed on the “data cleaning” process. Source: own elaboration.

5.3.3 Prototype implementation details

The MATLAB prototype includes several functions, which together implement the “data cleaning” process, that clean the data loaded by the “data acquisition” in the “exchange” variable (i.e. a matrix) used as the interface between them.

The “data cleaning” process is executed in a stepwise approach, that is, there is a first function focused on checking whether all the input data have a data type. This value is taken either from the ASTERIX record, such as in ASTERIX Cat.010 records (I010/020 #TYP), or is manually defined for other categories (e.g. Cat.010, Cat.020, Cat.021, etc.) while decoding the input data. So, all target reports or track updates with the data type value set to “NaN” are immediately removed from the “exchange” variable.

The next function is devoted to check whether the remaining target reports or track updates include the minimum set of ASTERIX data items to continue in the data processing. If this is not the case, they are removed. Otherwise, they are forwarded to the following process.

5.4 Data allocation

5.4.1 Process description

The “data allocation” process is in charge of allocating the already cleaned input data received in the “exchange” data structure into the internal data structures from where the rest of the data processing will continue.

This process populates either a “target report” or a “track update” internal data element per incoming target report or track update delivered by the “exchange” data structure in the internal data format. These data elements are bundles that grow as the data processing progresses. As such, they will incorporate, apart from the raw information (presented in section 5.2.2.1), transformed information, bias-corrected positions, location data, additional monosensor or multisensor information such as association, extrapolated or filtered or smoothed positions, etc.

Once the new data elements are built, they are assigned to the entity that represents the source which generated them. This means that plots, target reports and monosensor track updates, once converted into “target report” data elements, are allocated to the corresponding list of the entity that represents their generating source (i.e. the sensor). The same happens with the multisensor track updates which will end in the list of “track update” data elements of the entity representing the tracking system.

5.4.2 Processing details

5.4.2.1 Input data

The cleaned “exchange” data structure will be the input to this process.

5.4.2.2 Output data

The output will be a set of internal data elements, which are referred to as “target report” or “track update” data elements, grouped in lists that belong to entities that represent the generating sources (i.e. sensors or tracking systems).

5.4.2.3 Block diagram

This process can be graphically represented as:

Figure 5.5: Block diagram focussed on the “data allocation” process. Source: own elaboration.

5.4.3 Prototype implementation details

The implementation of this process is done by the same function that checks that the data to be allocated carries the minimum information required by the entity representing the generating sensor or the tracking system. This is so because data need to be assigned before checking whether they comply or not with the minimum information requirement for the type of source that generated them.

Thus, the matrix storing the “exchange” data structure is filtered per SAC/SIC to identify only those columns containing data from a particular sensor or tracking system whose content is going to be imported into its internal list of data elements. The SAC/SIC allows to identify the generating source and, as such, retrieving its information from the initial configuration. This fact permits, among other things, to mark the sensor as an active one, and to get the minimum set of ASTERIX data items before these columns are converted into a data element, which in this case are MATLAB objects instantiating elements of either the “target report” class or the “track update” one.

So, only columns that comply with the requirement concerning the minimum set of ASTERIX data items are imported into the system internal data structure.

5.5 Data presentation

5.5.1 Process description

The “data presentation” process is there to guarantee that the “data reading” processing block has been accomplished properly. This check relies on the display, per active system,

of the imported position data, which at the end is the one used for computing the “reference” trajectories.

The imported data are presented on the airport layout and depending on the source of the information, it can come in different coordinates or, in the sensor case, even mixing real detections with monosensor track updates generated by a local tracker. If the position information is provided in different coordinates, all them are displayed and compared among them.

After visualizing this information, the operator can decide to jump to the following processing block, that is, the “data preparation” where the actual data manipulation takes place or to discard the acquired data.

5.5.2 Processing details

5.5.2.1 Input data

The input to this process is the list of “target reports” or “track updates” data elements that have been allocated per generating source, that is, per sensor or tracking system.

5.5.2.2 Output data

The output is a set of figures and text files where the imported information in the internal data structures of the entities representing the generating sources are displayed.

5.5.2.3 Block diagram

This process can be graphically represented as:

Figure 5.6: Block diagram focussed on the “data presentation” process. Source: own elaboration.

5.5.3 Prototype implementation details

The MATLAB prototype implements a function that displays the position information of the “target report” or “track update” objects on top of the airport map. If it comes in various coordinates, all them are displayed. Please refer to section 8.4.1.2 for an example of the outcome of this process.

6

CHAPTER

DATA PREPARATION PROCESSING BLOCK

“Before anything else, preparation is the key to success.”

— Alexander Graham Bell

The trajectory reconstruction starts with the preparation of the data that is available and meaningful for its estimation. These data have already been imported to the system, but except for some completeness and integrity checks, they are taken as they are. Mainly because not all the needed information is yet available. Once the data is allocated to the entity representing the generating source, then further checks can be performed.

This is the starting point of this chapter whose first step selects the appropriate data taking into consideration some initial criteria that are usually user-defined. If the application of some selection criteria needs some previous processing, then they are postponed. If the sensor raw data is already known to have troubles (e.g. biases) and the user has an initial estimation, they can be directly corrected. Once the data is user-defined bias corrected, then it is transformed into the system reference coordinates, where the final selection is performed. At this point, the PET has the data that must use in the subsequent processing. Before any data association, the data is time-ordered. The association of the data into monosensor or multisensor associated trajectories allows finding the path that the target has followed along the airport and permits the classification of its modes of surface movement (MoSM). This information is key for the bias estimation and cancellation, which is performed after. Then, the estimated measurement model is completed by the noise estimation. As the bias cancellation might have modified the detection times, there is a need for time-reordering the data before the final association where all those target reports, yet un-associated, are given the last opportunity to become associated. All this process leads to the set of selected multisensor associated trajectories.

6.1 Data reading processing block overview

The “data preparation” is not only the processing block performing the selection and association of the data that undergo the trajectory reconstruction, but also the one in charge of homogenizing these incoming data before any further processing.

The more time is invested in their preparation, the easier and more precise the next processing will be. Among all the processes that are grouped under the umbrella of this processing block, the coordinate transformation and the bias estimation are the most important ones. If they are not correctly performed, the impact will affect the remaining steps. This is the reason by are known as pre-processing activities, as they project all aligned data into a common reference frame.

The processing starts by selecting the initial set of data that are the target of the performance analysis. The objective is to get rid of the not needed data as soon as possible, so the processing load and storage capacity are minimized. The selection is divided into two steps:

- The initial selection, which is performed by the “initial data selection” process.
- The final selection, which is carried out by the “final data selection” process, which is then executed once the needed computations, i.e. the coordinate transformation and the known bias correction are performed.

Once the initial set of data is defined, if there are user-defined biases, they are removed at this moment by the “known bias cancellation” process. The use of multisensor data implies a coordinate transformation into a common reference coordinate system where all data coming from all sensors is presented, compared, aligned, and fused. This is the task of the “coordinate transformation” process.

Once the data is in a common reference coordinate system and time aligned, it is possible to apply some global geographical filtering which is done, as already mentioned, by the “final selection” process.

The next step consists in ordering the data in time, so a time-ordered list with all the plots, target reports or monosensor track updates from each contributing sensor is built. This is performed by the “time ordering” process which is the responsible for delivering the input for the “data association” process, which is focused on grouping plots, target reports and monosensor track updates belonging to a certain target and coming from one or several sensors. This is to create monosensor or multisensor trajectories, respectively. This process can also implement a monosensor or multisensor trajectory filter. Finally, if the target associated to the trajectory has an AA, it is possible to retrieve information of such mobile and then, exploit it in the subsequent steps.

The following step consists in locating each target report or track update in one and only one segment of the airport. This task is accomplished by the “data location in the airport” process and it is a pre-requisite for some of the subsequent bias estimation algorithms.

Another pre-requisite, which is accomplished by the “movement model identification” process, is the identification of the different ground movement models that match the different segments of the trajectory that a mobile (aircraft or vehicle) has performed

around the airport. This is also known as the motion model or MoSM segmentation. Each target report and track update is then assigned with a ground movement. This information apart from the fact that it provides very useful information for the reconstruction process, it also facilitates the estimation of several biases. This knowledge concerning the MoSM is exploited by bias estimation algorithms when selecting the data they process. Hence, target reports which are classified as belonging to a particular manoeuvring segment might be discarded.

Afterwards, the “bias estimation and cancellation” process uses the selected multisensor associated trajectories for estimating all the biases. This estimation is eased by the knowledge of the airport map where stop points, and constant velocity segments are of especial interest. Thus, first sensor-based biases are estimated in a global way and then, target-based biases are computed in a local way.

Next, the noise is estimated by the “Noise estimation” subprocess, which again takes advantage of the airport map for estimating the noise features. Finally, depending on the magnitude of the estimated and cancelled biases it is possible that there is a need for an additional time re-ordering and an association refinement.

All the previous tasks form what is called the pre-processing, which also includes: the selection of the data to be further processed, the estimation of certain sensor parameters (e.g. the scan period for rotating sensors) or the completion of incomplete measurements.

This process can be graphically represented as:

Figure 6.1: Block diagram of the “data preparation” processing block. Source: own elaboration.

6.2 Initial data selection

6.2.1 Process description

The main target of the “initial data selection” process is to define the data set of interest on which the performance analysis is going to be carried out. It has a double objective, on one side, to limit the scope of the analysis to the intended purpose and, on the other, to minimize the load of the performance evaluation tool by filtering out all those data that have no special interest. The minimization is not only foreseen on the CPU load needed by the tool but also on the memory used to store all internal data structures.

The selection criteria are based on user-provided information that is already available and therefore, it does not need any processing. These data selection criteria can be based on:

- Set of sensors providing the data of interest (i.e. SAC/SIC filtering).
- Data type of the data of interest.
- Time window.
- Height band.
- Real detections vs. monosensor track extrapolations.
- Identity information, such as AA, Mode 3A, flight ID, track number (TN), etc.

Additionally, the selection strategy can be chosen, which means that the selection of a target report or a track update can be done only if all the selected criteria are passed (AND) or if any of them is passed (OR).

It is relevant to mention at this point that the selected data needs to be enough for allowing the convergence of all the estimation algorithms that are put in place during all the performance evaluation process. Consequently, there is a need for finding the right trade-off between the amount of data to achieve an acceptable quality in the performance evaluation process, and the resources needed to process it. It is also fundamental that these data are distributed in a homogeneous way all along the area of interest of the performance analysis to avoid a biased result of those algorithms.

6.2.2 Processing details

6.2.2.1 Input data

The input that arrives to this process is the output of the “data allocation” process, that is, the internal data structure fully filled in. This internal data structures consists of the data managed by the entity representing the PET itself and all those data allocated to the list of entities that represent the generating sources (i.e. active sensors or tracking systems). These sources are expected to manage their own list of “target report” or “track update” data elements previously allocated to them.

6.2.2.2 Output data

The “initial data selection” process provides as its output the same data structures it receives but without those data elements (“target reports” or “track updates”) that are not considered of interest for the performance analysis. Therefore, the list of data elements allocated to those sensors or tracking systems, which are not considered of interest, are emptied and the data elements removed. However, the entities representing sensors or tracking systems are kept.

6.2.2.3 Block diagram

This process can be graphically represented as:

Figure 6.2: Block diagram focussed on the “initial data selection” process. Source: own elaboration.

6.2.3 Prototype implementation details

At this point of the data processing, the prototype implements the previously defined selection criteria except for the one related to the generating sources (i.e. sensor or tracking system). This criterion is not implemented because it can be done at configuration level, where all the possible contributing systems are configured with their parameters. Therefore, if the input from a sensor or a tracker needs to be discarded, it is enough just not to configure them as part of the possible generating sources.

Coming back to the implemented selection criteria, they are performed one after another following the strategy that the more selective criteria are implemented first and then the more particular ones. So, the prototype starts by selecting the target reports whose “datatype” belongs to the ones selected in the “initial data selection” configuration file. Then, the ones that fall inside the user-defined time window are kept. Next, the height criterion is implemented. Depending on the sensor, the source of this height can be different (e.g. barometric or geometric) and, as such, the criterion is applied for the different height measurements. Up to this point, the three criteria are applied all together following an AND selection strategy. The rest of the selection criteria are related to identification and they are focussed on the AA, the Mode 3A and the TN. For these three criteria, it is possible to choose the selection strategy, that is, if they are combined with an AND operator or and OR one. Each of these selection criteria is applied for all the active generating sources. Finally, there are some additional criteria that are also applied depending on the sensor type, such as the selection of only actual detections for radar sensors.

In order to avoid executing many selection operations on the list of data elements per generating source, the entities representing the latter include a “data element selector” that is updated by each selection criterion, and once all them are applied, it is used for updating the list of data elements per source entity by just keeping the selected ones. Consequently, with only one update operation per list, the data element selection is performed.

6.3 Known bias cancellation

6.3.1 Process description

Quite often, the ATC operators in charge of evaluating the quality of the surveillance chain approximately know the biases of the sensors that take part in the surveillance performance evaluation. If this is the case, they can be directly removed from the measurements before any pre-processing is done. This is possible if the bias terms are known in the measurement domain, otherwise the input data must be converted to the projected domain before the already known bias terms can be removed.

The “known bias cancellation” process works only on a per sensor basis as this correction exploits a prior knowledge of sensor-based biases, so there is no impact on data imported from any tracking systems. Hence, this process receives from the “initial data selection”, the list of selected plots, target reports or monosensor track updates that contains all the information coming from the opportunity traffic that has been detected by the sensor.

This list is built per source but it is not time-ordered, as plots, target reports or track updates are stored in the same order as they are received or read by the “data reading” processing block. However, there is no need for a time-ordered list for removing already known biases, because they are sensor-based and, as such, all the data coming from the affected sensor suffer from these errors.

If these known biases are expressed in the same coordinates as the input data (i.e. measurement domain), they can be directly removed while loading them in memory before any further action; otherwise, this task has to be performed once these data are converted to the coordinates in which the biases are known. This is usually done in the data fusion centre coordinates (i.e. projected domain).

6.3.2 Processing details

6.3.2.1 Input data

The input to the “known bias cancellation” process is the not time-ordered list of plots, target reports and track updates per contributing sensor which have passed the user-defined selection criteria implemented by the previous process that has been presented in section 6.2.2.2.

It is worth noting that the initial raw data will never be modified all along the data processing as it is the information that will be compared with the reconstructed information that is obtained at the end of all the data processing.

6.3.2.2 Output data

The output of this process will be the same internal data structure as the one provided as input but adding new information to the data elements (i.e. plots, target reports or monosensor track updates) belonging to the list of those sensors whose biases are known. This new information mainly consists in the “corrected” measurement components and it is represented by the following expression:

$$\{Corr_Meas(Corr_ToA_k^{S_i})\} \quad (6.1)$$

These corrected components are obtained by removing the known biases from the raw measurements. This information is attached to the “target report” data element representation (presented in section 5.2.2.1) as it is shown in the following expression:

$$\begin{aligned} TR^{S_i}(t_k^{S_i}) = \\ = \{Meas(ToA_k^{S_i}), Corr_Meas(Corr_ToA_k^{S_i}), Id_{data}, Assoc_{info}, Time, Others\} \end{aligned} \quad (6.2)$$

Where S_i is a sensor whose biases or at least part of them are known. From this point onward, it will be the corrected information the one that will be processed and used by the algorithms, if available. The “corrected” information includes both the corrected measurement vector and its associated corrected covariance matrix, as it is shown hereafter:

$$Corr_Meas(Corr_ToA_k^{S_i}) = \{\underline{x}_C(Corr_ToA_k^{S_i}), Cov_C[\underline{x}_C(Corr_ToA_k^{S_i})]\} \quad (6.3)$$

Where:

- $Corr_ToA_k^{S_i}$: sensor reported “time of applicability” that is corrected with the user-defined time bias value ($T_{Bias}^{S_i}$) as shown below:

$$Corr_ToA_k^{S_i} = ToA_k^{S_i} - T_{Bias}^{S_i} \quad (6.4)$$

- $\underline{x}_C(Corr_ToA_k^{S_i})$: “corrected” vector that is composed of the corrected vector components, which are computed by removing the user-defined bias ($C_{Bias}^{S_i}$) from the “raw” measurement components ($C_k^{S_i}$), as shown below for a generic “corrected” component (i.e. $Corr_C_k^{S_i}$):

$$Corr_C_k^{S_i} = C_k^{S_i} - C_{Bias}^{S_i} \quad (6.5)$$

- $Cov_C[\underline{x}_C(Corr_ToA_k^{S_i})]$: “corrected” covariance matrix of the “corrected” vector. Depending on the knowledge concerning the correction to be applied, this might imply keeping the raw covariance matrix or enlarging the latter by a constant value.

As mentioned before, the generation of this additional information is mandatory in order to keep the initial information unchanged, as it will be the one evaluated. So, whenever the corrected information is available, the raw measurements will just remain for the final performance assessment.

6.3.2.3 Block diagram

This process can be graphically represented as:

Figure 6.3: Block diagram focussed on the “known bias cancellation” process. Source: own elaboration.

6.3.3 Prototype implementation details

The prototype implements this process per sensor as each of them has a different measurement model with different bias terms.

Therefore, the “sensor” class includes a measurement model which includes not only the bias terms but also the noise ones. This measurement model is particularized per sensor type (e.g. SMR, SSR, MLAT, etc.), so the derived sensor classes define a method for user-defined bias cancellation that takes into consideration the particularities of each sensor family.

These methods take the raw data provided by the sensors and correct them with the user-defined bias values, generating the so-called “corrected” data. The latter will be used to create a data structure, which at least contains a time value, a position vector and a covariance matrix, that is attached to the data elements remaining in the list of the sensor whose known biases are now corrected. This new structure is created regardless the user-provided values, so if no value is provided then the raw one is copied. As already mentioned, raw data are needed “as provided by the sensor” for the final performance analysis.

6.4 Coordinate transformation

6.4.1 Process description

The coordinate transformation is a mandatory pre-processing step of any data fusion process. In the surveillance domain, it is common to receive data from different sources, which do not share the same coordinate reference framework. Thus, radars provide their measurements in polar or spherical coordinates, ADS-B in geodetic coordinates and MLAT in Cartesian or geodetic coordinates, as it was presented in sections 2.2.1.4, 2.3.2.5 and 2.4.1.4 respectively.

Usually, they provide the information into local reference frames (ref. 2.1.4.1.1) although they can also provide it in global ones (ref. 2.1.4.1.2). However, they are often different from the one used by the data fusion algorithm which, in the ATC domain, consists in a stereographic projection centred at the location of the data fusion centre. In the case of an airport, this point is normally the ARP. All this defines the common or system reference framework that was introduced in section 2.1.4.1.3.

If the contributing surveillance sensors use a clearly defined reference frame, changing coordinate systems is a straightforward task, although not always free from introducing some errors that can be estimated and assessed. This way the information provided in different coordinates must be translated into a common coordinate framework, where all the calculations are performed. This is what is done by the “coordinate transformation” process. It is worth noting that it refers to both the measurement or state vector and their covariance matrices.

The “known bias cancellation” process delivers the list of the remaining “corrected” data elements (i.e. plots, target reports and monosensor track updates) per sensor to the “coordinate transformation” with all the known biases removed. This can be done if the known biases are in the same coordinates as the raw plots. Otherwise, there is a need for performing a previous coordinate transformation. This process should also complete those measurements which, although having all the required information, may lack some pieces that can be re-generated during this process.

In any case, the incoming measurement information needs to be converted into the coordinates where the whole data fusion process is accomplished. As already mentioned, in the case of the airport, it consists in a stereographic projection centred at the ARP of the airport. The coordinate transformation equations are the ones already presented in sections 2.2.1.4.2 and 2.3.2.5.2 for measurement vectors.

Similar to the way the “known bias cancellation” process attaches the output of its processing to the remaining data elements in the list managed per sensor or tracking system, the “transformed” data are added to these elements.

6.4.2 Processing details

6.4.2.1 Input data

The input to the “coordinate transformation” process is the list of selected plots, target reports and monosensor track updates per sensor, and the list of selected multisensor track updates per tracking system. None of these lists is expected to be time-ordered. Moreover, inputs coming from sensors whose biases are known are already corrected with the user-defined bias values.

6.4.2.2 Output data

The output of this process is the “transformed” information, which is attached to the input data. This is represented by the following expression:

$$TR^{S_i}(t_k^{S_i}) = \{Meas(ToA_k^{S_i}), Corr_Meas(Corr_ToA_k^{S_i}), Transf_Meas(Corr_ToA_k^{S_i}), \\ Id_{data}, Assoc_{info}, Time, Others\} \quad (6.6)$$

Where $Transf_Meas(Corr_ToA_k^{S_i})$ is the additional information generated by the “coordinate transformation” process consisting in the stereographic projection into the system reference framework of the corrected measurement $Corr_Meas(Corr_ToA_k^{S_i})$. This can be represented in the following way:

$$Transf_Meas(Corr_ToA_k^{S_i}) = \{\underline{x}_T(Corr_ToA_k^{S_i}), Cov_T[\underline{x}_T(Corr_ToA_k^{S_i})]\} \quad (6.7)$$

Where:

- $\underline{x}_T(Corr_ToA_k^{S_i})$: “transformed” measurement vector at time $Corr_ToA_k^{S_i}$, which can be represented as:

$$\underline{x}_T(Corr_ToA_k^{S_i}) = Transform_vector\{\underline{x}_C(Corr_ToA_k^{S_i})\} \quad (6.8)$$

The $Transform_vector\{\underline{x}\}$ function applies the equations presented in sections 2.2.1.4.2 and 2.3.2.5.2 to the incoming measurements. The specific equations will depend on the coordinates in which the raw measurement is provided.

- $Cov_T[\underline{x}_T(Corr_ToA_k^{S_i})]$: covariance matrix of the transformed measurement vector at time $Corr_ToA_k^{S_i}$. This covariance matrix needs also to be transformed to the appropriate domain. This fact can be shown as:

$$Cov_T[\underline{x}_T(Corr_ToA_k^{S_i})] = Transform_matrix\{Cov_C[\underline{x}_C(Corr_ToA_k^{S_i})]\} \quad (6.9)$$

Please note that the rest of the terms that appear in eqn. 6.6 where presented in sections 5.2.2.1 and 6.3.2.2.

Once all data are represented in the common data fusion, they are then ready to enter the following data processing steps.

6.4.2.3 Block diagram

This process can be graphically represented as:

Figure 6.4: Block diagram focussed on the “coordinate transformation” process. Source: own elaboration.

6.4.3 Prototype implementation details

The prototype implements the appropriate coordinate projections and transformations in order to convert the information provided by the input sensors and tracking systems into the common or system reference framework to be used by the data fusion algorithm that later reconstruct the trajectories. The coordinate transformation equations are the ones already presented in sections 2.2.1.4.2 and 2.3.2.5.2.

6.5 Final data selection

6.5.1 Process description

This process finalizes the selection task started by the “initial data selection”. It performs those data selections that relay on some of the intermediate processes that have been described before in sections 6.3 and 6.4. This is the case of any geographical filtering which might be needed in case of having sensors like a SSR which have a coverage of hundreds of kilometres or radars in general when providing the position information in polar coordinates centred in the radar location.

Once all the data is transformed in the same coordinate reference framework, the geographical filtering can be applied without the risk of not applying it in a correct way depending on the sensor providing the data. So, the “transformed” data is a pre-requisite for this process.

Based on this “transformed” data, different geographical filters can be designed by changing the shape of the area of interest where the plots, target reports or track updates are retained. The simplest filtering area shape is the circular one.

Finally, it is assumed that any known timestamp bias correction on the data will be negligible compared to the duration of the selection time window, so there is no need to implement such selection criteria at this point of the processing.

6.5.2 Processing details

6.5.2.1 Input data

The input that arrives to this process is the internal data structure selected by the “initial data selection” process, which is enlarged with two types of information: one linked to a correction on the position measurement values based on a prior knowledge of the biases that affect the sensor measurement process and the other related to the transformation of all the position information into the same coordinate reference framework, so they can be displayed together in a homogeneous way. However, these data still miss the application of those selection criteria that rely on the information added by the two new pieces of information.

As mentioned before, these internal data structures include all those data elements (i.e. target reports or track updates) allocated to the list that belongs to the entities that represent the generating sources (i.e. active sensors or tracking systems).

6.5.2.2 Output data

Until now, the data elements lists were managed at a contributing source level. However, this process adds a transversal dimension (i.e. tool-wise) to the selection criteria. In any case, the output keeps the same internal data structure but removes all those data elements that do not fulfil any geographical criteria that is defined by the user at tool level and, as such, they are of no interest for the performance analysis. Any other criteria that relays on the result of either the “known bias cancellation” or the “coordinate transformation” processes can also be implemented at this point.

6.5.2.3 Block diagram

This process can be graphically represented as:

Figure 6.5: Block diagram focussed on the “final data selection” process. Source: own elaboration.

6.5.3 Prototype implementation details

The prototype implements a function that reads the details of the final data selection criteria, which are provided through a text file, that in this case is a geographical filter that defines an area of circular shape around the ARP of the airport. Therefore, every data element (i.e. target report or track update) that is outside this circular area is marked to be excluded from the subsequent processing.

The way of implementing this functionality mimics the one followed by the “initial data selection” process, that is, it builds a final selector per source that identifies those data elements that should be further processed from the ones already selected by the initial selector built by the “initial data selection” process. The prototype does not remove those data elements from the source list, but from now onward it has a mean for not taking them into account, if needed.

6.6 Time ordering

6.6.1 Process description

Since surveillance sensors have different features, different means to timestamp their measurements and different time synchronization methods and references; there is a need

to express all them in the same timeline or time reference. This, together with the variable delay introduced by the communication network that connects the contributing systems with the airport premises and/or any other processing delay, makes the time of applicability (ToA)¹⁶² of the plot, target report or monosensor track update differs from the reception time (ToR)¹⁶³ at the receiving ground station or directly at the performance evaluation tool; and so, the order in which the different data are acquired would be different to their generation order.

Even more, some reported data from a sensor can suffer different delays depending, for example, on the technology that is used for detecting the target. This is the case that has been observed for multilateration systems that introduce a higher delay for some Mode S targets compared to other SSR ones, which are in their surroundings. This fact can even reach the point that later target reports overtake the newer ones at the entrance of the recording system, the tracking system, or the performance evaluation tool. This can cause some troubles for the association algorithms of some tracking systems as they work with very demanding performance requirements concerning their processing time that prevent them to introduce a buffer that can compensate these delays. For performance evaluation tools, this issue can be mitigated by processes like the one described in this section.

All this together, justify why there is a need for re-ordering these data with respect to the ToA or any other homogeneous time reference that can be available. Consequently, once all the data are corrected and presented in the common reference frame defined by the data fusion system, the next step is ordering them in time. Hence, based on the ToA, time of measurement or time of day (ToD) of each plot, target report or monosensor track update, the current data element list per source is time-ordered.

Thus, if the surveillance data come without ToA, then other time measurement should be used, like the reception time at a ground station or the recording time (Rectime)¹⁶⁴ at the “data acquisition” process. This is done at the cost of introducing an undetermined delay. This case happens when processing ADS-B data, as depending on the transponder version (i.e. ED102 or ED102A) and on the capability of the aircraft to be UTC synchronized, the time computed at the ground station can be the ToA or the ToR.

This processing eases the following steps like the “data association”, the “movement model identification” or the “data smoothing”. In the first case, because it facilitates one of the criteria for confirming that a plot or a target report belongs to a certain multisensor associated track or chain (target), that is, the time compatibility check. In the second one, because the motion model identification relies on a time-ordered list of measurements for applying its algorithms. Finally, the smoothing of the data also relies on a time-ordered list to avoid reprocessing or tracing back data.

¹⁶² The time of applicability (ToA) refers to the time when the information within the measurement was taken. It is the preferred time reference to be used in the data processing as it does not introduce any delay. The other time references usually introduce an unknown time delay to the processing that might be in some cases estimated.

¹⁶³ The reception time or time of reception (ToR) is the timestamp generated by any ground station which receives the message with the measurement information through any of the available ways to reach it. Depending on the surveillance technology there might be several ground stations that receives the incoming signal. This fact might be taken into consideration as it might cause problems to the processing logic as the same measurement information may be associated with different times.

¹⁶⁴ The recording time (Rectime) is the time stamp that the acquisition logic assigns to the incoming data. This is usually performed by the recording system in charge of acquiring the incoming data.

6.6.2 Processing details

6.6.2.1 Input data

The “time ordering” process receives as input the list of plots, target reports and track updates corrected by the “known bias cancellation” process and converted to the data fusion framework by the “coordinate transformation” process. This is a list that comes without any specific criterion concerning time ordering.

6.6.2.2 Output data

In this case, the output consists just in re-ordering the same information provided by the input. Consequently, the output list contains the same information but ordered in a different way. Instead of being based on the arrival order, it is based on the time of applicability.

6.6.2.3 Block diagram

The “time ordering” process can be graphically located in the “data preparation” processing block as shown in the figure below:

Figure 6.6: Block diagram focussed on the “time ordering” process. Source: own elaboration.

6.6.3 Prototype implementation details

The prototype defines a function that checks whether the data element list of a sensor is time-ordered, usually based on the ToD, or not. In case it is not, the function extracts the time-ordered sequence of times and provides the correct sequence of data element indexes. This sequence can then be used to correctly order the list of data elements of this sensor, which will be used by the subsequent processing steps. The time that is employed is the one available from the input data with the following priority: whenever it is available, the ToA is used, if not the ToR, and the last option would be the Rectime.

6.7 Data association

Since this process incorporates a complexity not comparable to the others that have been described up to now, this section has been enlarged with an additional subsection which addresses the overall problem definition. This approach will also be followed for the “bias estimation and cancellation” process.

6.7.1 Problem definition

The data association problem consists of the identification of the “more likely” plot or target report from the available ones that continues an already created and consolidated track. This track should be univocally linked with a target, either an aircraft or an airport mobile, such as: an airport vehicle, a stair truck, a bus, or a towing truck. By the “more likely” plot or target report, it is understood the one that reaches a certain degree of temporal, geographical and identity compatibility with the track, based on which the association decision can be confirmed or not. If the association decision is confirmed, the plot or target report is assigned to the track and later, the track internal states are updated with the information provided by this plot or target report. Please note, that this process should only deal with plots or target reports as monosensor track updates are already associated by the sensor local tracker. However, the association done at sensor level can be disregarded and then, the monosensor track updates should be considered as any plot or target report.

The association problem differs from one sensor to another as some of them impose some time constraints while others not. Thus, periodically scanning (or rotating) sensors, such as radars, can define a time window where to search the next target report; while, for example, ADS-B or multilateration systems do not implicitly have this constraint. It is true that nowadays, it is quite common to find multilateration systems (and also ADS-B systems) which are configured to deliver their target reports in a kind of periodical or regulated way (also known as throttle mode), instead of following a data driven approach, which was the common one when they were introduced fifteen to twenty years ago.

This process can be implemented in a stepwise approach. So, first a list of pre-selected target reports or system tracks can be identified and then, after some compatibility criteria are checked, the association to one or even to none of them, can be decided. There are association algorithms that allow a weighted association of several target reports to several tracks when the latter are found in what is called a “loss of resolution” situation (ref. ARTAS) and, as such, a clear decision cannot be taken.

In order to minimize the computational load, the pre-selection step tries to reduce the number of candidate plots or target reports to be associated with the track to be updated or vice versa, depending on the type of sensor which generated the target report. The pre-selection is generally implemented by both, a horizontal association window and a vertical one. The horizontal association window can take different shapes, such as a circle (the easiest one), a sector or a circle, a rectangular form, etc. Concerning the vertical association window, it usually takes the form of a height band.

If the sensor is a periodical one, then the track is the one that looks for the target report. In this case, the target report should also fall inside a time window centred in the expected time of the next update, which can be computed by taking into consideration the last update time and the known update period of the sensor. The track is then extrapolated to this next update time and the resulting position and height are used for centring the horizontal and vertical association windows, respectively.

On the other hand, if the sensor is not periodical, it is the plot the one that looks for the compatible tracks. In this case, there is no time window, as the target reports are asynchronous, and, as such, they do not follow any periodicity. Therefore, the plot search for the tracks that surround it. This search can also be done by the association windows

mentioned before. However, the size of these windows should cope with the fact that the tracks cannot be extrapolated to the target report time until they are pre-selected. Once this happens, the pre-selected tracks are extrapolated to the time of the plot or target report.

The second step deals with the pre-selected target reports or system tracks trying to identify either the candidate target report which is more likely to come from the track to be updated (periodical sensor) or the candidate track to which it is more likely that the target report to be associated belongs to (asynchronous sensor). In both cases, the situations that can be faced after the pre-selection are:

- There is no candidate target report or system track.
- There is only one candidate target report or system track.
- There are several candidate target reports or system tracks.

If there is no candidate target report or system track, then the track to be updated is in a miss state or coasted (periodical sensor), or the target report to be associated is either a false target report or it is a target report of a not yet confirmed new track (asynchronous sensor). In the second situation, the target report must be sent to the track initiation functionality, which will analyse whether it might belong to a new tentative track or not.

If there is only one candidate target report or system track after the pre-selection, then there are three association cases, namely:

- The target report belongs to the track.
- The target report comes from another target.
- The target report does not come from an actual target, that is, it is a false alarm.

The first situation is the ideal one, as the assignment will be correct. However, it is not the case for the other two and it is for those for which the association algorithm has to implement a strategy in order to minimize the probability of associating a target report, which might come either from another target or from a false alarm, to the system track. There are several strategies that can be followed to avoid these wrong associations that rely on a combination of criteria based on, for example:

- Horizontal distance.
- Vertical distance.
- Detection type.
- Identification data.

There are several possible strategies: one is to assign without any further assessment. However, there is a not negligible probability of associating a target report, which comes from a false alarm, to the system track. This case can be acceptable for target reports without identification. However, the usual case is to compute the likelihood of the association pair (target report-system track) and if it is above a certain threshold, then

confirming the association decision. In some cases, the problem can be addressed by managing several possible association hypotheses like in MHT filters [Bar-Shalom 95].

When there are several candidate target reports or system tracks, the complexity of the problem increases as the number of possible association cases grows with the number of pre-selected candidate target reports or system tracks. Even worse, if there is a situation of “loss of resolution” in which there are several system tracks, which pre-select some common target reports (periodical sensor), or the other way round, if there are several target reports to be associated (asynchronous sensor), which pre-select several common system tracks. All those situations require more sophisticated algorithms to confirm the association pairs that are more likely to be the correct ones.

Thus, there are different techniques that based on the distance between candidate target report or system track decide whether the association is plausible or not. Some examples of these association techniques are: the Nearest Neighbour (NN) algorithm, the Probabilistic Nearest Neighbour Association (PNNA), the Probabilistic Data Association (PDA), or the Joint Probabilistic Data Association (JPDA), for further information please refer to [Bar-Shalom 88].

Some of them weight the final score per association pair based on the identification information and some others take direct association decisions based on the AA or the Mode 3A, which might be then confirmed with geometric criteria. Moreover, there are sensors that provide information of their local tracking system which can also be used in difficult association situations. The preferred option is not to use this information as it couples the result of the performance evaluation tool with the sensor local tracking system. However, this choice can be left to the user who can allow it based on a configuration parameter.

The previous description is done from the point of view of a tracking system. When moving to the particular framework of a performance evaluation tool, the situation slightly changes as data are already available not only from the past but also from future of the association situation that needs to be solved. Additionally, there are no real-time constraints to perform the data association on the fly opposite to what happens with tracking systems. Therefore, the strategy can be partially changed as there is no immediate need to update a track which has to be quickly sent to the CWP in order to be displayed to the controller. This means that the information can be process in a different way as it will be presented in the following section.

6.7.2 Process description

The “data association” process goes through the list of time-ordered plots or target reports trying to find groups of plots or target reports belonging to the same target. This is done looking for some commonalities that allow them to be grouped together initially based on the identification information contained in the target reports themselves (e.g. Mode 3A code, AA, Flight Id, ACID, etc.) and then applying temporal and spatial compatibility criteria exploiting, for example, the ToA, the position, the Mode C, the height, etc. These criteria allow to confirm up to some extend the initial identity-based association. If there is no identification information, then the temporal and spatial criteria are used for performing the association. This is usually the case for non-cooperative sensors (see section 2.1.2 for further information).

Similar to what is proposed in [Besada 13], the association process is divided into two steps: the first one, which is carried out by this process, tries to group as many plots or target reports as possible, while the second one focusses on associating the remaining plots or target reports, that is, the complex associations.

However, there is a slight difference. In the first association step described in [Besada 13], only the very clear associations are performed by what is called the “gross association” and so, the rest are left to the “fine” association. The justification is on the constraints imposed by the further processing. In [Besada 13] there is a need for identifying straight segments with the requirement of minimizing the number of wrong associations as they might have a perverse effect in the subsequent bias estimation algorithms. This is the reason why any doubtful association is postponed to the “fine association”. This need comes from the particularities of the scenario where those algorithms are expected to work (i.e. en-route or TMA airspaces). In the case of this thesis, the airport layout adds some movement constraints that, in the end, allow the relaxation of the conditions on this “data association” process and, as such, the second step, which is carried out by the “association completion” process, only deals with the remaining cases once the data is bias corrected.

This process contributes to the “plot to trajectory association” function and ends with the creation of associated trajectories,¹⁶⁵ which bundle all the plots or target reports associated to the same target coming from either just one sensor or several of them. The first case is referred to as monosensor association, while the other is called multisensor association.

6.7.2.1 Monosensor association

This process groups together all the plots or target reports from a specific sensor which are likely to come from the same target.

The monosensor association might be completely performed by the PET, applying strategies like the ones commented in the previous section, or it might rely on the information provided within the monosensor track updates. Thus, there are sensors that perform a local tracking whose output can be forwarded to the following elements in the surveillance chain. This not only implies a local track number (LTN)¹⁶⁶ but also some kinematic information such as position or velocity.

Examples of sensor providing local tracking information are: SMR and multilateration systems. The latter might only provide extrapolated measurement instead of actual detections, which is not as convenient to the performance evaluation as actual measurements because the insights of the local tracking are usually unknown and, therefore, its statistical characterization might not be reliable. This behaviour is linked to the current trend of configuring multilateration systems as periodical or regulated sensors, see section 6.7.1.

¹⁶⁵ In some tools, the concept of “associated trajectory” is synonymous of “data chain” and the data association is referred to as “data chaining”.

¹⁶⁶ Generated by the surveillance sensor itself.

6.7.2.2 Multisensor association

This process groups together all the target reports from all contributing sensors that are thought coming from the same target. The “multisensor” association is eased if the target reports provide the identification of the reported target. Thus, the Mode 3A, the ICAO 24-bit address (i.e. AA), any LTN, or the callsign/flight ID/ACID are pieces of information that, conveniently used, help while identifying which target reports belong to which specific trajectory (target).

Additionally, other time and spatial information allows to confirm the initial identity-based association. This is usually based on temporal and spatial windows as already mentioned in section 6.7.1. The creation of a multisensor association track or chain is not as stringent as when done in an online tracker.

6.7.2.3 Multisensor trajectory selection

While managing multisensor associated trajectories, it is quite convenient to have the capability of selecting those ones that can be of interest for the subsequent processing, to reduce the computer load, and for testing purposes.

6.7.2.4 Target information retrieval

Finally, it is possible to enrich the multisensor associated trajectories with target information in case there is an identifier, such as the ICAO 24-bit address (i.e. AA) that uniquely identifies the target. With this piece of information, it is possible to obtain the aircraft type and model, the airline who owns it, and even a picture of it. In web pages like <http://www.airframes.org/> or <https://www.planespotters.net/>, it is possible to get this information. Once the aircraft type is available, then it is possible to retrieve information concerning the aircraft like: its physical dimensions or its wheelbase (see ANNEX II), the position of the different antennae on the fuselage (see ANNEX III), etc. This information is relevant for the subsequent data processing as it will be shown.

6.7.3 Processing details

6.7.3.1 Input data

The input data to this process is the full internal data structure, that is, the list of internal reports built by the PET and the data element list of finally selected time-ordered plots, target reports or monosensor track updates per sensor, and multisensor track updates from tracking systems.

6.7.3.2 Output data

The “data association” process performs some grouping on the sensor data and, so, each list of the finally selected time-ordered plots, target reports or monosensor track updates per sensor is now subdivided into smaller lists with those inputs that are assumed to come from the same target giving rise to monosensor trajectories. This is also done for the list of multisensor track updates managed by each tracking system. At PET level, the internal reports are grouped into multisensor trajectories per identified target. The not associated plots or target reports are kept in a list of un-associated data elements, which will be further processed by the “association completion” process (ref. 6.13) once more information is integrated and biases are removed.

6.7.3.3 Block diagram

The “data association” process can be graphically located in the “data preparation” processing block, as it is shown in the figure below:

Figure 6.7: Block diagram focussed on the “data association” process. Source: own elaboration.

6.7.4 Prototype implementation details

It is relevant to mention that data association is not one of the main research areas of this thesis and, consequently, not a lot of effort is devoted to this process. This means that this topic will remain as one of the future research lines (see section 9.3) as it is believed that there are other approaches than the currently found ones in the literature, which are mainly designed thinking on tracking system, that can be applied to improve and, in some way, decouple this task from the one done in tracking systems.

This process is implemented by a set of functions that progressively built monosensor associated trajectories per sensor, then multisensor trajectories at PET level as a combination of all the monosensor ones, and finally, enrich those multisensor trajectories with target information coming from external sources of information.

Therefore, each “sensor” class defines a method that performs the monosensor association on the data element list that it manages. This method is perfectly tuned to the information available in each sensor family. In this case, the local tracking information provided by SMR and multilateration sensors is exploited, while for the others (i.e. ADS-B) identification information, such as the AA, is used. All those methods are called by the PET on a sensor basis.

Consequently, the multisensor associated trajectories are build relying on the monosensor tracks generated by the sensors. The PET implements a function that decides on the more reliable sensor from the point of view of the monosensor trajectory generation and upgrades all its monosensor trajectories into multisensor ones. One point to be noted is that multisensor trajectories are manage at PET level, while monosensor ones are done at sensor level.

Once all the monosensor trajectories from the reference sensor are upgraded to multisensor trajectories, then the function goes through all the monosensor tracks of the remaining contributing sensors searching for other compatible monosensor tracks to associate with the already multisensor ones. This compatibility check is done mainly on

time overlap and geometric distance. If the monosensor trajectory is found compatible, then it is added to the list of monosensor trajectories per multisensor one. The data elements are then converted into internal reports of the PET by just creating a pointer from the internal report to the corresponding sensor data element. At the end of this process, all the monosensor trajectories, which have not been associated with a multisensor one, are upgraded to multisensor and copied to the PET list of multisensor trajectories. Every time a monosensor trajectory is associated to a multisensor one, all the information of the former is incorporated to the latter. This way, at the end of the process, each multisensor trajectory bundles all the available information of the target it represents.

Then, the PET implements a multisensor associated trajectory filter that just keeps the desired ones. This is useful for the rest of the processing as it can be focussed only on the information of interest either for testing or evaluation purposes.

Finally, for the selected multisensor associated trajectories, the identity information is used for retrieving data concerning the targets they represent. These data includes: aircraft dimensions, e.g. length, width, and height (see ANNEX II), the distance from the front wheel to the rear wheel axis, which is known as the wheelbase (see also ANNEX II), or the position of some antennae such as the GNSS or the SSR one (see ANNEX III). Later this information will be exploited by the target report location on the airport capability, the bias estimation algorithms, or the trajectory reconstruction.

6.8 Data location in airport

6.8.1 Process description

The target of the “data location in airport” process, as its own name mentions, is the location on one airport segment of each data element, no matter if it is a plot, a target report or a monosensor track update coming from a sensor, or a multisensor track update generated by a tracking system.

6.8.2 Processing details

6.8.2.1 Input data

This process is mainly driven by the execution of the algorithm that makes the assignment of an airport segment to each data element (i.e. the MHPF algorithm, in this case, see section 6.8.3.1 for details). Regardless of its implementation, this algorithm is expected to work on a multisensor associated trajectory basis. So, the main input of this process is the set of the selected multisensor associated trajectories and their associated internal reports.

6.8.2.2 Output data

The process goes through all the selected multisensor associated trajectories, and one-by-one, it calls the MHPF algorithm, which retrieves the list of data elements associated with that trajectory, and then performs the allocation of an airport segment to each data element. Additionally, it determines the “path” followed by the target.

6.8.2.3 Block diagram

The “data location in the airport” process can be graphically located in the “data preparation” processing block, as it is shown in the figure below:

Figure 6.8: Block diagram focussed on the “data location in the airport” process. Source: own elaboration.

6.8.3 Prototype implementation details

The implementation of this process relies on the algorithm that performs the mapping of every single data element into an airport segment. The proposal presented in this thesis, i.e. the MHPF algorithm, is described hereafter:

6.8.3.1 The multi-hypothesis path finder algorithm

The Multi Hypothesis Path Finder (MHPF) is the proposed algorithm in charge of assigning the proper airport segment to each data element associated to the time-ordered list of every selected multisensor associated trajectory. The algorithm runs in the forward direction, that is, from older data elements to newer ones. Its name comes from the fact that whenever the filter is not able to locate the data element in only one segment, this decision is postponed, and all the feasible hypotheses are opened and maintained until one of them is selected as the best hypothesis. Once this happens, all the pending data elements, which have not yet been assigned to an airport segment, are assigned with the airport segments belonging to the most likely hypothesis.

6.8.3.1.1 The MHPF algorithm update step

The MHPF algorithm has four states, depending on whether it is “located”, that is, one or more close segments have been already found in any of the searches, or not; and depending on whether it is “undecided”, that is, the algorithm is not able to perform the assignment of the data element to an airport segment, or not.

The algorithm enters in a loop over the list of all the data elements belonging to a multisensor associated trajectory. When the algorithm is started, it has no initial information concerning where the first element is located, so a global or full search is launched. This initial global search is repeated with every new data element that is processed until there is an actual location on one or several airport segments. In this moment, the MHPF becomes “located”. Once this happens, the search is relaxed and only performed on a limited area around those close airport segments. Therefore, the result after processing every data element is either an assignment to an airport segment or a

situation in which the assignment decision cannot be taken. The latter leads the algorithm to an “undecided” situation.

The target is to clarify all those doubtful assignments as the data elements in the time-ordered list are progressively processed. The strategy is based on opening hypotheses whenever the assignment decision cannot be taken. The hypothesis generation considers all the feasible possibilities, which are then pruned based on an independent path criterion. This criterion is the one that makes the complexity of this algorithm affordable. The hypotheses are kept opened until the doubtful situation is clarified.

The selection of the accepted hypothesis is triggered whenever a data element can be uniquely located in one airport segment and is based on the one with the highest score among the surviving hypotheses. Once this happens, the assignments proposed by the accepted hypothesis are confirmed and, consequently, all the data elements pending their airport segment are assigned. The algorithm finishes once all the data elements are processed and get their own airport segment.

6.8.3.1.2 The global airport segment search

Every time the MHPF algorithm starts processing the data elements of a multisensor associated trajectory, there is a need for locating them within the airport map. However, this is a blind search as there is no previous location information.

This global search is the one that carries out this initial location within the airport taking advantage of the knowledge of the airport layout. This search takes the list of all the airport segments and starts measuring the distance from each of them to the target report or track update under processing. The output of this search is the list of closest segments and the distance to each of them. This distance is computed to one of the extreme points, if the target report or track update cannot be perpendicularly projected on the segment, and directly to the segment, otherwise. The search is limited to a configurable distance from the target report or track update to keep the result meaningful.

The output of this initial search can drop three different situations:

- There is no close segment found.
- There is one close segment found.
- There are several close segments found.

If no close segment is found in the configured maximum distance threshold, then the target report or track update is directly assigned to airport segment “0”. This segment does not exist in the definition of the airport segments and it is used as the value for defining that the target report or track update is not yet located on an airport segment. It is relevant to mention that a target report or track update is considered as “located” in an airport segment when the former can be perpendicularly projected on the latter. Otherwise, the target report or track update is considered as “not located” in that segment and, if it cannot be located in any other, then, it has to be assigned to airport segment “0”. This value is assigned to all those target reports and track updates which are found in such a situation.

When the initial search finds just one segment, there are two possibilities, either the target report or track update is “located” in this segment or not. If it is located, then there is no doubt that it can be uniquely assigned to this segment, otherwise it must be assigned to segment “0”. All those data elements which are “located” and, as such, can be perpendicularly projected to the segment, allow to later compute what is called the “separation” distance (ref. 6.9). This distance is the cornerstone for the subsequent sensor-based bias estimation algorithm, so it must be carefully computed.

If the full search, finds several airport segments within the configured maximum distance threshold, then the MHPF algorithm cannot take a decision concerning where target report or track update is located. If this happens, then the hypotheses management process is triggered.

This full search can also be launched in case the algorithm is lost and it needs to re-“located” in the airport map.

6.8.3.1.3 The local airport segment search

Once the MHPF algorithm is “located”, there is no need to perform a full search for the following target reports or track updates in the list and, therefore, only a local or partial one is performed.

This local search considers the list of the previous close segments calculated by the previous search, either global or local, to look for the close segments that are used for extending the “independent” paths with the new reachable airport segment. The search of the new close segments is started from the list of previous close segments, which is then continued by those adjacent segments that are found following the airport layout (i.e. graph). The search is limited to a searching area whose surface is modulated by the covariance matrix of the position information of the target report or track update. At this level, no assessment of the search result is performed, that is, all the new airport segments will be considered in the further processing.

If more than one close segment is found, then the hypotheses management process is triggered.

6.8.3.1.4 The hypothesis management

Whenever the segment search finds more than one candidate airport segment, the MHPF algorithm cannot take a decision. So, it must postpone the decision concerning the location of this target report or track update. This situation justifies why a hypothesis management process has been introduced. To keep its workload under control, this process is divided in several phases, the first one is the hypothesis generation, then there is a first selection of hypothesis based on the feasible paths extensions, and finally, only the best hypothesis is chosen per “independent” path extension.

The hypothesis generation creates a hypothesis per airport segment, in case there was no previous hypothesis created, or as many children hypotheses per surviving parent hypothesis as new close segments were found. So, all the possible combinations of the surviving parent hypotheses and the new close segments are initially built.

Then, there is a phase which rates each generated child hypothesis with a score that is linked to the feasibility of the “path extension” associated to such hypothesis. The MHPF

algorithm, apart from keeping the list of airport segments per target report or track update, also maintains the “path” followed by the target which is consolidated every time a hypothesis management process is closed after selecting the best hypothesis. This “path” groups all the airport segments in which the target has been located all along its movement through the airport. The score assigned to the path extension is based on how feasible the proposed path is for going from the “independent” path extension of the parent hypothesis to the new close segment. The feasibility of the path is computed taking into consideration some assumptions and constraints, which are the base for this initial hypothesis pruning. Thus, the path is built under the assumption that the aircraft only moves forward, which is not always true as it can be the case when leaving the stand and a push-back operation is carried out. This is something that is noted as a future work (see section 9.3), when talking about trajectory classification. Another constraint is related to the turns or sudden movements that an aircraft can perform. Finally, this path management needs to be able to cope with “ping-pong” situations in the boundaries of segments mainly when the multisensor associated trajectory groups data from sensors suffering different biases. The fact that the “data allocation to an airport segment” process is a pre-requisite to the “bias estimation and cancellation”, prevents us to perform this path generation with unbiased data.

Even with this initial pruning, the number of generated hypotheses can exponentially grow when crossing intersections. In fact, in an initial design of this algorithm, no limitation was imposed to the hypothesis generation and, as such, there were situations in which, when biases heavily affected the inputs, the number of hypothesis grew exponentially and even with a pruning algorithm based on a weight function, the situation reached the point in which due to the high number of hypothesis, and the fact the pruning threshold was initially set to a fixed value, all the weights fell below this threshold and, consequently, all them were removed.

After analysing such situation, where the hypothesis generation considered and maintained all the possible combinations of feasible paths extensions (i.e. parent hypotheses) with all the closest segments found by the airport segment search. It was realized that it was possible to further prune most of these hypotheses based on the fact that all of them shared the same “independent” path extension (that is, the same succession of airport segments, regardless the assignment target report/track update-airport segment) and that the optimal assignment, per independent path, was the one showing the highest score, that is, the one with the shortest distance to the “independent” path extension. So, at the end, the issue was reduced to comparing the score of the unique surviving hypotheses per “independent” path, plus the option of staying in the initial segment from which the hypotheses management was launched. This approach made the issue affordable as the hypothesis generation, which initially led to the explosion of the number of hypothesis, now grows linearly instead of exponentially.

Based on the previous reasoning, a further pruning is proposed, which tries to identify the “independent” path extensions. These extensions are characterized by adding a new segment which is not previously found in the path extension of the parent hypothesis. The number of “independent” paths extensions can grow because of the extension of any of the previous ones. This is something usual when reaching intersections. Once all of them are computed, the set is enlarged with the path extension consisting in remaining in the first segment from which all path extensions start. This is usually the last segment added to the consolidated path at MHPF algorithm level. This hypothesis copes with the

probability that the target is staying at a certain position but due to the measurement noise it is located in various airport segments.

The next step is to select only the best hypothesis per “independent” path extension which is done based on the score given to each hypothesis.

Finally, the set of surviving hypotheses is maintained until the doubtful situation disappears, when this happens, a decision can be taken.

6.8.3.1.5 The “best” hypothesis selection

The “best” hypothesis selection decision is postponed until there is an undoubtful location of one of the following target reports or track updates, the probability of one of the hypotheses is above a certain threshold or there is a reason to force the decision based on the probability of the latest hypotheses. Whenever one of these situations happen, all the pending the assignments that were left undetermined are resolved.

In case the local search finds only one close segment, all the remaining hypotheses are evaluated and the one showing the best score is chosen. There are cases in which two hypotheses show a quite similar score that makes the decision error prone. It was realized that at certain intersections, the algorithm was identifying as the actual path, one which implied taking a bifurcation for then re-joining the actual path because the distances to the initial segments of the bifurcation and to the last ones of the re-joining were smaller than the ones to the corresponding straight segments. This issue is represented in the following picture:

Figure 6.9: Issues found at intersections

This issue was solved by introducing a correction factor that penalized the longest path when having similar global scores, so the shortest (and correct) path was favoured.

During all the time the MHPF algorithm is in the “undecided” situation and, therefore, the hypothesis management process is active, all surviving hypotheses are assessed at the

end of the processing of each target report or track update to check whether there is one which stands out from the rest. This is done by means of a probability value which is computed considering the scores of all the surviving hypotheses. If the value of any of them is above a configurable threshold, which is selectable by the user, the hypothesis management process is closed, and that particular hypothesis is selected as the “best” hypothesis and its segment assignments are confirmed.

Lastly, there are situations in which there is a need for triggering the selection of the “best” hypothesis. This happens when:

- Reaching the end of the data element list of the multisensor trajectory while the MHPF algorithm is still “undecided”, that is, there are still living hypotheses.
- The target is leaving the airport and so, the data elements become associated to segment “0”.
- The algorithm gets lost.

Whenever a situation like the described ones happens, the algorithm is abruptly forced to resolve the uncertain situation by providing the more likely hypothesis. Although the testing has shown a good performance in these situations, the assignment in these cases should be checked. This explains why it is recommended that the system prompts a warning to the user when ending like this.

6.8.3.2 Prototype implementation of the MHPF algorithm

The MATLAB prototype implements completely the logic described in section 6.8.3.1. Therefore, the set of selected multisensor associated trajectories is processed one-by-one by the MHPF algorithm, which retrieves the list of internal reports (i.e. internal data structure used for the representation of target reports and track updates within the PET), and one after another, it processes all the internal reports. This means that the first information that needs to be obtained is their position on the airport surface. This position and its uncertainty are then provided to the search functions which applying the search techniques already described in sections 6.8.3.1.2 and 6.8.3.1.3. These functions find all the close segments within a certain distance from the target report or track update.

Then, depending on the number of close segments that are found, the hypothesis management process might be launched, which counts on several functions for achieving its commitment. Thus, there is function for generating the initial hypothesis, which relies on another function (i.e. “path extension”) that computes the “path” to be followed for reaching the initial set of close segments and its feasibility.

The “path extension” function also provides the distance to the new close segment through the airport layout, the nodes which are traversed, and the length from the previous target report or track update to the new close segment following the chosen path. The function uses the airport nodes of the last airport segment assigned to the confirmed path of the target, which are referred as “initial” nodes, to start searching for those adjacent segments that can connect this last segment with the new close segment in case they are not directly connected. So, the first thing that is done is to obtain the nodes which are connected with each of the initial ones. It is worth noting that the algorithm can block nodes in order to prevent loops and processing the same nodes several times. These nodes will allow to identify the adjacent segments, which are then assessed as potential

extensions of the path, as there might be several paths that connect both segments. This assessment includes a “distance” criterion that checks that the search is kept limited to a certain distance to avoid endless searches or not optimal paths. Once a segment is considered reachable through the “distance” criterion, then a check on the feasibility of a potential turn is performed. If it is considered that the aircraft can turn there, the adjacent segment is added to the “path extension”. Then, the nodes of the adjacent segment become “initial” nodes and hence, they are added to the list of initial nodes. Similarly, the set of blocked nodes is also updated. The process ends when the list of initial nodes is completely processed. At that moment, the set of “path extensions” that have been found (in case of any) is processed and the one with the shortest distance is kept. If the function ends without a possible path extension, then it is assumed that the close segment is not reachable through that initial node. This information will be conveniently exploited by the hypothesis management to discard such hypotheses.

Next, there is a function for updating the set of surviving hypotheses. This function also relies on a function similar to “path extension” to analyse all the possible extensions to reach the new close segment from the last segment of the confirmed path. The “update” hypotheses function clones the previous surviving hypotheses and enlarge them with the new possibilities (segments and nodes) that the new target report or track update brings. As already mentioned, the number of surviving hypotheses is multiplied by the number of new close segments. However, the function also performs a pruning before releasing the set of feasible hypotheses. This pruning is based on the distance computed by the “path extension” function that, whenever a close segment is considered as not reachable, sets the distance to “infinite” as well as when it is considered that there is a not feasible turn between segments.

Once all the feasible hypotheses are identified, their paths are analysed in order to find the so-called “independent” paths or path extensions. This is done by another function which performs an iterative procedure that sets the longest path (in number of segments) as the first “independent” path and starts analysing whether the rest of them can be considered as sub-paths of this initial path or they are new “independent” paths. If a new “independent” path is found, the procedure is restarted with the remaining paths as now they can be a sub-path of several “independent” paths. This procedure is repeated until all the feasible paths are processed. Then, all the paths are classified either as an “independent” path or as a sub-path of one of those. The following step is to go through all the paths belonging to a certain “independent” path to find the one with the shortest distance. The hypothesis linked to this path will be the one selected for this “independent” path. This procedure is repeated for all the “independent” paths so, in the end, the number of surviving hypothesis equals the number of “independent” paths. By doing this, the computational load is kept affordable without loss of relevant information.

After every pruning procedure, the prototype performs an update of the probability of the surviving hypotheses. This information is later exploited by the hypothesis selection function which tries to determine the “best” hypothesis, which is not always possible. Depending on the situation, the selected hypothesis can be the only one remaining, the one with the highest probability or the unique one above a certain threshold. The second case happens when the hypothesis management process ends after a search that only provides a close segment and the third one, when there is a hypothesis that clearly stands out from the rest. If the selection function does not find any “best” hypothesis, the MHPF algorithm stays “undecided” and continues processing the following target report or track update aiming at reaching an undoubtful situation.

If the algorithm reaches the not “undecided” state, then the assumptions stated in the “best” hypothesis are translated to the algorithm and, consequently, all the target reports or track updates, whose location was postponed, become located in an airport segment. On one way or another, this will be the final result of the algorithm.

6.9 Movement model identification

6.9.1 Process description

The “movement model identification” process classifies each selected multisensor trajectory as one of a set of predefined “trajectory” types and then, it segments the trajectory based on the model that better matches the predominant surface movement. These models also belong to a predefined set of “surface movement” models (for the ones proposed in this thesis check 0). The trajectory segments are defined, and their associated models are identified taking into consideration the manoeuvres that the target undergoes while moving around the airport. A defined trajectory segment consists of one or several continuous airport segments. The manoeuvre that happens in each trajectory segment is a consequence of the accelerations that are suffered by the target. These accelerations can be either longitudinal or transversal. With the value of these accelerations, it is possible to perform an initial “mode of surface movement” (MoSM) segmentation of the multisensor associated trajectory that divides it into pieces, each of them assigned with only one movement model, the predominant one.

Therefore, the objective of this process is to estimate both accelerations, which are first used in the trajectory segmentation, and then, while processing the target reports or track updates for enriching their kinematic information. To achieve this objective, there is a need for obtaining some kinematic magnitudes that will allow to approximate the accelerations. Additionally, there is a not less important objective which is that the “separation” distance¹⁶⁷ has to be computed as it is a pre-requisite for the “bias estimation and cancellation” process.

Once the “path” followed by the target is known and the location of the target reports and track updates in an airport segment is obtained, it is also easy to locate them in an airport area (e.g. runway XX, taxiway YY, stand ZZ, platform area of terminal building A, movement area, manoeuvring area, etc.). Based on this information, it is possible to classify what type of trajectory the target is performing. This classification can take into account from what area the trajectory starts, to what area it arrives, what turns it performs, what speed it reaches, etc. The objective of this classification is to determine what operational phase the aircraft is performing as it will constraint the possible surface movement models. Just as an example, if it is known that the aircraft is parked in a stand of the terminal building and it starts moving, possibly, this movement will start with a segment in which the target is being pushed-back and, hence, it will be moving backwards. This movement usually continues with a turn which will leave the aircraft aligned in a taxiway or in an area of the platform from where it will start moving forward towards the runway from where it will take off. With a good “trajectory” classification, many of the subsequent data processing can be better tuned.

¹⁶⁷ The separation distance is a concept that will be further explained in section 6.10.7.2.1.

Please note that this initial MoSM segmentation might be further improved by using a segment classifier relying on the “mode probabilities” of the smoothers that are used as part of the trajectory reconstruction. However, this information is not yet available.

6.9.2 Processing details

6.9.2.1 Input data

The data that is consumed by the “movement model identification” process is the set of multisensor associated trajectories whose data elements have been assigned to an airport segment and whose path has been identified as provided by the “data location in the airport” process.

6.9.2.2 Output data

The main output of this process is the identification of the longitudinal and transversal manoeuvring areas that allows modelling the surface movements that the target undergoes while travelling along its trajectory through the airport. Other important computation is the “separation” distance, which will be added to the list of location data that this process generates and attaches to the data elements. Finally, another outcome of this process is the classification of the multisensor associated trajectory into one of a predefined set of “trajectory” types.

6.9.2.3 Block diagram

The “movement model identification” process can be graphically located in the “data preparation” processing block, as it is shown in the figure below:

Figure 6.10: Block diagram focussed on the “movement model identification” process. Source: own elaboration.

6.9.3 Prototype implementation details

The implementation that the prototype does of this process relies on several design decisions, namely:

1. The kinematic information needed for estimating or approximating the longitudinal and transversal accelerations is provided by sensors such as the SMR or the multilateration system. If this information is not available, then an algorithm based on the position of the target reports per segment must be implemented or an initial trajectory reconstruction based on the data from one of the contributing sensors might be used. However, these latest options have not been implemented.
2. The kinematic information of each multisensor associated trajectory is stored in data structures called “dynamic” segments.¹⁶⁸ There is a dynamic segment per airport segment that belongs to the trajectory path. Opposite to an airport segment, a dynamic segment is created based on the movement of the certain target, that is, the initial node, the final node, the segment orientation, etc. are computed taking into account the actual target movement, not the airport design. The information computed and stored in a dynamic segment can be used to identify the possible manoeuvres that the target is undergoing. Moreover, it is worth mentioning that any identified surface movement within a multisensor associated trajectory might extend from one to several dynamic segments.
3. The surface movement model is identified without precisely computing the longitudinal and transversal accelerations. The information that is needed is: on one side, the variations on the mean speed value between segments and on the other, the changes in the orientation between adjacent segments. Thus, a change in the mean speed of two adjacent segments, reflects a longitudinal acceleration, while a change in the orientation of the segments anticipates a transversal acceleration that leads to a turn, which can be characterized with a turn radius and a turn centre.
4. The multisensor associated trajectories are processed on an airport segment basis taking into consideration the order in which they are traversed by the target.

So, the algorithm starts retrieving the set of target reports and track updates allocated to the first segment of the trajectory path, then it extracts their kinematic information and finally, it computes the mean and variance of the ground speed (V) of all of them. This information allows to assess whether there has been a significant change in the ground speed between segments, if this happens, then it is understood that around the boundary of those segments there has appear a longitudinal acceleration leading to a longitudinal manoeuvre. Otherwise, the target is assumed either to continue with a “reasonably” constant speed or to stand still in a certain position.

¹⁶⁸ Note that a dynamic segment is a data structure linked to the trajectory, opposite to what has been referred to as “segment” or “airport segment” that is each of the pieces from which the airport map is built. Both data structures are coupled as a “dynamic segment” is created from the information stored in one and only one airport segment.

The prototype identifies different aircraft ground movements types related to longitudinal manoeuvres, namely:

- **Undetermined:** there is no clue that allows to classify the longitudinal ground movement as any of the others. This is normally related to the lack of context information, that is, either from the previous segment or from the next one.
- **Accelerating:** the aircraft is increasing its speed more than a predefined threshold compared to its previous or next segment (i.e. $\Delta V \geq V_{Th}$).
- **Decelerating:** the aircraft is decreasing its speed more than a predefined threshold compared to its previous or next segment (i.e. $\Delta V \leq -V_{Th}$).
- **Constant speed:** there is no significant change in the aircraft speed compare to the previous or next segment (i.e. $-V_{Th} \leq \Delta V \leq V_{Th}$).
- **Stopped:** the aircraft speed is close to zero (i.e. $V \leq V_{Th_Stopped}$).

Where:

$$\Delta V = V_{current} - V_{previous} \quad (6.10)$$

Being:

- $V_{current}$: ground speed on the current segment.
- $V_{previous}$: ground speed on the previous segment in the trajectory path.
- V_{Th} : user-defined threshold that determines when a change in ΔV is considered as caused by a longitudinal acceleration.
- $V_{Th_Stopped}$: user-defined threshold that identifies a standing aircraft.

There are situations in which it is not possible to identify the longitudinal ground movement when just going forward, that is, comparing the speed of the previous segment with the current one. In such a case, the prototype delays the decision concerning this segment until the next one is processed, if at that moment, the longitudinal ground movement cannot be identified, then it is marked as “undetermined”.

Next, the information concerning the changes in the orientation of adjacent segment is analysed. This analysis starts from the information stored in the airport segment, which does not consider the target movement. So, the first step is to identify whether each segment is aligned with the target movement or not. If it is already aligned, nothing needs to be done. Otherwise, it needs to be flipped. Once the segments are correctly aligned with the target movement, the analysis of the orientation of the adjacent ones can start as, at this point, they are completely coupled with the target heading (γ). This can lead to the following aircraft ground movements types related to transversal manoeuvres:

- **Undetermined:** there is no clue that allows to classify the transversal ground movement as any of the others. This is normally related to the lack of context information, that is, either from the previous segment or from the next one.

- **Constant heading:** the aircraft is going straight ahead although some tolerance (α_{Th}) is admitted (i.e. $|\Delta\alpha| \leq \alpha_{Th}$ or $2\pi - \alpha_{Th} \leq |\Delta\alpha|$)
- **Rotation left:** the aircraft is turning left around itself. This type is included but it is currently not used by the prototype.
- **Turn left:** the aircraft is turning left while moving forward (i.e. $-\pi \leq \Delta\alpha \leq 0$ or $\pi \leq \Delta\alpha \leq 2\pi$, and $\Delta\alpha$ is outside the range to be considered a constant heading movement).
- **Rotation right:** the aircraft is turning right around itself. This type is included but it is currently not used by the prototype.
- **Turn right:** the aircraft is turning right while moving forward (i.e. $0 \leq \Delta\alpha \leq \pi$ or $-2\pi \leq \Delta\alpha \leq -\pi$, and $\Delta\alpha$ is outside the range to be considered a constant heading movement).

Where:

$$\Delta\alpha = \alpha_{next} - \alpha_{current} \quad (6.11)$$

Being:

- α_{next} : orientation of the next segment in the trajectory path.
- $\alpha_{current}$: orientation of the current segment.
- α_{Th} : user-defined threshold that defines when a change in $\Delta\alpha$ is considered as caused by a transversal acceleration.

As before, there are situations in which it is not possible to identify the transversal ground movement or it might be wrongly classified when just processing in the forward direction, that is, when comparing the orientation of the next segment with the current one. Thus, when just looking forward, it is possible to classify a segment as turning when analysing the last segment of a constant heading transversal movement. This fact can be detected and corrected when checking backwards, that is, comparing the orientation of the current segment with the previous one. Only in case there is no information from both segments, the next and previous ones, the transversal ground movement of the current segment cannot be identified and, consequently, it is marked as “undetermined”.

The next step computes the turn radius and the turn centre in case the “previous” segment is classified as a “turning” one. This computation takes the initial node of the previous segment and the two nodes of the current one, and applying basic algebra obtains both values.

Once both longitudinal and transversal manoeuvres are classified and all the kinematic parameters are computed, the “dynamical” segment can be built. This process is then repeated for all the airport segments belonging to the multisensor trajectory path.

All the information generated by this process is also attached to the location information of the data elements. The stored information refines the one previously computed. This way, the data elements have a flag in order to indicate whether the target report or the

track update is within a longitudinal manoeuvring area, in a transversal one, in both or in none of them. This information allows to define the following MoSM:

- **Uniform motion model:** constant velocity (i.e. $a_L \cong 0$) and constant heading (i.e. $a_T \cong 0$).
- **Uniform accelerated motion model:** constant longitudinal acceleration (i.e. $a_L \cong \text{const}$) and constant heading (i.e. $a_T \cong 0$).
- **Uniform circular motion model:** constant velocity (i.e. $a_L \cong 0$) and constant turn rate (i.e. $a_T \cong \text{const}$).
- **Accelerated circular motion model:** constant longitudinal acceleration (i.e. $a_L \cong \text{const}$) and constant turn rate (i.e. $a_T \cong 0$).
- **Stopped model:** constant speed of zero (i.e. $|V| \leq V_{Th_stopped}$) and constant heading (i.e. $a_T \cong 0$).
- **U-Turn model:** constant speed of zero (i.e. $|V| \leq V_{Th_stopped}$) and constant turn rate (i.e. $a_T \cong \text{const}$).

Being:

- a_L : aircraft longitudinal acceleration approximated by ΔV .
- a_T : aircraft transversal acceleration approximated by $\Delta \alpha$.

The lack of a model for turning without speed is recorded as further work in section 9.3.

A crucial piece of information which is computed for convenience within this process is the “separation” distance, which is the distance of the target report of track update position to the centre line of the assigned airport segment as long as the perpendicular projection falls on the segment.

Concerning the trajectory classification, the prototype foresees the use of this information, but it is not currently performing the classification. However, the list of initially identified trajectory types includes the following values:

- **Undetermined:** whenever the trajectory type cannot be identified.
- **Parked:** the aircraft stands in a parking position in the platform, in a hangar or in a stand of the terminal building.
- **Arrival:** the aircraft is arriving to the airport and is going towards a parking position in the platform or a stand of the terminal building.
- **Departure:** the aircraft is leaving the airport and is going from its parking position to the runway where it will take off.
- **Arrival-Departure:** the aircraft is performing a full turnaround which includes an arrival leg, a variable duration stay at a parking position and a departure leg.

- **To hangar:** the aircraft is going from a parking position or a stand to a hangar. This can be seen as a movement that is internal to the airport.

At this point, the prototype is able to generate all the information needed to start the “bias estimation and cancellation” process, which is a key one and that will be presented in the following section.

6.10 Bias estimation and cancellation

6.10.1 Problem definition

As presented in CHAPTER 2, surveillance sensors measure the target position using different technologies that introduce different errors. Some of them are of random nature and some others present a more predictable behaviour. The latter is the case of biases or systematic errors. As already mentioned, these are constant or slowly variant errors that affect the measurements on a regular basis. It is this behaviour, the one that allows their estimation and subsequent cancellation.

The main issue with biased measurement is their effect on tracks or in our case, in reconstructed trajectories, as they destabilize the subsequent data fusion process, which is translated in split or erratic reconstructed trajectories. The fact that biases impact sensors in different ways, discards the assumption of aligned measurements, which is required by the data fusion algorithm and, wherefore, bias correction becomes a pre-requisite for data fusion processes, the same way as coordinate transformation.

The correct statistical modelling of measurement error components together with effective bias estimation algorithms enables the estimation and subsequent correction of a great part of the systematic errors, therefore enabling the alignment of all sensor measurements and measurement fusion in a common reference frame.

Thus, the aim is first estimating the “constant” biases of these different sensors, therefore enabling their removal. This aim will be supported by the modelling of the measurement sources presented in CHAPTER 2. This will be done by first introducing a classification of the bias terms that can be found in any measurement model keeping in mind that this classification will be coupled with the proposed bias estimation strategy. Then, different bias estimation algorithms are going to be presented within the overall bias estimation architecture. If these algorithms converge to reasonable values, the measurement can be corrected by removing them. This will guarantee an acceptable level of remaining biases.

6.10.2 Bias types

This section classifies the bias terms for the different sensor types listed in CHAPTER 2, into several groups based on the error source and on how they can be estimated together.

6.10.2.1 Sensor-based biases

Sensor-based biases affect the same way all the measurements from any detected target as these biases only depend on the sensor that generated the measurements. These are error that, once estimated, allow a general cancellation from all measurements. This is the

case of range and azimuth biases for radars and of time biases for any sensor that makes use of a time reference which might be not properly synchronized with the others.

Quite often, this type of biases are found together with target-based ones (see next section), which affect the measurements in exactly the same manner, preventing an independent estimation of both of them. These observability problems are very well known as it is mentioned, for example, in [Besada 04b], where an observability analysis is performed for all potential sensor combinations, or in [Soto 05], where this issue is taken into consideration all along the bias estimation process leading to algorithms that cope with them.

6.10.2.2 Target-based biases

Opposite to sensor-based biases, target-based ones only depend on the target, consequently, they are usually either depending on the target physical characteristics or on the performances of some airborne equipments.

In the first group, it is included:

- Location of ATC antennae, which are the ones used for transmitting the Mode A/C replies or the Mode S transmissions, either the squitters (i.e. short or long) or the replies to Mode S interrogations. These antennae are found in different locations along the fuselage of the aircraft usually in quite different locations, like close to the cabin and in the rear part of the fuselage close to the tail. Depending on the flight phase, the different ATC antennae are used. Thus, while flying they are used indistinctively, that is, one or the other, never together, in order to avoid reply losses due to occlusions in transmission like the ones caused by the wings. If the aircraft is on the ground, then normally, only the upper antenna is used as it will always have better signal propagation conditions.
- Location of GNSS receiver antennae, which are the ones used for receiving the satellite signals that allow the computation of the target position over the Earth. These antennae are usually located on the upper part of the fuselage closer to the centroid of the aircraft than the ATC ones. In any case, their position on the fuselage is different and, therefore, they must be considered separately. However, currently there is the possibility that the transponder corrects the reported position taking into account the position of the GNSS antenna or that the distance to it is reported to the ground by a DAP.

On the second group of potential target-based biases, there is:

- Transponder induced bias, which models the consequences of the allowed tolerance on the answering delay of the replies to Mode A/C or Mode S interrogations. This delay tolerance is of $\pm 0.5\mu\text{s}$ for Mode A/C replies and of $\pm 0.25\mu\text{s}$ for Mode S replies, which translated into distance implies a range error of $\pm 75\text{m}$ and $\pm 37.5\text{m}$, respectively.

There is a third group of target-based biases which are also sensor dependent as they are affected by how the sensor internal measurement logic works. This is the case of SMR sensors whose position extraction logic may be affected by occlusions (e.g. a wing might not be seen by the sensor, a building might be hiding the target or there is a blanking area that removes the echoes received from this area) or by the fact that the airport is not on

top of a perfect plane and, as such, there might be parts of it where the echoes from the targets are better received than others. This issue is presented and addressed in [García 03] or [García 05], where some techniques are proposed for blob-to-track association.

6.10.3 Bias estimation strategy

In the literature there are different approaches for bias estimation. Some examples focussed on the ATC domain are: [Cruz 92], [Besada 00], [Besada 04a], [Besada 12], or [Jarama 17] for the en-route area; [Besada 04b], [Soto 05], or [Besada 05] more focussed on the airport area; and, in a more generic context, one can find [Lin 04], [Lin 05], or [Rafati 07]. The more extended approaches are the ones in which all bias terms are estimated together with or without the target state vector, and those in which this process is done on a stepwise approach in which some of them are removed first and, sequentially, the remaining ones.

The research line followed in my research group (GPDS-SSR-UPM) has mainly been the second one, that is, the stepwise approach, and this is the base for the proposal done on this thesis. Additionally, the optimization of the processing time and the computer load has also guided these proposals in order to define algorithms which perform well while working with periodic updates avoiding the overhead of a batch processing. These estimators allow a constant accumulation of the information that is provided by every new measurement and, at the same time, offer the possibility of solving the bias estimator on-demand whenever is needed.

The bias estimation strategy is based on the one presented in [Soto 05] but modified taking into consideration the knowledge acquired while processing real data. This strategy is going to be outlined in this section, while its particular details are presented in section 0. It is worth noting that some assumptions on which the measurement models were based on have shown some deficiencies that might require an additional processing that has been highlighted as a future research area in section 9.3.

6.10.3.1 Stand-alone bias estimation

The bias estimation strategy relies on the availability of unbiased measurements coming from a sensor, the “reference” sensor. This is not usually the case as already highlighted when presenting sensor measurement models in CHAPTER 2. So, the first objective is trying to convert a biased sensor in an unbiased one. However, before there is a need for finding such sensor. The main criterion for this selection is the absence of target-based biases, at least theoretically, but there are some other features which are considered, like how well-known its measurement model is or how frequent its updates are.

Once the “reference” sensor is chosen, then the next step is to estimate its sensor-based biases. If its measurement model is mature enough, it is possible to build a bias estimator to estimate the bias values. One of the requirements of this estimator is that it has to estimate the bias values in a stand-alone manner. This can be done if it is possible to rely on some external information that can be exploited by the bias estimation algorithm in order to properly project the impact of the sensor biases in different directions, so they become observable and, as such, they can be estimated. In the airport case, this external information can be the airport map, and, in particular, the geographic information of the

centre lines of all the airport segments. If this estimator can be built, then it is possible to get an estimate of the bias values.

The final stage of this first objective is the removal of the estimated bias values from measurements delivered by the sensor.

6.10.3.2 Bias estimation based on measurement differences

The second objective is to estimate the biases of the rest of the sensors. This is done on a sensor-by-sensor basis. The strategy in this case is to take advantage of the unbiased measurements of the reference sensor to estimate the biases of the one under evaluation. This is implemented by a bias estimation algorithm that works with measurement differences, which are computed as the difference between the measurement provided by the biased sensor and the one provided by the reference one. Both measurements must be aligned in time, so the older one is extrapolated to the time of the newest. These measurement differences, if the reference sensor measurements have been properly bias corrected, are only a function of the biases of the sensor under evaluation. This fact allows to first estimate and then cancel them from the measurements of this under evaluation sensor.

6.10.3.3 Bias estimation architecture

The process is presented in the following figure:

Figure 6.11: High level description of the bias estimation. Source: own elaboration.

The innovative approach taken in this thesis relies on exploiting even more the airport information, not only the layout, but also some of the movements that are performed by the aircraft while arriving at the airport or while departing from it or, for example, other locations like the stands, the holding point or the stop bars. The location of the holding points or the stops bars usually implies an area where moving aircraft tend to stand for a while. This fact can be exploited as it will be detailed later. Another innovative proposal consists on exploiting the aircraft identity information to obtain the aircraft type and from the aircraft type obtain some of the unknown information that is considered a target-based bias, such as the ATC or GNSS antenna position. This way, this target-based bias terms can be removed from the bias estimation process easing the estimation of the other ones.

6.10.3.4 Time bias estimation

One important aspect that it is fundamental to bear in mind is the time bias. If the sensor is not well synchronized or the timestamp is not properly done, the information that it provides is then referred to another time instant, which is therefore translated in an along trajectory error.

When its information is fused with the one coming from other sensors, it introduces an instability along the longitudinal axis of the target movement. The impact of this bias in position is modulated by the aircraft ground speed, so if there is no ground speed this impact disappears. This is important to keep in mind while estimating the other target-based biases. The main issue with this bias is that it is difficult to estimate as it is usually coupled with other target-based bias, such as the GNSS or ATC antenna position. The estimation can be done once the measurements are bias corrected, so the only remaining error is the one due to the time bias.

The strategy that is followed in this thesis is to assume that the time reference is provided by the “reference” sensor, so this sensor has no time bias and then any remaining longitudinal bias in straight segments, where the aircraft is moving at a constant speed, is associated to an uncorrected time bias on the measurements coming from the remaining contributing sensors. If this is the case, the time bias of the other contributing sensor can be estimated and the ToA of the measurements corrected. Depending on the sensor, this can be a target-based bias, like for an ADS-B sensor due to the uncompensated latency, or a sensor-based one, like for multilateration systems.

6.10.4 Bias estimation methods

The processing of the measurements or the measurement differences is performed by estimation algorithms like the ones presented in [Soto 04] for estimating SSR biases, namely:

- Least Squares Estimator (LSE).
- Mean Square Error (MSE) estimator.
- Kalman filter.

These are all linear algorithms which admit a recursive form as it is also shown in [Soto 04]. For more details on such algorithms please refer to [Kay 93].

All these algorithms rely on a linear modelling of the estimation problem. If the latter can be achieved, then the output of them will converge to the actual value. It is expected that the two last ones, that is, the MSE estimator and the Kalman filter provide a better estimation than the LSE estimator as they not only integrate the measurements but also their covariance matrices.

6.10.5 Process description

The focus of the “bias estimation and cancellation” process is placed on the alignment of all the received measurement data at this point of the processing. This process is also

called registering in the literature, see [Cruz 92] or [Nabaa 99]. As already mentioned, this is a pre-requisite for any data fusion process as it is the reference trajectory reconstruction, and thereupon, the quality of the final trajectory reconstruction will heavily depend on the success in the bias removal performed by this process.

Thus, the objective of this process is to estimate and then cancel any constant or quasi-constant error from the plots, target reports or monosensor track updates. These errors can be found either in the time dimension or in the spatial one as commented in previous sections. So, different approaches can be followed. Depending on the coupling of these biases, the estimation strategy will be different. Regardless the chosen strategy, the final objective is ending with a set of data that are unbiased. This will be the main output to the next processing block, that is, the “trajectory reconstruction”, which will be detailed in CHAPTER 7.

This process together with the “noise estimation” and the “sensor measurement model” processing block (ref. 4.7.1) implements the “sensor model update” function that is introduced in section 3.6.2.3.

6.10.6 Processing details

6.10.6.1 Input data

The data that is consumed by the “bias estimation and cancellation” process is the set of multisensor associated trajectories whose path has been identified by the “data location in the airport” process and that its data elements have been assigned to an airport segment. Additionally, the “movement model identification” has assigned each of these data elements to one of the defined longitudinal and transversal manoeuvring areas and has also computed the “separation distance”, which will be exploited within this process.

6.10.6.2 Output data

The output of this process is the bias corrected set of data elements belonging to the group of selected multisensor associated trajectories.

6.10.6.3 Block diagram

The “bias estimation and cancellation” process can be graphically located in the “data preparation” processing block, as it is shown in the figure below:

Figure 6.12: Block diagram focussed on the “bias estimation and cancellation” process. Source: own elaboration.

6.10.7 Prototype implementation details

The overall bias architecture presented in section 6.10.3.3 is now particularized with the proposals introduced by this thesis. It also takes into account the more common sensors in the airport domain and what their relative importance is. The mathematical details are presented here after. Additionally, there are some proposals for the issues found when testing the initially proposed solutions and assumptions.

6.10.7.1 Selection of the reference sensor

The surveillance sensors that nowadays are commonly found¹⁶⁹ in an airport are:

- SMR radars: which are found in medium or large size airports.
- MLAT systems: which are found in medium or large size airports.
- ADS-B systems: which can be found in any airport due to their reduced cost.
- SSR radars: which are rarely found in airports as they usually have visibility constraints that prevent them to be located close to the airport.
- INCS sensors: which are still rarely deployed for tracking purposes except for coverage gap filling in case of windmills or any other research purpose.

Based on these considerations, the selection of the reference sensor can only be between SMR radars or MLAT systems. The rest cannot assume such a role as they are not part of the common surveillance sensor set of a medium/large size airport and in the case of ADS-B, its issues related to timestamping prevent it to be considered as the reference. The day in which these timestamping issues are solved, ADS-B will become an interesting candidate for this role.

Based on the fact that SMR radars present a more mature technology compared to MLAT systems and, linked to this, that they have a more generally accepted measurement model, the SMR is going to be selected as reference sensor as first option. In case the airport does not have an SMR sensor, then it would be an MLAT system the chosen one. The issue with MLAT systems is that apart from the fact that they do not have a commonly agreed measurement model, they have target-based bias terms like the ATC antenna location. However, as already mentioned, this bias term can be obtained, if the aircraft identity is known, and, consequently, it can be removed.

6.10.7.2 SMR bias estimation

6.10.7.2.1 Stand-alone calibration of the reference sensor

The stand-alone bias estimation that is proposed for the SMR relies on the one described in [Soto 05] for a real-time bias estimation system in the context of an airport surveillance data fusion system. The proposal in [Soto 05] assumed that aircraft follow taxiways axes. This seems to be a reasonable assumption as it can be seen in the following photograph:

¹⁶⁹ Sensors such as magnetic loops, miniradars or video camera systems are not considered as they are not widespread and additionally, they are barely used for tracking purposes.

Figure 6.13: Aircraft following centre line at Brussels airport. Source: own photograph.

Thus, using the information contained in a precise airport map, it is possible to calculate a separation distance (d). This distance measures how far the estimated position delivered by the SMR is from the centre line of the segment where the aircraft is located. There is a relationship between this distance and the SMR biases, which allows their estimation and subsequent cancellation. The following picture shows how the separation distance is computed:

Figure 6.14: Computation of the separation distance. Source: own elaboration.

In this thesis, the initial measurement model for the SMR radar has been modified in order to cope with some discrepancies that has been detected while testing the algorithm with real data. As already mentioned in section 2.2.3.3, there might be some issues when extracting the centroid of the aircraft by processing the blobs belonging to the target image. In that paper and in our initial model presented in section 2.2.3.3, this issue was assumed to be integrated within the noise components (i.e. $w_i^\rho(k)$, $w_i^\theta(k)$). Now, a new bias term is going to be included to cope with it, so the range and azimuth measurements can be modelled by the following expressions:

$$\rho_i^m(k) = \rho_i(k) + b_i^\rho + b_{i,j}^\rho(k) + w_i^\rho(k) \quad (6.12)$$

$$\theta_i^m(k) = \theta_i(k) + b_i^\theta + b_{i,j}^\theta(k) + w_i^\theta(k) \quad (6.13)$$

Where:

- $\rho_i(k)$, $\theta_i(k)$: true range (ρ) and azimuth (θ) values corresponding to the i -th SMR at time k .
- $b_i^\rho(k)$, $b_i^\theta(k)$: offset biases for the range (ρ) and azimuth (θ) corresponding to the i -th SMR at time k .
- $b_{i,j}^\rho(k)$, $b_{i,j}^\theta(k)$: offset biases for the range (ρ) and azimuth (θ) corresponding to the i -th SMR and the j -th target at time k .
- $w_i^\rho(k)$, $w_i^\theta(k)$: measurement noises for the range (ρ) and azimuth (θ) corresponding to the i -th SMR at time k .

The objective that is pursued by the introduction of $b_{i,j}^\rho(k)$, $b_{i,j}^\theta(k)$ is to compensate the effect of a slowly variant error component, which can be due to several causes, like:

- The lack of a well distributed set of multisensor associated trajectories.
- The presence of a not linear component in the bias terms.
- Troubles in the SMR extraction logic that computes the aircraft location (i.e. the centroid) due to some occlusions, a partial illumination of the target or a target manoeuvre.
- The impact of an airport which is not located in a perfect plane on the target image detected by the sensor.

This issue is modelled as a target-based bias because the size and shape of the aircraft might have an important impact on the unmodelled effect, which also depends on the sensor. Hence, it will be estimated per trajectory once the sensor-based biases are removed in case it is deemed necessary. This decision can be taken assessing the result of the initial correction, that is, the one based on the sensor-based biases. As introduced in [Chen 87], a “rambling factor” can be used for testing a non-zero mean of the separation distance. If the result shows a value above a threshold, it is understood that a further correction is needed. This correction is performed on a segment basis, so the concept of the “dynamic” segment introduced in the previous process is going to be extended to store

the estimation of these new bias terms, which are also applied on a segment basis, similar to the approach proposed in [Besada 09] that consists in computing and removing WAM biases per cell. The difference in this case is that part of the bias terms computed in a segment will be carried over to the downstream one. An in-depth analysis of the root cause of this effect is suggested as a future work in section 9.3.

The rest of the error model is the same as the one presented in section 2.2.3.3, which means that the measurement noise errors are considered a realization of a random white process with associated constant covariance matrix N given by:

$$N = \begin{bmatrix} \sigma_\rho^2 & 0 \\ 0 & \sigma_\theta^2 \end{bmatrix} \quad (6.14)$$

Where:

- $\sigma_\rho, \sigma_\theta$: standard deviation of the range noise ($w_i^\rho(k)$) and the azimuth noise ($w_i^\theta(k)$) corresponding to the i -th SMR.

Please note that the subscript i has been removed from N for the sake of clarity.

Some of the constraints that were highlighted on [Soto 05] are not present in the context of a performance evaluation tool. Thus, the requirement concerning the automatic location of the target in only one segment can be relaxed, as the MHPT algorithm removes the uncertainty concerning the location of the target by assigning univocally a segment to each measurement.

Once the measurement model is updated, the next step is to analyse how sensor-based biases are projected into the “separation” distance. This is done as shown below:

$$d(k) = H(k) \begin{bmatrix} b_i^\rho \\ b_i^\theta \end{bmatrix} \quad (6.15)$$

Where:

- $H(k)$ is the projection matrix of the sensor-based biases onto the “separation” distance, whose expression is:

$$H(k) = [-\sin(\alpha(k) - \theta_i^m(k)) \quad \rho_i^m(k) \cos(\alpha(k) - \theta_i^m(k))] \quad (6.16)$$

Being:

- $\alpha(k)$ is the segment orientation as depicted in Figure 6.14. Positive angles are defined from the North to the East.

Based on the previous information, an estimator can be built. As it is done in [Soto 05], the chosen estimator is a Best Linear Unbiased Estimator (BLUE), which takes the following recursive form:

$$\begin{bmatrix} \hat{b}_i^\rho \\ \hat{b}_i^\theta \end{bmatrix} = \left(\sum_k H^T(k) S^{-1}(k) H(k) \right)^{-1} \left(\sum_k H^T(k) S^{-1}(k) d(k) \right) \quad (6.17)$$

Where:

- $S(k)$: projected covariance matrix of the noises on “ d ” direction (i.e. $d(k)$ variance), given by:

$$S(k) = H(k) N H^T(k) \quad (6.18)$$

Once the “separation” distance ($d(k)$) is retrieved from the location data of the measurement, the projection matrix ($H(k)$) and the projected covariance matrix ($S(k)$) are calculated. After that, they can be included in the BLUE. For the sake of clarity, the summations are renamed as:

$$A = \sum_k H^T(k) S^{-1}(k) H(k) \quad (6.19)$$

$$B = \sum_k H^T(k) S^{-1}(k) d(k) \quad (6.20)$$

This way of expressing the estimator allows an easy incorporation of each of the measurements as they are processed. This is quite convenient in case the convergence of the estimator needs to be analysed.

Once all the measurements have been processed, then the estimator can be resolved, which in terms of A and B, looks like:

$$\begin{bmatrix} \hat{b}_i^\rho \\ \hat{b}_i^\theta \end{bmatrix} = (A)^{-1} B \quad (6.21)$$

Resolving the estimator provides a bias estimate that is used to directly cancel them from the measurements coming from the sensor, so if the sensor measurement model is precise enough, this bias estimation leads in mean to the real bias value, making the measures unbiased.

As previously mentioned, the values obtained might not allow to perfectly align all the measurements. This effect is usually more observable the further you check from the sensor. This is the reason why the processing might incorporate a target-based bias estimator that aims at mitigating this undesirable effect. It is designed to work per segment traversed by the target and to facilitate the convergence of the segment estimators the values computed in a previously traversed segment are propagated to the following one by means of a Bayesian combination. So, the estimator would be the same but this time only working on the measurements allocated in the segment that it is processing. The combination would look like:

$$\begin{bmatrix} \hat{b}_{i,S_j}^\rho \\ \hat{b}_{i,S_j}^\theta \end{bmatrix} = \left(A + C_{S_{j-1}}^{-1} \right)^{-1} \left(B + C_{S_{j-1}}^{-1} \begin{bmatrix} \hat{b}_{i,S_{j-1}}^\rho \\ \hat{b}_{i,S_{j-1}}^\theta \end{bmatrix} \right) \quad (6.22)$$

Where:

- C_{S_j} is the covariance matrix of the bias estimate computed for segment j .

If there is a need for reducing the weight of the previous estimate in the combination it is always possible to increase C_{S_j} by adding a covariance matrix.

At this moment, the corrected SMR measures (which may be assumed to be unbiased) can be used to estimate bias terms of other sensors. This assumption will be applied in the following sections.

6.10.7.2.2 Two or more SMR

Before moving to the following section, the case of having more than one SMR sensor is addressed. The proposal in this circumstance is to perform the same procedure as the one described above but with the difference that, once done, the measurements of each SMR sensor are considered as sensor-based bias corrected.

Then, a strategy similar to the one that is going to be presented in the following section is proposed. This strategy relies on a bias estimator based on measurement differences. In this case, the target is to estimate any potential time-bias that might exist between any SMR sensor and the reference SMR sensor that is assumed as providing the time reference.

6.10.7.3 Multilateration system bias estimation

The measurement model of the MLAT system assumes that there are no sensor-based biases and that there is a target-based bias related to the difference between the ATC antenna position and the centroid of the aircraft. However, the multisensor environment in which these sensors work introduces an additional time bias term (ref. 2.6.1) that copes with the lack of synchronization between the clock of the MLAT system, which is used for the timestamp of MLAT measurements, and the one used as the reference. This time bias is a sensor-based bias term as it should affect all the measurements the same way. This term is included in the initial measurement model of radar sensors (ref. 2.2.1.2) eqns. 2.07 and 2.08, but it is also applicable to multilateration systems.

Hence, the measurement model for MLAT position measures (X^m, Y^m) for the j -th aircraft is given by (ref. 2.4.1.2):

$$X_j^m(k) = X_j(k) - V_j^x(k)\Delta t_i + \Delta l_j^x + w_j^x(k) \quad (6.23)$$

$$Y_j^m(k) = Y_j(k) - V_j^y(k)\Delta t_i + \Delta l_j^y + w_j^y(k) \quad (6.24)$$

Where:

- $X_j(k), Y_j(k)$: true position of the j -th target in Cartesian coordinates at time k .
- $V_j^x(k), V_j^y(k)$: velocity vector of the j -th target in Cartesian coordinates at time k .
- Δt_i : time offset bias that represents a discrepancy between the MLAT internal clock and the time reference of the reference sensor.
- $\Delta l_j^x, \Delta l_j^y$: offset biases that depend on the antenna position for the j -th aircraft and also on its orientation (γ).

- $w_j^x(k)$, $w_j^y(k)$: measurement noise errors for the x and y components corresponding to the position measurement for the j -th aircraft at time k .

Δl_j^x and Δl_j^y represent the projection into both Cartesian coordinates, i.e. x and y , of the offset bias due to the discrepancy between the ATC antenna location and the aircraft position reference point or centroid (Δl_j). Similar to the SSR and ADS-B cases, Δl_j can also be modelled as a uniform random variable between -40m and 40m. Thus, their values are given by:

$$\circ \quad \Delta l_j^x = \Delta l_j \cos \gamma \quad (6.25)$$

$$\circ \quad \Delta l_j^y = \Delta l_j \sin \gamma \quad (6.26)$$

This can be graphically seen in the following picture:

Figure 6.15: Antenna position induced bias. Source: own elaboration.

It is worth noting that nowadays Δl should be a known value provided by the aircraft manufacturer, if this is the case, it is referred as $\overline{\Delta l_{ATC}}$.

Δt_i is the time bias whose effect is propagated to the x and y position by means of the ground speed components in both directions (i.e. $V_j^x(k)$, $V_j^y(k)$). This effect is totally coupled with the one described by Δl_j^x and Δl_j^y in straight segments. This is graphically shown in the following picture for a $\Delta t_i > 0$:

Figure 6.16: Time offset effect. Source: own elaboration.

Taking into consideration that the estimation strategy differs depending on the availability of the value of Δl , that is, whether $\overline{\Delta l_{ATC}}$ is known or not, and also on the percentage of aircraft with a known $\overline{\Delta l_{ATC}}$, a slightly different local bias estimator is proposed in each of the cases.

In case $\overline{\Delta l_{ATC}}$ is unknown, the objective of the local bias estimator is to estimate Δl and the selected bias estimation method is based on measurement differences [Besada 04a]. This method works on a trajectory basis and, hence, it expects the MLAT measurements to be associated to a multisensor associated trajectory. The measurement differences are obtained calculating the difference between the position provided by the MLAT sensor and the position coming from the reference unbiased sensor, in this case, a SMR. In consequence, the bias terms present in the measurement differences will be related exclusively to the pair “MLAT system-aircraft” and so they must be locally estimated.

These two positions need to be aligned in time by extrapolation (e.g. using a velocity estimation coming from the local tracking of the SMR or the MLAT system, or from a monosensor reconstruction in case it is performed after correcting the biases of the SMR, which is our choice). In this case, the chosen estimator is also a BLUE. The aircraft dependent x and y offsets ($\Delta x(k), \Delta y(k)$) at time k can be expressed in terms of Δl , assuming negligible noises, as:

$$\begin{bmatrix} \Delta x(k) \\ \Delta y(k) \end{bmatrix} = \begin{bmatrix} x_{MLAT}(k) - x_{SMR}(k) \\ y_{MLAT}(k) - y_{SMR}(k) \end{bmatrix} = H(k)\Delta l \quad (6.27)$$

Where:

- $H(k)$: projection matrix of the bias on the Cartesian coordinates, whose expression is:

$$H(k) = [\sin \gamma(k) \quad \cos \gamma(k)]^T \quad (6.28)$$

With:

- $\gamma(k)$: aircraft axis orientation at time k .

Based on the sensible assumption that aircraft move following the segment axis, as shown in Figure 6.13. This matrix can be refined using the segment orientation, i.e. $\alpha(k)$, instead of the target orientation, whenever the target is located in a straight segment. This information is provided by the “movement model identification” process. Taking this into consideration, the refined projection matrix will be:

$$H(k) = [\sin \alpha(k) \quad \cos \alpha(k)]^T \quad (6.29)$$

Opposite to what it is highlighted in [Soto 05], there is no problem with the fact that the target could be located into several segments as the previous process has removed the uncertainty concerning the segment where the target is located. Additionally, it provides the information concerning the manoeuvres that the target might be performing, this allows the selection of those measurement differences that are in straight segments and, therefore, only the latest expression of $H(k)$ is used for the computation of the estimate avoiding some undesirable effects. The algorithm also applies a safeguard area whenever a straight segment ends with a transversal manoeuvre, which depends on the target length.

Thus, the implemented BLUE has the following expression:

$$[\Delta \hat{l}] = (A)^{-1} B \quad (6.30)$$

and A and B have a similar expression as before:

$$A = \sum_k H^T(k) S^{-1}(k) H(k) \quad (6.31)$$

$$B = \sum_k H^T(k) S^{-1}(k) \begin{bmatrix} \Delta x(k) \\ \Delta y(k) \end{bmatrix} \quad (6.32)$$

Where:

- $S(k)$ is the covariance matrix of the measurement difference, which can be obtained as:

$$S(k) = \begin{bmatrix} \sigma_{x,MLAT}^2 & 0 \\ 0 & \sigma_{y,MLAT}^2 \end{bmatrix} + G(k) \begin{bmatrix} \sigma_{\rho,SMR}^2 & 0 \\ 0 & \sigma_{\theta,SMR}^2 \end{bmatrix} G^T(k) \quad (6.33)$$

With:

- $\sigma_{x,MLAT}$, $\sigma_{y,MLAT}$: standard deviations of the MLAT measurement noise in the x and y directions. This information might be provided by the sensor in I020/500 subfield #2, which can also include the covariance between x and y (i.e. σ_{xy}).
- $G(k)$: projection matrix of the SMR noises on the Cartesian coordinates for the k -th measurement.

When $\overline{\Delta l_{ATC}}$ is available, the local bias estimator aims at estimating Δt_i instead of Δl . So, the bias estimator is modified in order to integrate the new information (i.e. $\overline{\Delta l_{ATC}}$) and proceed to the estimation of Δt_i .

In this case, the chosen estimator is also a BLUE. The aircraft dependent x and y offsets ($\Delta x(k), \Delta y(k)$) at time k can be expressed in terms of Δt_i and $\overline{\Delta l_{ATC}}$, assuming negligible noises, as:

$$\begin{bmatrix} \Delta x(k) \\ \Delta y(k) \end{bmatrix} = \begin{bmatrix} x_{MLAT}(k) - x_{SMR}(k) \\ y_{MLAT}(k) - y_{SMR}(k) \end{bmatrix} = H(k)\Delta t_i + \begin{bmatrix} \sin \gamma(k) \\ \cos \gamma(k) \end{bmatrix} \overline{\Delta l_{ATC}} \quad (6.34)$$

Reordering the previous expression, the measurement difference can be corrected to incorporate the prior knowledge of the difference between the ATC antenna location and the aircraft centroid (i.e. $\overline{\Delta l_{ATC}}$):

$$\begin{bmatrix} \Delta x(k)' \\ \Delta y(k)' \end{bmatrix} = \begin{bmatrix} \Delta x(k) \\ \Delta y(k) \end{bmatrix} - \begin{bmatrix} \sin \gamma(k) \\ \cos \gamma(k) \end{bmatrix} \overline{\Delta l_{ATC}} = H(k)\Delta t_i \quad (6.35)$$

Where:

- $H(k)$: projection matrix of the time bias on the Cartesian positions, whose expression is:

$$H(k) = [-V^x(k) \quad -V^y(k)] = [-V(k)\sin \gamma(k) \quad -V(k)\cos \gamma(k)]^T \quad (6.36)$$

With:

- $\gamma(k)$: the aircraft axis orientation at time k .
- $V(k)$: the ground speed of the target at time k .

As previously stated, this projection matrix can be refined using the segment orientation ($\alpha(k)$) instead of the target orientation, whenever the target is located in a straight segment, which leads to the following expression:

$$H(k) = [-V_j(k)\sin \alpha(k) \quad -V_j(k)\cos \alpha(k)]^T \quad (6.37)$$

Thus, the implemented BLUE has the following expression:

$$[\Delta \hat{t}] = (A)^{-1}B \quad (6.38)$$

and A and B have a similar expression as before (eqns. 6.31 and 6.32) with the following only change:

$$\begin{bmatrix} \Delta x(k) \\ \Delta y(k) \end{bmatrix} \xrightarrow{\text{into}} \begin{bmatrix} \Delta x(k)' \\ \Delta y(k)' \end{bmatrix} \quad (6.39)$$

and $S(k)$ keeps the same expression as in eqn. 6.33.

Once the measurement model has be stated and the proposed local bias estimation methods have been presented, then, the overall processing can be described.

The first thing that is clarified before processing all the MLAT-supported multisensor associated trajectories is whether $\overline{\Delta l_{ATC}}$ is generally available or not. This information will decide the estimation strategy and the local bias estimator that is going to be used for the MLAT measurements of the different trajectories.

If the scenario does not include enough aircraft with a known $\overline{\Delta l_{ATC}}$, then the local bias estimator based on estimating Δl will be used for all the trajectories, otherwise the one aiming at estimating Δt will be used for the trajectories with a known $\overline{\Delta l_{ATC}}$ and for the others, once they are time-bias corrected, the estimator aiming at estimating Δl will be used. Both strategies are going to be detailed hereafter while describing the overall process.

Once the bias estimation strategy is selected, every time a new MLAT target report is processed, the bias estimation algorithm analyses whether this target report can be used for updating the MLAT local bias estimator. This analysis checks whether the target is within a transversal manoeuvring area. If it is the case, the MLAT target report is discarded; otherwise, the algorithm search for the closest (in time) target report from the reference sensor, the SMR in this case, and then taking advantage of the monosensor reconstruction performed on the SMR plots, extrapolates the SMR plot into the time of the MLAT target report.

Then, depending on the bias estimation strategy, the algorithm will use the appropriate formulae to build the measurement difference first and then, the projection matrix ($H(k)$) and the projected covariance matrix ($S(k)$).

All this information is used to feed the corresponding BLUE estimator, which accumulates the information coming from all the computed measurement differences from MLAT target reports located within non-transversal manoeuvring areas in the already presented A and B expressions. Once, all the measurement differences are processed by the BLUE, it can be resolved using the final values of A and B .

If the strategy is the one set when there are not enough known values of $\overline{\Delta l_{ATC}}$ (to later on obtain a good global time bias estimate for the sensor), then $\Delta \hat{l}$ estimates the difference between the ATC antenna position and the centroid of the aircraft together with the effect on position of any time bias. The value of $\Delta \hat{l}$ is used to correct the position of the MLAT measurements belonging to this particular aircraft. In this case, the correction is carried out as shown hereafter:

$$\begin{bmatrix} x(k) \\ y(k) \end{bmatrix} = \begin{bmatrix} x_{MLAT}(k) \\ y_{MLAT}(k) \end{bmatrix} - H(k)\Delta \hat{l} \quad (6.40)$$

The measurement covariance matrix (N) is enlarged to cope with the uncertainty in the target orientation that is used in the projection matrix (i.e. $H(k)$) for those measurements located in a transversal manoeuvring area. The value of the target orientation is taken for the SMR (monosensor) reconstruction performed with the SMR measurements as well as its variance (σ_γ^2). This is reflected in the following expression:

$$N = \begin{bmatrix} \sigma_{x,MLAT}^2 & 0 \\ 0 & \sigma_{y,MLAT}^2 \end{bmatrix} + P(k)\sigma_\gamma^2 P^T(k) \quad (6.41)$$

Where:

- $P(k)$: projection matrix of the uncertainty in the target orientation into the Cartesian coordinate framework, whose expression is:

$$P(k) = [\Delta \hat{l} \cos \gamma(k) \quad -\Delta \hat{l} \sin \gamma(k)]^T \quad (6.42)$$

In case the number of trajectories with a $\overline{\Delta l_{ATC}}$ is large enough, the process is divided into several stages as described here below:

1. Calculating $\Delta \hat{t}$ only for those targets, which should be the majority, whose $\overline{\Delta l_{ATC}}$ is known. This is done by a local estimator like the one described by eqns. 6.35-38, which runs per trajectory that is MLAT supported.
2. Obtaining $\Delta \hat{t}_{SENSOR}$ by means of a global estimator that uses each of the $\Delta \hat{t}$ values coming from the local estimators executed in the previous stage as a measurement. This estimator is run when all the local estimators of the targets with a known $\overline{\Delta l_{ATC}}$ are executed, and it looks like:

$$\Delta \hat{t}_{SENSOR} = \left(\sum_j \frac{1}{\sigma_{\Delta t}^2(j)} \right)^{-1} \left(\sum_j \frac{\Delta t(j)}{\sigma_{\Delta t}^2(j)} \right) \quad (6.43)$$

$$\sigma_{\Delta \hat{t}_{SENSOR}}^2 = \left(\sum_j \frac{1}{\sigma_{\Delta t}^2(j)} \right)^{-1} \quad (6.44)$$

If the estimation is good enough, that is, $\sigma_{\Delta \hat{t}_{SENSOR}}^2$ is below a certain threshold, $\Delta \hat{t}_{SENSOR}$ is applied to all MLAT target reports, so their ToA is corrected and the uncertainty in the estimation is propagated to the measurement covariance matrix applying the same speed used for the extrapolation of the reference sensor measurement.

$$N'_{MLAT} = \begin{bmatrix} \sigma_{x,MLAT}^2 & 0 \\ 0 & \sigma_{y,MLAT}^2 \end{bmatrix} + \begin{bmatrix} V_x^2(k) & V_x V_y \\ V_x V_y & V_y^2(k) \end{bmatrix} \sigma_{\Delta \hat{t}_{SENSOR}}^2 \quad (6.45)$$

At this point, all the trajectories belonging to aircraft with a known $\overline{\Delta l_{ATC}}$ are fully characterized. It only remains the correction of the measurement position, which is done by:

$$\begin{bmatrix} x(k) \\ y(k) \end{bmatrix} = \begin{bmatrix} x_{MLAT}(k) \\ y_{MLAT}(k) \end{bmatrix} - H(k) \overline{\Delta l_{ATC}} \quad (6.46)$$

Like in the case where most of the $\overline{\Delta l_{AC}}$ are unknown, the measurement covariance matrix is enlarged to cope with the uncertainty on the target orientation for those measurements located in transversal manoeuvring areas as presented before:

$$N = N'_{MLAT} + P(k) \sigma_{\gamma}^2 P^T(k) \quad (6.47)$$

3. The final stage aims at estimating the Δl of the remaining MLAT trajectories with unknown $\overline{\Delta l_{ATC}}$. This is done by the local bias estimator aiming at Δl , that is $\Delta \hat{l}$, but the difference now is that all the measurements of these trajectories have already been time bias corrected, so the only remaining bias is Δl .

If the proposed measurement model is accurate enough, all the corrected measurement would be now referred to the centroid of the aircraft and also time aligned with the reference sensor. The results obtained by using this algorithm are presented in CHAPTER 8.

6.10.7.4 ADS-B system bias estimation

The measurement model that is proposed for ADS-B is similar to the one presented for MLAT systems except for the presence of the “uncompensated latency” (UL), which is a target-based time bias already described in section 2.3.2.3, instead of a sensor-based time bias like the one introduced when describing the MLAT measurement model.

Similar to the case of the MLAT system, the ADS-B measurement model also includes a target-based position bias that represents the difference between the location of the GNSS antenna and the centroid of the aircraft. Note that the position of this antenna is usually different than the position of the ATC one.

Hence, the measurement model of ADS-B position measures (X^m, Y^m) for the j -th aircraft at time k can be assumed as (ref. 2.3.2.3):

$$X_j^m(k) = X_j(k) - V_j^x(k)\Delta t_{UL,j} + \Delta l_j^x + w_j^x(k) \quad (6.48)$$

$$Y_j^m(k) = Y_j(k) - V_j^y(k)\Delta t_{UL,j} + \Delta l_j^y + w_j^y(k) \quad (6.49)$$

Where:

- $X_j(k), Y_j(k)$: x and y components of the true stereographic projection of the position of the j -th target provided by ADS-B at time k .
- $V_j^x(k), V_j^y(k)$: velocity vector of the j -th target projected in the stereographic plane at time k .
- $\Delta t_{UL,j}$: time bias due to the transponder UL for the j -th aircraft.
- $\Delta l_j^x, \Delta l_j^y$: offset bias components which depend on the GNSS antenna position for the j -th aircraft and also on its orientation (γ).
- $w_j^x(k), w_j^y(k)$: measurement noise errors for the x and y components corresponding to the position measurement for the j -th aircraft at time k .

The proposal in this case is to apply a similar approach to the one followed on the MLAT case, that is, a local bias estimator (i.e. $\Delta \hat{l}$ or $\Delta \hat{t}$) based on the processing of measurement differences with respect to a reference sensor, like the one described by eqns. 6.27-6.33. However, as it is assumed that there is no sensor-based time bias, the local bias estimator is chosen since the beginning based on the availability or not of $\overline{\Delta l_{GNSS}}$ (i.e. the distance

from the position of the GNSS antenna to the centroid of the aircraft) for the trajectory being process. Like in the MLAT case, it is expected that $\overline{\Delta l_{GNSS}}$ is known for most of the aircraft. A difference compared to WAM is that the navigation system can also correct the effect of the difference between the position of the GNSS antenna and the centroid of the aircraft, and additionally, it can also provide an estimation of this difference in one of the BDS registers of the Mode S transponder. In such a case, the UL can be directly estimated.

If $\overline{\Delta l_{GNSS}}$ is unknown and it has not been already removed by the navigation system, then the strategy aims at finding $\Delta \hat{l}$, which will include both, the difference between the position of the GNSS antenna and the centroid of the aircraft, and the time bias effect on the reported position. This value is then removed from the ADS-B position measurements belonging to the multisensor associated trajectory that is being processed. This is done by applying eqns. 6.40-41.

On the other hand, if $\overline{\Delta l_{GNSS}}$ is available or it has already been compensated by the navigation system, then the approach is to directly obtain $\Delta \hat{t}_{UL}$, as the position bias is assumed to be already compensated. This value is then removed from the ToA of all the ADS-B measurements belonging to the multisensor associated trajectory that is being processed.

Something that has been detected while processing ADS-B real data is that the position data do not seem to come from independent measurements. This fact represents a trouble for the processing as it introduces a correlation between measurements that is not present in the rest of the sensors. The selected choice in this thesis has been to detect them and not use them in the bias estimator.

As before, if the measurement model is good enough, the bias estimator gives an estimate of either the difference between the GNSS antenna position and the centroid of the aircraft plus the effect on the position due to the UL or directly an estimation of the UL.

As shown above, if $\overline{\Delta l_{GNSS}}$ is known for the aircraft, then the UL can be estimated. It is significant to mention that nowadays, this time bias is the warhorse while fusing ADS-B reports with the data coming from any other surveillance technology that are used for feeding tracking systems, no matter if they are for air or ground surveillance. So, if the estimation is good enough, it can be forwarded to the downstream surveillance system, e.g. the air tracker in case of a departing fight, which can use it for improving the accuracy of ADS-B position measurements.

Currently, systems like ARTAS [ARTAS] allow the definition of a mean timestamp bias value and its standard deviation as a mitigation mean for the UL. Both values are configurable per MOPS version (i.e. V0, V1 or V2 corresponding with DO-260, DO-260A and DO-260B, respectively). Therefore, the same mean value is subtracted from the ToA of all ADS-B measurements coming from transponders that share the same MOPS version, while the standard deviation is used for propagating the uncertainty concerning this mean value to the other components of the ADS-B measurement vector, mainly the position and the course. Hence, the possibility of forwarding the value of the UL to trackers represents an interesting opportunity for the PET as a complementary system for trackers.

6.10.7.5 Secondary radar bias estimation

The secondary radar (SSR) concept includes both, conventional SSR and Mode S radars. They are addressed together because they share their measurement model except for some minor details, which will be conveniently highlighted. The measuring technique is the same in both cases, although Mode S radars, as more modern systems, are expected to provide better performances.

6.10.7.5.1 Secondary radar measurement model

The measurement model has five bias terms, three sensor-based and two target-based, and an additional time bias, which accounts for any deviation between the internal clock that is used for the timestamping of the plots and the overall time reference that, in this case, it is assumed to be provided by the SMR clock.

The ones related with the sensor represent not only a range bias and an azimuth bias, as in the SMR case, but also the radar gain, which represents the linear variation of biases with range (ref. 2.2.2.3 and 2.2.4.3). The others, which are aircraft-based, represent the transponder induced bias (i.e. erroneously calibrated response delay), and the difference between the ATC antenna position and the centroid of the aircraft.

Hence, the expressions for the range (ρ^m) and azimuth (θ^m) measurements are modelled as:

$$\rho_i^m(k) = [1 + \epsilon_i^\rho(k)]\rho_i(k) + b_i^\rho(k) + \Delta R_j + f(\vec{V}_j(k), \Delta t_i) + \Delta l_j^\rho + w_i^\rho(k) \quad (6.50)$$

$$\theta_i^m(k) = \theta_i(k) + b_i^\theta(k) + f(\vec{V}_j(k), \Delta t_i) + \Delta l_j^\theta + w_i^\theta(k) \quad (6.51)$$

Where:

- $\rho_i(k), \theta_i(k)$: true range (ρ) and azimuth (θ) values corresponding to the i -th radar at time k .
- $\epsilon_i^\rho(k)$: sensor-based scale bias for the range (ρ) corresponding to the i -th radar at time k . This is the term that accounts for the radar gain.
- $b_i^\rho(k), b_i^\theta(k)$: sensor-based offset biases for the range (ρ) and azimuth (θ) corresponding to i -th radar at time k .
- ΔR_j : transponder induced error corresponding to the j -th aircraft. It is an aircraft-based offset bias.
- $\vec{V}_j(k)$: velocity vector of the j -th target at time k .
- Δt_i : time bias due to a lack of synchronization with respect to the time reference for the i -th SSR, which can be considered as another sensor-based bias.
- $\Delta l_j^\rho, \Delta l_j^\theta$: offset bias terms that depend on the ATC antenna position ($\overline{\Delta l_{ATC}}$) and the orientation (γ) of the j -th aircraft. Consequently, they represent aircraft-based bias terms.

- $w_i^\rho(k)$, $w_i^\theta(k)$: measurement noises for the range (ρ) and azimuth (θ) corresponding to the i -th radar at time k that are generally modelled as zero-mean white Gaussian random sequences with corresponding variances σ_ρ^2 and σ_θ^2 , which are assumed mutually independent of each other.

From the complete model presented in section 2.2.4.3, only the terms linked to the radar's eccentricity are omitted as they are assumed negligible due to the expected proximity of the secondary radar to the airport. Likewise, the radar gain $(\epsilon_i^\rho(k))$ is also considered negligible for the same reason. Moreover, the effect of a potential time bias on the range and azimuth measurements is introduced by means of the term $f(\vec{V}_j(k), \Delta t_i)$ for the sake of completeness. However, any computation of this time bias would be done in Cartesian coordinates as already described in section (ref. 2.2.1.2) eqns. 2.07 and 2.08.

In this case, sensor-based biases are processed in a global way, because they affect all the measurements from all the aircraft within the SSR coverage. Consequently, they will be estimated by a global bias estimator like the one presented for the MLAT time bias. However, the target-based biases are locally processed for each aircraft since they only affect this aircraft. Thus, a local bias estimator is also needed, one for each selected multisensor associated trajectory. This implies that the estimation process of all biases (dependent and non-dependent) needs to take care of this matter.

Therefore, the SSR bias estimation method is comprised of two estimators. The first estimator is a local one as it goes through all the measurement belonging to a multisensor associated trajectory, so the estimated parameters are both target and sensor dependent. Consequently, there is a local estimator per selected multisensor associated trajectory. Depending on the available information, this estimator will slightly change its layout and the estimated parameters.

The second SSR estimator is a global one as it takes advantage of the estimates provided by the local estimators to compute the sensor-based biases, that is, only bias terms depending on the SSR. This estimator will admit measurements, that is, local estimates with different dimensions, depending on the nature of the parameters estimated by the local estimators.

All this process is done in a sequential order. First, all the local estimators are executed and then, the global estimator is fed with their results. Compare to what it is stated in [Soto 05], there is no need for a third estimator that combines the previous information with the current one, as all the information is already available and, additionally, in a PET, there is no pressure on using the bias estimates as soon as they are available for the correction of the new incoming measurements. In fact, the bias cancellation is scheduled at the very end when all the available information is already integrated.

6.10.7.5.2 Local bias estimation

The implemented method for the local bias estimator is also based on measurement differences [Besada 04a] and its final expression will depend on the available information. As in the previous sections, the method consists in calculating the difference between the position provided by the SSR and the corrected measures coming from the (assumed unbiased) "reference" SMR. In consequence, the bias terms present in the measurement differences will be related exclusively to the SSR. As before, the two measurements also

need to be aligned in time by extrapolation. The x and y offsets at time k , i.e. $\Delta x(k)$ and $\Delta y(k)$, are expressed in terms of the i -th sensor and the j -th aircraft biases, assuming negligible noises, as:

$$\begin{bmatrix} \Delta x(k) \\ \Delta y(k) \end{bmatrix} = \begin{bmatrix} x_{SSR}(k) - x_{SMR}(k) \\ y_{SSR}(k) - y_{SMR}(k) \end{bmatrix} = H(k) \begin{bmatrix} b_i^\rho + \Delta R_j \\ b_i^\theta \\ \Delta l_j - V(k)\Delta t_i \end{bmatrix} = H(k) \begin{bmatrix} \delta R_j \\ b_i^\theta \\ \Delta l_j' \end{bmatrix} \quad (6.52)$$

This equation shows the projection on the Cartesian coordinates of all the identified bias terms. As it shows, the transponder induced bias (ΔR_j) cannot be distinguished from range bias (b_i^ρ) if there was only one target. So, the objective is to estimate their sum (δR_j). In a similar way, the presence of any uncompensated time bias (Δt_i) will be perfectly coupled with the position bias due to the difference in the position of the ATC antenna and the aircraft centroid (Δl_j), meaning that they initially have to be estimated together.

The bias projection matrix, $H(k)$, is more complex than in the previous cases, due to a larger number of estimated parameters. Its expression has the form:

$$H(k) = \begin{bmatrix} \sin \theta_i^m(k) & \rho_i^m(k) \cos \theta_i^m(k) & \sin \gamma(k) \\ \cos \theta_i^m(k) & -\rho_i^m(k) \sin \theta_i^m(k) & \cos \gamma(k) \end{bmatrix}$$

Again, a BLUE is also implemented as in the previous cases that, in the case of not knowing $\overline{\Delta l_{ATC}}$ (ref. 6.10.7.3), takes the following aspect:

$$\begin{bmatrix} \delta \hat{R} \\ \hat{b}^\theta \\ \Delta \hat{l} \end{bmatrix} = (A)^{-1} B \quad (6.53)$$

and A and B have a similar expression as before:

$$A = \sum_k H^T(k) S^{-1}(k) H(k) \quad (6.54)$$

$$B = \sum_k H^T(k) S^{-1}(k) \begin{bmatrix} \Delta x(k) \\ \Delta y(k) \end{bmatrix} \quad (6.55)$$

Where:

- $S(k)$: covariance matrix of the measurement difference, which is given by:

$$S(k) = G_{SSR}(k) \begin{bmatrix} \sigma_{\rho,SSR}^2 & 0 \\ 0 & \sigma_{\theta,SSR}^2 \end{bmatrix} G_{SSR}^T(k) + G_{SMR}(k) \begin{bmatrix} \sigma_{\rho,SMR}^2 & 0 \\ 0 & \sigma_{\theta,SMR}^2 \end{bmatrix} G_{SMR}^T(k) \quad (6.56)$$

With:

- $\sigma_\rho, \sigma_\theta$: standard deviations of the SSR and SMR measurement noise in the ρ and θ dimensions.

- $G(k)$: projection matrices of the SSR and SMR noises on the Cartesian coordinates for the k -th measurement.

As mentioned before, if there is any time bias (Δt) due to a discrepancy between the time reference used for timestamping the SSR measurements and the one considered as the reference (i.e. the SMR one), then $\Delta \hat{l}$ will estimate the join effect of both biases, that is the different between the location of the ATC antenna and the aircraft centroid, together with the impact of the time bias ($\Delta l'_j$).

Again, if $\overline{\Delta l_{ATC}}$ is known, Δt can be estimated and then removed from the measurements as it was presented for the MLAT case (ref. 6.10.7.3). Likewise, Δt is treated as a sensor-based bias and, as such, a better estimation can be obtained by integrating the information coming from the other trajectories, which is what it is proposed in this thesis. So, the bias estimator is modified to integrate the new information (i.e. $\overline{\Delta l_{ATC}}$) and proceed to the estimation of Δt .

The corrected aircraft dependent x and y offsets at time k , i.e. $\Delta x'(k)$ and $\Delta y'(k)$, can be expressed in terms of Δt and $\overline{\Delta l_{ATC}}$, assuming negligible noises, as:

$$\begin{bmatrix} \Delta x(k)' \\ \Delta y(k)' \end{bmatrix} = \begin{bmatrix} \Delta x(k) \\ \Delta y(k) \end{bmatrix} - \begin{bmatrix} \sin \gamma(k) \\ \cos \gamma(k) \end{bmatrix} \overline{\Delta l_{ATC}} = H(k) \begin{bmatrix} \delta R_j \\ b_i^\theta \\ \Delta t \end{bmatrix} \quad (6.57)$$

Where:

- $H(k)$: projection matrix of the new estimated parameters on the Cartesian positions when Δt is decoupled from $\overline{\Delta l_{ATC}}$, whose expression is:

$$H(k) = \begin{bmatrix} \sin \theta_i^m(k) & \rho_i^m(k) \cos \theta_i^m(k) & -V(k) \sin \gamma(k) \\ \cos \theta_i^m(k) & -\rho_i^m(k) \sin \theta_i^m(k) & -V(k) \cos \gamma(k) \end{bmatrix} \quad (6.58)$$

With:

- $\gamma(k)$: aircraft axis orientation at time k .
- $V(k)$: ground speed of the target at time k .

This projection matrix can be refined using the segment orientation, $\alpha(k)$, instead of the target orientation, $\gamma(k)$, whenever the target is located in a straight segment, which leads to the following expression:

$$H(k) = \begin{bmatrix} \sin \theta_i^m(k) & \rho_i^m(k) \cos \theta_i^m(k) & -V(k) \sin \alpha(k) \\ \cos \theta_i^m(k) & -\rho_i^m(k) \sin \theta_i^m(k) & -V(k) \cos \alpha(k) \end{bmatrix} \quad (6.59)$$

So, the expression of the BLUE estimator will change into:

$$\begin{bmatrix} \delta \hat{R} \\ \hat{b}^\theta \\ \Delta \hat{t} \end{bmatrix} = (A)^{-1} B \quad (6.60)$$

and A and B have a similar expression as before:

$$A = \sum_k H^T(k) S^{-1}(k) H(k) \quad (6.61)$$

$$B = \sum_k H^T(k) S^{-1}(k) \begin{bmatrix} \Delta x'(k) \\ \Delta y'(k) \end{bmatrix} \quad (6.62)$$

and $S(k)$ keeps the same expression as in eqn. 6.56.

6.10.7.5.3 Global bias estimation

In the case of the SSR, the presence of sensor-based biases introduces the need for a global bias estimation, which takes a slightly different form depending on the bias estimation strategy that is selected. This strategy considers the percentage of trajectories with a known $\overline{\Delta l_{ATC}}$ as it will be described below.

Focussing on the case in which the local bias estimator that is mainly used is the one providing the joint estimation, that is $\Delta \hat{l}$. The global bias estimation takes the same form as the one described in [Soto 05]. Therefore, it will be fed by:

$$\begin{bmatrix} \delta \hat{R} \\ \hat{b}^\theta \\ \Delta \hat{l} \end{bmatrix}_j \longrightarrow \begin{bmatrix} \delta \hat{R} \\ \hat{b}^\theta \end{bmatrix}_j \quad (6.63)$$

$$S(j) = \begin{bmatrix} s_{11} & s_{12} & s_{13} \\ s_{21} & s_{22} & s_{23} \\ s_{31} & s_{32} & s_{33} \end{bmatrix} = A^{-1} \longrightarrow \hat{C}(j) = \begin{bmatrix} s_{11} & s_{12} \\ s_{21} & s_{22} \end{bmatrix} \quad (6.64)$$

Please note the change in the index, as the global estimator is updated once per trajectory. In this case, $\delta \hat{R}$ is also considered an estimate of b^ρ with the special feature that it is biased by the presence of the transponder induced bias (ΔR_j), which is a bias that can be modelled as a uniform random variable (see below). Consequently, the covariance matrix associated with the provided information should be modified to consider this uncertainty:

$$M(j) = \hat{C}(j) + \begin{bmatrix} \sigma_{\Delta R}^2 & 0 \\ 0 & 0 \end{bmatrix} \quad (6.65)$$

Where:

- $\sigma_{\Delta R}$: standard deviation of the transponder induced bias (ΔR_j) for any aircraft, whose value is:

$$\sigma_{\Delta R} = \frac{75}{\sqrt{3}} \text{ m} \quad (6.66)$$

Finally, in this case, the global bias estimator and its covariance matrix would look like:

$$\begin{bmatrix} \hat{b}^\rho \\ \hat{b}^\theta \end{bmatrix} = \left(\sum_j M^{-1}(j) \right)^{-1} \left(\sum_j M^{-1}(j) \begin{bmatrix} \delta \hat{R} \\ \hat{b}^\theta \end{bmatrix} \right) \quad (6.67)$$

$$C(j) = (C^{-1}(j-1) + M^{-1}(j))^{-1} = \left(\sum_j M^{-1}(j) \right)^{-1} \quad (6.68)$$

On the other hand, if it is assumed that $\overline{\Delta l_{ATC}}$ is known for all the trajectories, then the local bias estimator that is mainly used is the one providing just an estimate of the time bias, that is $\Delta \hat{t}$. Therefore, the global bias estimator needs to be changed, as now there are three sensor-based biases, opposite to the previous case in which there were only two out of three. As a result, there is no need for selecting those components belonging to sensor-based biases as all the estimates from the local bias estimators are sensor-based. The new expression of the global bias estimator, in this case, would be:

$$\begin{bmatrix} \hat{b}^\rho \\ \hat{b}^\theta \\ \Delta \hat{t} \end{bmatrix} = \left(\sum_j M^{-1}(j) \right)^{-1} \left(\sum_j M^{-1}(j) \begin{bmatrix} \delta \hat{R} \\ \hat{b}^\theta \\ \Delta \hat{t} \end{bmatrix} \right) \quad (6.69)$$

$$C(j) = (C^{-1}(j-1) + M^{-1}(j))^{-1} = \left(\sum_j M^{-1}(j) \right)^{-1} \quad (6.70)$$

Where:

- $M(j)$: covariance matrix associated to each estimate coming from a local bias estimator, which is enlarged with the prior variance of the transponder induced bias (ΔR):

$$M(j) = S(j) + \begin{bmatrix} \sigma_{\Delta R}^2 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix} = A^{-1} + \begin{bmatrix} \sigma_{\Delta R}^2 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix} \quad (6.71)$$

6.10.7.5.4 Overall bias estimation process

After stating the measurement model, the different local bias estimation methods, and the different layout of the global bias estimator depending on the availability of certain information then, the overall processing can be described.

As before, the first aspect that needs to be clarified before processing one-by-one all the SSR-supported multisensor associated trajectories is whether $\overline{\Delta l_{ATC}}$ is generally available or not. This information will determine the estimation strategy and the local bias estimator that is going to be generally used for processing the SSR measurements of the different trajectories.

If the scenario does not have enough aircraft with a known $\overline{\Delta l_{ATC}}$ then, the local bias estimator based on estimating Δl is used for all the trajectories. Otherwise, the one aiming at estimating Δt is used for the trajectories with a known $\overline{\Delta l_{ATC}}$ and for the others, once they are time-bias corrected, the estimator that is used is the one estimating Δl . Both strategies are going to be detailed hereafter while describing the overall process.

Once the bias estimation strategy is selected, every time a new SSR target report belonging to a multisensor associated trajectory is processed, the bias estimation

algorithm analyses whether this target report can be used for updating the SSR local bias estimator of such trajectory. This analysis checks whether the target is within a transversal manoeuvring area. If it is the case, the SSR target report is discarded; otherwise, the algorithm searches for the closest target report (in time) from the reference sensor, the SMR in this case, and then taking advantage of the monosensor reconstruction performed on the SMR plots, extrapolates the SMR plot into the time of the SSR one. Then depending on the bias estimation strategy, the algorithm uses the appropriate formulae to build the measurement difference first and then, the projection matrix, $H(k)$, and the projected covariance matrix, $S(k)$.

All this information is used to feed the corresponding BLUE estimator, which accumulates the information coming from all the computed measurement differences from SSR plots located within non-transversal manoeuvring areas in the already presented A and B expressions. Once, all the measurement differences for a trajectory are processed by the BLUE, it can be resolved using the final values of A and B .

If the strategy is set considering that there are not enough $\overline{\Delta l_{ATC}}$ known values to later on obtain a good global time bias estimate for the sensor, then the local bias estimator is the one focussed on estimating the difference between the ATC antenna position and the centroid of the aircraft ($\Delta \hat{l}$), which will also includes the time bias effect.

Once all the local bias estimators are executed, their estimates are sent to the global bias estimator, which will have to deal with input vectors (and their associated covariance matrices) that contains two sensor-based bias estimates and one target-based as it is shown in eqns. 6.63 and 6.64. The result of the global bias estimator will be merged with the one provided by the local bias estimator in order to build the final correction as it is shown hereafter:

$$\begin{array}{ccc} \text{GLOBAL ESTIMATE} & & \text{LOCAL ESTIMATE} \\ \left[\begin{array}{c} \hat{b}_i^\rho \\ \hat{b}_i^\theta \end{array} \right]_{\Delta \hat{l}} & \longleftrightarrow & \left[\begin{array}{c} \delta \hat{R} \\ \hat{b}^\theta \\ \Delta \hat{l} \end{array} \right]_j \end{array} \quad (6.72)$$

$$C_{\Delta \hat{l}} = \begin{bmatrix} c_{11} & c_{12} \\ c_{21} & c_{22} \end{bmatrix} \longleftrightarrow S(j) = \begin{bmatrix} s_{11} & s_{12} & s_{13} \\ s_{21} & s_{22} & s_{23} \\ s_{31} & s_{32} & s_{33} \end{bmatrix} \quad (6.73)$$

Taking into consideration that:

$$\begin{aligned} \delta R &= b^\rho + \Delta R \longrightarrow \Delta \hat{R} = \delta \hat{R} - \hat{b}^\rho \\ \sigma_{\Delta R}^2 &= s_{11} + c_{11} \end{aligned} \quad (6.74)$$

Then, the final correction would be done in stages as shown below:

1. All the SSR measurement are corrected using the sensor-based bias estimate (indicated with the SB and C subscripts) provided by the global bias estimator.

$$\left[\begin{array}{c} x_{SSR}(k) \\ y_{SSR}(k) \end{array} \right]_{SB,C} = \left[\begin{array}{c} x_{SSR}(k) \\ y_{SSR}(k) \end{array} \right] - H(k) \left[\begin{array}{c} \hat{b}_i^\rho \\ \hat{b}_i^\theta \end{array} \right]_{\Delta \hat{l}} \quad (6.75)$$

Where:

- $H(k)$: projection matrix of the sensor-based biases on the Cartesian coordinate framework, whose expression is:

$$H(k) = \begin{bmatrix} \sin \theta_i^m(k) & \rho_i^m(k) \cos \theta_i^m(k) \\ \cos \theta_i^m(k) & -\rho_i^m(k) \sin \theta_i^m(k) \end{bmatrix} \quad (6.76)$$

The measurement covariance matrix is enlarged to cope with the uncertainty of the global bias. This is reflected in the following expression:

$$N'_{SSR} = G(k) \begin{bmatrix} \sigma_{\rho,SSR}^2 & 0 \\ 0 & \sigma_{\theta,SSR}^2 \end{bmatrix} G^T(k) + H(k) C_{\Delta \hat{l}} H^T(k) \quad (6.77)$$

Where:

- $G(k)$: projection matrix of the uncertainty of the SSR noises into the Cartesian coordinate framework.
2. Then, the target-based biases are removed from all the measurement associated to the trajectories with SSR support, which are now sensor-based bias corrected, as shown below:

$$\begin{bmatrix} x_{SSR}(k) \\ y_{SSR}(k) \end{bmatrix}_c = \begin{bmatrix} x_{SSR}(k) \\ y_{SSR}(k) \end{bmatrix}_{SB,C} - H(k) \begin{bmatrix} \Delta \hat{R} \\ \Delta \hat{l} \end{bmatrix} \quad (6.78)$$

Where:

- $H(k)$: projection matrix of the target-based biases on the Cartesian positions adapted to the new estimated parameters, whose expression is:

$$H(k) = \begin{bmatrix} \sin \theta_i^m(k) & \sin \gamma(k) \\ \cos \theta_i^m(k) & \cos \gamma(k) \end{bmatrix} \quad (6.79)$$

With:

- $\gamma(k)$: aircraft axis orientation at time k .

Please note that the value of $\Delta \hat{l}$ is used to correct the position of the SSR measurements belonging to the j -th aircraft and no time bias is neither estimated nor corrected.

The measurement covariance matrix is enlarged to cope with the uncertainty of the local bias. This is reflected in the following expression:

$$N''_{SSR} = N'_{SSR} + H(k) \begin{bmatrix} \sigma_{\Delta R}^2 & 0 \\ 0 & s_{33} \end{bmatrix} H^T(k) \quad (6.80)$$

At this point, if the proposed measurement model is accurate enough, the initial SSR measurement for the j -th aircraft would be referred to the centroid of the aircraft and time

aligned with the reference sensor. For the rest of the processing the bias corrected measurement $\begin{bmatrix} x_{SSR}(k) \\ y_{SSR}(k) \end{bmatrix}_c$ and its covariance matrix (N''_{SSR}) are used.

Opposite to the previous case, if the strategy is set considering that there are enough $\overline{\Delta l_{ATC}}$ known values to achieve a good global time bias estimate for the sensor, then the process follows a stepwise approach. First, the local bias estimator focussed on estimating the time bias (i.e. Δt) is executed on all the trajectories with a known $\overline{\Delta l_{ATC}}$ to obtain, using the global bias estimator described by eqns. 6.69 and 6.70, the corresponding estimate $\Delta \hat{t}$. Then all the SSR measurements are time-bias corrected based on the value of $\Delta \hat{t}$ and its uncertainty (variance) is propagated to the positions. Next, the trajectories with an unknown $\overline{\Delta l_{ATC}}$ are processed using the local estimator that estimates Δl using its time-bias corrected measurements and uncertainties while computing the measurement differences. All these local bias estimates are then process by the global bias estimator described by eqns. 6.67 and 6.68 obtaining an estimate of the sensor-based biases of this second group of trajectories. Finally, the two estimates of the sensor-based biases are optimally combined to obtain the value that will be used in eqn. 6.75. For the rest, these trajectories follow the same process as the one just described above this paragraph.

Once all the local bias estimators of the first step are executed, their estimates are sent to the global bias estimator, which processes them together with their associated covariance matrices as it is shown in the previous section, eqns. 6.63 to 6.68. The result of the global bias estimator is then divided as shown below:

$$\begin{bmatrix} \hat{b}^\rho \\ \hat{b}^\theta \\ \Delta \hat{t} \end{bmatrix} \begin{array}{c} \longrightarrow \\ \longrightarrow \end{array} \begin{bmatrix} \hat{b}^\rho \\ \hat{b}^\theta \\ \Delta \hat{t} \end{bmatrix}_{\Delta \hat{t}} \quad (6.81)$$

The second step takes advantage of the time-bias estimate ($\Delta \hat{t}$), which is assumed to have converged enough, to correct the timestamp of all the SSR measurements:

$$t_{SSR,C} = t_{SSR} - \Delta \hat{t} \quad (6.82)$$

and their covariance matrices from all the SSR-supported trajectories. Using the fact that:

$$\sigma_{\Delta \hat{t}}^2 = c_{33} \quad (6.83)$$

Where:

- c_{33} : element of the bias estimate covariance matrix $C(j)$, presented in eqn. 6.70, corresponding to the variance of $\Delta \hat{t}$, that is, the element in position (3,3).

The covariance matrix of the SSR measurement is then enlarged to consider the propagation of the uncertainty in the timestamp correction. This is translated into:

$$N'_{SSR} = N_{SSR} + \begin{bmatrix} V_x^2(k) & V_x V_y \\ V_x V_y & V_y^2(k) \end{bmatrix} \sigma_{\Delta \hat{t}}^2 \quad (6.84)$$

Next, the remaining trajectories, which have an unknown $\overline{\Delta l_{ATC}}$, are processed using the local bias estimator that estimates Δl , which is fed with the time-bias corrected SSR

measurement and their enlarged covariance N'_{SSR} . These estimates feed a global bias estimator that combines all them in order to obtain $\begin{bmatrix} \hat{b}^\rho \\ \hat{b}^\theta \end{bmatrix}_{\Delta \hat{t}}$ by means of using eqns. 6.67 and 6.68.

Finally, the sensor-based bias estimates computed from the last group of multisensor associated trajectories are then merged with the ones computed by the other global bias estimator, that is, $\begin{bmatrix} \hat{b}^\rho \\ \hat{b}^\theta \end{bmatrix}_{\Delta \hat{t}}$ and its covariance matrix, which is obtained from A as it is shown below:

$$S_{\Delta \hat{t}}(n) = \begin{bmatrix} s_{11} & s_{12} & s_{13} \\ s_{21} & s_{22} & s_{23} \\ s_{31} & s_{32} & s_{33} \end{bmatrix} = A^{-1} \longrightarrow \hat{C}_{\Delta \hat{t}}(n) = \begin{bmatrix} s_{11} & s_{12} \\ s_{21} & s_{22} \end{bmatrix} \quad (6.85)$$

For performing this merge, a Bayesian combination is proposed:

$$\begin{bmatrix} \hat{b}^\rho \\ \hat{b}^\theta \end{bmatrix} = \left(\hat{C}_{\Delta \hat{t}}^{-1}(m) + \hat{C}_{\Delta \hat{t}}^{-1}(n) \right)^{-1} \left(\hat{C}_{\Delta \hat{t}}^{-1}(m) \begin{bmatrix} \hat{b}^\rho \\ \hat{b}^\theta \end{bmatrix}_{\Delta \hat{t}} + \hat{C}_{\Delta \hat{t}}^{-1}(n) \begin{bmatrix} \hat{b}^\rho \\ \hat{b}^\theta \end{bmatrix}_{\Delta \hat{t}} \right) \quad (6.86)$$

Whose covariance matrix is:

$$C = \left(\hat{C}_{\Delta \hat{t}}^{-1}(m) + \hat{C}_{\Delta \hat{t}}^{-1}(n) \right)^{-1} \quad (6.87)$$

Where:

- m : number of selected multisensor associated trajectories that have a known $\overline{\Delta l_{ATC}}$.
- n : number of selected multisensor associated trajectories that have an unknown $\overline{\Delta l_{ATC}}$.

Once these sensor-based biases are computed, they are removed from all the SSR measurements, but the approach followed is different depending on the local bias estimator that was used.

If the local biases of the trajectory were computed with the estimator that estimated Δt , then there is no need to correct the difference between the position of the ATC antenna and the aircraft centroid. However, there is still the transponder induced bias (ΔR_j), which can be estimated and corrected once these sensor-based biases are removed. The way to proceed is again based on stages:

1. The first stage consists in removing the remaining sensor-based biases as the time bias was removed in the previous step. The way of carrying out this operation is equivalent to the one done on the previously described stage 1 (see eqns. 6.75 to 6.77) where the only difference is based on the use of the estimate obtained after the Bayesian combination described in eqn. 6.86, which is summarized below:

$$\begin{bmatrix} \hat{b}^\rho \\ \hat{b}^\theta \end{bmatrix}_{\Delta \hat{t}} \longrightarrow \begin{bmatrix} \hat{b}^\rho \\ \hat{b}^\theta \end{bmatrix} \quad (6.88)$$

2. The second stage removes the target-based biases from all the measurement associated to each of the trajectories with SSR support, which are now sensor-based bias corrected, as shown below. Note that opposite to the previous case, now there is only one remaining target-based bias:

$$\begin{bmatrix} x_{SSR}(k) \\ y_{SSR}(k) \end{bmatrix}_{c,j} = \begin{bmatrix} x_{SSR}(k) \\ y_{SSR}(k) \end{bmatrix}_{SB,c,j} - H(k) \Delta \hat{R}_j \quad (6.89)$$

Where:

- $H(k)$: projection matrix of the target-based bias on the Cartesian reference framework, whose expression is:

$$H(k) = \begin{bmatrix} \sin \theta_i^m(k) \\ \cos \theta_i^m(k) \end{bmatrix} \quad (6.90)$$

The measurement covariance matrix is enlarged to cope with the uncertainty of the local bias estimate. This is reflected in the following expression:

$$N''_{SSR} = N'_{SSR} + H(k) \sigma_{\Delta R}^2 H^T(k) \quad (6.91)$$

For the remaining trajectories, the bias removal is done following the same process as in the one described when the strategy was set considering that there were not enough $\overline{\Delta l_{ATC}}$ known values. So, eqns. 6.75 to 6.80 apply.

The fact that the SSR is a sensor with a reduced update rate compared to the others, with less accurate measurement, a reduced coverage of the airport and with both target-based and sensor-based biases makes the convergence of all the estimators (local and global ones) slower than in the case of the other sensors. It is worth mentioning that the reference sensor can be changed to have a larger overlap region where to compute the measurement differences. A MLAT sensor should offer more overlap than the SMR.

6.10.7.6 INCS system bias estimation

These sensors currently lack of a mature measurement model that can be used for estimating sensor-based biases as they are expected not to have target-based biases because they are non-cooperative sensors and, as such, no transmission from the target is expected.

Another important aspect to consider is that these are quite recent sensors and, consequently, data are not generally available for research purposes as the performances are still subject to improvement in most of the cases. Real data are only available under very strict non-disclosure agreements (NDA).

Finally, under the umbrella of the INCS acronym, there are different families of sensors that share the fact that they are independent and non-cooperative sensors but apart from this, they do not have to share anything else, so a common way of characterizing them seems to be difficult to obtain.

Taking all the previous considerations into account and similar to what was proposed in [Žáčik 17], the approach that has been followed is that a measurement model like the one used for multilateration systems is going to be used for them, that is, only local biases are

expected and, as such, only local bias estimators as the one described by eqns. 6.27 to 6.33 are going to be used.

6.10.7.7 Other considerations

If the airport has many holding points, this situation can be taken into consideration to estimate biases. Aircraft are forced to wait at these points until the ground controller allows them to proceed to the runway or to the intersection that is protected with such mean. This time can be exploited for target-based bias estimation.

In this situation, if the aircraft is stopped and, consequently, its ground speed is negligible, then the effect of any time bias disappears and in the case of a MLAT system, for example, only the target-based bias (i.e. the difference between the position of the ATC antenna and the aircraft centroid) is present.

If the aircraft stands for enough time in a position, this can be exploited to estimate the Δl , which can be later removed. The proper way of exploiting this fact is by detecting these periods of time when the aircraft is stopped and then by applying the same algorithms as presented in section 6.10.7.3 estimating Δl , which can then be fed to the other algorithms as the value for $\overline{\Delta l_{ATC}}$. These periods of time can be detected by the “movement model identification” process, which can identify them while analysing each trajectory, and afterwards, it can inform the “bias estimation and cancellation”, which is the process that can conveniently exploit them.

6.11 Noise estimation

6.11.1 Process description

The “noise estimation” process approximates the statistical behaviour of the sensor measurement noise. It aims at providing an estimate of the probability density function (pdf) of the measurement noise or, if not possible, an estimate of the mean and covariance matrix of the noise vector in either the sensor coordinates or in the system ones.

Furthermore, it gives an indication on how successful the “bias estimation and cancellation” process has been removing the systematic errors as the mean value is expected to be zero. Hence, if it is not the case, there might still be some residual biases that have not been removed. This might point to an issue with either the bias estimation algorithm, which might not have converged, or the spatial distribution of the processed data, which might be concentrated in several areas, or the sensor measurement model, which may lack some other bias components.

This process relies on both the “bias estimation and cancellation” and on the “movement model identification” processes for accomplishing its task and it feeds the “sensor measurement model” ancillary processing block. This process contributes to the implementation of the “sensor model update” function (ref. 3.6.2.3). The statistical characterization of the sensor measurement noise can be used in the current processing so it can substitute the static value provided by the user or it can be used in subsequent executions of the scenario.

6.11.2 Processing details

6.11.2.1 Input data

The input to this process is the position information of the target reports belonging to the selected multisensor associated trajectories that have reach this point. The position information is expected to be unbiased and to be located in an airport segment meaning that the separation distance has already been computed.

6.11.2.2 Output data

The “noise estimation” process delivers a statistical characterization of the sensor measurement noise, which can be provided in the form of a pdf, a histogram or a mean and a variance value. This information can be directly consumed by the current data processing while reconstructing the target trajectory or it can be sent to the “sensor measurement model” ancillary processing block in order to be used in future executions of the scenario.

6.11.2.3 Block diagram

The “noise estimation” process can be graphically located in the “data preparation” processing block, as it is shown in the figure below:

Figure 6.17: Block diagram focussed on the “noise estimation” process. Source: own elaboration.

6.11.3 Prototype implementation details

The approach that is followed by the prototype relies on identifying the target reports that are located only in straight segments (for preventing any anomalous behaviour due to transversal manoeuvres) and then, extracting their “separation” distance. If the sensor is a rotating one, it is possible to project this measurement on the sensor local coordinates.

The values of the “separation” distance of all the selected target reports are displayed in a histogram, which roughly assesses how well the “bias estimation and cancellation” process performed. Afterwards, this “separation” distance is projected in either the sensor local coordinates or in the system ones. The pdf of the sensor measurement noise in such coordinates is again approximated by means of a histogram and finally, the mean and the variance of each measurement component is computed.

6.12 Time re-ordering

6.12.1 Process description

The “time re-ordering” process aims at redistributing the data elements associated to the selected multisensor associated trajectories considering the time bias correction performed by the “bias estimation and cancellation” process. If the estimated time bias is large enough, it might influence the order in which target reports are processed. This correct ordering is important when estimating the reference trajectory.

6.12.2 Processing details

6.12.2.1 Input data

The input is the set of selected multisensor associated trajectories which are bias corrected.

6.12.2.2 Output data

The output is the same set of selected multisensor associated trajectories that were received as input. However, their data elements are now time-ordered based on the bias corrected time obtained as a result of the “bias estimation and cancellation” process.

6.12.2.3 Block diagram

The “time re-ordering” process can be graphically located in the “data preparation” processing block, as it is shown in the figure below:

Figure 6.18: Block diagram focussed on the “time re-ordering” process. Source: own elaboration.

6.12.3 Prototype implementation details

The prototype re-executed the function already mentioned in section 6.6.3 that checks whether the list of data elements of the set of selected multisensor associated trajectories is time-ordered, usually based on the time of day (ToD), or not. In case it is not, the function extracts the time-ordered sequence of times and provides the correct sequence of data element indexes. This sequence can then be used to correctly order the list of data

elements of the selected multisensor associated trajectories, which will be used by the subsequent processing steps. The time that is employed is the same one as the one employed in section 6.6.3 but time bias corrected. This means that, whenever it is available, the ToA is used, if not the ToR, and the last option would be the Rectime.

6.13 Association completion

6.13.1 Process description

This process concludes the data association of those plots or target reports that were not associated by the “data association” process and likewise also contributes to the “plot to trajectory association” function. Hence, the target reports which are not associated by this process are sent to the final list of un-associated target reports that will be used for estimating the probability of false alarm (PFA) or for computing the percentage of associated target reports from the total number.

If a target report reaches this point without being associated is because either it does not have any identity information (i.e. it is a primary target report), or the one that has is not discrete (i.e. no AA or a non-discrete Mode 3A) and, as such, it does not allow to apply any discarding or selecting criteria. Additionally, it is also possible that the reason of not being associated yet is because there were doubts when trying to associate it during the “data association” process and it was preferred to delay the association until all the information was available and the biases were corrected.

At this moment of the data processing, the data elements that remain are the ones associated either to the selected multisensor associated trajectories or the not associated ones. To achieve the objective of this process, there is a need for correcting the not associated target reports by applying the same bias correction as the one suffered by the rest of the target reports associated to the trajectory. Depending on the sensor that has provided them, it might suffer from sensor-based biases, target-based ones or both of them.

The process goes through the full list of un-associated target reports and tries to find all the selected multisensor associated trajectories that are compatible in time and in space with each target report under analysis. If one or more trajectories are found that match both criteria, then the selected target report is bias corrected using first the sensor-based biases then, once the analysis is focussed on each compatible trajectory, the target report is corrected with the target-based biases calculated for such trajectory. The target report is then aligned with the trajectory and a thorough analysis is performed to score how likely is that it comes from the same target that the trajectory.

Once the process has been repeated for all the compatible tracks, then a decision is taken based on the score that is given to each candidate association. The decision can be associating the target report to one of the candidate tracks, associating to several of them or considering that it is a false alarm.

6.13.2 Processing details

6.13.2.1 Input data

The input data to this process is the set of selected multisensor associated trajectories and the list of initially un-associated target reports generated by the “data association” process (ref. 6.7).

6.13.2.2 Output data

The output is the same set of selected multisensor associated trajectories whose list of data elements (i.e. plots, target reports or monosensor track updates) might have been enlarged with one or several previously un-associated target reports and the final list of un-associated target reports.

6.13.2.3 Block diagram

The “association completion” process can be graphically located in the “data preparation” processing block, as it is shown in the figure below:

Figure 6.19: Block diagram focussed on the “association completion” process. Source: own elaboration.

6.13.3 Prototype implementation details

The prototype does not implement this process as the real data provided for testing purposes do not include un-associated data. Even the SMR data already come locally associated with a Local Track Number that can be exploited by the association processes.

7

CHAPTER

TRAJECTORY RECONSTRUCTION PROCESSING BLOCK

“Innovation requires an experimental mindset”

— Denise Morrison

The trajectory reconstruction finishes when all previously generated pieces of information are put together and then, the proper smoothing algorithm exploits them to approximate the actual path followed by the target, which is represented by the multisensor reconstructed trajectory.

Smoothing algorithms are the cornerstone of this crucial step. Their performance, which clearly depends on the performance of the previous processes, will determine not only the success of this processing block, but also the reliability on the whole performance evaluation process.

Moreover, this processing block includes a classification of the reconstructed trajectory based on the likelihood that it does not come from an actual target and, therefore, it is a false track. This classification can also identify the type of false track (e.g. a split track, a reflexion, a side lobe, etc.). Another process tries to complete the missing information in the reconstructed trajectory samples in case it could not be obtained from the respective input. Finally, there is a process responsible for performing any remaining activity.

All in all, the trajectory reconstruction, which is performed by the “data smoothing” process, is the main topic and, as such, almost the full chapter is devoted to it. First the problem of obtaining the reference trajectory from opportunity traffic is presented and then, two of the more extended smoothing techniques are described, that is, the spline-based and the Bayesian ones. The latter will be the choice of this work.

7.1 Reference trajectory estimation

This chapter concentrates on presenting the state of the art concerning trajectory reconstruction techniques as a way of obtaining “reference” trajectories to be used for performance evaluation purposes. In order to obtain “reference” trajectories, there are different approaches to follow. The most common ones are:

- Performing a trajectory reconstruction based on opportunity traffic.
- Using special test vehicles with improved measuring equipment and flight trials.
- Running simulations.

The first approach is the one that will keep all the attention of this research. It tries to estimate the actual position along time of a mobile (vehicle or aircraft) by fusing all the surveillance information received from different sensors during a certain period, which should be long enough for the internal algorithms to converge. As already mentioned, this process is implemented by a data fusion algorithm that expects that the inputs share the same coordinate framework and that they are bias corrected.

The second approach consists on equipping a ground vehicle or an aircraft with a high-precision measurement equipment¹⁷⁰ (e.g. a D-GPS receiver, accelerometers, gyroscopes) which records the kinematic data (position, velocity, acceleration, etc.). Then, after finishing the test, the recorded data are compared with those provided by surveillance sensors and by the tracking system, obtaining the corresponding performance indicators. The main drawback of this approach is that it is quite expensive and very constrained to a certain test scenario, so their use is quite limited. Only in situations where the aim is to assess a certain aspect of a surveillance sensor, which is usually in operation, this approach is recommended.

Finally, simulations have the advantage of knowing the actual trajectory as it is used for generating the surveillance data provided by the simulated sensors. They are mainly intended for “in factory” performance tests in which the system, normally a tracker, is fed with simulated surveillance data to measure its performances. An additional issue with this approach is that simulations are based on models and these models are not always as close as desired to the reality. Consequently, in some cases, the detected problems might not be caused by the system itself but by the simulation model. This fact implies that some care must be taken when extracting conclusions from this kind of approaches.

7.1.1 The reference trajectory

This concept represents the result provided by trajectory reconstruction techniques, the data gathered by a high precision measurement equipment or the trajectory model used by a simulator for generating synthetic surveillance data.

It tries to determine the “ground truth” concerning the movement that an aircraft has performed all along the airport. It is referred to as “reference” because it will be compared with the data provided by either noisy surveillance sensors, like the ones presented in

¹⁷⁰ With also a frequent update that guarantees a reasonable sampling rate of the measured state variables.

section 1.3.1, or trackers or SDP systems (ref. 1.3.3). This comparison allows to calculate some sensor/tracker performance indicators to be used while assessing the behaviour of both sensors and trackers. These reference trajectories are also called “ideal” trajectories as they are supposed to represent the actual trajectory that the aircraft have followed. Depending on the chosen approach, the reference trajectory will be closer or further to the actual one.

Mathematically talking, the reference trajectory is a 4D trajectory which means that apart from defining the position in the three spatial coordinates (x, y, z) , it also includes the time dimension (t) . Currently, there is a trend in the ATM domain that takes into account the time in the calculations as mentioned when introducing the TBO concept in section 1.5. This fact outlines the importance of the computation of these trajectories also in the aerodrome domain.

7.1.2 Reference trajectory estimation from opportunity traffic

From the different approaches that have been presented, the focus in this thesis is placed on those techniques based on opportunity traffic (ref. 1.3.5), which consists of mobiles that have moved around a delimited airspace¹⁷¹ during a defined period of time. Moreover, all the information concerning the opportunity traffic that is processed takes the form of multi-sensor data.

7.1.2.1 Statement of the problem

The problem to be solved is a multisensor data fusion one aiming at the estimation of the aircraft kinematic state (e.g. position, velocity and acceleration) at a certain time instant having access to past but also to future noisy observations provided by more than a surveillance sensor of different types such as radars, ADS-B, MLAT, etc. of the traffic that has undergone a pre-defined space volume during a certain period of time.

As presented in CHAPTER 2, these noisy observations are normally affected by different errors such as: measurement noise, quantization noise, systematic errors, timestamp issues, etc. They also arrive at a different pace and in different coordinates depending on the generating source. Converting the information from the different data sources to the same coordinate framework is the task of the “coordinate transformation” process (ref. 6.4), while systematic errors are expected to be estimated and cancelled by bias estimation algorithms as the ones implemented by the “bias estimation and cancellation” process (ref. 6.10). Consequently, the aim of these trajectory reconstruction techniques is to filter out the remaining error terms from the inputs and generate improved target state estimates applying some prior knowledge. Therefore, its output, that is, the reference trajectory, can provide a better accuracy than a reference obtained from only one of the single data sources by itself and, as such, it can be assumed as the “reference” to be used for comparison purposes.

The fact that available information encompasses past but also future measurements introduces a constraint in the processing. Until the recording containing all the

¹⁷¹ This is defined as a volume around the area of interest.

measurements has not been closed, the computation cannot start. This restriction can be relaxed in some situations depending on the system architecture.

When the reconstruction scenario is an airport, there is further information that can be exploited, such as the airport map, the airport configuration, aircraft ground movement models, etc. All this information imposes further constraints to the aircraft ground movements that can be used for improving the aircraft kinematic state estimation. Thus, the aircraft state estimation can take advantage of taxiway directions, the airport layout (the centre of the taxiways or the location of holding points) and other airport elements.

Moreover, any prior knowledge of the aircraft (e.g. dimension, antennae position, wheelbase) can also be used to improve the quality of the final reconstructed trajectory.

7.1.2.2 Advantages and disadvantages

Trajectory reconstruction from opportunity traffic has several advantages from the other approaches that have been presented in section 0. The reasons why these techniques are preferred are based on the following features:

- Low cost.
- Possibility of covering huge areas.
- Types of vehicles that can be considered.
- Consideration of abnormal and not expected situations.
- Not invading.

Thus, it has low cost because the data used for computing the reference trajectories is already available for the tool as they have been previously processed by the surveillance function.

This approach has different scenarios where it can be applied. The most common one is the en-route/TMA domain, but the aerodrome domain is also a candidate for applying these techniques. The former is the one associated to SASS-C and the latter is expected to become an extension of it. This work will focus on the second one. For the en-route airspace the possibility of covering huge areas is clear. However, for the airport, the areas to be covered are the ones defined by the coverage of the surface surveillance sensors that serve the airport.

Concerning the vehicles, this approach does not make any distinction among the different mobiles that can be found in the area under analysis with the only exception that they have to be visible to the surveillance sensors and tracking systems.

Additionally, this approach includes all the situations that can happen in the ATC environment. Consequently, the same situations that operational tracking systems suffer while processing the air traffic data are also suffered by trajectory reconstruction techniques, so the same non nominal situations or abnormal ones are taken into consideration.

Finally, this approach can be implemented in a pseudo-online fashion or in an offline one.

However, it also has some drawbacks, so it is fundamental to analyse the situation to be faced in order to choose the best approach. Therefore, it is relevant to keep in mind that:

- They present a transient period.
- Special situations cannot be forced.
- They work in non-real time.

The first drawback derives from the fact that there is a need for the convergence of bias estimators before start reconstructing trajectories. Moreover, there is a need for having the whole trajectory before computing the reference trajectory.

The technique cannot by default schedule certain abnormal situations while reconstructing the trajectories. Therefore, the situations that are addressed are the ones that have happened during the time period and within the coverage area that are under analysis. Other situations are not initially considered. This is the reason why flight trials or test vehicles are still used as they have the ability of forcing these situations in case they do not appear as part of the regular movements that happen in the airspace or airport area.

Due to the fact that trajectory reconstruction techniques are based on the knowledge of both past and future, this introduces an additional constraint as the PET has to wait until the batch is closed in order to start computing the reference trajectories. Furthermore, this process is also affected by the need for having bias estimates before entering in the trajectory reconstruction process as it is described in the “performance evaluation” process (ref. CHAPTER 4).

7.2 Taxonomy of current techniques for reference trajectory reconstruction

7.2.1 Introduction

There are different types of techniques that can be applied to obtain reconstructed trajectories using opportunity traffic. The main difference between them is the a priori knowledge about the data to be processed in order to obtain the reference trajectory. These data are the measurements provided by surveillance sensors.

7.2.1.1 Non-Bayesian techniques

These techniques make no a priori assumption of the data to be processed. Neither aircraft movement model nor measurement error model. So, they are more robust than the ones that assume a certain model but, conversely, their performances are inferior. There are plenty of techniques that are non-Bayesian. In this work, the interest is placed on Spline functions, which are introduced in section 7.3 as a clear exponent of this group. However, they are just presented for the sake of completeness because the focus of this thesis is on the following group, i.e. on Bayesian techniques.

7.2.1.2 Bayesian techniques

These techniques are based on a target state estimation based on its measurement-conditioned posterior distribution. Prior distributions of the target state, i.e. $p(x)$, are updated by means of measurements of the actual target (z) to form the posterior distribution, i.e. $p(x|z)$. This implies that two model assumptions must be made, the likelihood $p(z|x)$ and the prior $p(x)$.

Once they are available, the measurement-conditioned posterior distribution can be obtained by applying the conditional probability rule:

$$p(x|z) = \frac{p(z|x)p(x)}{p(z)} \quad (7.1)$$

Where:

- x : target state vector.
- z : sensor measurement.

The techniques of this group, which are going to be considered, are the ones related to optimal filtering and optimal smoothing. In the case of taking into consideration suboptimal techniques, the focus will be placed on interacting multiple models (IMM) ones.

7.3 Spline-based techniques

7.3.1 Spline history

The name of “spline” comes originally from an East Anglian dialect [Wikipedia]. It referred to the wooden strips used by early designers of ships hulls and aircraft for drawing the curvature they wanted to implement into their design. Thus, these elements provided an interpolation of the key points in the design obtaining smoothed curves which were aimed at having minimum strain energy.

These curves were exploited by a ship drafting technique called “lofting”, which was also imported by the British aeronautical industry during World War II. This technique transfers the designed curves from plan to wood, which is then cut for advanced woodworking that finishes with a scaled model of the ship, or a part of it such as the hull or the keel. The curves were defined by means of lead weights called ducks, which were in charge of holding down the metal or wooden strips. Ducks were moved to make the strips go through the desired points achieving the desired curve. Once the curve was obtained, it was traced on a paper or directly on wood. This way the desired curves were achieved. An example of this technique is shown in the following picture:

Figure 7.1: Example of the lofting technique. Source: [Lofting]

It was in 1946, after World War II, when Isaac Jacob Schoenberg introduced the notion of the mathematical splines in his paper [Schoenberg 46]. He realised that for small deflections, the shape of the physical spline could be described by a piecewise cubic polynomial. In the beginning, the use of splines was very limited. It was the advent of industrial applications based on them and the increase in the computer performance what boosted the use of splines. Thus, from the traditional industries that exploited them (i.e ship building and aircraft design), splines jumped also to the automotive industry. It was in the sixties when splines obtained the attention of the research community, mainly from the automotive industry, having a great development. Authors such as: De Boor, Casteljau, Bézier, Birkhoff and Garabedian; devoted their efforts in developing the spline theory. Great applications have also been found in the field of computer-aided design and computer graphics. There are many books on this subject such as [Bartels 87] and later [Lengyel 12].

In the last two decades, splines have also expanded to other knowledge fields such as signal and image processing applications, where authors like Unser have especial relevance [Unser 98][Unser 99]. Thus, it has been shown that splines offer the best cost-performance trade-off among available interpolation methods.

7.3.2 Evolution of interpolation techniques

7.3.2.1 Polynomial interpolation

Interpolation consists of taking a number of points¹⁷² (n) on a curve $y = g(x)$ and find another function that goes through all of them. This interpolating function is supposed to have a better behaviour than the interpolated one. When this function takes the form of a polynomial $p(x)$ of degree $n - 1$, we talk about polynomial interpolation.

Therefore, polynomial interpolation is one of the initial techniques for interpolating a given dataset by means of a polynomial to exploit the good features of the latter. Among these features, it can be cited that polynomials are easy to evaluate, differentiate and

¹⁷² Usually referred as data sites or control points.

integrate, which make them great candidates for approximating other functions. In many situations, this fact allows to represent a complicated function by a simpler one or in others to define a smooth curve through experimental data points.

Consequently, the target is to find a smooth curve which passes through some given points. In this case, this smooth function takes the form of a polynomial function with the following appearance:

$$p(x) = a_n x^n + a_{n-1} x^{n-1} + \cdots + a_2 x^2 + a_1 x + a_0 \quad (7.2)$$

Each point represents an additional constraint when computing the interpolating polynomial. Thus, for every point (x_i, y_i) the following equation adds to the linear system to be solved:

$$p(x_i) = a_n x_i^n + a_{n-1} x_i^{n-1} + \cdots + a_2 x_i^2 + a_1 x_i + a_0 = y_i \quad (7.3)$$

A polynomial of degree n has $n + 1$ degrees of freedom, so for solving the system and finding the coefficients $a_n, a_{n-1}, \dots, a_2, a_1, a_0$; $n + 1$ points are needed.

$$(x_0, y_0), (x_1, y_1), \dots, (x_n, y_n) \quad (7.4)$$

Once all the points are available, a system like the one below is formed, and its solution provides the unknown coefficients that determine the polynomial.

$$\begin{pmatrix} x_0^n & x_0^{n-1} & \cdots & x_0 & 1 \\ x_1^n & x_1^{n-1} & \cdots & x_1 & 1 \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ x_n^n & x_n^{n-1} & \cdots & x_n & 1 \end{pmatrix} \begin{bmatrix} a_n \\ a_{n-1} \\ \vdots \\ a_0 \end{bmatrix} = \begin{bmatrix} y_0 \\ y_1 \\ \vdots \\ y_n \end{bmatrix} \quad (7.5)$$

$$M \underline{a} = \underline{y} \quad (7.6)$$

There are several methods for performing the computation of the interpolating polynomial. The most common ones are the Lagrange form and the Newton form.

The Lagrange form is based on computing a set of basis functions that relies on the set of data sites $(x_0, x_1, x_2, \dots, x_{n-1}, x_n)$ instead of using the functions $x^n, x^{n-1}, \dots, x^2, x, 1$. Each basis function is dependent on the set of data sites. Thus, the i -th Lagrange polynomial for x_i has the following appearance:

$$l_i(x) = \prod_{\substack{j=0 \\ j \neq i}}^n \frac{x - x_j}{x_i - x_j} \quad (7.7)$$

This basis function vanishes in all the data sites except in x_i where it takes the value 1. Hence, for an arbitrary function g , the interpolating polynomial at the data sites $(x_0, x_1, x_2, \dots, x_{n-1}, x_n)$ looks like this:

$$p(x) = \sum_{i=0}^n g(x_i) l_i(x) = \sum_{i=0}^n y_i l_i(x) \quad (7.8)$$

One drawback of this form is that any change in the data sites implies that all the basis functions have to be recalculated.

On the other hand, the Newton form is based on divided differences [De Boor 01]. This form overcomes the drawback presented by the Lagrange form as it allows to include new data sites without a big effort. The Newton form of the polynomial of degree n , which agrees with the arbitrary function g at the data sites $(x_0, x_1, x_2, \dots, x_{n-1}, x_n)$, is:

$$p(x) = \sum_{i=0}^n (x - x_0) \cdots (x - x_{i-1}) [x_0, x_1, \dots, x_i] g \quad (7.9)$$

Where:

- $[x_0, x_1, \dots, x_i]g$: k -th divided difference of the function g at the data sites $x_0, x_1, \dots, x_i$.

However, the issue that appears with polynomials is that as we incorporate new points and though, the degree of the interpolating polynomial increases,¹⁷³ a problem of oscillation at the edges of the interpolating interval occurs. This effect is known as Runge's phenomenon, due to its discoverer Carl David Tolmé Runge. Therefore, this means that going to higher degrees does not imply an improvement in the accuracy of the interpolating function.

Hence, unless the underlying function is conveniently sampled (Chebyshev sites), high-order polynomial interpolation is unstable. This is the case when interpolating equidistant points as the ones provided by surveillance sensor every update period (one second in the airport area and every several seconds in the en-route airspace).

7.3.2.2 Piecewise polynomial interpolation

There is a straightforward solution to avoid Runge's phenomenon that is using piecewise polynomial functions or “pp” functions. So, instead of using just one polynomial function $p(x)$, the idea is to define a piecewise function $f(x)$, which is composed of several polynomial functions $p_i(x)$, each one between each pair of data sites, that is:

$$f(x) = \begin{cases} p_1(x), & \text{if } x_0 \leq x < x_1 \\ p_2(x), & \text{if } x_1 \leq x < x_2 \\ \vdots & \vdots \\ p_n(x), & \text{if } x_{n-1} \leq x < x_n \end{cases} \quad (7.10)$$

Therefore, a piecewise polynomial of order k with break sequence ξ (necessarily strictly increasing) are defined in [De Boor 01a] as any function f that, on each of the half-open intervals $[\xi_j, \xi_{j+1})$ agrees with some polynomial of degree $< k$. This way of defining piecewise polynomials makes the function right-continuous.¹⁷⁴ This definition also implies a change in the notation as the data sites (x_i) are referred to as break points (ξ_i) .

It must be highlighted that this change in the notation also allows to define interpolating piecewise polynomials in which the extremes are not the data sites to be interpolated. This

¹⁷³ Unless the new point is included in the $n - 1$ degree interpolating polynomial to be upgraded.

¹⁷⁴ This fact is a convention, but it seems quite extended.

means that the interpolating function may have less pieces than data sites. These pieces will connect in the break points and will go through all the data sites. So, it is a good option separating both concepts.

Depending on the degree of the polynomials used to connect the discrete break points, the resulting curve will be nearly smooth. Thus, it is common to find the following application of piecewise polynomial functions:

- Piecewise linear interpolation.
- Piecewise quadratic interpolation.
- Piecewise cubic interpolation.

Piecewise linear interpolation consists in connecting the break points by means of straight lines. Similarly, piecewise quadratic and cubic interpolation use quadratic and cubic polynomials for connecting the break points, that is, parabolas and cubic curves. Other types of piecewise polynomial applications are also possible, but they are less common.

Thus, the good features of cubic polynomial have extended the use of piecewise cubic interpolation in many fields such as computer aided design and computer graphics. These features are:

- Lowest degree polynomial that can support an inflection.
- Well behaved numerically.

Additionally, as it has already been mentioned, polynomials with a degree higher than three tend to be very sensitive to the position of the data sites (or break points when talking about piecewise interpolation) and so they may not show a sufficiently smooth behaviour.

Focusing on piecewise cubic polynomials, the aim is to find the set of cubic polynomials that together define the interpolating function $f(x)$. This is a classical numerical problem which implies solving a system of a few linear equations which depends on the number of segments to be joined.

Thus, in the case of having $M + 1$ break points, M segments are defined and so M cubic polynomials must be found. As every cubic polynomial $p_i(x)$ has 4 coefficients $(a_{3,i}, a_{2,i}, a_{1,i}, a_{0,i})$ which have to be computed, the total number of equations which are needed for solving the system is $4M$. In order to solve this linear system, $4M$ constraints have to be imposed to the system. These constraints come from the continuity conditions which are demanded to the resulting interpolating function.

The conditions that are demanded per segment are:

- Enforcing C^0 continuity at each extreme of the segment, so the function passes through the pair of break points which define it. This introduces two constraints to the function:
 - $p_i(x_{i-1}) = y_{i-1}$
 - $p_i(x_i) = y_i$

- Enforcing C^1 continuity, that is, continuity in the slopes of the polynomials that join. This demand adds an additional constraint:

$$\circ \quad p'_i(x_i) = p'_{i+1}(x_i)$$

- Enforcing C^2 continuity, which means continuity in the curvature of the polynomials that join. This fact adds another equation to the system:

$$\circ \quad p''_i(x_i) = p''_{i+1}(x_i)$$

- Additionally, there is a need for including two additional equations in the extremes of the domain of the interpolating function to obtain a determined system. Depending on the chosen criterion, different kinds of piecewise functions can be obtained.

If all these equations are put together, the resulting linear system has four coefficients per segment. Nowadays, this fact should not be a problem for current computers unless the number of equations is enormous.

However, it is possible to reduce the complexity of the problem by parameterizing the equations which consists in using as the dependent variable a parameter that takes values in the segment $[0..1]$ instead of using the x values. This fact is exploited by numerical computation tools to reduce the computational burden.

The result of the linear system is called a “natural cubic spline”. But the main drawback is the fact that any change in the equations implies having to solve the whole system again, that is, the curve is determined by one set of linear equations $M\mathbf{a} = \mathbf{y}$. This means that there is no local control so changes in one part of the interpolating function may affect distant parts of it. This fact opens the door to other schemes which provide local shape control. The most important are:

- Piecewise cubic Hermite interpolation or Hermite spline.
- Piecewise cubic Bessel interpolation.
- Akima's interpolation.
- Catmull-Rom splines.
- Cardinal splines.

Strictly speaking none of these schemes are cubic splines as they do not guarantee C^2 continuity, however this condition has been relaxed along time and nowadays, most of them are included when talking about splines.

These schemes offer the possibility of having complete control of the values and slopes in the extremes of each piece which implicitly provides local control of the interpolating function. Thus, each segment is calculated separately, so to obtain the whole curve the calculations must be performed per piece that is repeated over all breaking points. These are the equations that have to be solved per each piece (i):

- $p_i(x_{i-1}) = y_{i-1}$
- $p_i(x_i) = y_i$
- $p'_i(x_{i-1}) = s_{i-1}$
- $p'_i(x_i) = s_i$

This means using C^0 and C^1 continuity constraints at the break points. Thus, all these schemes offer different ways of defining the C^1 continuity constraints (s_{i-1}, s_i) . These values can be provided by the user or can be calculated taking into consideration other break points, such as (x_{i-2}, y_{i-2}) , (x_{i+1}, y_{i+1}) , or also a shape parameter as it is the case for Cardinal splines.

7.3.2.3 Spline interpolation

Splines are piecewise polynomial functions whose pieces are joined in a smooth fashion. Thus, they are a generalization of polynomials that overcome the main drawbacks of the latter such as the already mentioned Runge's phenomenon.

Schoenberg defined them as “pp” functions of order k on some interval with $k - 2$ continuous derivatives on the connection points which were called “knots” [Schoenberg 46]. The order of a spline refers to the maximum order of all the pieces that form the spline function (i.e. $f(x)$). Thus, all polynomials which define the spline function must be of order k or below. There is a need for clarifying the difference between degree and order of a polynomial. This clarification will be provided in the next section.

The initial definition was relaxed by Carl De Boor [De Boor 01] who provided a more general view on what a spline was. His definition broadened the concept of spline as he considered that a spline function of order k with knot sequence t was any linear combination of B-splines of order k and knot sequence t . As it will be described in the coming sections, spline functions define a function space $(\mathcal{S}_{k,t})$ where the B-splines are the basis functions.

7.3.3 Splines functions

7.3.3.1 Spline function space

The spline functions or simply splines define a function space, where polynomials and piecewise polynomials are included, that is called $\mathcal{S}_{k,t}$ [De Boor 01]. For defining such a space, the spline theory introduces the concept of “order” of a polynomial. Thus, polynomials of order k include all polynomials of degree $k - 1$ and below. This way the set of these polynomials has the structure of a vector space, which may be endowed with a norm and an inner product. In a function space, the vectors are functions and, in this case, these functions are piecewise polynomials.

This fact introduces another term in the equation, which is that all vector spaces have at least a basis. In this case, there are several bases that can span the whole space and that are linearly independent. This means that any function belonging to the $\mathcal{S}_{k,t}$ space can be expressed as a linear combination of the basis elements. Some examples of these bases are B-splines, Bernstein polynomials or the truncated power basis.

In this work, the focus is going to be placed in B-splines which will be introduced in section 7.3.4.

7.3.3.2 Spline boundary conditions

Any of the interpolating schemes presented in section 7.3.2.2 and also the “pure” spline functions require the definition of the boundary conditions on the extremes of the interpolating interval. This boundary conditions affect the first derivative (s_0, s_n) and sometimes the second one (f_0'', f_n''). Some examples of these conditions are:

- The first derivatives at the extremes of the underlying function g are known: ($s_0 = g'(x_0), s_n = g'(x_n)$). This is called the complete cubic spline interpolant of g .
- The second derivatives at the extremes of the underlying function g are known: ($f_0'' = g''(x_0), f_n'' = g''(x_n)$).
- Natural spline interpolation: ($f_0'' = f_n'' = 0$).
- Not-a-knot condition [De Boor 66]: s_0, s_n are chosen so that the first and the last interior knots are not active.
- Computing an estimate for: $g'(x_0), g'(x_n), g''(x_0), g''(x_n)$.
- Defining different values at the two ends of the interval.

Depending on the information available concerning the underlying function, the constraints that must be imposed on the extremes and the application of the spline function, one can choose a boundary condition or another.

7.3.3.3 Splines curves

Spline functions have been initially presented for one dimension. However, these ideas can be extended in order to describe arbitrary curves in two- and three-dimensional spaces. Even more, this theory can also be extrapolated to surfaces (e.g., NURBS).

In this thesis, 2D and 3D splines curves could have been used to describe the trajectory followed by an aircraft not only on the surface of the airport but also in the final approach phase of the flight. However, the final decision was to focus on Bayesian solutions.

Until now the dependent variable y has been described by means of the independent one, x . In order to leave x, y and z variables to the three spatial dimensions, a new independent variable, t , is going to be introduced. This new variable is considered a parameter in most of the situations but, in this case, it will represent the time. This way each spatial dimension is described as follows:

- $x = x(t)$
- $y = y(t)$
- $z = z(t)$

So, a 2D plot (x, y) is really one view of a three dimensional one (x, y, t) . The same for the 3D case (x, y, z) .

7.3.4 B-splines

The B-spline functions (B_{jkt}) are piecewise polynomial functions that can form a basis for the $\$_{k,t}$ space already introduced in section 7.3.3.1. Therefore, all the functions which are included in the $\$_{k,t}$ space can be expressed as a linear combination of the corresponding B-splines. They are a good choice of a basis due to their minimal support, their smoothness, and the fact that they provide a domain partition of unity. A basis for the $\$_{k,t}$ space, which is composed of B-splines, is dependent on the knot sequence t and on the order k of the spline functions to be represented.

Thus, B-splines are defined by the knot they refer to (j), their order (k) and the knot sequence (t). Usually, the knot sequence is omitted as it can be inferred from the context. So, it is common to represent the B-spline function as B_{jk} or even as B_j instead as B_{jkt} . There are two ways of deriving the expression which defines the B-splines of order k for a certain knot sequence, t . One is based on divided differences and the other is based on a recursion law. In this work, the second option is the one be presented.

The recursion law states that [De Boor 01]:

$$B_{jk} = \omega_{jk} B_{j,k-1} + (1 - \omega_{j+1,k}) B_{j+1,k-1} \quad (7.11)$$

Where:

$$\omega_{jk}(x) := \frac{x - t_j}{t_{j+k-1} - t_j} \quad (7.12)$$

Being the initial condition:

$$B_{j1}(x) = \begin{cases} 1, & \text{if } t_j \leq x \leq t_{j+1} \\ 0, & \text{otherwise} \end{cases} \quad (7.13)$$

When developing all these expressions, the graphical result is the one presented in the following figure:

Figure 7.2: Normalized B-splines of order $k = 1, 2, 3, 4$. Source: [B-splines]

So, based on B-splines, the functions belonging to the $\mathcal{S}_{k,t}$ space can be expressed in the following manner:

$$f(x) = \sum_{j=1}^n \alpha_j \cdot B_{jk}(x) \quad (7.14)$$

Where:

- α_j : B-splines coefficients of $f(x)$ with respect to the B-spline basis (B_{jk}) .

7.3.5 Spline representation

There are two ways of representing spline functions: one based on the fact that they are “pp” functions, and another which exploits that B-splines are a basis for the $\mathcal{S}_{k,t}$ space. Although they are ways of representing the same thing, their different features make one more suitable than the other depending on the framework where they are used. So, both are going to be presented in the following sections.

7.3.5.1 ppform

This representation exploits the fact that splines are “pp” functions. Thus, the “ppform” of a spline needs to define the following parameters:

1. k : maximum order of the piecewise polynomials that compose the spline function.
2. l : number of pieces of the interval.
3. $\xi_1, \xi_2, \dots, \xi_{l+1}$: strictly increasing sequence of break points, where the pieces join.
4. $f_1, f'_1, f''_1, f_2, f'_2, f''_2, \dots, f_{l+1}, f'_{l+1}, f''_{l+1}$: the continuity conditions in each of the break points (i.e. value, first derivative, second derivative). It is usual to present this information in matrix form.

Summing up, this form is based on considering the splines as a series of polynomial pieces with the corresponding constraints in the extremes.

7.3.5.2 B-form

On the other hand, the B-form exploits the fact that splines are functions that belongs to the function space $\mathcal{S}_{k,t}$, so they can be represented by means of the corresponding B-spline basis. In this case, the following parameters are needed:

1. k : maximum order of the B-splines that compose the spline function.
2. n : number of B-splines coefficients.
3. $t = (t_i)_{i=1}^{n+k}$: sequence of knots which can be partially coincident.
4. $\alpha = (\alpha_i)_i^n$: B-spline coefficients of the spline function with respect to the B-spline basis $(B_i)_i^n$.

Each form has its advantages and disadvantages so, depending on the application, one or the other can be used.

7.3.6 Application of spline functions

There are two clear applications of the spline functions: interpolation and approximation. The target when using splines for interpolation is to provide the values between the interpolation sites in order to build a continuous function, which can have different objectives. On the other hand, approximation is done for obtaining a function from the $\mathcal{S}_{k,t}$ space that is closed enough to the approximated one, which does not belong to the function space. There are different criteria for defining the approximating function, which will depend on the specific needs of the problem. Therefore, there are “interpolating splines” and “approximating splines”.

7.3.6.1 Interpolating splines

As already mentioned, interpolating consists in recovering the underlying function by providing the intermediate values between interpolation sites normally adding some constraints on the smoothness of the solution. This is done because the only information that is available is some samples of the interpolated function. Hence, the interpolating spline and the interpolated function coincide in value at the interpolating sites.

In this application, the aim is to find a spline that matches the samples (values at the interpolating sites) of the function to be interpolated, $g(x)$, and that satisfies the same conditions as g is known to satisfy. Additionally, it is expected that it presents a better behaviour in other features, which are interesting for the subsequent processing of the data. This better behaviour is usually related to the smoothness and continuity conditions (i.e. continuous derivatives).

This application is preferred when the samples are known to be good enough so no additional processing (i.e. filtering or smoothing) is needed for removing errors. In this work, this could be the case when computing predicted trajectories based on the knowledge of the airport layout and on the ground route to be followed by the aircraft.

7.3.6.2 Approximating splines

The other main application of splines is the approximation of functions that do not belong to the $\mathcal{S}_{k,t}$ space. So, the aim in this case is using a spline instead of the original one exploiting the good features of the former.

Another situation in which approximation is needed is when it is known that the data to be approximated are heavily affected by noise. In this situation, the target is to define a spline with enough degrees of freedom in order to approximate the underlying function (tendency), but not enough to follow the noise which is assumed to be varying with time quicker than the function (i.e. the noise has a bandwidth larger than the underlying function or signal). This is the case that will be faced in this work when trying to reconstruct the actual trajectory followed by an aircraft while moving around the airport based on measurements provided by surveillance sensors.

Unlike interpolating splines, the curve to be defined in this case passes near to the data but not necessarily through them. There are two schemes that can be applied for reaching the best approximating spline which are going to be presented next.

7.4.6.2.1 Smoothing splines

This scheme tries to find a trade-off between the smoothness of the approximating spline and how close it is from the data to be approximated. These are conflicting objectives as the closer we get to the data the less smooth the approximating function will be.

The error model that it is usually applicable is:

$$y_i = g(x_i) + \varepsilon_i \quad (7.15)$$

Where:

- $g(x)$: function to be approximated.
- ε_i : measurement noise which can be modelled as a white Gaussian noise process with zero mean and σ^2 variance.

As it has been mentioned previously, the aim is to recover $g(x)$ by building a spline function, i.e. $f(x)$, that minimizes the following expression which implements the balance between smoothness of the function (second term) and closeness to the data (first term).

$$p \sum_{i=1}^N \left(\frac{y_i - f(x_i)}{\sigma} \right)^2 + (1 - p) \int_{x_1}^{x_N} \left(f^{(m)}(t) \right)^2 dt \quad (7.16)$$

The latter expression depends on the value that it is assigned to p . It will determine which feature is the dominating one while obtaining the approximating spline. This approximating curve can go from a pure interpolation scheme in case $p = 1$, to a broken line in case $p = 0$.

As it can be shown [De Boor 01], the solution to this problem turns to be a “natural” spline of order $k = 2m$ with simple knots at $x_2, x_3, \dots, x_{N-1}$. In this case, as the focus would be placed on cubic splines, which means $k = 4$, the natural boundary conditions are $f''(x_1) = f''(x_N) = 0$.

7.4.6.2.2 Least-squares approximant

The other scheme for obtaining an approximating spline to noisy data is the one based on the least-squares approximation by means of a spline function.

The target in this case is to find the spline from the $S_{k,t}$ space which minimizes the distance to the data in the least square sense. This is done by computing the appropriate B-splines coefficients $(\alpha_i)_1^n$ of such a spline. In order to compute them, the system of normal equations is formed with the assumption that the B-spline basis $(B_i)_1^n$, which is going to be used, is relatively well conditioned. It can be proved that this is true for moderate k [De Boor 01].

The system has the following appearance:

$$\sum_{j=1}^n \langle B_i, B_j \rangle \alpha_j = \langle B_i, g \rangle \quad (7.17)$$

Where:

- $\langle B_i, B_j \rangle = \int_{x_1}^{x_{j+k}} B_i(x) B_j(x) dx$ represents the inner product.

Solving this system of equations provides the values of the B-spline coefficients (α_j) that define the least-squares approximation spline to the noisy data.

7.3.7 Application of spline curves to the ATM world

There are several situations in the ATM world in which splines can be used to generate, predict, and reconstruct a trajectory performed by an aircraft irrespective of whether it is in the air or on the ground.

The focus in this work is placed on the ground, moreover the airport is the framework of all the research.

7.3.7.1 Trajectory generation

Splines can be used to obtain a first look of the trajectory to be followed by an aircraft from its stand to the runway or vice versa. The generation of this trajectory can also be exploited by controller ground support tools for trajectory prediction purposes for safety nets (conflict prediction), in surface routing algorithms, and for auxiliary functions such as bias estimation. This application is based on interpolating splines as the position of the points is well defined as they belong to the airport map.

In a broader context, these algorithms are present in most of the trajectory prediction (TP) modules that can be found in current FDP systems and that support the MTCD or CMON controller tools. These tools were introduced when FDP systems were presented (ref. 1.2.3.1).

7.3.7.2 Trajectory filtering

Interpolating splines also appear within tracking systems with road constraints as it is presented in [Hansberg 08] or more recently in [López-Araquistain 19].

7.3.7.3 Trajectory reconstruction

Approximating splines can be used for building an estimate of the actual trajectory followed by an aircraft while moving around the airport. In this case, the data is affected by errors as it comes from surveillance sensors measurements. The estimated trajectory becomes what it is called the reference trajectory, already mentioned at the beginning of this chapter, and its normal use is for comparing it to the measurements provided by surveillance sensors or the output of surveillance tracking systems.

7.4 Bayesian Estimation Techniques

7.4.1 Introduction

These techniques try to take advantage of the prior information that is available concerning the target to generate the optimal estimation of the system state variables or the system state vector.

Moreover, these techniques take on board results coming from both, the estimation theory and the control theory. As previously presented, the estimation theory is in the core of the algorithms that have already been presented in section 6.10, where biases were estimated in order to be removed from the measurements. This fact was presented as a prerequisite for the further processing that is needed to finally reach the reference trajectory followed by the aircraft or the mobile. This is so because computing the reference trajectory is a data fusion process, where the estimation along time of what is called the “state” of the system is needed.

It is in this point where the control theory provides its support. Thus, there are several concepts that are imported from this theory which underpin the presentation of the problem. They are mainly focussed on the definition of the system internal description.

The target of the control theory is to obtain the inputs to a system that allow obtaining certain outputs from it, that is, finding the way of influencing the behaviour of the system under control.¹⁷⁵ This fact implies having a “quite” good knowledge of the behaviour of the system. The depth of this knowledge depends on the application, where the control is needed, so it can vary considerably from one to another.

Within the control theory, there are two main approaches:

- The classical control theory.
- The modern control theory.

The former generally describes a system by means of frequency domain techniques while the latter uses time domain ones.

The classical control theory is a consequence of the burst of development associated to World War II. It is focussed on analysing the system to be controlled from an input-output point of view using high-order differential/difference equations or, equivalently, its Laplace transform (transfer function). This approach is based on a “black box” model of the system, see Figure 7.3a. Consequently, it is generally focussed on a frequency domain analysis, so being related to the concepts of transfer function, the impulse or step response of the system and the graphical representation of the input-output behaviour of systems using Bode diagrams, pole-zero plots and other representations. It is very useful when analysing observability, stability, and controllability of SISO (Single-input Single-output) systems. This approach is mainly related to authors such as Bode, Nyquist, Nichols, and Evans.

¹⁷⁵ There are different means of controlling the system, i.e. open-loop or closed-loop methods.

On the other hand, it was in the 1950's when there was a change in the approach to be followed for modelling the systems to be controlled that resulted in the development of the state space representation. This approach tries to model the system as a “white box” in which its internal states are of interest for describing the behaviour of the system, see Figure 7.3b. Thus, this representation is closer to the physical reality than the classical one and it can also consider an improved statistical description of the system behaviour, which opens the door to better applications. All these changes gave rise to what is known as the modern control theory and authors like Wiener [Wiener 49] and Kalman [Kalman 60] boosted this new way of modelling systems.

Figure 7.3: Black box (a) and white box (b) models. Source: own elaboration.

The incorporation of improved statistical models introduces the link with the estimation theory. Thus, from the control theory, a model for the internal representation of the system is obtained, while it is the estimation theory the one that provides the methods for estimating the internal states and the outputs of such a model.

The same way as in the control theory there are two main approaches. In the estimation theory, there are also two philosophies, namely:

- The classical estimation theory.
- The Bayesian estimation theory.

The classical estimation theory relies on the fact that the parameters to be estimated have a fixed value, assuming no prior knowledge of these values, while the Bayesian one assumes that the unknown parameters are a realization of a random variable (or process) that follows a certain prior distribution. The latter is the philosophy that is used in the techniques to be described in the coming sections. This way, the prior knowledge about the parameters or system state variables is exploited to define better estimators.

7.4.2 System classification

A system is a collection of elements that interacts with its environment via a set of input variables $u(t)$ (or $u(k)$ if it is a discrete-time system) and output variables $y(t)$ (or $y(k)$) to perform a common goal.

Systems can be classified in different ways depending on the properties that characterize them. The most important classifications from the point of view of this thesis are the ones presented hereafter.

7.4.2.1 Autonomous versus controlled systems

An autonomous system is the one that is input free. Opposite to a controlled system, which is the one that has at least an input signal or variable that modulates its output, so the

expected value can be reached. Taking into consideration the aircraft, they are controlled systems, as there is a pilot inside who directs them towards their intended destination. However, the inputs provided by the pilot (e.g. engine thrust or direction wheel) are unknown to the surveillance sensors and, for extension, to the whole surveillance chain.

7.4.2.2 Continuous-time versus discrete-time systems

Continuous-time systems are characterized by a time evolution where the indices $t \in \mathbb{R}$, which is a continuous set. On the other hand, discrete-time systems evolve with time indices $k \in \mathbb{Z}$. In other words, a system is continuous-time (discrete-time) when its inputs/output variables are continuous-time (discrete-time).

A general representation of the observations can be:

$$y(t) = f(u(t), t) \quad \text{or} \quad u(t) \rightarrow y(t) \quad (7.18)$$

$$y(k) = f(u(k), k) \quad \text{or} \quad u(k) \rightarrow y(k) \quad (7.19)$$

Most of real-world systems are continuous-time, while discrete-time systems are difficult to find in nature. Some examples of the continuous-time systems are physical systems such as: vehicles, pendulums, springs, electromagnetic signals, etc. Examples of discrete-time systems are bank accounts, national economy growth models, digital words, etc. However, with the advent of digital computers, it has appeared a new type of discrete-time system, i.e. sampled data systems which are obtained due to the discretization of continuous-time systems. As a result, the observations are sampled at discrete, equally spaced time intervals, so the time index (k) is integer-valued. These systems can be represented by a mathematical model such as:

$$y(kT) = f(u(kT), kT) \quad \text{or} \quad y(k) = f(u(k), k) \quad (7.20)$$

Where:

- T : sampling period which is usually omitted for the sake of clarity.

In the case of inherent discrete-time systems, there is no need to represent the sampling period.

7.4.2.3 Continuous-state, discrete-state and hybrid systems

These terms appear as a way of classifying systems considering the values that their state variables can take on. Hence, a discrete-state system is the one whose state variables can take on discrete values from a finite set (e.g. $X_k \in \{\text{red}, \text{blue}, \text{black}\}$). A continuous-state system is the one whose state variables take values on a continuous set (e.g. $X_k \in \mathbb{R}$). Finally, a hybrid system is the one whose state consists of both discrete and continuous-state variables (e.g. an aircraft which has continuous-state variable like position, velocity and acceleration, and a discrete state variable for defining the current motion model) or, in other words, it is a system with both continuous and discrete dynamics.

7.4.2.4 Static versus dynamic systems

Depending on the way of expressing the time evolution of a system, it can be classified as static, in case this evolution can be stated by means of algebraic equations, or as dynamic or dynamical,¹⁷⁶ when its evolution with time is described by means of a differential or difference equation depending on the continuous-time or discrete-time nature of the system.

This property can also be stated taking into consideration the dependency of the outputs on the inputs. Thus, a static system is the one whose output depends only on its current input while the output of a dynamic system depends not just on the current input (at t) but also on the past inputs that go from $-\infty$ to t . The latter can be solved by knowing the so-called state of the system at one previous time instant. As it will be shown in 7.4.3, this matter is related with the need for having the initial conditions to be applied to the set of difference/differential equations.

7.4.2.5 Linear versus nonlinear systems

A system is called linear when the relation between its inputs and its outputs satisfy the additivity and homogeneity properties. That is:

$$\text{If } \left. \begin{array}{l} u_1(t) \xrightarrow{\text{yields}} y_1(t) \\ u_2(t) \xrightarrow{\text{yields}} y_2(t) \end{array} \right\} \text{ then } \left\{ \begin{array}{l} u_1(t) + u_2(t) \xrightarrow{\text{yields}} y_1(t) + y_2(t) \\ \alpha u_1(t) \xrightarrow{\text{yields}} \alpha y_1(t) \quad \forall \alpha \in \mathbb{R} \end{array} \right. \quad (7.21)$$

In other textbooks, this is also equivalent to satisfy the superposition property.

$$\text{If } \left. \begin{array}{l} u_1(t) \xrightarrow{\text{yields}} y_1(t) \\ u_2(t) \xrightarrow{\text{yields}} y_2(t) \end{array} \right\} \text{ then } \alpha u_1(t) + \beta u_2(t) \xrightarrow{\text{yields}} \alpha y_1(t) + \beta y_2(t) \quad \forall \alpha, \beta \in \mathbb{R} \quad (7.22)$$

This has a consequence which is that:

$$\text{If } u(t) \equiv 0 \xrightarrow{\text{yields}} y(t) \equiv 0 \quad (7.23)$$

The same is applicable for discrete-time system with just changing the t by the k index.

Otherwise, the system is nonlinear.

7.4.2.6 Time varying versus time invariant systems

A system is called time-invariant if the output to a certain input does not change over time. That is:

$$u(t) \xrightarrow{\text{yields}} y(t) \quad \Rightarrow \quad u(t - t_0) \xrightarrow{\text{yields}} y(t - t_0) \quad \forall t_0 \quad (7.24)$$

¹⁷⁶ Within this text the terms dynamic system and dynamical system are going to be used as synonyms.

$$u(k) \xrightarrow{\text{yields}} y(k) \quad \Rightarrow \quad u(k - k_0) \xrightarrow{\text{yields}} y(k - k_0) \quad \forall k_0 \quad (7.25)$$

Strictly speaking all systems are time-varying, because you can always find a bigger system that contains the system under analysis, which is time-varying, so the variation of the former affects the latter. In fact, in mechanical systems all parameters suffer from aging, which causes the system to slowly change its behaviour along time. However, if the observation interval is short enough, the time-invariant assumption can hold.

7.4.2.7 Deterministic versus stochastic systems

A deterministic system is the one whose behaviour is defined without uncertainty by means of a mathematical model. When uncertainty appears in the system model, then it turns into a stochastic system.

This uncertainty can take the form of system noise, when the model is not precise enough or it presents disturbances, or measurement noise when the system internal states are measured by noisy sensors or sensors which are not able to appropriately measure the corresponding states due to observability problems.

7.4.2.8 With time-delay versus without time-delay systems

A system is said to be “without time-delay” when its output at t or k depends just on the input at t or k . These systems are also called memoryless. This is also equivalent to be independent of the input except at t or k . Otherwise, the system has memory or present a time-delay.

The study of systems with memory is much more mathematically involved than the ones without time-delay, so they are going to be avoided in this work.

7.4.2.9 Causal and non-causal systems

As defined in [Oppenheim 97]: “a system is causal if the output at any time depends only on values of the input at the present time and in the past”. These systems are also called physical systems or non-anticipative as their output does not tell anything about future values of the input.

This way, aircraft movement models are causal as they do not anticipate anything related to the action that the pilot is going to take in the near future. Linking them with the previous ones, memoryless systems are causal, while not all causal systems are memoryless as they can be affected by past values. However, the systems that are of interest for this thesis (i.e. PET) might not comply with the causality constraint as they work with data that have been previously recorded in order to obtain the so-called “reference” trajectories.

7.4.3 Dynamical systems

The term “dynamical system” groups most of the systems that are of interest for engineering works. Hence, this is the case for a PET, as aircraft and airport vehicles belong to this group. This fact allows us to apply the most representative models for analysing their behaviour along time.

For understanding the meaning of the term “dynamical system”, it is mandatory to understand what features are assumed when the system is characterised as dynamical or dynamic. In this point, it is interesting the interpretation that is done in [Polderman], which says that “The adjective dynamical refers to phenomena with a delayed reaction, phenomena with an aftereffect, with transients, oscillations, and, perhaps, an approach to equilibrium”. This description remarks how important is the time evolution for these systems.

This relationship is formalized by means of differential or difference equations which explain that the evolution of the internal variables ($x(t)$ or $x(k)$) is not instantaneous but there is a transient before reaching the expected state.

Therefore, dynamical systems can be described in two different ways that are aligned with the two main approaches that have been presented concerning the control theory; that is, the (“classical”) external view, more descriptive; and the (“modern”) internal view, more explanatory.

The external view is closer to the classical control theory, so it is focussed on the description of the system from an input/output point of view or as it is also referred to as a “black box” model (see Figure 7.3a). The relation among the inputs and the output can be modelled by different means depending on the features of the system. Hence, the input-output relation of dynamical systems is generally described in the time domain by means of differential equations whether the system is continuous or by means of difference equations if it is discrete. In the most general case, the mathematical model of such a system in the continuous-time case is given by:

$$\begin{aligned} \frac{d^n y(t)}{dt^n} + a_{n-1} \frac{d^{n-1} y(t)}{dt^{n-1}} + \dots + a_1 \frac{dy(t)}{dt} + a_0 y(t) \\ = b_m \frac{d^m u(t)}{dt^m} + b_{m-1} \frac{d^{m-1} u(t)}{dt^{m-1}} + \dots + b_1 \frac{du(t)}{dt} + b_0 u(t) \end{aligned} \quad (7.26)$$

And it takes the following expression for the discrete-time case:

$$\begin{aligned} y(k+n) + a_{n-1} y(k+n-1) + \dots + a_1 y(k+1) + a_0 y(k) \\ = b_m u(k+m) + b_{m-1} u(k+m-1) + \dots + b_1 u(k+1) + b_0 u(k) \end{aligned} \quad (7.27)$$

Where

- n : order of the system.
- y : output.
- u : input.

This relation can also be described in the frequency domain by means of the transfer function of the system, which in the continuous-time case is given by:

$$G(s) = \frac{Y(s)}{U(s)} = \frac{b_m s^m + b_{m-1} s^{m-1} + \dots + b_1 s + b_0}{s^n + a_{n-1} s^{n-1} + \dots + a_1 s + a_0} \quad (7.28)$$

Or, in the discrete-time case:

$$G(z) = \frac{Y(z)}{U(z)} = \frac{b_m z^m + b_{m-1} z^{m-1} + \dots + b_1 z + b_0}{z^n + a_{n-1} z^{n-1} + \dots + a_1 z + a_0} \quad (7.29)$$

In the first expression (7.28), the Laplace transform is taken, and in the second one (7.29) the Z-transform. This view is of less importance to this work, as it is the internal state the one that is to be estimated and predicted to obtain the “reference” trajectory.

On the other hand, the internal view is focussed on explaining the inside workings of the system following a “white box” approach (see Figure 7.3b) where the internal variables, the inputs and the outputs are mainly related through the classical mechanics.

Figure 7.4: System representation. Source: own elaboration.

This relationship is formalized by means of first-order differential/difference equations, which describe the evolution of the internal variables ($x(t)$ or $x(k)$). This view is linked with the state-space representation typical of the modern control theory and it takes the following general form for the continuous-time:

$$\begin{aligned} \frac{dx_1(t)}{dx} &= x_2(t) \\ \frac{dx_2(t)}{dx} &= x_3(t) \\ &\vdots \\ \frac{dx_n(t)}{dx} &= -a_0 x_1(t) - a_1 x_2(t) - \dots - a_{n-1} x_n(t) + u(t) \end{aligned} \quad (7.30)$$

Which can be expressed in the state space form given by:

$$\begin{bmatrix} \dot{x}_1(t) \\ \dot{x}_2(t) \\ \vdots \\ \dot{x}_{n-1}(t) \\ \dot{x}_n(t) \end{bmatrix} = \begin{bmatrix} 0 & 1 & 0 & \dots & \dots & 0 \\ 0 & 0 & 1 & 0 & \dots & 0 \\ \vdots & \vdots & 0 & \ddots & 0 & \vdots \\ \vdots & \vdots & \vdots & \ddots & 1 & 0 \\ 0 & 0 & \dots & \dots & 0 & 1 \\ -a_0 & -a_1 & -a_2 & \dots & \dots & -a_{n-1} \end{bmatrix} \begin{bmatrix} x_1(t) \\ x_2(t) \\ \vdots \\ x_{n-1}(t) \\ x_n(t) \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \\ \vdots \\ 0 \\ 1 \end{bmatrix} u(t) \quad (7.31)$$

$$y(t) = [1 \quad 0 \quad 0 \quad \dots \quad \dots \quad 0] \begin{bmatrix} x_1(t) \\ x_2(t) \\ \vdots \\ \vdots \\ x_{n-1}(t) \\ x_n(t) \end{bmatrix} \quad (7.32)$$

Similar expressions can be obtained for discrete-time dynamical systems:

$$\begin{aligned} x_1(k+1) &= x_2(k) \\ x_2(k+1) &= x_3(k) \\ &\vdots \\ x_n(k+1) &= -a_0x_1(k) - a_1x_2(k) - \dots - a_{n-1}x_n(k) + u(k) \end{aligned} \quad (7.33)$$

Being its state space form given by:

$$\begin{bmatrix} x_1(k+1) \\ x_2(k+1) \\ \vdots \\ \vdots \\ x_{n-1}(k+1) \\ x_n(k+1) \end{bmatrix} = \begin{bmatrix} 0 & 1 & 0 & \dots & \dots & 0 \\ 0 & 0 & 1 & 0 & \dots & 0 \\ \vdots & \vdots & 0 & \ddots & 0 & \vdots \\ \vdots & \vdots & \vdots & \ddots & 1 & 0 \\ 0 & 0 & \dots & \dots & 0 & 1 \\ -a_0 & -a_1 & -a_2 & \dots & \dots & -a_{n-1} \end{bmatrix} \begin{bmatrix} x_1(k) \\ x_2(k) \\ \vdots \\ \vdots \\ x_{n-1}(k) \\ x_n(k) \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \\ \vdots \\ \vdots \\ 0 \\ 1 \end{bmatrix} u(k) \quad (7.34)$$

$$y(k) = [1 \quad 0 \quad 0 \quad \dots \quad \dots \quad 0] \begin{bmatrix} x_1(k) \\ x_2(k) \\ \vdots \\ \vdots \\ x_{n-1}(k) \\ x_n(k) \end{bmatrix} \quad (7.35)$$

For a detailed explanation of how these expressions are obtained please refer to [Gelb 74].

Additionally, there is a formal definition of a dynamical system (Σ) as a mathematical model in which the objects of interest are functions of time and the universe is a function space, see [Polderman] for a more detailed description. Thus, the dynamical system is defined by means of a triple:

$$\Sigma = (\mathbb{T}, \mathbb{W}, \mathfrak{B}) \quad (7.36)$$

Where:

- $\mathbb{T}$: *time axis*, which is a subset of $\mathbb{R}$, if the system is continuous-time; and a subset of $\mathbb{Z}$, if the system is discrete-time.
- $\mathbb{W}$: set called the *signal space*, *phase space* or *state space*. It defines the look like of the variables whose evolution in time is going to be described. In our case, it takes the form of a finite-dimensional vector space, although it can be an infinite-dimensional function space.

- $\mathfrak{B}$: *behaviour*¹⁷⁷ of the system and consists in the family of time trajectories taking their values in the signal or state space. These trajectories must comply with the fixed rules or laws that govern the system.

In our case, the dynamical systems are the aircraft; the time axis $\mathbb{T}$ is an interval of $\mathbb{Z}$ as the system state is periodically sampled by sensors; the state space is $\mathbb{R}^N$ with N depending on the state variables that are needed for representing the state of the system; and finally, $\mathfrak{B}$ is the set of trajectories that defines the time evolution of the system. These trajectories are defined by the aircraft motion model and its initial state.

Therefore, it seems appropriate to consider that a moving aircraft is a dynamical system, due to the fact that its behaviour changes with time as one of its crucial features, and that it depends on what has happened previously. This behaviour is related to the movement model that the aircraft is following in each moment. This leads us to a further step forward as the aircraft can follow different possible movement models or modes, becoming then a hybrid system. This fact will be addressed when talking about manoeuvring targets in section 7.4.6.3.

7.4.4 The state-space model

As already mentioned, the state-space technique represents the modern approach to system control theory and its applications. This representation, which is also known as the “time-domain approach”, consists of a mathematical model of the dynamical system, which is composed of a set of variables (inputs, outputs and state variables) that are related by first-order differential/difference equations as previously shown (eqns. 7.30-7.32 for continuous-time systems; and, eqns. 7.33-7.35 for discrete-time systems).

The state-space model (SSM) consists of two mathematical equations, namely:

1. The state equation.
2. The observation equation.

Ergo, there are different ways of obtaining such a mathematical representation. It depends on the information concerning the system that is available, which goes from differential or difference equations to transfer functions; and on the measurements obtained from it.

This approach is very convenient for representing high-order dimensional and complex system as it makes tractable what would otherwise be a notationally intractable analysis. Thus, if we try to model such a system in the “frequency-domain”, assuming i inputs and o outputs, the number of Laplace transforms that are needed for representing the same information about the system is: $i \cdot o$. Additionally, there are limitations imposed by the “frequency-domain” approach that do not appear in the state space representation.

The way in which dynamical systems are represented allows to easily apply numerical algorithms, mainly from the area of numerical linear algebra, which are extremely

¹⁷⁷ This is also defined as a subset of $\mathbb{W}^{\mathbb{T}}$, which defines the collection of all maps from $\mathbb{T}$ to $\mathbb{W}$.

efficient for numerical calculations. Thus, if the dynamical system is linear and time invariant, its representation can be expressed in matrix form.

Therefore, the SSM tries to describe the internal working of the dynamical system by means of a set of linear first-order differential/difference equations, which can be presented in matrix form, that together with the fact that inputs, outputs and state variables are presented in vector form allows to reach a more compact representation as it can be seen if applied to the previous set of equations (eqns. 7.31 and 7.34 for continuous-time and discrete-time respectively):

$$\frac{d}{dt}\underline{x}(t) = A(t)\underline{x}(t) + B(t)\underline{u}(t) \quad \text{or} \quad \underline{x}(k+1) = A(k)\underline{x}(k) + B(k)\underline{u}(k) \quad (7.37)$$

There are several concepts which are of paramount importance for better understanding this representation that are going to be introduced hereafter.

The “state” of a dynamical system is the minimum set of variables that completely characterize the current behaviour of the system for the purpose of predicting its future evolution. Thus, the “state space” is the set where the state takes value. This set is generally a vector space, which leads to another important concept, the “state vector” ($\underline{x}(t)$ or $\underline{x}(k)$). This is the form in which the internal state is represented that groups all the state variables ($x_j(t)$ or $x_j(k)$). The vector form simplifies the modelling and analysis of the system offering a convenient and compact representation as it is shown in eqn. 7.37. Finally, the components of this a vector are the “state variables”. These variables should be the smallest possible subset of system variables capable of representing the entire state of the system at any given time. The minimum number of state variables is defined by the order of the system's differential/difference equations or the order of the denominator of its transfer function. The state variables are independent, so no one can be a linear combination of the others.

It is important to note that the definition of the state of a dynamical system is non-unique, so there can be different possible representations of its internal description, which leads to different choices of state variables, although it must keep the same input-output relation. These other representations are obtained by applying linear transformations to the initial one.

7.4.4.1 Multivariate time series

The problem of estimating the “reference” trajectory that an aircraft has followed within the airport begins with the analysis of multivariate time series data $\{Z_1, Z_2, \dots, Z_N\}$, which are provided by surveillance sensors while observing the aircraft moving around the airport. Hence, each sensor provides a set of observations from each target, which are measured at successive time instants, normally, at a nearly constant rate related to their illumination period.¹⁷⁸

This set of observations or measurements is a stochastic process where each of the observations (Z_k) at every time interval¹⁷⁹ can be modelled as a random variable with its

¹⁷⁸ Illumination period or scan period or processing period, as currently, new sensors such as MLAT or ADS-B, which are not rotating ones, do not have a constant illumination period due to their way of working. In these cases, it is assumed that the sensor has a processing period when all the measurements computed/received from target are processed to provide a kind of combined measurement.

¹⁷⁹ There are also continuous-time versions, but they are not of interest for this work.

own probability distribution related to the measurement process followed by the sensor providing them. It is relevant to note that this set of observations generally measures the value of the output variables $\{Y_1, Y_2, \dots, Y_N\}$ of the dynamic system, which are directly related to the internal state of the system $\{X_1, X_2, \dots, X_N\}$, as it has been outlined in section 7.4.3.

As time series data, they show a natural temporal ordering. This fact can be exploited to improve their analysis, as observations close together in time will have more in common than observation further apart. Additionally, taking into consideration the forward direction of time, current values can be defined as a consequence of past values but not of future ones.

7.4.4.2 Markov processes

One of the simplest approaches for describing the behaviour of a dynamic system is assuming that it can be modelled as a first-order Markov process, that is, it complies with the Markov property that, roughly speaking, says that given the present state, the future states and the past states are independent. Using a graphical representation:¹⁸⁰

Figure 7.5: First-order Markov process. Source: own elaboration.

The Markov property can be summarized by the following mathematical expression:

$$P(X_{k+1}|X_k, X_{k-1}, \dots, X_1) = P(X_{k+1}|X_k) \quad (7.38)$$

Which states that the conditional probability of the future state of the system just depends on its current value and not on the previous system states.

The representation on the figure above shows that the joint probability of the sequence of states can be factored in the following way:

$$P(X_{1:T}) = P(X_1) \prod_{k=2}^T P(X_k|X_{k-1}) \quad (7.39)$$

Where the notation $X_{1:T}$ for referring to $\{X_1, X_2, \dots, X_T\}$ has been used.

The state can belong to a discrete (finite or countable) state-space or to an infinite one. When the state is considered to change in discrete-time instants (k) and its value is restricted to a set of predefined values, the model is usually called a Markov chain. The set of all the possible states and the transition probabilities between them completely define a Markov chain. When the changes in the state occur at any time instant, the model is called a “Semi-Markov process” [Nunn 77].

There are several ways of extending the Markov models, one of them is taking into consideration that higher-order interactions between states may exist. So, the state in k

¹⁸⁰ Also called Bayesian network, belief network, probabilistic graphical model, or probabilistic independence network.

can be influenced by the “ n ” states that precede it (i.e. from $k - 1$ to $k - n$). In this case, the process is said to follow a “ n^{th} -order Markov model”.

7.4.4.3 Hidden Markov model

The hidden Markov model (HMM) represents another extension of the Markov model that consists on incorporating a further assumption which is that the states X_k are not directly observable (in some way they are hidden) and so, they have to be estimated by means of observations of the behaviour of the system Y_k , which are a consequence of the hidden state (i.e. X_k). These states can evolve with time.

Additionally, the model assumes that the sequence of states $\{X_1, X_2, \dots, X_T\}$ is a Markov process, so it satisfies the Markov property. Consequently, as the evolution of the state is hidden from the observer, it has to be estimated through the sequence of observations $\{Y_1, Y_2, \dots, Y_N\}$, which also satisfy the Markov property with respect to the states. Therefore, given X_k , Y_k is independent from all the previous states and observations. Both properties are mathematically expressed as:

$$P(X_{k+1}|X_k, X_{k-1}, \dots, X_1) = P(X_{k+1}|X_k) \quad (7.40)$$

$$P(Y_k|X_{k-1}, X_{k-2}, \dots, X_1, Y_{k-1}, Y_{k-2}, \dots, Y_1) = P(Y_k|X_k) \quad (7.41)$$

As said in [Ghahramani 01]: "A hidden Markov model is a tool for representing probability distributions over sequences of observations". This fact allows to present the joint distribution of a sequence of states and observations factored in the following expression:

$$P(X_{1:T}, Y_{1:T}) = P(X_1)P(Y_1|X_1) \prod_{k=2}^T P(X_k|X_{k-1})P(Y_k|X_k) \quad (7.42)$$

Which makes use of the same notation as in the previous section.

This expression can be represented in graphical form as:

Figure 7.6: Hidden Markov process. Source: own elaboration.

One additional assumption of the model is that the hidden variable is discrete. In this case, the observations represent the observable part of the model. This model is completely defined by:

- $X_k \in \{V_1, V_2, \dots, V_N\}$: set of values that the hidden state can take on.
- π : probability distribution over the initial state $P(X_1)$ that is for $k = 1$.

- A : $N \times N$ state transition matrix whose elements ($a_{i,j}$) represent the probability of transitioning from state i to state j , that is, $P(X_{k+1} = j | X_k = i)$.
- The output model, which defines $P(Y_k | X_k)$, when the observations are continuous or a $N \times L$ observation (or emission) matrix when they are discrete (L possible values).

As it is presented in [Rabiner 86], there are three main problems to be solved when using HMMs, namely:

- The computation of the probability of an observation sequence.
- The estimation of the hidden state sequence (inference).
- The estimation of the parameters of the model (learning).

Additionally, the more suitable algorithms to solve them are also presented in [Rabiner 86], which are the forward-backward procedure, the Viterbi algorithm, and the Baum-Welch method, respectively.

Although this model might be convenient for representing the aircraft movement model, as it matches with the definition of a “hidden” state as it takes discrete values (e.g. uniform motion, constant acceleration, turning, etc.) that may be inferred through the position measurements provided by the sensors. It lacks a way for also integrating the continuous states, such as the position, velocity, acceleration, etc. of the aircraft.

7.4.4.4 Linear-Gaussian state-space models

The linear-Gaussian state-space model is the continuous-state version of the HMM. In this model, X_k is a continuous “hidden” state variable or more generally a L -dimensional state vector from which M -dimensional real-valued observation vectors $\{Y_1, Y_2, \dots, Y_T\}$ are obtained. Like the HMM, the linear-Gaussian state-space model also complies with the Markov property for both, the state and the observations, which can be mathematically stated as:

$$P(X_{k+1} | X_k, X_{k-1}, \dots, X_1) = P(X_{k+1} | X_k) \quad (7.43)$$

$$P(Y_k | X_{k-1}, X_{k-2}, \dots, X_1, Y_{k-1}, Y_{k-2}, \dots, Y_1) = P(Y_k | X_k) \quad (7.44)$$

This model takes its name from three defining features, namely:

1. It assumes that the evolution of the state is linear, so it can be presented in matrix form.
2. It introduces the uncertainty in either the state evolution (system dynamics) or the state observation (sensor measurement model) assuming a Gaussian distribution in both cases.
3. It is based on the state-space representation of dynamic systems.

Taking into account the aforementioned features, X_k is not constrained to a limited number of states, but determined by a system model such as the one shown in eqns. 7.31

or 7.34, which includes a stochastic component (w_k) that introduces the uncertainty within the model:

$$X_{k+1} = A_k X_k + B_k U_k + w_k \quad (7.45)$$

Where:

- A_k : state transition matrix.
- B_k : input matrix.
- U_k : input observation vector.
- w_k : plant or process noise, which is restricted to be a zero-mean Gaussian white noise with covariance matrix Q .

On the other hand, the observation model, which is the one that relates the current state with the current observation, takes the following form:

$$Y_k = C_k X_k + D_k U_k + v_k \quad (7.46)$$

Where:

- C_k : projection matrix.
- D_k : input matrix.
- v_k : measurement or observation noise, which is restricted to be a zero-mean Gaussian white noise with covariance matrix R .

These equations refer to a controlled stochastic linear system with additive noise. However, a more general model will be presented in section 7.4.4.7 to reflect the impact of the assumptions concerning the system that are going to be made. Like the previous model, the joint probability of states and observations has the same appearance:

$$P(X_{1:T}, Y_{1:T}) = P(X_1)P(Y_1|X_1) \prod_{k=2}^T P(X_k|X_{k-1})P(Y_k|X_k) \quad (7.47)$$

And, the same graphical representation:

Figure 7.7: Linear-Gaussian state-space model. Source: own elaboration.

This model is completely defined by:¹⁸¹

- $X_k = \{X_k^{(1)}, X_k^{(2)}, \dots, X_k^{(L)}\}$: set of state variables that define the state. The superscript means the vector component where the state variable is located.
- A : $L \times L$ state transition matrix which defines the evolution of the system dynamics based on the corresponding motion or system model from the current state X_k to state X_{k+1} .
- C : $L \times M$ projection transition matrix which translates the states into the space of observations or measurement space.
- $(\underline{w}_k, Q)$: stochastic characteristics of the process noise; that is, the mean vector and its covariance matrix.
- $(\underline{v}_k, R)$: stochastic characteristics of the observation noise; that is, the mean vector and its covariance matrix.
- (X_1, V) : the initial distribution is assumed as a Gaussian random vector with mean X_1 and covariance V , which is uncorrelated with the noise variables.

Like the HMM, there are two main problems to be solved when using Linear-Gaussian state-space models, namely:

- The estimation of the hidden state sequence (inference).
- The estimation of the parameters of the model (learning).

Focussing on the first one, it is known that the optimal solution to this problem is provided by the Kalman filter [Kalman 60], which is presented in ANNEX I section A.4. This model can be used for vehicles that do not present changes in their movement model, that is, vehicles without any discrete state. However, this is not the general case for airport vehicles as they change their movement model from one to another all along their movement through the airport. Consequently, this model does not yet fully describe the behaviour of the targets.

7.4.4.5 Markovian jump linear system

This kind of model is an evolution of the linear-Gaussian state-space model, as it adds the possibility of taking into consideration several system models or dynamical models for the system, which can change along time. Another way of describing this fact is assuming that the system changes the parameters of its dynamical model based on a discrete mode. This is a good model for those systems that operates in multiple modes as it is the case of manoeuvring targets¹⁸² at an airport.

There are two distinguishing features on this kind of systems which are outlined in the name of the model. On one hand, the different dynamical models that are considered are linear, and on the other hand, the evolution of the discrete mode is governed by a

¹⁸¹ For the sake of clarity, the input signal is not included, i.e. the system is considered autonomous.

¹⁸² Aircraft or airport vehicles (tows, follow me, buses, catering vehicles, refuelers, etc.).

stochastic process that is described as a Markov process, which is statistically independent from the state variables.

The systems that comply with this model belong to a larger class known as hybrid systems, which have been introduced in section 7.4.2.3.

In this case, the system model has an additional dependency with the discrete mode r , which drives the linear dynamical model that is used at each time step k . This fact is shown in the following equations:

$$X_{k+1} = A_{r_k} X_k + B_{r_k} U_k + w_k \quad (7.48)$$

$$Y_k = C_{r_k} X_k + D_{r_k} U_k + v_k \quad (7.49)$$

Where:

- r_k : discrete mode at time k .

The joint probability of states and observations, now must also include the effect of the different modes, so now the appearance changes to:

$$P(X_{1:T}, Y_{1:T}, r_{1:T}) = \left(P(r_1) \prod_{i=2}^T P(r_i | r_{i-1}) \right) \cdot \left(\prod_{l=1}^L P(X_1^{(l)}) \prod_{j=2}^T P(X_j^{(l)} | X_{j-1}^{(l)}) \right) \cdot \prod_{k=1}^T P(Y_k | X_k^{(1)}, \dots, X_k^{(L)}, r_k) \quad (7.50)$$

Graphically, it can be represented by:

Figure 7.8: Markovian jump linear system. Source: own elaboration.

The model is defined when the following parameters are available.

- $r_k \in \{M_1, M_2, \dots, M_m\}$: discrete mode values at time k .
- $\pi_{j,k}$: probability of the discrete mode of taking a value of M_j at time k .

- $\{A_{i,k}, B_{i,k}, C_{i,k}, D_{i,k}\}$: parameters of the i -th model at time k .

Like in the previous model, the focus in this case is also on the inference of the state of the system.

7.4.4.6 Switching State-Space model

This model is a generalization of the HMM and the SSM in which the different models that the system can follow transition in a discrete manner from one linear motion model to another. However, the models themselves are not limited to being linear as it is possible to find in the literature nonlinear switching SSM.

In most of the cases, using a Markovian Jump Linear system is enough, and only in those cases where there might be a big improvement, nonlinear models may be taken into consideration. To cover this situation, the next section introduces a general expression for those systems that do not meet the restrictions of the previous models.

7.4.4.7 General State-Space models

The previous sections constrain the behaviour of the system, the way in which system models switch between them, and the observation model by means of assumptions that are sometimes difficult to find in real-life statistical signal processing problems. These constraints are now going to be relaxed.

Therefore, in order not to restrict the model and trying to avoid approximation methods, a more general expression of a SSM for modelling time series data is going to be presented assuming a multitarget-multisensor environment, that is, an environment where there are many targets that are detected by more than one sensor. Therefore, the system model and the observation model must reflect this matter, as it can be seen in their general expression presented hereafter:

$$X_{k+1} = \varphi_i(k, X_k, U_k, w_k) \quad (7.51)$$

$$Y_k = \phi_j(k, X_k, U_k, v_k) \quad (7.52)$$

Where:

- φ_i : nonlinear state transition function corresponding to the i -th system model.
- ϕ_j : nonlinear projection function corresponding to the j -th sensor.
- w_k and v_k : are independent and identically distributed noise processes.

These expressions (conveniently tuned) allow to cope with those systems whose nonlinear motion models are affected by not just an additive Gaussian noise but by other more general error types. Furthermore, the observation model is also unconstrained.

This model will be the base for presenting some state estimation algorithms such as the Extended Kalman Filter (EKF), the Unscented Kalman Filter (UKF) and the Particle Filter (PF). The expressions of the first two filters can be found in ANNEX I sections A.5 and A.6, respectively.

7.4.5 Stochastic hybrid system models

As its own name reflects, one important characteristic of these models is the generalization of the concept of hybrid systems to go beyond continuous and discrete deterministic dynamics by means of taking into consideration probabilistic phenomena. This fact allows a more realistic modelling of some system behaviours.

In the initial formulation of hybrid systems [Branicky 96][Lygeros 03], there was no space for randomness, although it is in the essence of any system.

For some initial applications, this fact was acceptable, but for the case under consideration in this thesis, taking into consideration randomness or uncertainty is a must. In surface surveillance, stochastic features are very important as the observations provided by sensors are affected by uncertainty and, even more, the movement models are by far not exact.

Thus, the uncertainty has been introduced following different approaches:

- Perturbating the evolution of the system mode (i.e. r_k), which is a discrete state, while preserving a deterministic behaviour for the continuous states (i.e. X_k) of the system.
- Replacing the deterministic behaviour of the continuous states (i.e. X_k) of the system determined by ordinary differential/difference equations by its stochastic counterpart, keeping a deterministic evolution of the discrete state (i.e. r_k).
- Resetting the continuous states (i.e. X_k) after a change in the system mode (i.e. r_k) is another approach. The new continuous state would take a stochastic value. This fact can appear merged with the previous approaches.

Taking into consideration these three approaches, there have appeared different models. Each one with its own particularities. An overview of the ones that can be found in the literature is developed in [Pola 03]. Systems such as bouncing balls, automotive control systems, highway systems, digital controllers, etc. and why not aircraft and ground vehicles can be modelled this way. The models that are more relevant to this thesis are presented hereafter.

7.4.5.1 Piecewise deterministic Markov processes

Piecewise deterministic Markov processes (PDMP) [Davis 93] are a particular case of stochastic hybrid system in which randomness appears in the transition between its discrete states. Such transitions can happen when the state reaches the boundary of the state space, or randomly, following a certain stochastic process.¹⁸³ Between transitions, the evolution of the continuous state is determined by, in general, nonlinear ordinary differential equations. After each transition, the continuous state is reset following a probability distribution which is independent of the continuous state.

¹⁸³ This stochastic process is usually modeled as a Poisson one.

7.4.5.2 Switched diffusion processes

This is another model associated to the general concept “stochastic hybrid system”. The term “switching diffusion processes” (SDP) [Ghosh 97] describes systems whose continuous state is represented by a stochastic differential equation (SDE). However, its discrete state evolution is modelled by a continuous-time Markov process and there are no jumps in the continuous state after each mode transition.

7.4.5.3 Stochastic hybrid systems

The concept “stochastic hybrid system” (SHS) groups systems with very different features which share that their state has discrete and continuous variables and that randomness appears somewhere in their model.

The issue with this term is that there are different definitions of SHS in the literature. Thus, [Hu 00] provided a formal model for SHS in which stochastic difference equations substitute deterministic ones and reset maps were generalized to probabilistic distributions, which described the random nature after a change in the discrete state. However, discrete state transitions were triggered by deterministic events.¹⁸⁴ On the other hand, [Hespanha 04] presents a definition of SHS in which transitions between discrete states occur based on stochastic events that happen with a probability that could be related to both discrete and continuous components of the hybrid state. This approach is similar to a PDMP except for the fact that the transition between modes can depend also on the continuous state.

Additionally, a straightforward definition of SHS can be found in [Cassandras 07] which says that a SHS is a "Hybrid system with stochastic continuous dynamics governed by stochastic differential equations and with random discrete mode transitions governed by Markov chains". This definition is perfectly suitable to the topic of this thesis as the movement of aircraft and airport vehicles can be described as a succession of different movement models (modes) that change along time and whose dynamic model description includes some type of uncertainty.

This model is also used in Air Traffic Control applications such as aircraft tracking in TMA [Seah 09a] areas because aircraft follow similar take-off and landing patterns, which can be exploited by state estimators based on these models. Some estimation algorithms for aircraft tracking are also presented in [Seah 09a], which proposes a state-dependent mode transition matrix in their SDTHE algorithm.

7.4.5.4 Stochastic linear hybrid systems

Stochastic linear hybrid systems (SLHS) [Balakrishnan 04][Seah 09] represent a subgroup of SHS in which the continuous state evolution is described by a set of stochastic linear difference equations and transitions between modes are governed by stochastic guards which are a set of polyhedral partitions of a real vector space. This vector space is defined by [Seah 09] as the union of the continuous state space and a random parameter space which helps defining the stochastic guards. These guards introduce the uncertainty in the system mode, which is different from the one found in the continuous state. [Seah 09] also

¹⁸⁴ These events take the form of deterministic conditions or guards.

lists other areas where this model has been used such as: vehicle tracking and remote sensing, transportation system modelling, communication networks, etc.

The problem of estimating the discrete and continuous state of a SLHS in applications to aircraft tracking is presented in [Hwang 06], where only the case with continuous system output data is considered, and a modified IMM that uses the mean of the residual to increase the difference between the likelihood of the correct mode and those of the other modes is proposed (i.e. RMIMM).

7.4.5.5 Constrained stochastic linear hybrid systems

This is a further refinement of SLHS where apart from taking into consideration guard conditions and stochastic linear difference equations, this concept also exploits different constraints which are imposed by the environment to the systems whose behaviour is to be modelled. Thus, in [Mann 11] it can be found that "Constrained stochastic linear hybrid systems (CSLHS) are stochastic linear hybrid systems that have state equality constraints on the dynamics of each mode".

Like in the previous model, a CSLHS propose a discrete-time model for a SHS with some particularities, which are gathered by the equality constraints. This way, this model shares with the latter that the evolution of the continuous state is described by stochastic difference equations and that the system mode evolution is governed by stochastic guards (or conditions that contain random parameters).

This model describes quite well the aircraft ground movements as they are constraint by airport taxiways and runways. In fact, in [Mann 11] and [Mann 12] an application of this model is proposed for taxiway conformance monitoring where several alternatives are compared and a modelled-fault constrained innovation hybrid estimator conformance-monitoring algorithm (MFCIHE) is proposed. Furthermore, the CSLHS model has also been used for state estimation and fault detection and identification applied to the management of a water tank in [Mann 13].

7.4.6 Movement models

The target of this section is to present the models that describe the aircraft movements on the surface of an airport. Most tracking/smoothing algorithms rely on models of the target's dynamic behaviour for filtering out random errors as it is the case for the algorithms that are implemented in this thesis. This is the reason why it is mandatory to describe the ground movement models that are utilized in this work.

For the purpose of this thesis, aircraft are autonomous systems in the sense that, from the point of view of the surveillance system, there is no control over them, so a priori no input signal is considered while processing the data to be analysed.

The definition of the model also includes the different ground motion modes or ground movement models that can be performed by the aircraft. The definition of such modes assumes that aircraft are real physical systems, so known physical laws apply (uniform motion, constant acceleration, constant turn rate, stopped, etc.) except when occasionally perturbed by an external force.

In the next subsections, the general terms used for modelling the behaviour of aircraft and airport vehicles are going to be presented and some typical aircraft motion models and their mathematical equations will be shown.

7.4.6.1 Aircraft trajectory reconstruction particularities

In the framework of this thesis, where one of the main topics is aircraft ground trajectory reconstruction, there are some particularities that are worth taking into account as they restrict the working framework allowing certain approximations but without severely affecting the quality of the estimations.

Some of the particularities or assumptions that can be outlined are:

1. Aircraft are modelled as hybrid systems.
2. Aircraft can be physically modelled based on their aircraft type.
3. Aircraft kinematics and parameters are periodically sampled by noisy surveillance sensors.
4. Different sensors may have different aircraft reference points.
5. Aircraft are manoeuvring targets.
6. Aircraft movement models can be linear or nonlinear.
7. The airport environment is a multitarget-multisensor environment.
8. The airport layout constrains the aircraft movements.

Going back to the first two points, aircraft are systems that are composed of many parts which interact among them to achieve a certain common goal, usually moving both cargo and passengers from one point to another. There are different ways of modelling an aircraft depending on the target of the analysis where the model is going to be used. One option is to simplify it to a point target,¹⁸⁵ which heavily impose constraints to the model. Another possibility is to model it as a physical system that moves following several movement patterns, with certain dimensions (length, width, and height), shape, colour, and other parameters of interest such as identifiers, equipage or aerodynamic characteristics. In this thesis, the focus is placed on aircraft kinematics, where the aircraft can be modelled as a hybrid system. This is so because among the state variables which define the aircraft motion, there are continuous ones, such as position, velocity and acceleration,¹⁸⁶ but also discrete ones such as the movement model (e.g. stopped, constant velocity, constant turn rate, etc.). This fact, together with some physical features, will be considered when designing the aircraft state estimators for trajectory reconstruction.

Concerning the third point, aircraft are seen by sensors that provide periodical noisy measurements of their position while moving around the airport. This is reflected in a discrete-time observation model. However, the trajectory that they follow is better modelled with a continuous-time system state model, but there is a clear preference to

¹⁸⁵ This is the case when the aircraft is modelled in an en-route environment for tracking purposes.

¹⁸⁶ But not just, also the speed or the turn rate or angular acceleration are other possible state variables. As it was said, the system state is not unique.

process this information by discrete-time state estimation algorithms, such as the Kalman filter (ref. ANNEX I section A.4). Consequently, it is more suitable to represent the time evolution of the system state by means of difference equations rather than using differential ones. This fact explains why it is preferred to use the time index k instead of t all along the document.

It is fundamental to mention that not all the sensors measure the same aircraft position as they have different aircraft reference points. For example, the ADS-B position, unless corrected, provides the position of the GNSS antenna, not the centroid of the aircraft as it is expected from an SMR, or similarly the MLAT measures the position of the ATC antenna (ref. 2.3.2.2 and 2.4.2.2). This information must be considered, if known, or estimated otherwise, to refer all the position information to the same reference framework. This correction is carried out by the “bias estimation and cancellation” process as it impacts the uncertainty of the corrected measurement that finally reaches the “trajectory reconstruction” processing block.

Another point to consider is that aircraft are manoeuvring targets, that is, their dynamical behaviour changes along time among several possible motion models, which is something implicit in their definition as hybrid systems. One of the most common approaches to deal with this issue is to model aircraft as systems with Markovian switching coefficients. This is the case of Markovian jump linear systems presented in section 7.4.4.5. Thus, the aircraft is then considered as a multiple-model system whose dynamical behaviour is therefore characterized by a set of motion models $r_k \in \{1, 2, \dots, m\}$, which have a given probability of switching between them at every observation or scan period $p_{i,j} = P(r_k = j | r_{k-1} = i)$. Hence, the evolution of the motion models is assumed to be governed by a finite-state Markov chain. These motion models are defined based on the regular surface movements which, as already mentioned before, go from stopped, to accelerated movements, passing through constant velocity or constant turn rates. These movement models are assumed to be in effect during the scan period ending at k , which goes from $(k - 1, k]$.

The sixth point introduces the fact that there might be situations in which the motion models are linear but there are others in which they are clearly nonlinear, which introduces much more complexity to the process. The first situation is the preferred one, however there are mitigation techniques in the case that there is a need for including a clearly nonlinear motion model. Several solutions exist for the latter, which go from approximating the model to a linear one, assuming a loss of precision on the estimation, to solutions in which the model is taken as it is and some additional computations are performed.

The airport environment is a perfect example of a multitarget-multisensor environment, as it concentrates many mobiles, not just aircraft, and usually count on more than one sensor for managing its surface. Among these sensors, there can be radar systems (SMR, PSR, SSR, PRM, Mode-S, ASDE, miniradars, etc.), MLAT systems, ADS-B stations, video camera systems, magnetic loops, etc. Such a variety of sensors is taken into consideration when selecting the corresponding observation/measurement model. Each type of sensor has its own measurement model, whose parameters are particularized per sensor as not all the sensors of the same type show the same performances (biases and noise statistics). This parameter tuning activity is of great importance not only for tracking systems but also for trajectory reconstruction algorithms.

Finally, the airport imposes some constraints to the movement of the aircraft and vehicles. Unlike the en-route domain, which offers more flexibility to aircraft, the airport reduces the aircraft movements to the one allowed by the airport layout¹⁸⁷ and its current configuration (e.g. active runways, taxiway directions and limited speeds). This fact can be exploited to improve the targets state estimation.

7.4.6.2 Process and observation models

The movement models that are assumed within this thesis can be generally grouped into two sets: linear and non-linear models. The general formulation of each set is going to be presented together with their assumptions. The details of the ones employed in this work are provided in section 7.4.6.3.

For each of the sets, both the system model and the observation model are presented. Note that, in the literature, these models are also known as process model and measurement model, respectively.

7.4.6.2.1 Linear movement models

This set can model most of the movement models of aircraft around the airport. The general aspect of the process model in discrete-time filtering notation is:

$$x(k+1) = Fx(k) + Gw(k) \quad (7.53)$$

Where:

- F : state transition matrix, which extrapolates the state.
- G : noise projection matrix, which projects the noise into the state space.
- $w(k)$: process noise sequence with covariance matrix Q .

On the other hand, the observation model, which is the one that relates the current state with the current observation, takes the following form:

$$z(k) = Hx(k) + v(k) \quad (7.54)$$

Where:

- H : projection matrix, which projects the state into the measurement space.
- $v(k)$: observation noise sequence with covariance matrix R .

Assumptions:

- In a multisensor environment, there will be an observation model per each sensor, which can be identified by subindex j , (i.e. H_j and v_j). This is not reflected in order to avoid cluttering the formulae.
- No external input, i.e. $u(k)$, is expected.

¹⁸⁷ Also referred in other parts of the document as airport map.

- $w(k)$ and $v(k)$ are independent zero-mean white-noise sequences with the following statistics:

$$E[w(k)] = 0 \text{ and } E[v(k)] = 0 \quad (7.55)$$

$$E[w(k)w(j)^T] = Q \delta(k - j) \text{ and } E[v(k)v(j)^T] = R \delta(k - j) \quad (7.56)$$

$$E[v(k)w(j)^T] = 0 \quad \forall k, j \quad (7.57)$$

Where:

- $\delta(n)$: Kronecker delta function.
- $w(k)$ and $v(k)$ are assumed to show a Gaussian distribution with the following characteristic:

$$w(k) \sim N(0, Q) \text{ and } v(k) \sim N(0, R) \quad (7.58)$$

- The initial state $x(0)$ is independent of both, the process noise and the measurement noise and it is assumed as being Gaussian distributed:

$$x(0) \sim N(x_0, P_0) \quad (7.59)$$

Where:

- x_0 : initial mean value for $x(0)$.
- P_0 : covariance matrix of $x(0)$.

Some examples of linear motion models are:

- Stationary or stopped (ref. 7.4.6.3.1).
- Constant or nearly constant velocity (ref. 7.4.6.3.2).
- Constant or nearly constant acceleration (ref. 7.4.6.3.3).
- Right/Left turn at a known turn rate (ref. 7.4.6.3.4).

Summarizing, these equations refer to an autonomous stochastic linear system with additive zero-mean white Gaussian noise.

7.4.6.2.2 Nonlinear movement models

These models cover all nonlinear ones that are not correctly represented by a linearized version of them. Thus, the process model is described as:

$$x(k + 1) = f(x(k)) + w(k) \quad (7.60)$$

Where:

- $f(x)$: nonlinear state transition function.
- $w(k)$: independent and identically distributed process noise sequence.

The measurement model takes the following form:

$$z(k) = h(x(k)) + v(k) \quad (7.61)$$

Where:

- $h(x)$: nonlinear projection function.
- $v(k)$: independent and identically distributed measurement noise sequence.

Assumptions:

- No external input, i.e. $u(k)$, is expected.
- $w(k)$ and $v(k)$ are independent noise sequences.
- $w(k)$ and $v(k)$ show their own distribution.
- The initial state $x(0)$ is independent of both, the process noise and the measurement noise. Nothing is assumed about its distribution.

Some examples of nonlinear motion models are:

- Turn rate estimation (ref. 7.4.6.3.5).
- Three-wheel model (ref. 7.4.6.3.6).

Summing up, these equations refer to a stochastic nonlinear system with additive noise. However, the assumptions for the noises of linear systems are usually accepted for these systems as it happens for the ones just listed above.

7.4.6.3 Aircraft ground movement models

Although there is a great variety of ground movement models ranging from simple ones, just considering kinematics and a point target, to very complex ones in which forces are taken into consideration in extended targets [Coetzee 06][Coetzee 11]. In this section, the most common dynamics that can be followed by aircraft and airport vehicles are presented as they will be exploited by the state estimation algorithms that will be presented at the end of this chapter. These dynamics are the ones corresponding to the most common ground movements models that can be found while observing the traffic that moves around an airport.

Moreover, any surface trajectory can be roughly approximated by a succession of segments of constant velocity¹⁸⁸ that are then followed by either constant acceleration segments, or turns,¹⁸⁹ or standing periods. With these “basic” pieces, which are linked to a motion mode or a movement model, a simplified ground trajectory can be built. Each motion mode or movement model is going to be presented in high level in the following subsections. One important aspect to bear in mind is that all these basic pieces are kinematic tracking models as they do not assume any control over the control inputs of

¹⁸⁸ These segments are also called uniform motion or non-maneuvring segments.

¹⁸⁹ These segments are referred to, in general, as manoeuvre segments or manoeuvring segments.

the aircraft (i.e. pilot acting over the engine thrust or over the steering wheel), which are then modelled stochastically.

Thus, the process noise or plant noise is chosen to consider the expected manoeuvrability of the aircraft when following a certain model. The aim is to minimize the effect of an unexpected manoeuvre due to an unknown acceleration, which can be caused by any human control input, an automatic guidance system or any weather effect such as wind. There are two main approaches to address the uncertainty concerning these unexpected manoeuvres. Both assume that the start is unknown. One assumes that the manoeuvre can be modelled as a random process and the other assumes that it is not random, but it can be detected and/or estimated when happening. The former approach can be further divided into two alternatives depending on whether the statistical properties of the random process can be considered a white noise or a correlated noise.

Examples of the random process approach, which assumes a white noise, are the nearly constant velocity model (section 7.4.6.3.2) or the nearly constant acceleration model (section 7.4.6.3.3) in either of its versions, that is coming from continuous-time white noise or assuming a piecewise constant white noise. On the other hand, the Singer's model [Singer 70] is the leading exponent of the correlated noise random process approach. Other examples are the Helferty's model [Helferty 94] or Bar-Shalom and Fortmann model [Bar-Shalom 88]. Finally, the non-random approach takes advantage of manoeuvre detection algorithms to either change the parameters of the model, or correct or augment the state estimate in order to better absorb the detected manoeuvre.

In general, the chosen basic models assume a 2D representation of the aircraft unless the taking off or the landing phases are taken into consideration. In the latter, the altitude or the height should be included.

There is a lot of information concerning dynamic models, in [Li 03] there is a complete survey on dynamic models which goes from the one presented in this thesis to more complex ones. Additionally, a derivation of the expressions, which are going to be presented in the following sections, is obtained by solving the initial continuous differential equations presented in section 7.4.3. This derivation can follow two approaches: the discrete-time equivalent or the direct discrete-time counterpart. For details see [Li 03], footnote 1.

7.4.6.3.1 Stationary or stopped

This movement model is the one which describes those periods of time in which the modelled system (i.e. aircraft or ground vehicle) is stopped at any position. This means that the position state variables do not evolve while any velocity state variable is zero. The system state and measurement equations are:

$$\underline{x}(k+1) = F\underline{x}(k) + G\underline{w}(k) \quad (7.62)$$

$$\underline{z}(k) = H\underline{x}(k) + \underline{v}(k) \quad (7.63)$$

Where:

- $\underline{x}(k)$: state vector, which has the following general layout:

$$\underline{x}(k) = [x \quad V_x \quad y \quad V_y]^T \quad (7.64)$$

However, this layout may lead to bad conditioned matrices, which might cause the appearance of instabilities in the computations. For avoiding these situations, the state vector might be reduced to the following expression:

$$\underline{x}(k) = [x \quad y \quad \theta]^T \quad (7.65)$$

- F : state transition matrix for the stopped movement model, which looks like:

$$F = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix} \quad \text{or} \quad F = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \quad (7.66)$$

The second expression is the one to be used when instability appears in the calculations. It is matched to the second expression of the state vector, that is, the one with just position and heading state variables (eqn. 7.65).

- $\underline{z}(k)$: measurement vector, which is given by:

$$\underline{z}(k) = [x \quad y]^T \quad (7.67)$$

- H : projection matrix in charge of translating the state into the measurement space. Depending on the state vector layout and assuming that the measurement just provides position information, it will look like as:

$$H = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \quad \text{or} \quad H = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \end{bmatrix} \quad (7.68)$$

- $\underline{w}(k)$ and $\underline{v}(k)$: independent and identically distributed noise sequences.

It is relevant to mention that this model presents some issues when used with other models in a hybrid state estimator, such as an IMM, a MMAE or a GPB algorithm unless the dimension of the state vector is adjusted conveniently.

7.4.6.3.2 Constant or nearly constant velocity

This movement model is known as “white-noise acceleration model” [Li 03] and it is a second-order model¹⁹⁰ that describes those periods of time in which aircraft move at a constant speed and follow a constant course (straight line). The uncertainty in this model is introduced by a zero-mean white noise acceleration. Depending on the approach, the nearly constant velocity (CV) model assumes that this acceleration is piecewise constant between the sampling instants, or that it changes right before the sampling instant. The associated system state and measurement equations, for the first approach, are:

$$\underline{x}(k+1) = F\underline{x}(k) + G\underline{w}(k) \quad (7.69)$$

$$\underline{z}(k) = H\underline{x}(k) + \underline{v}(k) \quad (7.70)$$

Where:

¹⁹⁰ A second-order model incorporates position and velocity components.

- $\underline{x}(k)$: state vector, which has the following general layout:

$$\underline{x}(k) = [x \quad V_x \quad y \quad V_y]^T \quad (7.71)$$

- F : state transition matrix for the CV movement model, which is given by:

$$F = \begin{bmatrix} 1 & T & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & T \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (7.72)$$

- G : noise projection matrix into the state space, which is given by:

$$G = \begin{bmatrix} T^2/2 & 0 \\ T & 0 \\ 0 & T^2/2 \\ 0 & T \end{bmatrix} \quad (7.73)$$

- $\underline{w}(k)$: white Gaussian process noise sequence that is characterized by:

$$\underline{w}(k) = [w_x(k) \quad w_y(k)]^T \xrightarrow{\forall i,j} E[\underline{w}(i) \underline{w}^T(j)] = Q \delta(i - j) \quad (7.74)$$

Where Q is its covariance matrix that is given by:

$$Q = G \begin{bmatrix} \sigma_{CV}^2 & 0 \\ 0 & \sigma_{CV}^2 \end{bmatrix} G^T = \sigma_{CV}^2 \begin{bmatrix} T^4/4 & T^3/2 & 0 & 0 \\ T^3/2 & T^2 & 0 & 0 \\ 0 & 0 & T^4/4 & T^3/2 \\ 0 & 0 & T^3/2 & T^2 \end{bmatrix} \quad (7.75)$$

With:

- $\delta(i - j)$: Kronecker delta function.
- T : sampling period.
- σ_{CV}^2 : variance that characterize the piecewise constant random acceleration of the aircraft in both dimensions, i.e. in x and y , (assuming the same value). The acceleration is independent from period to period. As it will be shown by the end of this section, this is just one possible valid and straightforward approximation, although there are others.
- H : projection matrix in charge of translating the state into the measurement space. Assuming the state vector layout previously shown and considering that the measurement space only includes position, it looks like as:

$$H = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \quad (7.76)$$

- $\underline{v}(k)$: white Gaussian measurement noise sequence that is characterized by:

$$\underline{v}(k) = [v_x(k) \quad v_y(k)]^T \xrightarrow{\forall i,j} E[\underline{v}(i) \underline{v}^T(j)] = R \delta(i - j) \quad (7.77)$$

Where R is its covariance matrix that is given by:

$$R = \begin{bmatrix} \sigma_x^2 & 0 \\ 0 & \sigma_y^2 \end{bmatrix} \quad (7.78)$$

With:

- σ_x : standard deviation of the measurement error in the x dimension.
- σ_y : standard deviation of the measurement error in the y dimension.

If the sensor providing the measurement just delivers position and not velocity.

Additionally, if the second approach is followed, the process noise is modelled as a continuous-time white noise acceleration, which leads to a different covariance matrix that is obtained if the continuous-time system state equation:

$$\frac{\partial}{\partial t} X(t) = AX(t) + Bw(t) \quad (7.79)$$

is integrated all along the sampling period (T). For further details please visit [Bar-Shalom 07] chapter 6. Finally, the result is:

$$Q = E[\underline{w}(i) \underline{w}^T(j)] = S_{CV} \begin{bmatrix} T^3/3 & T^2/2 & 0 & 0 \\ T^2/2 & T & 0 & 0 \\ 0 & 0 & T^3/3 & T^2/2 \\ 0 & 0 & T^2/2 & T \end{bmatrix} \quad (7.80)$$

Note that S_{CV} is the power spectral density, not the variance of the continuous-time white noise process (i.e. $w(t)$), which has been discretised, as in the previous case. It is relevant to remark that the CV model implies that the magnitude of the process noise is “small” enough to consider it a “uniform motion” model. In fact, there are manoeuvring models based on this model that just increase the noise level to allow the absorption of certain manoeuvres and so, being able to follow any target performing them.

7.4.6.3.3 Constant or nearly constant acceleration/deceleration

This movement model is known as “Wiener-process acceleration model” [Li 03]. It is a third-order model¹⁹¹ that describes those periods of time in which aircraft change their speed along a straight segment. As before, there are two versions of the constant or nearly constant acceleration (CA) model: the “Wiener-sequence acceleration model”, and the “white noise jerk model”. The former assumes that the derivative of the acceleration (i.e., the “jerk”) is constant between the sampling instants while, in the latter, the jerk changes just before the sampling instant. The associated system state and measurement equations, for the first version, are given by:

$$\underline{x}(k+1) = F\underline{x}(k) + G\underline{w}(k) \quad (7.81)$$

$$\underline{z}(k) = H\underline{x}(k) + \underline{v}(k) \quad (7.82)$$

¹⁹¹ A third-order model incorporates position, velocity, and acceleration components.

Where:

- $\underline{x}(k)$: state vector, which has the following general layout:

$$\underline{x}(k) = [x \quad V_x \quad a_x \quad y \quad V_y \quad a_y]^T \quad (7.83)$$

- F : state transition matrix for the CA movement model, which is given by:

$$F = \begin{bmatrix} 1 & T & T^2/2 & 0 & 0 & 0 \\ 0 & 1 & T & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & T & T^2/2 \\ 0 & 0 & 0 & 0 & 1 & T \\ 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix} \quad (7.84)$$

- G : noise projection matrix into the state space, which is given by:

$$G = \begin{bmatrix} T^2/2 & 0 \\ T & 0 \\ 1 & 0 \\ 0 & T^2/2 \\ 0 & T \\ 0 & 1 \end{bmatrix} \quad (7.85)$$

- $\underline{w}(k)$: white Gaussian process noise sequence that is characterized by:

$$\underline{w}(k) = [w_x(k) \quad w_y(k)]^T \xrightarrow{\forall i,j} E[\underline{w}(i) \underline{w}^T(j)] = Q \delta(i - j) \quad (7.86)$$

Where Q is its covariance matrix that is given by:

$$Q = G \begin{bmatrix} \sigma_{CA}^2 & 0 \\ 0 & \sigma_{CA}^2 \end{bmatrix} G^T = \sigma_{CA}^2 \begin{bmatrix} T^4/4 & T^3/2 & T^2/2 & 0 & 0 & 0 \\ T^3/2 & T^2 & T & 0 & 0 & 0 \\ T^2/2 & T & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & T^4/4 & T^3/2 & T^2/2 \\ 0 & 0 & 0 & T^3/2 & T^2 & T \\ 0 & 0 & 0 & T^2/2 & T & 1 \end{bmatrix} \quad (7.87)$$

With:

- $\delta(i - j)$: Kronecker delta function.
- T : sampling period.
- σ_{CA}^2 : variance that characterize the random increments in the acceleration of the aircraft in both dimensions (assuming the same value).
- H : projection matrix in charge of translating the state into the measurement space. Assuming the state vector layout previously shown and considering that the measurement space only includes position, it looks like as:

$$H = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \end{bmatrix} \quad (7.88)$$

- $\underline{v}(k)$: white Gaussian measurement noise sequence as defined in section 7.4.6.3.2.

As in the previous model, the process noise can be approximated using a continuous-time white noise assumption, which leads to a different covariance matrix. These values are obtained if the continuous-time system equation (7.79) is integrated all along the sampling period (T) giving the following result [Bar-Shalom 07]:

$$Q = E[\underline{w}(i) \underline{w}^T(j)] = S_{CA} \begin{bmatrix} T^5/20 & T^4/8 & T^3/6 & 0 & 0 & 0 \\ T^4/8 & T^3/3 & T^2/2 & 0 & 0 & 0 \\ T^3/6 & T^2/2 & T & 0 & 0 & 0 \\ 0 & 0 & 0 & T^5/20 & T^4/8 & T^3/6 \\ 0 & 0 & 0 & T^4/8 & T^3/3 & T^2/2 \\ 0 & 0 & 0 & T^3/6 & T^2/2 & T \end{bmatrix} \quad (7.89)$$

This result belongs to the “White-noise jerk model” mentioned above [Bar-Shalom 01]. Like in the previous model, S_{CA} is the power spectral density of $w(t)$, and the noise level should be “small” enough to keep the name “constant or nearly constant acceleration model”. Because bigger values of the process noise make the model good for tracking other manoeuvres apart from the constant acceleration one.

7.4.6.3.4 Right/Left turn at a known turn rate

Turning models change the perspective, compared to the previous ones, as they rely on the target kinematics instead of on random processes when describing the target movement. Previously the focus was on modelling the time correlation between samples more than really describing the trajectory that the target was following. Now the approach changes.

The models that are introduced in the following sections belong to a more general model referred to as “Constant Turn” (CT) model, which assumes some of the typical particularities of a coordinated turn,¹⁹² which are that the aircraft moves at a constant speed while it turns at a constant turn rate. There are two possible versions of this model depending on the prior knowledge about the turn rate (or angular rate), namely:

- The “turn right” (TR) or “turn left” (TL) models, which assume that the value of the angular rate and the turn direction are known.
- The “constant turn rate” (CTR) model, which estimates the value of the angular rate and its turning direction.

The main difference between them is the appearance of an additional term in the state vector, i.e. the turn rate (ω), that impacts both the state transition matrix and the projection matrix, apart from the fact that the first one becomes linear while the second is clearly nonlinear.

¹⁹² A coordinated turn is a manoeuvre that implies more complex assumptions than the ones presented in this work. It considers other dynamical aspects that are of no relevance to this work.

Focussing on the TR/TL models, they are second-order models that introduce some uncertainty assuming an acceleration in x and y due to the uncertainty in the already known turn rate. These accelerations are modelled as a zero-mean white Gaussian noise. As before, there are two approaches, the direct discrete-time counterpart (piecewise constant acceleration), and the discrete-time equivalent (acceleration as an impulse). The associated system state and measurement equations, for the first approach, are:

$$\underline{x}(k+1) = F\underline{x}(k) + G\underline{w}(k) \quad (7.90)$$

$$\underline{z}(k) = H\underline{x}(k) + \underline{v}(k) \quad (7.91)$$

Where:

- $\underline{x}(k)$: state vector, which has the following layout:

$$\underline{x}(k) = [x \quad V_x \quad y \quad V_y]^T \quad (7.92)$$

- F : state transition matrix for either the TR model or the TL model, which is given by:

$$F = \begin{bmatrix} 1 & \frac{\sin \omega T}{\omega} & 0 & -\frac{1 - \cos \omega T}{\omega} \\ 0 & \cos \omega T & 0 & -\sin \omega T \\ 0 & \frac{1 - \cos \omega T}{\omega} & 1 & \frac{\sin \omega T}{\omega} \\ 0 & \sin \omega T & 0 & \cos \omega T \end{bmatrix} \quad (7.93)$$

- G : noise projection matrix into the state space, which is given by:

$$G = \begin{bmatrix} T^2/2 & 0 \\ T & 0 \\ 0 & T^2/2 \\ 0 & T \end{bmatrix} \quad (7.94)$$

- $\underline{w}(k)$: white Gaussian process noise sequence that is characterized by:

$$\underline{w}(k) = [w_x(k) \quad w_y(k)]^T \xrightarrow{\forall i,j} E[\underline{w}(i) \underline{w}^T(j)] = Q \delta(i-j) \quad (7.95)$$

Where Q is its covariance matrix that is given by:

$$Q = G \begin{bmatrix} \sigma_{TX}^2 & 0 \\ 0 & \sigma_{TY}^2 \end{bmatrix} G^T = \sigma_{TX}^2 \begin{bmatrix} T^4/4 & T^3/2 & 0 & 0 \\ T^3/2 & T^2 & 0 & 0 \\ 0 & 0 & T^4/4 & T^3/2 \\ 0 & 0 & T^3/2 & T^2 \end{bmatrix} \quad (7.96)$$

With:

- $\delta(i-j)$: Kronecker delta function.

- T : sampling period.
- σ_{TX}^2 : variance that characterizes the random increments in the x and y components of the aircraft acceleration. "X" can be "R" or "L" depending on the turning direction. If $\omega > 0$ the model to be applied is the TL model, otherwise the TR one. Note that it is assumed that the value is the same for both components (x, y) and both models (TR or TL).
- H : projection matrix in charge of translating the state into the measurement space. Assuming the state vector layout previously shown and considering that the measurement space only includes position, it looks like as:

$$H = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \quad (7.97)$$

- $\underline{v}(k)$: white Gaussian measurement noise sequence as defined in section 7.4.6.3.2.

Conversely, if the discrete-time equivalent approach is used [Li 03], then:

$$Q = cov(\underline{w}(k)) = S_{TX} \begin{bmatrix} \frac{2(\omega T - \sin \omega T)}{\omega^3} & \frac{1 - \cos \omega T}{\omega^2} & 0 & \frac{\omega T - \sin \omega T}{\omega^2} \\ \frac{1 - \cos \omega T}{\omega^2} & T & -\frac{\omega T - \sin \omega T}{\omega^2} & 0 \\ 0 & -\frac{\omega T - \sin \omega T}{\omega^2} & \frac{2(\omega T - \sin \omega T)}{\omega^3} & \frac{1 - \cos \omega T}{\omega^2} \\ \frac{\omega T - \sin \omega T}{\omega^2} & 0 & \frac{1 - \cos \omega T}{\omega^2} & T \end{bmatrix} \quad (7.98)$$

Like in the previous models, S_{TX} is the power spectral density of the continuous-time white noise process $w(t)$.

7.4.6.3.5 Turn rate estimation

The other version of the CT model estimates the value of the turning rate, so it implies including a new component in the state vector, i.e. the angular velocity or turn rate. This change impacts the rest of elements that define the model as all the matrices will have to be enlarged to cope with the new component.

Opposite to the previous models that were linear, this one is a nonlinear model with means an increased complexity. As such, there is a need for acquiring its discrete-time linear approximate model, which will be needed in case the state is to be estimated based on a discrete-time linear filtering technique, which is quite often the case when implementing an EKF, for example. Hence, this implies the discretization of the continuous-time model, which is described in [Gustafsson 96], where two ways of reaching a discrete-time model are proposed, namely:

- Discretized linearization.
- Linearized discretization.

The discretized linearization first linearizes the nonlinear equations of the state and afterwards discretizes the resulting linearized differential equations. The solution is then found via integration. This approach is quite extended as it is not difficult to obtain.

Contrarily, the linearized discretization first discretizes the model equations and then linearized the discrete-time model. The issue with this approach is that it is not always possible to reach a tractable discretized expression of the continuous-time model as it means solving a nonlinear differential equation. However, if found, it would provide more accurate results than the other approach as shown in [Gustafsson 96]. Luckily, the differential equation involved in CTR models are manageable [Roth 14], so this approach is the selected one for the models to be presented in this section.

Apart from the choice of previous approaches, there is another decision to be taken, which is the coordinates in which the state vector, or mainly its velocity components, are going to be described. Although in this work, the decision is to express the velocity in Polar coordinates as it is justified in section 7.4.6.4. For the sake of completeness, both choices are going to be presented hereafter. Firstly, the one based on a Cartesian representation of the velocity, and then, the one that accounts for the Polar representation.

These models will also differentiate from the previous ones on the election of the components of the noise vector, which will introduce the manoeuvrability in the model. These components are the longitudinal acceleration (along-track) and the transversal one (across-track), also known as tangential acceleration and normal acceleration, respectively. These models are also called “curvilinear motion models” in [Li 03] and one example can be found in [Best 97]. The second component is substituted in some cases by the angular acceleration as it is proposed in [Roth 14].

Last, the models consider two options when modelling the change in the accelerations, as before. That is, the acceleration can be assumed to take a piecewise constant value between sampling instants or to acts as an impulse just before each sampling instant as in [Roth 14]. This will be reflected when presenting the noise projection matrices.

The representation in this case is going to be slightly changed due to the nonlinear nature of the model equations.

Constant turn in polar coordinates

The associated system and measurement equations for this model are:

$$\underline{x}(k+1) = f(\underline{x}(k)) + G(\underline{x}(k))\underline{w}(k) \quad (7.99)$$

$$\underline{z}(k) = H\underline{x}(k) + \underline{v}(k) \quad (7.100)$$

Where:

- $\underline{x}(k)$: state vector, which has the following layout:

$$\underline{x}(k) = [x \quad y \quad V \quad \theta \quad \omega]^T \quad (7.101)$$

With:

- V : ground speed.

- θ : heading.
- ω : turn rate.
- $f_p(\underline{x}(k))$: nonlinear state dynamics for the CT model in polar coordinates, which is given by:

$$f_p(\underline{x}(k)) = \begin{bmatrix} x + \frac{2V}{\omega} \sin\left(\frac{\omega T}{2}\right) \cos\left(\theta + \frac{\omega T}{2}\right) \\ y + \frac{2V}{\omega} \sin\left(\frac{\omega T}{2}\right) \sin\left(\theta + \frac{\omega T}{2}\right) \\ V \\ \theta + \omega T \\ \omega \end{bmatrix} \quad (7.102)$$

- $G_p(\underline{x}(k))$: noise projection matrix into the state space for a state vector in polar coordinates, depending on the assumption regarding the change in the value of the acceleration, there are two options which are given by:

$$G_p^1 = \begin{bmatrix} 0 & 0 \\ 0 & 0 \\ T & 0 \\ 0 & 0 \\ 0 & T \end{bmatrix} \quad or \quad G_p^2 = \begin{bmatrix} \frac{T^2}{2} \cos(\theta) & 0 \\ \frac{T^2}{2} \sin(\theta) & 0 \\ T & 0 \\ 0 & \frac{T^2}{2} \\ 0 & T \end{bmatrix} \quad (7.103)$$

Where:

- G_p^1 assumes an acceleration that acts as an impulse just before each sampling instant.
- G_p^2 considers a piecewise constant acceleration between the sampling instants.
- $\underline{w}(k)$: white Gaussian process noise sequence that is characterized by:

$$\underline{w}(k) = [a_t(k) \quad \alpha(k)]^T \xrightarrow{\forall i,j} E[\underline{w}(i) \underline{w}^T(j)] = Q \delta(i - j) \quad (7.104)$$

Where Q is its covariance matrix that is computed as:

$$Q = G_p \begin{bmatrix} \sigma_{a_t}^2 & 0 \\ 0 & \sigma_{\alpha}^2 \end{bmatrix} G_p^T \quad (7.105)$$

With:

- $\delta(i - j)$: Kronecker delta function.
- $\sigma_{a_t}^2$: variance of the tangential acceleration.

- σ_a^2 : variance of the angular acceleration.
- H : projection matrix in charge of translating the state into the measurement space. Assuming the state vector layout previously shown and considering that the measurement space only includes position, it will look like as:

$$H = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \end{bmatrix} \quad (7.106)$$

- $\underline{v}(k)$: white Gaussian measurement noise sequence as defined in section 7.4.6.3.2.

Constant turn in Cartesian coordinates

The associated system and measurement equations for this model are:

$$\underline{x}(k+1) = f(\underline{x}(k)) + G(\underline{x}(k))\underline{w}(k) \quad (7.107)$$

$$\underline{z}(k) = H\underline{x}(k) + \underline{v}(k) \quad (7.108)$$

Where:

- $\underline{x}(k)$: state vector, which has the following layout:

$$\underline{x}(k) = [x \quad y \quad V_x \quad V_y \quad \omega]^T \quad (7.109)$$

With:

- V_x, V_y : x and y components of the ground speed.
- ω : turn rate.
- $f_c(\underline{x}(k))$: nonlinear state dynamics for the CT model in Cartesian coordinates, which is given by:

$$f_c(\underline{x}(k)) = \begin{bmatrix} x + \frac{V_x}{\omega} \sin(\omega T) - \frac{V_y}{\omega} (1 - \cos(\omega T)) \\ y + \frac{V_x}{\omega} (1 - \cos(\omega T)) + \frac{V_y}{\omega} \sin(\omega T) \\ V_x \cos(\omega T) - V_y \sin(\omega T) \\ V_x \sin(\omega T) + V_y \cos(\omega T) \\ \omega \end{bmatrix} \quad (7.110)$$

- $G_c(\underline{x}(k))$: noise projection matrix into the state space for a state vector in Cartesian coordinates, depending on the assumption regarding the change in the value of the acceleration, there are two options which are given by:

$$G_c^1 = \begin{bmatrix} 0 & 0 \\ 0 & 0 \\ T \cos(\theta) & 0 \\ T \sin(\theta) & 0 \\ 0 & T \end{bmatrix} \quad \text{or} \quad G_c^2 = \begin{bmatrix} \frac{T^2}{2} \cos(\theta) & 0 \\ \frac{T^2}{2} \sin(\theta) & 0 \\ T \cos(\theta) & 0 \\ T \sin(\theta) & 0 \\ 0 & T \end{bmatrix} \quad (7.111)$$

Where:

- G_c^1 assumes an acceleration that acts as an impulse just before each sampling instant.
- G_c^2 considers a piecewise constant acceleration between the sampling instants.
- $\underline{w}(k)$: white Gaussian process noise sequence that is characterized by:

$$\underline{w}(k) = [a_t(k) \quad \alpha(k)]^T \xrightarrow{\forall i,j} E[\underline{w}(i) \underline{w}^T(j)] = Q \delta(i - j) \quad (7.112)$$

Where Q is its covariance matrix that is computed as:

$$Q = G_c \begin{bmatrix} \sigma_{a_t}^2 & 0 \\ 0 & \sigma_{\alpha}^2 \end{bmatrix} G_c^T \quad (7.113)$$

With:

- $\delta(i - j)$: Kronecker delta function.
- $\sigma_{a_t}^2$: variance of the tangential acceleration.
- σ_{α}^2 : variance of the angular acceleration.
- H : projection matrix in charge of translating the state into the measurement space. Assuming the state vector layout previously shown and considering that the measurement space only includes position, it will look like as:

$$H = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \end{bmatrix} \quad (7.114)$$

- $\underline{v}(k)$: white Gaussian measurement noise sequence as defined in section 7.4.6.3.2.

7.4.6.3.6 Three-wheel movement

Until now the movement models have not considered that aircraft are tricycles or three-wheel systems with the front wheel being the steering one and the rear ones the powered ones. This information can be exploited to obtain a better movement model.

The approximation that has been taken is to simplify the aircraft movement model to a kinematic bicycle model, which is equally accepted for tricycles [Klančar 17] by just

collapsing the pair of single wheels at the centre of the rear axle. The derivation of the model can be found in [Snider 09].

These are models, which are quite used in the control of mobile robots and for autonomous vehicle navigation, that impose several assumptions such as:

- Wheels have no lateral slip.
- Only the front one is steerable.
- Vehicle moves on the XY plane (two-dimensional environment).

The model that is proposed in this case tracks the motion of the centre of the rear axle and it is represented in the picture below.

Figure 7.9: Three-wheel model of an aircraft. Source: own elaboration.

The model provides a relationship between the aircraft turn rate (ω) and the front wheel steering angle (δ) through the ground speed (V) and the wheelbase (L), which is the distance from the front wheel to the rear axle. This relationship is the following:

$$\omega = \frac{V \tan \delta}{L} \quad (7.115)$$

Considering this, the associated system and measurement equations for this model are:

$$\underline{x}(k+1) = f(\underline{x}(k)) + G(\underline{x}(k)) \underline{w}(k) \quad (7.116)$$

$$\underline{z}(k) = H \underline{x}(k) + \underline{v}(k) \quad (7.117)$$

Where:

- $\underline{x}(k)$: state vector, which has the following layout:

$$\underline{x}(k) = [x \quad y \quad V \quad \theta \quad \delta]^T \quad (7.118)$$

With:

- V : ground speed.
- θ : heading.
- δ : steering angle.
- $f_{3w}(\underline{x}(k))$: nonlinear state dynamics for the three-wheel model in polar coordinates, which is given by:

$$f_{3w}(\underline{x}(k)) = \begin{bmatrix} x + \frac{2L}{\tan \delta} \sin\left(\frac{V \tan \delta}{2L} T\right) \cos\left(\theta + \frac{V \tan \delta}{2L} T\right) \\ y + \frac{2L}{\tan \delta} \sin\left(\frac{V \tan \delta}{2L} T\right) \sin\left(\theta + \frac{V \tan \delta}{2L} T\right) \\ V \\ \theta + \frac{V \tan \delta}{L} T \\ \delta \end{bmatrix} \quad (7.119)$$

- $G_{3w}(\underline{x}(k))$: noise projection matrix into the state space for a state vector in polar coordinates, depending on the assumption regarding the change in the value of the acceleration, there are two options which are given by:

$$G_{3w}^1 = \begin{bmatrix} 0 & 0 \\ 0 & 0 \\ T & 0 \\ 0 & 0 \\ 0 & T \end{bmatrix} \quad or \quad G_{3w}^2 = \begin{bmatrix} \frac{T^2}{2} \cos(\theta) & 0 \\ \frac{T^2}{2} \sin(\theta) & 0 \\ T & 0 \\ 0 & \frac{VT^2}{L} \sec^2 \delta \\ 0 & T \end{bmatrix} \quad (7.120)$$

Where:

- G_{3w}^1 assumes an acceleration that acts as an impulse just before each sampling instant.
- G_{3w}^2 considers a piecewise constant acceleration between the sampling instants.
- w_k : white Gaussian process noise sequence that is characterized by:

$$\underline{w}(k) = [a_t(k) \quad \alpha(k)]^T \xrightarrow{\forall i,j} E[\underline{w}(i) \underline{w}^T(j)] = Q \delta(i - j) \quad (7.121)$$

Where Q is its covariance matrix that is computed as:

$$Q = G_{3w} \begin{bmatrix} \sigma_{a_t}^2 & 0 \\ 0 & \sigma_{\alpha}^2 \end{bmatrix} G_{3w}^T \quad (7.122)$$

With:

- $\delta(i - j)$: Kronecker delta function.
- $\sigma_{a_t}^2$: variance of the tangential acceleration.
- σ_a^2 : variance of the angular acceleration.
- H : projection matrix in charge of translating the state into the measurement space. Assuming the state vector layout previously shown and considering that the measurement space only includes position, it will look like as:

$$H = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \end{bmatrix} \quad (7.123)$$

- $\underline{v}(k)$: white Gaussian measurement noise sequence as defined in section 7.4.6.3.2.

This model is the one that is used as the support for the corresponding filters and smoothers for the case of manoeuvring targets.

7.4.6.4 Coordinate selection

The coordinate selection is a key choice, as it is mentioned in [Li 03], because there are plenty of derived effects linked to it, such as: the selection of the target dynamic model, the accuracy of the approximations that are done, the measurement coordinate conversions, and some other issues that directly depend on them.

The state vector to be considered in the proposed PET is the one presented in 7.118, which is composed of position components, velocity components in polar coordinates and the steering angle. This is for the case of manoeuvring targets, although this state vector can cope well with other models, from the ones presented in section 7.4.6.3, by just tuning it properly.

The rationale of such decision is based on some practical aspects but also in the better performance that state estimation algorithms shows when processing this type of information. The latter is analysed and stated in [Gustafsson 96], where state vectors described in polar coordinates are shown to provide better performances than the equivalent ones in Cartesian coordinates.

Moreover, the fact that aircraft moving around airport may stand for a while or may be parked, also justifies the selection of the polar coordinates as at low speeds the heading is conveniently tracked, while for Cartesian coordinates there might appear some instabilities.

7.4.7 State estimation

As it has been stated when describing the SSM (ref. 7.4.4), there are three main problems that can be faced depending on the objective of the analysis that is supported by the time series data (observations), namely:

- The estimation of the state of the system.

- The extraction of the models to be used for modelling the system.
- The estimation of the parameters of the chosen models.

The first one is included in what is called inference, while the other two are considered as learning problems.

The topic to be addressed within this thesis is included in the first type of problems and it can be defined as finding a performant algorithm for state estimation while processing recorded surveillance data from multiple sensors concerning multiple targets. In other words, the aim is to design a state estimator that operates in a multitarget -multisensor environment with offline surveillance data that is previously pre-processed for both converting them to the same reference framework and removing the existing biases.

In state estimation, the variables to be predicted are part of the state of a dynamic system, e.g. an aircraft. When this estimation is performed in a multitarget-multisensor environment, it implies solving two important issues:

- Measurement origin uncertainty.
- Uncertainty introduced by the noises that affect the system and observation processes.

The first issue comes from the fact that the airport environment includes the presence of not real targets caused by clutter and false alarms, which are generated by the sensors. The second one is intrinsic to the modelling of the system and measurement processes provided for the targets and the sensors, respectively.

Hence, the state estimation problem for a unique target can be graphically represented by:

Figure 7.10: Block diagram of state estimation. Source: [Bar-Shalom 95]

If this is performed on-line, then the problem is called a “tracking” problem, which can be graphically represented in the following way:

Figure 7.11: Block diagram of a tracking system. Source: [Bar-Shalom 95]

When the state estimation problem is performed offline or pseudo-online on recorded data, then it is called a smoothing problem. This is shown graphically on the following picture:

Figure 7.12: Block diagram of a smoothing system. Source: own elaboration.

Moreover, the problem of obtaining the state estimate of the targets is a classical data fusion problem which implies having to address all the typical issues, such as:

- Appearance of false alarms or clutter.
- Coordinate conversions.
- Need for bias estimation and cancellation.
- Outliers or out of sequence measurement.

In the case of a PET, the environment is multisensor-multitarget, which means that there is a large number of targets that can be detected by more than one sensor. This fact can be represented as follows:

Figure 7.13: Block diagram of a generic PET. Source: own elaboration.

7.4.8 Optimal estimation problems

In state estimation theory [Kay 93], there are four main types of problems depending on the information available and on the requested information:

- Filtering.
- Prediction.
- Smoothing.
- Interpolation.

One must choose among the three first options when the state estimation problem is constraint on already predetermined time instants, which are usually defined by the detection time of the observations. Moreover, there is also the interpolation problem where the state estimation is requested at any time between two defined ones. All of them have in common that they provide a state estimation of a target but in different conditions. These conditions are going to be outlined in the following subsections.¹⁹³

7.4.8.1 Filtering

A filtering problem is faced when the state estimate is computed at the time when the last measurement was received, taking into consideration the information contained in such measurement; that is, if the measurement has been received at instant k , the filtered state estimate at instant k is computed taking into consideration all the previous information, which is normally summarized by the last state estimate,¹⁹⁴ and the last measurement received at instant k .

Using probability notation, this can be expressed as:

$$\hat{\underline{x}}_f(k|k) = E[\underline{x}(k)|\underline{z}(k), \dots, \underline{z}(1)] \quad (7.124)$$

Graphically:

Figure 7.14: Filtered state estimate. Source: own elaboration.

¹⁹³ Please note that from this point on, the control theory notation is going to be substituted by the common one in tracking. Thus, the following changes are expected:

- Vectors are identified by an underscore, e.g. $\underline{x}$, and estimates by a circumflex accent, e.g. $\hat{\underline{x}}$.
- Measurement: $y \rightarrow z$
- Modes: $r_k \rightarrow M(k)$

¹⁹⁴ In the case the Markov property applies.

7.4.8.2 Prediction

A prediction problem consists in estimating a future value of the state based on the measurements already received until the current moment. Assuming that instant k represents the future instant when the state wants to be estimated, and l represents the time instant when the last measurement was received (i.e. $k > l$), this fact can be mathematically represented as:

$$\hat{\underline{x}}_p(k|l) = E[\underline{x}(k)|\underline{z}(l), \dots, \underline{z}(1)] \quad (7.125)$$

Graphically:

Figure 7.15: Predicted state estimate. Source: own elaboration.

7.4.8.3 Interpolation

An interpolation problem consists in generating the state estimate at any time j between two predefined instants (i.e. k and $k + 1$) when a state estimate was already provided. This problem is faced when a state estimate is needed in a time which does not belong to the initial time series.

Mathematically, this state estimation problem can be expressed as:

$$\hat{\underline{x}}_i(t|k, k + 1) = E[\underline{x}(t)|\underline{x}(k), \underline{x}(k + 1)] \quad \forall t \in \mathbb{R} \wedge \forall k \in \mathbb{N} \wedge t_k < t < t_{k+1} \quad (7.126)$$

An example of such situation is when the state estimate must be computed at the time of measurements provided by a sensor different from the one that was used for calculating the initial state estimate, such as in section 6.10.7.3, when interpolating the state estimate corresponding to the reference sensor reconstructed trajectory at the time when the MLAT measurement were taken.

This situation is graphically represented in Figure 7.16, where the greyed red circles represent the (MLAT) measurements that define the time instants when the interpolated state estimates, i.e. red diamonds, are computed. The blue circles and blue diamonds represent the measurements from the reference sensor and the corresponding state estimates, respectively. The latter can be obtained by either filtering (ref. 7.4.8.1) or smoothing (ref. 7.4.8.4) depending on the constraints of the addressed problem.

Figure 7.16: Interpolated state estimate. Source: own elaboration.

7.4.8.4 Smoothing

Finally, a smoothing problem is found when the generation of the state estimate at time instant k is supported not only by past and present information (i.e. information up to time instant k) but also by future information (i.e. from time instant k onward). This is the case when an offline processing is performed based on recorded data or, at least, delayed several minutes. A clear example is the computation of reference trajectories in a PET.

Mathematically speaking, this state estimation problem can be expressed in a similar way as in the previous problem but just bearing in mind that now $l > k$:

$$\hat{x}_s(k|l) = E[x(k)|z(l), \dots, z(k), \dots, z(1)] \quad (7.127)$$

Which can be shown in a graphical way as:

Figure 7.17: Smoothed state estimate. Source: own elaboration.

7.4.9 Types of smoothing problems

Taking into consideration the main objective of this thesis, which is to reconstruct the actual trajectory that a target has followed around the airport from a set of recorded data, the focus is going to be placed on the smoothing problem. Data smoothing provides the best solution to this state estimation problem as all the information concerning the targets is already available and so it can be exploited. This kind of algorithms allows to take into consideration not just the past and present information but also the future one with respect to the moment in which the state estimate is computed.

However, none of the other problems, i.e. filtering, interpolation and prediction, are going to be forgotten as they take part in the solution to the smoothing problem or in the trajectory reconstruction presentation. This is the reason why some filtering algorithms are presented as part of ANNEX I.

Coming back to the smoothing problem, a further classification is presented based on the parameter selection defined by the underlying state estimation problem. These parameters are:

- N : number of observations involved in the state estimation problem.
- l : index of the last observation that is used in the computation.
- k : index of the observation when the estimation is desired.
- Δ : delay (in number of observations) between k and l that is waited before computing the smoothed estimate.

Based on this criterion, the following types of smoothing problems can be found.

7.4.9.1 Fixed-interval smoothing

This type of smoothing problem assumes that all data are already available, so l is known and fixed, and it is assumed to coincide with N , which does not grow with time. Therefore, the smoothed state estimation for a fixed-interval problem is obtained as k goes from 0 to N taking into consideration the information in all the available data:

$$\hat{\underline{x}}_s(k|l) = \hat{\underline{x}}(k|N) \quad (7.128)$$

This is the case faced when estimating reference trajectories by offline processing recorded data. Consequently, this would be the most relevant case for this work. The rest of the smoothing types are just presented for the sake of completeness.

7.4.9.2 Fixed-point smoothing

There are other situations in which a different smoothing problem can be found. If the aim is to improve the state estimation at the time instant when a certain observation (k_0) is received, while observations are still been received, the fixed-point smoothing is the type of problem that applies. Hence, the fixed parameter is now the index k_0 of the observation that defines the time instant when the state estimation is desired, while N increases with time t . Thus, the expression would be:

$$\hat{\underline{x}}_s(k|l) = \hat{\underline{x}}(k_0|N(t)) \quad (7.129)$$

This type is of little importance for the topic addressed by this thesis as the reconstructed trajectory should cover from the first measurement to the last one, with the same especial interest in any of the intermediate time instants.

7.4.9.3 Fixed-lag smoothing

Finally, the last one is the fixed-lag smoothing, which addresses the situation in which both the index of the observation (k) when the state estimation is desired, and the number of observations to be used for its estimation (N), increase with time t . In this case, the constant parameter is the delay in number of observations (Δ) between the index of the observation that defines the estimation time and the index of the last observation (l) included in the computation. This is represented as:

$$\hat{\underline{x}}_s(k|l) = \hat{\underline{x}}(l - \Delta|l) = \hat{\underline{x}}(N(t) - \Delta|N(t)) \quad (7.130)$$

This type of smoothing is interesting in the case that a pseudo on-line performance evaluation processing is desired.

7.4.10 Smoothing algorithms

Based on the characteristics of the reference trajectory reconstruction problem, the fixed-interval smoothing type is the one that better matches. So, the objective of this section is to introduce the smoothing algorithms that better address the fixed-interval smoothing problem.

These smoothing algorithms must exploit all the available information concerning the target state, $\underline{x}(k)$, from the first observation that has been associated to this target to the last one, which might be defined by either the disappearance of the target from the domain of interest of the system or from the coverage of the sensors, or by reaching the end of the available information in case of a recording. As such, each state estimate is computed based on the same number of observations (N). Moreover, there will be a state estimate per received observation, i.e. $k \in [0, \dots, N]$. This fact allows an offline data processing which can be organized in batches, if the computation load exceeds the available processing power.

In most of the cases, the aim is finding the optimal smoothing algorithm, which implies specifying a suitable criterion of optimality. The latter will define the look-like of the algorithm to be used. However, the optimal solution to the problem might not be feasible as it might mean a resource need which grows exponentially with the number of observations to be processed per target. This is typically the case when processing a SHS (ref. 7.4.5). So, it is taken into consideration when designing the proposed smoothing solution for the trajectory reconstruction that is presented in section 7.8.3.4.

The next subsections describe two recursive algorithms which provide the optimal state estimation for a smoothing problem in which the system does not present different motion modes. These algorithms are the “two-filter smoothing” and the “forward-backward smoothing”.

7.4.10.1 Two-filter smoothing

This algorithm is also called, in [Gelb 74], the “forward-backward filter” formulation of the optimal smoother because it provides the smoothed estimate of the state, i.e. $\hat{\underline{x}}(k|N)$,¹⁹⁵ as a suitable combination of two optimal filters, one going forward in time (“forward filter”), which exploits all the information until and including the k -th observation, and provides the estimate $\hat{\underline{x}}(k)$; and the other going in the opposite direction, that is, backward in time, from the last observation associated to the target (N) upto the observation $k + 1$. Thus, the latter is called the “backward filter”, as it operates on all data in the future compared to the k -th observation, and it provides the estimate $\hat{\underline{x}}_b(k)$. This way, both estimates jointly exploit all the available information. In order to maintain the estimates uncorrelated, the forward estimation also includes the current measurement which is not processed by the backward estimation.

Therefore, the smoothed estimate $\hat{\underline{x}}(k|N)$ is computed by optimally combining the forward and the backward estimates; that is, $\hat{\underline{x}}(k)$ and $\hat{\underline{x}}_b(k)$. This computation also calculates the estimated error covariance matrices, which are: $P(k)$ and $P_b(k)$, respectively.

It is worth noting that there are some assumptions concerning the time reversibility of the measurement process that must be done in order to obtain $\hat{\underline{x}}_b(k)$. These assumptions are required on the transition matrix $F(k, M(k))$ and on the “initial conditions” of both, the backward state estimate, i.e. $\hat{\underline{x}}_b(k)$, and the backward error covariance matrix, i.e. $P_b(k)$. Please refer to the following subsection to have a better understanding of this issue.

The expression which provides the smoothed estimate for the forward-backward filter is:

$$\hat{\underline{x}}(k|N) = P(k|N)[P^{-1}(k)\hat{\underline{x}}(k) + P_b^{-1}(k)\hat{\underline{x}}_b(k)] \quad (7.131)$$

Where:

$$P(k|N) = [P^{-1}(k) + P_b^{-1}(k)]^{-1} \quad (7.132)$$

The details about the way to derive these expressions can be found in [Fraser 69] or in [Gelb 74].

Regarding the performance, this formulation of the optimal smoother is considered to be less efficient than the one that is going to be presented in the next subsection. However, due to some other advantages, it is considered of interest for this work.

7.4.10.1.1 Backward filters and time reversibility

As it has been outlined in the previous section, it is relevant to analyse the reversibility of the underlying processes as it will allow us to know if there is a backward expression for the forward filter.

Concerning time reversibility, the concept can be presented for both, deterministic processes and stochastic processes. Focussing on deterministic process, they are time

¹⁹⁵ Please note that for the sake of clarity the subindex s has been omitted as it is kind of redundant with the current representation of the smoothed state estimate, i.e. $\hat{\underline{x}}(k|N)$, where the conditionality on all the associated measurements, past and future ones, is represented by N .

reversible if the time-reversed or backward process satisfies the same dynamical equations as the forward-time process. This means that the equations are symmetric under a change in the direction of time. This is the case for classical mechanics which apply to the movement models where past, present, and future are entirely contained in a single timeless equation.

Concerning stochastic processes, they are time-reversible when their statistical properties are invariant to the direction of time [Chang 06]. This can also be stated considering that the probability distribution of the original process $\{X(k_1), X(k_2), \dots, X(k_n)\}$ is the same as the one shown by the time-reversed one $\{X(-k_1), X(-k_2), \dots, X(-k_n)\}$.

Stochastic processes are characterized depending on the dimension of the state. Thus, univariate stochastic linear processes are time reversible if they are Gaussian distributed [Weiss 75] or if they comply with the conditions stated in [Cheng 99]. However, this is not true in general for multivariate linear stochastic processes. [Tong 05] and later [Chang 06] address the multivariate case.

Focussing on Markov processes, there is a theorem which states that if $\{X(k)\}$ is a stationary Markov process with transition probabilities p_{ij} , and stationary probabilities π_i , then the backward time process is also Markov with transition probabilities:

$$p_{ij}^* = \frac{\pi_j}{\pi_i} p_{ji} \quad (7.133)$$

In general, $p_{ij}^* \neq p_{ji}$. However, in the case that $p_{ij}^* = p_{ji}$ (which means that $\pi_i p_{ij}^* = \pi_j p_{ji}$), the process is also time-reversible.

The important fact for this work, as it is said in [Tong 05], is that "If the series is time-reversible, then the projection forward in time is equivalent to the projection backward in time". This can be exploited by backward algorithms while computing the smoothed state estimate.

A relevant example concerning a linear system with Markovian switching coefficients is addressed in [Blom 90] where the reversion in time of the stochastic difference equation in a hybrid space with a Markovian solution is obtained by a Martingale approach. The results of this publication are applied to a fixed interval smoothing of partial observation of a suddenly manoeuvring object (Trajectory smoothing).

7.4.10.2 Forward-backward smoothing

The "forward-backward smoothing" algorithm is a compact formulation of the optimal smoother that was presented by Rauch, Tung and Striebel in [Rauch 65]. This formulation is also known as the RTS smoother.

This algorithm uses the information provided by a forward filter when computing in the reverse direction the smoothed state estimate. Therefore, this implies a stepped approach which starts with the computation of all the "filtered" and "predicted" state estimates, i.e. $\hat{x}(k)$ and $\hat{x}(k+1|k)$; and their associated error covariance matrices $P(k)$ and $P(k+1|k)$. Once obtained, they are stored for a later use. After that, there is a backward pass where the smoothed estimates are computed in the reverse direction.

The formula which provides the smoothed state estimate is:

$$\hat{\underline{x}}(k|N) = \hat{\underline{x}}(k) + A(k)[\hat{\underline{x}}(k+1|N) - \hat{\underline{x}}(k+1|k)] \quad (7.134)$$

Where:

$$A(k) = P(k)F^T P^{-1}(k+1|k) \quad (7.135)$$

On the other hand, the error covariance matrix is computed by the following expression:

$$P(k|N) = P(k) + A(k)[P(k+1|N) - P(k+1|k)]A^T(k) \quad (7.136)$$

The derivation of these formulae can be obtained from [Rauch 65] or [Gelb 74].

Finally, there are different flavours of this smoothing algorithm depending on the underlying forward filter. As such, there is the Unscented RTS smoother which is presented in [Sarkka 08], the Cubature Kalman smoother that is introduced in [Arasaratnam 11] or a version that relies on Fourier Hermite series for the expansion of nonlinear functions, which is described in [Sarmavuori 12]. These are just several examples of a few smoothing algorithms based on the RTS-smoother thought for nonlinear systems.

7.4.10.3 Suboptimal strategies

Taking into consideration that aircraft can be modelled as mode-switching systems, i.e. hybrid systems, the optimal approach would be not feasible to be implemented in general, and, even more, if the number of observations grows considerably, as it implies a huge amount of smoothers working in parallel. Specifically, this number grows exponentially with the number of measurements to be processed (N) in order to cover all the possible combinations of movement models that the target may show in the whole interval. As a matter of example, if there are N samples and r possible modes are assumed, then the number of possible combinations and, therefore, the number of smoothers for reaching the optimal state estimate is r^N .

In such a situation, which is the common one when trying to reconstruct the trajectory followed by a target, the solution is to make use of suboptimal algorithms. The same way as it is done when filtering the measurements coming from this kind of systems, the preferred suboptimal solution is chosen to be based on IMM filters and their time-reversed counterparts. Thus, the objective is then to find the time-reversed IMM filter for such a system and then use the two-filter formulation for computing the smoothed state estimate or to find the forward-backward smoothing algorithm that properly addresses Markovian switching systems.

There are many examples of suboptimal solutions to this problem that can be found in the literature following both approaches. Hence, the two-filter algorithm approach is already used in [Blom 90] while testing their conclusion on time reversibility by the use of an IMM filter. Other contributions applying a similar IMM-based approach are [Helmick 93], [Helmick 95] or [García 07]. The latter describes the initial version of the algorithm used for trajectory reconstruction within the already mentioned SASS-C tool (ref. 3.3.3.1).

Moreover, there is also a large number of publications that rely on the other approach, that is, the forward filter-backward smoothing. This is the case for example of [Koch 00]

who propose a hierarchy of suboptimal retrodiction¹⁹⁶ algorithms that generalize the RTS fixed-interval smoothing to MHT applications employing IMM methods (IMM-MHT). Other examples are [Nandakumaran 09] or [Nadarajah 12], who propose a new smoothing method which involves forward filtering followed by a backward smoothing while maintaining the fundamental spirit of the IMM. Both contributions are initially focus on the fixed-lag smoothing problem, although they can be easily extended to the fixed-interval one.

7.5 Combined techniques

Apart from the two general techniques presented for computing the reference trajectory in sections 7.3 and 0, there are plenty of hybrid techniques which try to obtain the best features from a combination of them, multiplying thus the possibilities of addressing the problem. In fact, the use of both, splines and Bayesian approaches, is complementary in the sense that the latter can provide better statistically characterized state estimates while the former can offer a more smoothed reconstruction from a graphical point of view. It is worth noting here that users of performance evaluation tools such as SASS-C, in general, do not really worry about the optimality of the proposed trajectory reconstruction but most of the time by its graphical representation.

As a practical example, it can be cited that MURATREC II [Neven 94] combines both, spline reconstruction and a jump-linear smoothing algorithm, in the output that it offered to the users.

7.6 Examples of trajectory reconstruction techniques based on opportunity traffic

This section compiles the features of the trajectory reconstruction modules of the performance evaluation systems/suites which were presented in section 3.3.3.1. For each of them, the information concerning the technique which they use for trajectory reconstruction is described in case of having enough information about it.

7.6.1 MURATREC

The “Multi Radar Trajectory Reconstruction” (MURATREC) is the trajectory reconstruction module which has been part of SASS-C since its beginning in the late 80's. There are currently two versions, i.e. MURATREC I and II, in the still working but deprecated SASS-C v6. The particularities of each of them are outlined in the following subsections.

7.6.1.1 MURATREC I

This software took part of the initial attempt to build a common performance evaluation tool within Europe for radar and tracker assessment. This attempt was led by

¹⁹⁶ “Retrodiction” is a technical term introduced by Olivier Drummond in [Drummond 97], which represents the antonym of “prediction”. Its definition states “retrodiction: the process of computing estimates of states, probability densities, or discrete probabilities for a prior time (or over a period of time) based on data up to and including some subsequent time, typically, the current time”.

EUROCONTROL in cooperation with NLR, which were the responsible for designing and delivering its first version.

This first version of MURATREC was based on the use of B-splines for curve fitting [Renes 85]. As outlined in section 7.3, B-splines are one of the possible alternatives for reference trajectory reconstruction as it mentioned in [Van der Kraan 90]. In this case, the election was to implement 4th-order B-splines and to subdivide the analysis period into dynamically adjustable time intervals. This tuning depended on the movement developed by the target (mainly in its variability) so, in manoeuvring situations, the time intervals were shortened while in steady situations they were enlarged. In order to guarantee the smooth reconstruction of the trajectory, a constraint was imposed at the beginning and the end of each time interval. This constraint forced the continuity of the position and the velocity at the joints.

The reconstructed trajectories were stored in the form of B-splines coefficients (α_j) which resulted from applying a least-square approximation (see section 7.4.6.2.2). The number of coefficients per time interval is linked to the dimensions of the trajectories, usually three for an en-route environment.

MURATREC I presented some troubles related to the reconstruction of velocities and accelerations although the position estimate was quite acceptable. The architecture of this module was divided into several steps associated with the pre-processing tasks (e.g. association and bias estimation) and the data fusion or smoothing activities. Such a process was split into the following steps [Van der Kraan 90]:

1. *Chaining Process*: it builds datasets from opportunity traffic data in which all the data concerning a certain target are grouped together. This facilitates any further processing such as: bias estimation and cancellation, outlier detection, trajectory reconstruction, etc. The plots which cannot be associated to any of the datasets are marked as "false targets" and they are removed from any further processing.
2. *Systematic error estimation*: this process evaluates all the surveillance data that have been associated in the previous step and estimates the systematic error which can affect the measurements. The estimation algorithms are based on the measurement model of the different radar sensors (PSR, SSR, combined, etc.) which should look like the one presented in CHAPTER 2. In this case, only radar data was evaluated whose measurement model included the following components:
 - Sensor location.
 - Bias in azimuth and range.
 - Time stamp error.
 - Co-located sensor error.

The bias estimation also includes target-based biases such as the one related to the reply time of the SSR transponder.

3. *Systematic error cancellation*: after the biases are estimated, they are removed from the measurements whenever the estimation algorithm has converged enough to

guarantee the reliability of the processing. Depending on the estimation algorithm, this convergence may take more or less time.

4. *Trajectory reconstruction*: next, the reference trajectory reconstruction is carried out. Input data are assumed to be bias free as they have undergone through the previous steps. Thus, it is at this point when the curve fitting based on B-splines is performed.

Therefore, the output of such a process is a set of reconstructed reference trajectories which are provided in B-form (see 7.3.5.2). These ideal trajectories are then used as a reference for assessing the performance of surveillance sensors and trackers.

7.6.1.2 MURATREC II

MURATREC II was an alternative trajectory reconstruction module to MURATREC I, which is also available in SASS-C v6. The alternative technique implemented in MURATREC II is a hybrid technique as it combines an initial spline-based reconstruction with a jump-linear smoothing algorithm. More information about the details of MURETREC II can be found in [Neven 94].

7.6.2 OTR

The “opportunity traffic reconstruction” (OTR) is the trajectory reconstruction module of the “trajectory reconstruction & evaluation suite” (TRES), which was one of the four interoperable functional suites that comprised SASS-C v7 and subsequent versions.

The trajectory reconstruction technique is based on a forward-backward filter scheme in which both, the forward filter and the time-reversed one, are IMM-based. The details of the reconstruction algorithms are presented in [García 07], which describes how the data to be reconstructed are processed in order to first, classify them into motion segment then, the parameters of each of those segments are computed and, finally, this information is exploited while performing the fixed-interval smoothing based on two IMM filters.

Further details of this module, which includes much more than the trajectory reconstruction strategy, can be found in [Besada 08a], [García 09] or [Besada 13]. Currently, the latest version of SASS-C v8 still implements the OTR module, which still relies on pretty the same algorithms that were described in the aforementioned references.

7.7 Effects due to non-ideal conditions

The moment of truth of a PET is when it is faced with real data. Only in this moment, the designer of the tool knows whether the algorithms and the proposed architecture are valid or not. This fact explains why the designed tool has been tested with real data since the moment they were available.

There are many situations in which some of the assumptions that were made during the design process are found to be incorrect and so some mitigation means must be included. This section lists the more common issues that might be faced (or the ones that the author has found) when designing and testing such a tool.

7.7.1 Biases or systematic errors

As already said, biases or systematic errors are constant or slowly varying errors which, in general, can be estimated and then removed from the data processing as it has been presented in section 6.10. This issue is quite common in air traffic surveillance due to the intrinsic measurement process performed by surveillance sensors, and also to the features of aircraft and their surveillance equipment (e.g. SSR or Mode-S transponders, and antenna positions).

Taking into consideration that the trajectory reconstruction is a data fusion process [Llinas 01], there are some previous steps that must be incorporated to extract the maximum benefit of the implemented data fusion algorithm. These previous steps are usually called in the literature the prerequisites for the data fusion, being one of them, the bias estimation and cancellation as could be seen in [Renes 85] and the other, the coordinate conversion into a common reference frame.

If biases are not correctly removed, the quality of the estimation can be dramatically compromised meaning that reconstructed trajectories might show an erratic behaviour or present some instabilities, which may affect the continuity of the reconstructed trajectory. Some of these troubles can be mitigated by enlarging the noise of the processed data. In other situations, a trajectory-based bias estimation adjustment might be required, which might discover some correlation between trajectories that, at the end, may be exploited. This is something that has been experienced while testing the SMR bias estimation and that it is included for further research (see section 9.3) although an initial mitigation is provided.

7.7.2 Non-Gaussian random noise

There is a very common approach when working with stochastic processes which is to assume that the noises are additive and present a Gaussian distribution, which usually simplifies the statistical analysis of the problem. Hence, both, system and measurement processes, are initially assumed to be Gaussian processes, which is not always the case. Whenever this assumption is not valid, the Gaussian estimation techniques should be substituted by others which do not assume that the underlying process distribution is Gaussian. Algorithms such as: unscented Kalman filters [Julier 97], cubature Kalman filters [Arasaratnam 09], or particle filters [Ristic 04] are examples of such techniques.

7.7.3 Out of sequence or late measurements

These are typical issues of surveillance data provided by remote sensors. “Out of sequence” measurements and “late” measurements represent troubles while receiving the data from a sensor and these data are fused with the one coming from other sensors. There are two main situations in which sensor data might arrive not in order, namely:

1. Saturation of the sensor due to an abnormal increase in the number of detections.
2. Congestion of the communication infrastructure.

The former is usually related to a failure in the sensor or a misconfiguration of the detection thresholds or in the detection logic. Thus, it prevents the sensor to timely deliver the data. Taking into consideration the second problem, it is worth pointing out that surveillance data are received at ATC centres or towers after crossing one or several communication networks,¹⁹⁷ which may suffer from congestion due to several causes. This issue means that the sensor data, which might have been timely delivered, will reach the data fusion centre later than others. Both situations are reflected by the fact that data arrive with a certain “extra” delay to the one suffered by data coming from other sensors. This extra delay, which impacts the ToR of the data at the data fusion centre may imply that other data with a more recent ToA overtake the ones suffering the delay and, as such, they are processed before. So, this is translated in a problematic change in the processing order, as the measurements that arrive late will become “out of” the expected sequence, which might affect the tracking system. Depending on the situation it might be worth processing them as “late” measurements or, on the other hand, throwing them (or at least their kinematic information) to the bin. Usually, this is handled by setting a threshold so, if the measurements are late but less than the threshold, their kinematic information is taken into consideration by conveniently updating the track state vector. Otherwise the measurements are considered as “too” late and consequently, their kinematic information is discarded although the rest (e.g. identity or ADD) might be used.

It is worth noting that a PET does not suffer from this trouble when computing the reconstructed trajectory as they rely on the ToA of the measurements. However, they should assess the delay that the incoming data suffers to highlight any sensor or communication difficulty as it might explain some performance issues shown by the tracking systems.

7.7.4 Outliers

Outliers represent the other side of the statistical modelling problem related to sensor measurements. Thus, out of sequence measurements suffer a discrepancy in the time delay distribution while outliers show a discrepancy in the probability distribution of the measurement values. Therefore, outliers are data that do not follow the regular pattern of the rest of the data showing a different probability distribution, which do not match the general one, and therefore, the algorithms are not expected to process them correctly unless they are prepared for it by, for example, considering a double-Gaussian error model as it is the case for ARTAS [ARTAS]. Hence, it is important to devote some effort to first, identify them as soon as possible, and then, to treat them to mitigate their impact as their effect on the rest of the processing might be quite significant.

This trouble is better addressed by a PET, while computing the reconstructed trajectory, as they can exploit not only the information from the past and present but also from the future. So, their impact compared to the one caused in tracking systems is much reduced.

7.7.5 Quantified measurements

This effect is caused by the lack of precision in the measurements or the need for rounding them in order to meet the precision of data format of the message that is used for sending

¹⁹⁷ Or passing through one or several communication lines.

them to the ATC centre or tower. When the size of the quantization error of the data format is negligible compared to the measured magnitude, this effect can be obviated. However, if this is not the case, as can be seen in the airport domain, it is recommended to assess its impact and, if necessary, to model it and take it into account in the measurement models of the affected sensors. This is usually mitigated by enlarging the covariance matrix of the measurements and modelling the quantization error as a uniform random variable of the size of the quantization step.

This problem is clearly appreciated when there are standing aircraft and, as such, it is possible to appreciate the quantization effect. This effect is clearly seen in ADS-B target reports as it is shown in section 8.4.1.2.3 Figure 8.9.

7.7.6 Correlated measurements

ADS-B measurements are found to be correlated meaning that not all the delivered measurements are really the result of a measurement process but the result of an extrapolation of previous information. As it was mentioned in section 2.3.2, ADS-B measurements are sometimes extrapolated by the aircraft navigation system to the time when they are supposed to be applicable.

This effect is clearly observable when processing ADS-B data coming from certain aircraft (see section 8.4.1.2.3 Figure 8.9). The issue with this extrapolated information is that it is not easy to statistically characterize it as the processing that is performed by the navigation system to obtain such extrapolation is unknown. This is the reason why, in this work, it is preferred to discard them, meaning that there is a need for introducing a previous processing before they are incorporated to the trajectory reconstruction.

7.7.7 Manoeuvring targets and not modelled movements

As already mentioned in section 7.4.6.3, aircraft and ground vehicles are manoeuvring targets so the use of multiple model filters for tracking or smoothing their trajectories is more than recommended. Related to this, there is a design decision consisting in selecting the movement models that are going to be included in the data processing. This is part of what is called the learning process of an estimation problem.

Thus, the number of models is limited, which implies that there can be movements or manoeuvres that happen and that are not modelled. This fact impacts on the performance of the trajectory reconstruction module. However, if this situation is clearly identified, it is possible to model it and include the new movement model.

Finally, it is relevant to mention that the inclusion of new models does not always impact positively on the performance of the data processing, so some trade-off must be reached. In fact, in the proposal defined in this thesis, the number of motion models is reduced to two, a manoeuvring one and a non-manoeuving one. The manoeuvring model is selected and tuned to the expected manoeuvre exploiting some previous information such as the airport map or the aircraft ground speed, while the non-manoeuving one is there to cope with the situation before or after the expected manoeuvre. This emulates what is called a variable structure (VS) algorithm aiming at reducing the uncertainty introduced by non-related movement models in the final state estimate.

7.7.8 High-Energy manoeuvres

The appearance of “high-energy manoeuvres” (HEM), which are the one performed by military aircraft, is barely seen in the airport domain. These manoeuvres are out of the regular manoeuvres that are performed by civilian aircraft and so, they are not modelled within the trajectory reconstruction module. This effect is more frequent in the en-route domain where military aircraft can perform differently to civil ones.

In case detected by tracking systems, the best option is not to track the data and just show the raw data. Like when processing outliers or out of sequence measurements by a PET, the smoothing of such a trajectory is easier than when they are tracked, due to the availability of past and future data.

7.7.9 Coordinate transformation

This is another prerequisite to data fusion as it is mandatory to refer all the information to the same reference framework. This is so, because there are usually several types of sensors, each of them providing their measurements in a certain coordinate reference framework, which is normally different to the data fusion one. This issue has already been considered in section 2.1.4.

At this point, it is worth mentioning that this transformation may introduce some error which should be estimated and assessed in order to conveniently enlarge the covariance matrix of the transformed measurements. However, with the current coordinate projection algorithms, the introduced error is negligible compared to the others.

7.8 Data smoothing

7.8.1 Process description

The “data smoothing” process is the core of the trajectory reconstruction processing block and this is the reason why it is going to be presented before “false target report analysis”. Moreover, there might be some results from the “data smoothing” process that are convenient for false target report identification and classification, meaning that the precedence of one process over the other is not fully clear.

The objective of this process is to generate the so desired reconstructed, reference or ideal trajectories that will be used as the reference toward which target reports and track updates are compared in order to assess the performance of sensors and tracking systems, respectively. This process relies on the previous ones in order to receive input data in the expected conditions, that is, in a common reference frame, time aligned, associated per target, and bias corrected.

Considering the modelling of aircraft, which are assumed to be partly constraint stochastic hybrid systems; the particularities of an environment like the airport; and, the type of data processing that is performed, i.e. off-line; the smoothing algorithm is designed.

Consequently, the proposal of this thesis regarding the trajectory reconstruction consists of a fixed-interval smoothing algorithm based on a IMM forward filter-IMM backward smoother method that implements two mode-matched filters in the case of the IMM forward and the consequent mode-matched smoothers in the case of the IMM backward smoother. These mode-matched filters and smoothers consist of a non-maneuvring mode and a manoeuvring one. Both modes are selectable, meaning that depending on context information, they can be changed to better adapt to, on one side, the expected manoeuvre, and on the other, to the uniform motion segment that may come. This way the algorithm can intrinsically cope with the unknown moment in which the manoeuvre starts. The details of this proposal are provided in section 7.8.3.

7.8.2 Processing details

7.8.2.1 Input data

The input is the set of selected multisensor associated trajectories, whose data elements are transformed into the common reference frame, time aligned, and bias corrected.

7.8.2.2 Output data

The output is the equivalent set of selected multisensor reconstructed trajectories.

7.8.2.3 Block diagram

The “data smoothing” process can be graphically located in the “trajectory reconstruction” processing block, as it is shown in the following figure:

Figure 7.18: Block diagram focussed on the “data smoothing” process. Source: own elaboration.

7.8.3 Prototype implementation details

The trajectory reconstruction smoothing algorithm proposed in this thesis is presented hereafter. This algorithm is implemented within the prototype together with many other tracking or smoothing algorithms that are either considered as building blocks of the trajectory reconstruction algorithm or as references to be compared with the latter for performance analysis.

The proposal is presented in a bottom-up approach, so the initial building blocks are first introduced referring to the corresponding sections of this document where their description can be found to avoid repeating the same information. These building blocks refer to the mode-matched filters and smoothers that are considered, and the movement models they implement. Later the whole architecture is introduced providing the way in which context information and building blocks are merged together to compute the final state estimates that will constitute the reconstructed trajectory.

7.8.3.1 Aircraft movement model selection

The first question that needs to be answered is how the aircraft movements are going to be modelled. There are several approaches to this issue, one is based on defining a movement model that can be conveniently tuned to cope with, on one hand, the segments in which the aircraft follows a uniform motion or it is stopped, and, on the other, the segments where the aircraft performs any “regular” manoeuvre or any other unforeseen movement. This approach usually relies on the constant or nearly constant velocity model (ref. 7.4.6.3.2) for modelling both situations, i.e. with and without manoeuvre, by just tuning some parameters of the model, which normally means enlarging the process covariance matrix (Q). The advantages of this approach rely on its simplicity, as with only one model all the movements are covered, and on the fact that there are less interactions between models. However, the main disadvantage is the lack of adaptation to the actual movement followed by the target, which for sure will reduce the accuracy of any state estimation algorithm based on it.

Moreover, there is also the approach which tries to define a movement model per aircraft regular manoeuvre. This approach relies on models like the ones defined in section 7.4.6.3 for describing the aircraft or ground vehicle movements, namely:

- Stationary or stopped model, for standing target (ref. 7.4.6.3.1).
- Constant or nearly constant velocity model, for targets moving at constant speed and with a constant heading (ref. 7.4.6.3.2).
- Constant or nearly constant acceleration model, for accelerating or decelerating targets with a constant heading (ref. 7.4.6.3.3).
- Right/Left turn at a known turn rate model, for turning targets at a known angular speed (ref. 7.4.6.3.4).
- Turn rate estimation model, for turning targets when their angular speed is not known (ref. 7.4.6.3.5).
- Three-wheel movement model, for turning aircraft at an unknown angular speed when there are some aircraft features that are known such as the wheelbase (ref. 7.4.6.3.6).

This approach loses the simplicity and generality of the previous one but opposite to it, it highly improves the matching of the chosen model to the actual target movement. This will have a direct positive impact on the accuracy of any state estimation algorithm that relies on it. However, the coupling between the chosen models might also negatively impact the state estimation.

The approach proposed in this thesis is a combination of the two previous ones. The rationale behind this decision is the intention of keeping the best properties of both approaches by making use of some external information provided by the previous processing blocks. Therefore, from the first approach, the reduced number of movement models avoiding the impact of the non-participating models on the overall performance is picked up. While from the second approach, the better matching of each of the listed models to the actual target movement is favoured.

The way of merging these two otherwise incompatible approaches is by performing a context-based selection of the two models on which the final state estimation will be based. This selection will aim at choosing the models that better match the expected behaviour of the target from the full list of available aircraft movement models. There will be one selected model for addressing non-maneuvring situations, and another for manoeuvring ones, so the number is kept limited.

Furthermore, the objective is also to take advantage of the context information to help in the state estimation. This is done by incorporating information from:

- Aircraft, such as the wheelbase, the location of the antennae or its physical dimensions.
- Airport, such as the proximity to a turn and its characteristics (e.g. turn radius), the presence of a holding point or the location into a straight segment.
- Sensor, such as the local tracking.

Some of this information is used for:

- Tuning the mode-matched filters and smoothers (e.g. aircraft wheelbase).
- Identifying the presence of a longitudinal or transversal manoeuvre (e.g. sensor local tracking and airport layout).
- Finding locations where the targets might stop for a while (e.g. the proximity to airport holding points).

It is worth noting that some of the movement models can be used for modelling several aircraft manoeuvres just by modifying the underlying parameters usually linked to the accelerations that the target might be suffering and which impacts on the process covariance matrix. This is similar to what was mentioned concerning the constant or nearly constant velocity model.

Finally, the novelty within the selection of the movement models is the use of the three-wheel model (ref. 7.4.6.3.6), which is also used in state-of-the-art tracking systems like ARTAS [Van Hintum 13], but also incorporating the exact information concerning the aircraft wheelbase.

7.8.3.2 Model-matched filters

Considering that the objective is always to keep a mode for manoeuvres and another one for non-maneuvring situations, the filter types may be different. The choice of the movement models is the following one:

- Non-maneuvring targets are:
 - Standing targets.
 - Targets moving at a constant speed and with a constant heading.
- Manoeuvring targets are:
 - Accelerating or decelerating targets with a constant heading.
 - Turning targets at a constant turn rate.
 - Turning target at a variable turn rate due to a changing ground speed.

For modelling non-maneuvring targets, two models are selected, the stationary or stopped for standing targets, and the constant or nearly constant model for the targets moving at a constant speed and with a constant heading.

On the other hand, the manoeuvring targets are modelled using the constant or nearly constant velocity model in case there is a longitudinal acceleration by just enlarging the process noise covariance matrix. However, for turning targets, the model depends on the target type, i.e. whether it is an aircraft or a ground vehicle, and in the case of an aircraft, on the availability of the wheelbase. Consequently, if the wheelbase is known, the movement model that is used is the one based on the three-wheel movement (ref. 7.4.6.3.6). Otherwise, the turn rate estimation model (ref. 7.4.6.3.5). Both models can be tuned to cope with longitudinal accelerations by enlarging the corresponding components of the process noise matrix.

Once the movement models are chosen, then there is a need for selecting the state estimator that is going to be used within the overall multiple-model smoothing approach. As already mentioned, the overall approach is based on a IMM forward filter-IMM backward smoother method which means that there is a need for selecting the modes that are going to be part of the IMM forward filter and that will likewise determine the active smoothers that will form the IMM backward smoother.

Among the aforementioned movement models, there are nonlinear models and so, they must be implemented by a nonlinear estimator such as the EKF or the UKF. In this work, the decision has been to count on a UKF mainly because of its good features when dealing with nonlinear models and because it was a less familiar filter to the author. The details of the UKF filter can be found in ANNEX I, section A.6.

The strategy that has been followed is to use the UKF formulation of the three-wheel movement model for all the mode-matched filters but removing all the unnecessary processing. This is done by reducing the size of the internal state vector and the corresponding covariance matrix. By doing this, the appearance of close to singular

covariance matrices while performing the unscented transform is prevented avoiding this way instabilities on the filter output.

Therefore, the particularities of the selectable mode-matched filters (and smoothers) are:

7.8.3.2.1 UKF for standing targets

This UKF filter (UKF-ST) implements the stationary or stopped model (ref. 7.4.6.3.1) with the following state vector:

$$\underline{x}(k) = [x \quad y \quad \theta]^T \quad (7.137)$$

With a process covariance matrix that is defined by:

$$Q_{ST} = \begin{bmatrix} \sigma_x^2 & 0 & 0 \\ 0 & \sigma_y^2 & 0 \\ 0 & 0 & \sigma_\theta^2 \end{bmatrix} \quad (7.138)$$

This is one of the filters considered for non-manoeuving targets when the ground speed of the target is below a certain threshold, that is, $V \leq V_{Th_2}$.

7.8.3.2.2 UKF for uniform motion

This UKF filter (UKF-UM) implements the constant or nearly constant velocity model (ref. 7.4.6.3.2) with the following state vector:

$$\underline{x}(k) = [x \quad y \quad V \quad \theta]^T \quad (7.139)$$

With a process covariance matrix that is defined by:

$$Q_{UM} = G_p^* \begin{bmatrix} \sigma_{a_t}^2 & 0 \\ 0 & \sigma_a^2 \end{bmatrix} G_p^{*T} \quad (7.140)$$

With:

- G_p^* : projection matrix as defined in section 7.4.6.3.5 without the last row.
- $\sigma_{a_t}^2$: variance of the tangential acceleration.
- σ_a^2 : variance of the angular acceleration.

This is the other filter that is considered for non-manoeuving targets whenever the ground speed of the target is above a certain threshold, that is, $V \geq V_{Th_1}$. The model considers a small σ_a^2 to cope with slight variations in the assumed constant heading.

7.8.3.2.3 UKF for uniform acceleration

This UKF filter (UKF-UA) relies on the constant or nearly constant velocity model (ref. 7.4.6.3.2) for estimating the state of a uniformly accelerated target. As before, the state vector looks like:

$$\underline{x}(k) = [x \quad y \quad V \quad \theta]^T \quad (7.141)$$

With a process covariance matrix that is enlarged to cope with the effect of a constant longitudinal acceleration:

$$Q_{UA} = FACTOR_{UA} \cdot Q_{UM} \quad (7.142)$$

This is one of the filters that is considered for a potential manoeuvring target whenever it is considered that the target could be in a longitudinal manoeuvring area.

7.8.3.2.4 UKF for uniform circular motion without wheelbase

This UKF filter (UKF-UC) relies on the turn rate estimation model (ref. 7.4.6.3.5) for estimating the state of a target that is turning at a constant ground speed. The target might be an aircraft without a known wheelbase or a ground vehicle. In this case, the state vector is enlarged as shown hereafter:

$$\underline{x}(k) = [x \quad y \quad V \quad \theta \quad \omega]^T \quad (7.143)$$

With a process covariance matrix that can be expressed as:

$$Q_{UC} = G_p \begin{bmatrix} \sigma_{a_t}^2 & 0 \\ 0 & \sigma_{\alpha}^2 \end{bmatrix} G_p^T \quad (7.144)$$

With:

- G_p : projection matrix as defined in section 7.4.6.3.5.
- $\sigma_{a_t}^2$: variance of the tangential acceleration.
- σ_{α}^2 : variance of the angular acceleration.

This is other filter that is considered for a potential manoeuvring target whenever it is considered that the target could be in a transversal manoeuvring area. The model considers a small $\sigma_{a_t}^2$ to cope with slight variations in the assumed constant ground speed.

7.8.3.2.5 UKF for uniform circular motion with wheelbase

This UKF filter (UKF-3W) relies on the three-wheel motion model (ref. 7.4.6.3.6) for estimating the state of a target that is turning at a constant ground speed. The target must be an aircraft with a known wheelbase. In this case, the state vector is slightly modified:

$$\underline{x}(k) = [x \quad y \quad V \quad \theta \quad \delta]^T \quad (7.145)$$

With a process covariance matrix that can be expressed as:

$$Q_{3W} = G_{3W} \begin{bmatrix} \sigma_{a_t}^2 & 0 \\ 0 & \sigma_{\alpha}^2 \end{bmatrix} G_{3W}^T \quad (7.146)$$

With:

- G_{3W} : projection matrix defined in section 7.4.6.3.6.
- $\sigma_{a_t}^2$: variance of the tangential acceleration.

- σ_{α}^2 : variance of the angular acceleration.

This is other filter that is considered for a potential manoeuvring target whenever it is considered that the target could be in a transversal manoeuvring area. The model considers a small $\sigma_{a_t}^2$ to cope with slight variation in the assumed constant ground speed.

7.8.3.2.6 UKF for accelerated circular motion without wheelbase

This UKF filter (UKF-AC) mimics the same filter as the one described in 7.8.3.2.4 except for a larger $\sigma_{a_t}^2$ value that aims at coping with not negligible tangential accelerations.

7.8.3.2.7 UKF for accelerated circular motion with wheelbase

This UKF filter (UKF-3WA) mimics the same filter as the one described in 7.8.3.2.5 except for a larger $\sigma_{a_t}^2$ value that aims at coping with not negligible tangential accelerations.

7.8.3.3 Model-matched smoothers

For each of the UKF filters presented in the previous section, it is possible to obtain its unscented Kalman smoother (UKS) counterpart. In this work, the UKS that are used in the overall smoothing approach are based on the results derived from [Sarkka 08].

7.8.3.4 Overall multiple-model smoothing approach

Finally, the overall multiple-model smoothing approach is presented. The implementation that has been used in this work relies in the formulae derived in [Nadarajah 12], which present a smoothing method based on a forward filter and a backward smoothing while maintaining the fundamental spirit of the interacting multiple model (IMM) filters, that is, an IMM-based RTS-type algorithm (ref. 7.4.9.1) which is initially design for fixed-lag smoothing. In our case, these formulae are extended for a fixed-interval smoothing problem as the one that is addressed in this work.

7.8.3.4.1 Forward filtering

The first step in any RTS-type algorithm is to run the forward filter. This proposal relies on an IMM filter that is a well-proven efficient state estimator for tracking manoeuvring targets as the ones that can be found moving around an airport. This filter relies in two or more modes that are designed to match the expected movements that the targets might perform. The details of the IMM filter can be found in ANNEX I section A.7.

The approach that is followed in this thesis is based on exploiting some prior knowledge or context information to prevent having modes that do not participate in the target movement but can negatively interfere in the state estimation. This information has already been presented in section 7.8.3.1 and a high-level view on how this information is used is presented in Table 21:

Table 21: Mode-matched filter selection based on context and prior information

IMM Mode	Target ground speed	Location in manoeuvring areas			
		Not in manoeuvring areas	In a transversal manoeuvring area	In a longitudinal manoeuvring area	In a longitudinal and transversal manoeuvring area
Non-manoevring	$V \geq V_{Th_1}$	UKF-UM	UKF-UM	UKF-UM	UKF-UM
Manoeuvring		UKF-UA	UKF-UC or UKF-3W	UKF-UA	UKF-AC or UKF-3WA
Non-manoevring	$V \leq V_{Th_2}$	UKF-ST	UKF-ST	UKF-ST	UKF-ST
Manoeuvring		UKF-UA	UKF-UC or UKF-3W	UKF-UA	UKF-AC or UKF-3WA

Therefore, every time the IMM filter process a measurement, the set of mode-matched filters is selected based on this information. This means introducing an extra logic to correctly manage the mode probabilities although in this work this management is kept to a minimum by grouping the mode-matched filters into two modes: the manoeuvring one and the non-manoevring one. This simplifies the processing while not introducing any noticeable negative effect.

Then, the data follow the regular IMM update cycle, which can also be found in references like [Li 93]. From this step, both, the output of the IMM filter and the output of each of the mode-matched filters, are stored, as this information will be needed by the mode-matched backward smoothers and the IMM backward smoother. Consequently, from the IMM filter, the combined state vector and its covariance matrix are kept, which are represented by:

$$\hat{x}(k|k) \quad \text{and} \quad P(k|k) \quad (7.147)$$

The output of the j -th mode-matched filter, that is, its state vector and its covariance matrix are:

$$\hat{x}^{0j}(k|k) \quad \text{and} \quad P^{0j}(k|k) \quad (7.148)$$

The initial values and the probability transition matrix are presented in ANNEX I section A.7.

7.8.3.4.2 Backward smoothing

Once the forward IMM filter is run, then all the forward information is available and the backward smoothing can start. The following steps are presented hereafter following a similar presentation as the one done for the IMM in ANNEX I section A.7.

Interaction (mixing)

The first step of the backward smoothing consists in the computation of the inputs that will enter each of the mode-matched smoothers that take part in the overall backward IMM smoothing process.

Thus, the mixed initial smoothed state $\hat{x}^{0j}(k+1|N)$ and covariance matrix $P^{0j}(k+1|N)$ for the j -th mode-matched smoother are obtained by combining the last outputs of the mode-matched smoothers calculated in the previous update time $k+1$. For the j -th mode-matched smoother, the output is composed by its smoothed state $\hat{x}^j(k+1|N)$ and its covariance matrix $P^j(k+1|N)$.

The mixed initial state is given by:

$$\hat{x}^{0j}(k+1|N) = \sum_{i=1}^M \mu_{i|j}(k+1|N) \hat{x}^i(k+1|N) \quad (7.149)$$

Where:

- $\mu_{i|j}(k+1|N)$: backward mixing probability, which is given by:

$$\mu_{i|j}(k+1|N) = \frac{1}{d_j} b_{ij} \mu_i(k+1|N) \quad i, j = 1, \dots, M \quad (7.150)$$

With:

- d_j : predicted backward mode probability.

$$d_j = \sum_{l=1}^M b_{lj} \mu_l(k+1|N) \quad (7.151)$$

- b_{ij} : backward transition probability for going from mode i to mode j .

$$b_{ij} = \frac{1}{e_i} \pi_{ji} \mu_i(k|k) \quad i, j = 1, \dots, M \quad (7.152)$$

Being:

- e_i : backward a priori probability.

$$e_i = \sum_{l=1}^M \pi_{li} \mu_l(k|k) \quad (7.153)$$

- π_{ji} : forward transition probability for going from mode j to mode i .
- $\mu_i(k|k)$: forward probability of the i -th mode at time k .

- $\mu_i(k+1|N)$: backward mode probability for mode i at time $k+1$.

The initial covariance matrix is given by:

$$\begin{aligned} P^{0j}(k+1|N) = & \sum_{i=1}^M \mu_{i|j}(k+1|N) \left\{ P^i(k+1|N) \right. \\ & \left. + [\hat{x}^i(k+1|N) - \hat{x}^{0j}(k+1|N)][\hat{x}^i(k+1|N) - \hat{x}^{0j}(k+1|N)]^T \right\} \end{aligned} \quad (7.154)$$

Mode-matched smoothing

The next step consists in smoothing each of the previously calculated initial inputs by means of its mode-matched smoother. Consequently, for a generic smoother " j " the equations for the smoothed mode-conditioned mean and covariance matrix would be:

$$\hat{x}_j(k|N) = \hat{x}_j(k|k) + A_j(k|N) \left(\hat{x}_{0j}(k+1|N) - \hat{x}_j(k+1|k) \right) \quad (7.155)$$

$$P_j(k|N) = P_j(k|k) - A_j(k|N) \left(P_{0j}(k+1|N) - P_j(k+1|k) \right) A_j^T(k|N) \quad (7.156)$$

Where:

- $\hat{x}_j(k|k)$: forward state for the j -th mode-matched filter at time k .

- $A_j(k|N)$: smoothing gain for the j -th mode-matched smoother.

$$A_j(k|N) = P_j(k|k)F_j^T(k)P_j^{-1}(k+1|k) \quad (7.157)$$

- $\hat{x}_j(k+1|k)$: predicted mean corresponding to the j -th mode-matched filter.
- $P_j(k+1|k)$: predicted covariance corresponding to the j -th mode-matched filter.

Mode probability update

The following step computes the smoothed mode probability for the j -th mode-matched smoother:

$$\mu_j(k|N) = \frac{\Lambda_j(k|N)\mu_j(k|k)}{\sum_i^M \Lambda_i(k|N)\mu_i(k|k)} \quad (7.158)$$

Where:

- $\Lambda_j(k|N)$: likelihood function that takes the form:

$$\Lambda_j(k|N) = \sum_{l=1}^M \pi_{li} \mathcal{N}(\hat{x}_j(k+1|N); \hat{x}_j(k+1|k), P_j(k+1|k)) \quad (7.159)$$

Estimate and covariance combination

Finally, the output of the IMM backward smoother is computed taking into consideration all the previous results. Thus, the estimate and covariance matrix look like:

$$\hat{x}(k|N) = \sum_{i=1}^M \mu_i(k|N)\hat{x}^i(k|N) \quad (7.160)$$

$$P(k|N) = \sum_{i=1}^M \mu_i(k|N) \left\{ P^i(k|N) + [\hat{x}^i(k|N) - \hat{x}(k|N)][\hat{x}^i(k|N) - \hat{x}(k|N)]^T \right\} \quad (7.161)$$

This is the final state estimate and its covariance matrix provided by the IMM forward filter-IMM backward smoother, which will be kept as the ideal position of the target at the time of the k -th measurement.

7.8.3.4.3 Overall smoothing architecture

This section tries to graphically represent the already cumbersome algorithms that have just been presented:

Figure 7.19: Detail of the IMM backward smoothing step for the k -th measurement. Source: own elaboration.

7.9 False target report analysis

7.9.1 Process description

The “false target report analysis” process aims at identifying the data elements associated to the selected multisensor associated trajectories which are likely to be a false target report, that is, they do not come from an actual target or ground vehicle. Selected multisensor reconstructed trajectories are also subject of this analysis as they might come from false target reports. In both cases, this process tries to also classify this false target report or false tracks into different categories, which might depend on the sensor. Consequently, radar might present false plots which are second turn around measurements, sidelobes, reflections, splits, etc. For false tracks, there might be a similar classification, usually depending on the false plots that originated them or at least contributed to their generation.

7.9.2 Processing details

7.9.2.1 Input data

The input is the set of selected multisensor associated or even reconstructed trajectories with their associated data elements and additionally, any un-associated plot or target report.

7.9.2.2 Output data

The output is the same set of selected multisensor trajectories that were received as input. However, now some of the data elements that are part of them might be classified as a false target report. Even more, some of the selected multisensor trajectories might be identified and classified as a certain type of false track.

7.9.2.3 Block diagram

The “false target report analysis” process can be graphically located in the “trajectory reconstruction” processing block, as it is shown in the figure below:

Figure 7.20: Block diagram focussed on the “false target report analysis” process. Source: own elaboration.

7.9.3 Prototype implementation details

This process is outside the scope of the prototype and therefore it has not been implemented.

7.10 Information completion

7.10.1 Process description

Once the trajectories are reconstructed, it remains to confirm that the generated information is complete. This is the target of the “information completion” process. So, depending on the sensors providing the input, there might be information that is useful to generate to ease the subsequent assessment. It is worth noting that the reconstructed information is provided in the system coordinate frame, which do not share the same coordinate system as the one used by the sensors. For example, if the information of a sensor is provided in geodetic (e.g. ADS-B or MLAT system), it is worth converting the reconstructed trajectory updates into this coordinate system to compare both pieces of information in the sensor coordinate frame. Something similar happens when the sensor provides the height information in a different type of height source (e.g. from a barometric source like the mode C, from geometric height measurements like the ones provided by a 3D radar or a MLAT system, etc.) than the one generated in the reconstruction process.

The same occurs when dealing with the output of tracking systems, as their system coordinates might differ from the ones used by the PET or, for example, they might incorporate more height sources to provide the height information. Thus, this information may come from internal computations at the tracker side, assumptions based on the location of the trajectory or a default height, if there is no better estimate.

Finally, there could be also some identity or extra information such as ADD or DAP that are missing and that can be completed with the information from target reports in the surroundings of the data to be completed. It is crucial to take care when completing the reconstructed information in order not to introduced wrong information or outdated one. So, if some of the information to be included has for example an estimate of its age, then this value should also be taken into consideration.

7.10.2 Processing details

7.10.2.1 Input data

The input is the set of selected multisensor reconstructed trajectories.

7.10.2.2 Output data

The output is the same set of selected multisensor reconstructed trajectories that were received as input. However, they are now completed with additional information that can improve or ease the further assessment to be performed by the “performance evaluation” processing block.

7.10.2.3 Block diagram

The “information completion” process can be graphically located in the “trajectory reconstruction” processing block, as it is shown in the figure below:

Figure 7.21: Block diagram focussed on the “information completion” process. Source: own elaboration.

7.10.3 Prototype implementation details

The prototype partially implements this process to be ready for a comparison in the sensor coordinate framework.

7.11 Trajectory post-processes

7.11.1 Process description

The last process of the “trajectory reconstruction” processing block aims at improving some of the already performed activities, like the movement model identification, once the reconstructed information is available. However, there are other activities that can now be performed, taking into consideration that the full information that can be extracted from sensors should be now integrated. Some activities, in line with the ones described in [Besada 13], are now scheduled:

- **Aircraft classification:** it relies on the calculated kinematic values and identified manoeuvres together with the other trajectory information for classifying the target as belonging to any of the already defined aircraft or ground vehicle types. This classification can partially cover the one reported in I062/300 (fleet identification) or in I062/380 #21 (emitter category).
- **Reconstructed trajectory reconnection:** until now, the PET has recreated what it is believed to be the ideal trajectory of the target reports which are associated together. However, there are many situations in which, due to occlusions, RF interferences or human mistakes, target reports are lost and, consequently, the trajectory performed by the target is divided into several not connected pieces. Depending on the length of these gaps and how the data association is carried out

by the PET, there might be a need for reconnecting these pieces of the trajectory together and group them into a single data structure. This is done by implementing some compatibility checks like the ones proposed in this text for performing the data association but now relying on reconstructed data. Care must be taken in order to avoid the reconnection of pieces of trajectory belonging to different targets.

- **Association with system tracks:** another activity that can be performed at this point is first, the association and then, the comparison with tracks coming from surface tracking systems. This comparison enables the assessment of the tracker performances, which will be carried out by the following processing block, i.e. the “performance evaluation”.
- **Trajectory enrichment with information from other data sources:** as already mentioned in CHAPTER 3, there are several strategies for assessing the performance of a sensor or a tracking systems. In this thesis, the focus is on reconstructing the reference trajectory based on opportunity traffic, however there are other such as simulated trajectories or the ones coming from high-precision location systems (e.g. from a D-GPS receiver). In this moment, it is possible to incorporate such information, which usually refers to a reduced set of trajectories, to improve the performance evaluation of the system under assessment.

After all this processing, it can be considered that all the needed information, for carrying out the performance evaluation of the sensors and tracking systems that take part in the surveillance chain, is already available and ready to be used by the “performance evaluation” processing block.

7.11.2 Processing details

7.11.2.1 Input data

The input is the set of selected multisensor reconstructed trajectories that have been completed by the previous process.

7.11.2.2 Output data

The output is a refined set of multisensor reconstructed trajectories that now incorporate classification information, association to system tracks and that might be enriched with information coming from another source of trajectory data.

It is worth noting that the number of trajectories may vary in case it is found that some of them belong to the same target and, as such, they are grouped together.

7.11.2.3 Block diagram

The “trajectory post-processes” can be graphically located in the “trajectory reconstruction” processing block, as it is shown in the figure below:

Figure 7.22: Block diagram focussed on the “trajectory post processes”. Source: own elaboration.

7.11.3 Prototype implementation details

The prototype only implements the association with system tracks as it allows to close the circle of the performance evaluation paradigm from opportunity traffic. The rest of the functionalities are considered out of the scope of this work.

CHAPTER 8

PERFORMANCE EVALUATION PROCESS RESULTS

“Excellence doesn't mean we have to be 100% better in any one thing: it means we strive to be 1% better in 100 things.”

— Jan Carlzon.

After having followed all the steps identified in the performance evaluation process, having gathered all the available data and having cleaned, prepared and processed them with the designed algorithms, we reach the moment of truth, i.e. the moment when the results are assessed to be fit for purpose. This is achieved by following two approaches, one qualitative and another one more quantitative.

The qualitative approach shows that, at first glance, the outcome of the performance evaluation process performs well. This means that the “data reading” processing block correctly imports the input, the “data preparation” selects the right data, associates them per target and aligns them before letting them transition to the “trajectory reconstruction” processing block. Conversely, the quantitative approach is focussed on obtaining performance indicators that allow to compare the PET outcome with different parameter configurations of itself and with the outcomes obtained by other algorithms that represent other design choices, which can be taken, or even with other tools.

In this chapter, the proposed process description, designed architecture, and estimation algorithms are instantiated in a particular implementation that aims at validating whether they are accurate and reliable enough to be used as the reference trajectory provider for a future PET that can be used in an ATC centre or tower to evaluate the performance of the surveillance chain.

8.1 Introduction

At this point, there is something that is worth mentioning which is that there is only one processing block in the proposed performance evaluation overall process that has not been described in a dedicated chapter, i.e. the “performance evaluation”, not because it is not relevant but because it is considered out of the scope of this research work. However, it contributes considerably to the final impression that the user will get from the tool. The way data is presented may change completely the user perception and understanding of the displayed information making it a relevant added value. Hence, the PET users will appreciate not only the accuracy and reliability of the provided results but also the way they visualize and understand the information that is generated. However, as it was shown in section 3.3.2, there are already several tools which are mainly focussed on these visualization and presentation aspects of the results reducing the relevance of such topic within this work.

Coming back to the purpose of this chapter, it is devoted to the validation of the proposed overall process and high-level architecture with the focus placed on the improvements provided by the new algorithms that are derived for path recovery, bias estimation, and trajectory reconstruction.

Consequently, these improvements go in the direction of extending the current use of the airport layout and the aircraft available information for modifying existing technologies, which are upgraded for coping with the particularities of the airport environment and the actual aircraft that are moving around.

The evaluation is supported by a real scenario, which is fed on one side with real data and on the other one with synthetic data. The details of such scenario will be presented hereafter.

8.2 Research Platform Description

The research platform has been described all along this document as it represents the instantiation performed by the author of the proposed performance evaluation overall process and the high-level architecture. It implements the designed algorithms that based on context information, such as the airport layout or the aircraft features, estimate sensor biases, identify the followed path and reconstruct the trajectory followed by the target.

The platform is implemented in MATLAB, not only taking advantage of the object-oriented capabilities that this programming language offers, but also exploiting the desktop environment that it provides, which facilitates the display of the processed data. Hence, the platform relies on this capability to offer some visualization means to follow the evolution of the processing that the surveillance data undergo.

This platform is fully functional although it does not implement all the processes that are described all along this work as not all the topics are considered as in the scope of this thesis. This is mainly due to two reasons, either they do not imply a research activity in the data processing field, or they do not represent a core step within the overall process. However, progressing on some of the not developed processes might be of interest for improving and completing the functionalities of a PET and, as such, they are mentioned as potential future research lines in CHAPTER 9.

Furthermore, the platform also implements some performance criteria for its own performance evaluation mainly while estimating biases and reconstructing trajectories.

8.3 Validation approach

As already mentioned, the criteria that are used for validating the proposed algorithms are based on qualitative and quantitative methods.

Focussing on qualitative methods, these are the options which are considered:

- Visual inspection is used for an overall assessment of the results of the whole process not only at its final stage (i.e. after going through all the implemented processes) but also at intermediate points where the progress of the algorithm can be checked. This can give a quick impression on how the implemented processing is performing.
- Comparison with externally calculated values also provides a good idea on the accuracy of the computed values. This approach is followed, for example, when evaluating the estimated speed value, which is compared with the mean value obtained from surveillance data (e.g. from SMR data), or the steering angle.

Moreover, quantitative criteria are used for computing performance indicators that can be used for comparing different algorithms in the same conditions, so decisions concerning which one performs better can be taken. Some measurements that are representative of how well the overall process is performed are:

- Mean value of the distance to the centre line of the segments where the target reports and system track updates, on one side, and the reconstructed trajectory samples, on the other, are allocated. This measurement represents one of the performance indicators that are checked while validating the proposed reconstruction algorithm mainly in non-transversal manoeuvring areas.
- Difference with respect to map values can be used for assessing the accuracy of estimated magnitudes like the heading, which can be compared with airport segment orientation, when the aircraft is located in straight segments.
- Root mean square (RMS) error. This is an indicator that can be used in two different situation, either when the distance from the measurements is compared with the extrapolated value or, in the more general case, when trajectories are created synthetically, and Montecarlo simulations are run in order to compute the RMS error of the reconstructed trajectory and also obtain the confidence interval of the final estimations.

Based on this combination of criteria, the results are compared in order to evaluated their performance.

8.4 Validation framework

8.4.1 Real Traffic Scenario: Lisbon Airport

8.4.1.1 Initial configuration

The real scenario that is used for the evaluation of the performance of the complete prototype is based on Lisbon airport (ICAO code: LPPT). The airport information that is needed for the computations includes the Airport Reference Point (ARP) or the airport map.

8.4.1.1.1 ARP

The ARP is the designated geographical location of an aerodrome and it provides a reference in the airport grounds that is used for coordinate conversion purposes mainly. This piece of information can be found in the AIP of the airport **¡Error! No se encuentra el origen de la referencia.** and it is presented hereafter:

Table 22: Airport information

	Lisbon airport (LPPT)		
	Latitude	Longitude	Height
ARP	38° 46' 27.00" N	09° 08' 03.00" W	114.0m

8.4.1.1.2 Airport map

The airport map information is obtained from the website <https://overpass-turbo.eu/>, which by means of an “Overpass query” extracts, from the maps available in www.openstreetmap.org, the data related to the airport like its runways, taxiways, stands, etc. in a .json file. This “export” functionality gives a list of elements that can be found in a certain area, filtered by the appropriate criteria. These elements are classified as:

- Nodes.
- Ways.

Which are then further tagged as:

- Taxiways.
- Runways.
- Apron.
- Gates.
- Parking positions.
- Buildings (e.g. terminal, hangar).

Then, the only thing that is needed is a parser that is able to extract the airport nodes and the airport segments. The nodes in the .json file are defined in geodetic coordinates while the ways provide the information concerning the succession of interconnected nodes that build up a taxiway or a runway. The next step is projecting the nodes on the stereographic plane centred on the ARP. Once this is done, the interconnection information is used to identify the single segments that build up the airport map. The result is then stored as several MS Excel files (i.e. .xls). Usually, one for the nodes (x and y projected coordinates) and another one for the segments (origin and destination nodes). This information is the one used by the prototype to build up the airport map. The result is the one presented in the following picture:

Figure 8.1: Airport map and sensor location

Figure 8.1 also presents the location of the sensors that are going to be characterized hereafter. Their position is presented in the following table:

Table 23: Sensor information

	Data format	Identification		Location		
	ASTERIX category	SAC	SIC	Latitude	Longitude	Height
SMR-Lisboa	Cat.010	104	30	38° 46' 45.816" N	09° 08' 22.729" W	0.0m
SSR-Lisboa	Cat.001	104	2	38° 45' 50.050" N	09° 08' 20.010" W	138.0m
WAM-Lisboa	Cat.020	104	32	38° 45' 49.000" N	09° 08' 22.000" W	114.0m
ADS-B	Cat.021	104	42	00° 00' 00.000" N	00° 00' 00.000" W	0.0m

8.4.1.1.3 Sensor measurement models

The measurement model that is used for each of the sensors is provided in the following tables. They present the noise standard deviation (STD) that is assumed for each of them. The first ones that are presented are the noise and biases related to radar sensors.

Table 24: Radar measurement model parameters

	Noise				Biases				
	Time STD	Range STD	Azimuth STD	Height STD	Time	Range	Range gain	Azimuth	Height
SMR	0.0s	5.0m	0.15°	0.0m	0.0s	0.0m	N.A.	0.0°	0.0m
SSR	0.0s	80.0m	0.5°	0.0m	0.0s	0.0m	0.0ppm	0.0°	0.0m

It is worth noting that the SSR is a quite noisy sensor if compared with the demanding performance requirements for surface surveillance. However, it is included because it can contribute to the transition between airport surveillance and TMA, which is not directly tackled within this thesis but pointed out as a future research line together with the incorporation of the height management (ref. 9.3).

A sensor that is nowadays taking a great importance in the airport domain is the MLAT. Table 25 presents the assumed noise values that are used in case the sensor does not provide the uncertainty of its measurements, which is the regular situation. As already mentioned in section 2.4.1.3, this information can be provided either in I020/500 or in I020/REF (PA).

Table 25: Multilateration measurement model parameters

	Noise				Biases			
	Time STD	Δx STD	Δy STD	Δz STD	Time	Δx	Δy	Δz
MLAT-Lisboa	0.0s	20.0m	20.0m	20.0m	0.0s	0.0m	0.0m	0.0m

Finally, the ADS-B sensor measurement model is presented. As it can be seen, it is considered that as an unbiased sensor except for the presence of a time bias which copes with synchronization issues among the different data sources.

Table 26: ADS-B measurement model parameters

	Noise				Biases
	Time STD	Δx STD	Δy STD	Δz STD	Time
ADS-B	0.0s	10.0m	10.0m	10.0m	0.0s

By default, all sensors are considered as unbiased as it can be appreciated from the tables. This allows to run the bias estimation algorithms from scratch to get a first estimation. However, if biases are known by the user, they can be provided to the PET and it will directly correct the incoming data before they are processed. This will lead the PET bias

estimation algorithms to compute not the absolute bias correction but a relative one to the one provided by the user.

The real data covers a duration of four hours but for the purpose of our validation the selected time will be shorten, usually to ten (i.e. 600s) or fifteen (i.e. 900s) minutes.

8.4.1.2 Data reading

This section aims at presenting the results of the “data reading” processing block, so the output of the “data presentation” process is provided. This is a point in the data processing in which a qualitative analysis can be performed.

As it can be appreciated in the following pictures, the incoming data is properly acquired, cleaned, and allocated to its generating source. In fact, the “data presentation” goes through all the generating sources (i.e. sensors) plotting the information allocated to them, so it knows how to handle this information and how to display it.

8.4.1.2.1 SMR plots

The next figure shows the result of the SMR data acquisition from Cartesian coordinates delivered in I010/042. This information represents the output of a sensor local tracker that provides not only updates, which are shown in red, but also extrapolations, which are shown in a different colour (i.e. orange). Updates are generated based on received measurements, while extrapolations are generated without measurement support based on the belief that measurements should have arrived.

Figure 8.2: SMR local track updates imported by the PET

At first glance, it looks like quite acceptable. However, when we pay a closer look on the data, see Figure 8.3, we can appreciate that data present erratic behaviours describing straight line trajectories in places where there are no taxiway, runway, or apron area. This fact points to extrapolations done using a uniform motion model whenever the measurements disappear. This clearly affects data accuracy and reliability, which becomes more than questionable. Additionally, these local track updates come without any estimation of its uncertainty, which prevents their further processing in a data fusion process.

Figure 8.3: Detail of the extrapolations performed by the SMR local tracker

Consequently, these local extrapolations are removed from the following processing, so only detections will be used. These detections are provided in polar coordinates (I010/040) and, as such, they need to be converted into the system coordinate framework.

Figure 8.4: Detections provided by the SMR sensor

Figure 8.4 shows the result when only detections are displayed. It can be seen that the strange straight-line extrapolations are not present, and the data is roughly found in the airport areas where they should be. This figure also highlights the fact that the SMR is a pure surface surveillance sensor, not covering neither the gliding path nor the take-off segment. As it will be shown in the following sections (ref. 8.4.1.3.6), when this recording was done, the SMR sensor was clearly biased.

8.4.1.2.2 MLAT target reports

The input from the MLAT sensor can be received either in Cartesian coordinates (I020/042) or in geodetic ones (I020/041). The prototype checks that the results are barely the same when projected in the system coordinates, so only the output in Cartesian coordinates is used as it is assumed to be the coordinate framework where the multilateration measurement is computed. Figure 8.5 also shows that the sensor covers not only the surface area but also the gliding path and the take-off segment. This fact makes the MLAT sensor an interesting one for improving the continuity between the surface surveillance and the air surveillance. Another relevant aspect is that it can help in better estimating any QNH correction (ref. 2.6.4) that could align barometric altitude measurements with geometric heights. However, this is proposed as a future research line (ref. 9.3).

Figure 8.5: MLAT sensor local tracking imported by the PET

Like the SMR, this sensor provides the output of a local tracking system. In fact, it only provides local tracks updates, so no raw measurement is delivered. However, opposite to the SMR, the uncertainty of each track update is statistically characterized by means of its standard deviation values, correlation coefficients or covariance components. These

pieces of information are delivered as part of the ASTERIX records in two different ways, either in I020/500 data items or as part of reserved expansion fields, i.e. I020/REF (PA), as it was presented in section 2.4.1.3. This is not the optimal way as the internal tracking process is unknown, but at least the information can be fused with the one coming from other sensors. A typical issue that is found when processing multilateration data is that the uncertainty values delivered by the sensor are quite optimistic, showing position STD values even below 1m. This is an issue for tracking systems (more notably if the STD estimation is not accurate enough) as they will give way more weight to the measurements coming from the MLAT system than to the others in the data fusion biasing the output towards the MLAT measurements. This is often solved by limiting the uncertainty values from below, for example, by setting a minimum STD value.

Figure 8.6: Detail of the extrapolations performed by the MLAT local tracker

The previous picture shows the effect of some problematic situations at local tracking level. Hence, it is possible to find some erroneous extrapolations of some target tracks, which clearly show impossible movements as they traverse buildings, do not follow taxiways, or represent extremely high speeds.

These drawbacks will need to be identified by the PET in order to remove the affected target reports from the subsequent processing as they will impact the trajectory reconstruction and, as such, the performance evaluation results.

8.4.1.2.3 ADS-B target reports

ADS-B together with MLT sensors represent the state of the art concerning surface surveillance sensors. ADS-B target reports provide position information in geodetic coordinates (I021/130). These target reports cover the surface area, the gliding path and the take-off segment like in the MLAT sensor case. Hence, they can also contribute to the QNH correction mentioned in the previous section. The uncertainty and integrity of the provided measurements depend on the transponder MOPS version and on the quality indicators that characterize this information (e.g. NIC, NUCp, PIC, etc.) [ICAO 9871].

Figure 8.7: ADS-B measurements imported by the PET

As already mentioned in section 2.3.2, ADS-B relies on airborne equipment (i.e. GNSS receiver and/or INS) to obtain the kinematics of the aircraft. It is well known that GNSS receivers are noisier than INS, but the latter can develop larger biases.

Figure 8.8: Detail of the ADS-B measurements

The use of different navigation systems can explain some of the erratic behaviours that target reports perform in this scenario like the ones highlighted in red in the previous

picture. This is usually linked to older (e.g. MOPS v0) or faulty transponders, which need to be reported to be filtered out from the performance evaluation. This is managed nowadays by black and white lists. EUROCONTROL initiatives, such as the creation of the “non-performing ADS-B transponder list” (NPAL) or the “surveillance radio frequency and avionics anomalies” (SURFAR) stakeholder group [SAFPA], go in this direction.

Figure 8.9: Quantization effect in ADS-B measurements

The previous figure highlights two different features of ADS-B measurements. On one hand, it brings to our attention the effect of quantization and, on the other, it shows pieces of trajectories which seems to be extrapolations. The quantization error can also explain the latter, but it might also come from an extrapolation that is done by the navigation system that might extrapolate the forwarded information to a certain time instant. These measurements are correlated measurements, and they should be removed or treated in a particular way by the PET.

8.4.1.2.4 SSR plots

SSR radars are sensors that can partially cover the airport. These are sensors that can also provide both, measurements and track updates. Measurements are provided in slant polar coordinates in I001/040 or I048/040, while local track updates are delivered in Cartesian coordinates in I001/042 or I048/042. Track updates and extrapolations are presented in Figure 8.10 using different colours as it was previously done for the SMR.

This type of radars are mainly air surveillance sensors although they can cover the approach and the departure to/from the airport. Usually, they barely see targets on the airport surface and, if it is the case, the quality of the position is far from the one provided by the other sensors. Just to recall at this point that the transponder induced bias, which was presented in section 2.2.4.2, introduces a systematic error that might reach several tens of meters. However, together with ADS-B and MLAT sensors, SSR radars can contribute to the transition between air and surface surveillance and can also contribute to the height alignment in the proximity of the airport where this correction is crucial.

Figure 8.10: SSR local track updates imported by the PET

If the focus is placed on the area around the airport (see Figure 8.11), it is possible to appreciate the approach and departure procedures followed by aircraft and some overflights that can be visually identified by the gap in their detections while they cross the cone of silence of the SSR sensor.

Figure 8.11: Detections provided by the SSR sensor in the proximity of the airport

Going down to the airport, there are only few SSR plots on the runway, probably up to/from a certain height from where the sensor starts detecting the targets. This fact, together with the lower accuracy of the measurements and the less frequent update, makes the SSR a not especially interesting sensor for surface surveillance. However, this thesis considers its integration in the overall procedure as a step further for a combined air-surface performance evaluation (ref. 9.3).

Figure 8.12: Detections provided by the SSR sensor around the airport

8.4.1.3 Data preparation

Once the imported data has been shown as properly allocated to the generating source entity by the “data presentation” process, the focus is placed on the processes that implement the contribution of this thesis concerning data pre-processing, that is, the multi-hypothesis path finder algorithm (i.e. “data location in airport” process) and the “bias estimation and cancellation” process, where the complete bias estimation strategy is implemented.

However, for the sake of completeness, the results obtained after every process that has been described all along this work are going to be either displayed or commented.

8.4.1.3.1 Initial data selection

The “initial data selection” process is configured with the following criteria:

- **Data Types:** SMR_Cat10, MLAT_Cat20, ADSB_Cat21, SSR_Plot_Cat01, SSR_Track_Cat01
- **Keep detection only (Radar):** yes
- **Time interval(s):** 43200, 600
- **Maximum Height(ft):** 3000, 3350
- **Selection strategy:**
- **Aircraft Addresses:**
- **3A Modes:**
- **ACIDs:**
- **Track Numbers:**

This means that all the data types that are found in the Lisbon scenario are accepted (e.g. SMR, MLAT, ADS-B and SSR data). Only radar detections are selected, so extrapolations are filtered out. The processed data last 10 minutes (600s) from the beginning of the scenario (43200s since midnight, i.e. 12:00:00 PM) and there is a maximum height filter to remove overflights that filters out any target report with a barometric height above 3000feet or a geometric height above 3350feet.

Concerning identification, there is no active criterion so all the data that pass the previous criteria are kept for further processing.

8.4.1.3.2 Known bias cancellation

Initially, this process is not activated as the intention is to compute the initial sensor-based biases of the contributing sensors. So, the initial bias values are all set to 0.

8.4.1.3.3 Coordinate transformation

All the data displayed in section 8.4.1.2 are now transformed into the system coordinate framework and initially filtered following the criteria specified in section 8.4.1.3.1. The result is presented in the following section showing not only the result of the “coordinate transformation” into the system framework but also the data finally selected.

8.4.1.3.4 Final data selection

The “final data selection” process is configured with the following criteria:

- **Domain of interest(m):** 5000

This means that any target report or track update, which is more than 5000m away from the ARP position, are filtered out. The objective is to only keep the targets which are moving around the airport, the ones approaching and the ones departing. Additionally, it also performs the time filter for the SSR sensor as it needs further processing than the other sensors as the detection time of SSR plots, when provided in Cat.001, is not coming as a ToD but coded in a different way (i.e. as a Truncated ToD). So, there is a need for a kind of ToD reconstruction which cannot be done at the time of the “initial data selection”.

The selected and converted information per sensor is presented in the following figures.

SMR data

The selected SMR measurements in system coordinates are presented hereafter:

Figure 8.13: Selected SMR detections

MLAT data

The selected MLAT track updates in system coordinates are presented hereafter:

Figure 8.14: Selected MLAT track updates

ADS-B data

The selected ADS-B measurements in system coordinates are presented hereafter:

Figure 8.15: Selected ADS-B measurements

SSR data

The selected SSR measurements in system coordinates are presented hereafter:

Figure 8.16: Selected SSR detections

8.4.1.3.5 Time-ordering

Except for the SMR, the rest of the sensors already send the data ordered in time. So, only the SMR data is time ordered.

8.4.1.3.6 Data association

For keeping the extension of this chapter under control, the evaluation of the contributions to this thesis are going to be focussed on three trajectories: a departure, an arrival and a taxiing aircraft going to its stand. First of all, the output of the “data association” process is presented for each of them. This is shown by drawing together all the monosensor tracks that contribute to a certain multisensor one. The data from each sensor is presented with a different colour. Consequently, MLAT data is presented in green, ADS-B is drawn in blue, SMR in red and SSR in cyan as in the previous figures.

Taxiing trajectory

The following trajectory shows an aircraft taxiing towards its stand. It is a trajectory that landed before the start of the recording and ends in a stand.

Figure 8.17: Overall picture of the taxiing trajectory

The following figure provides the details of the transversal manoeuvres that the aircraft performs. It is possible to appreciate that two out of the three sensors contributing to this trajectory provide local tracking information (i.e. the SMR and the MLAT). Something that can also be highlighted is the biased behaviour of the SMR that is clearly visible when approaching the first turn and when leaving the second one. A final aspect that can also be pointed out is that there seems to be a blanking area for the SMR sensor around

the terminal parking area. Therefore, it can be seen that the SMR plots instead of following the MLAT target reports as the aircraft gets closer to the finger, they seem to stay at the border of this potential blanking area.

Figure 8.18: Detailed picture of the taxiing trajectory

Departure trajectory

The following trajectory is a departure from runway 03 that starts from its stand:

Figure 8.19: Overall picture of a departure from runway 03

The following picture shows some interesting aspects concerning this trajectory. The first one is that there are some MLAT extrapolation issues in the beginning of the trajectory as well as some SMR outliers. The second one is the already mentioned biased behaviour of the SMR, which is more noticeable when the plots are further away from the SMR position. Finally, this represents a trajectory which is seen by the four sensors that cover the airport area.

Figure 8.20: Detailed picture of a departure from runway 03

Arrival trajectory

The following trajectory is an arrival to runway 03, which then reaches its stand:

Figure 8.21: Overall picture of an arrival to runway 03

Next, the detail of the turn while leaving the runway is shown, which seems to highlight a noticeable difference in the location of the GNSS antenna (ADS-B detections) with respect to both, the location of the ATC antenna (MLAT detections) and the aircraft centroid which is approximated by the SMR plots.

Figure 8.22: Detailed picture of an arrival to runway 03

8.4.1.3.7 Data location in the airport

This process associates an airport segment to each target report of the multisensor trajectory and computes the continuous path that the target has followed. This continuous path is described by the list of continuous airport segments that the aircraft has gone through. The difficulty of this algorithm comes from the management of the possible hypotheses as it must maintain them under an acceptable number keeping in mind:

- The huge number of small segments that composed the airport map.
- The different noises and biases that affect the sensors.
- The different time synchronization between sensors.
- The feasible ground movements.

Some of these issues can be mitigated with a refined model of the airport map. Therefore, this idea is recorded as one of the possible future activities (ref. 9.3). Substituting series of short turning segments by a circular segment or by an approximating spline (ref. 7.3.6.2) can considerably simplify and ease this process.

The results of this process corresponding to the three reference targets are shown here after.

Taxiing trajectory

This figure shows the complete continuous path followed by the taxiing target:

Figure 8.23: Continuous path of the taxiing trajectory

Some details of the computed continuous path:

Figure 8.24: Details of the continuous path followed by the taxiing aircraft

Departure trajectory

Next figure shows the complete continuous path followed by the departing target:

Figure 8.25: Continuous path of a departure from runway 03

Some details of the computed continuous path:

Figure 8.26: Details of the continuous path followed by the departing aircraft

Arrival trajectory

This figure shows the complete continuous path followed by the landing target:

Figure 8.27: Continuous path of an arrival to runway 03

Some details of the computed continuous path:

Figure 8.28: Details of the continuous path of an arrival to runway 03

Once the continuous path is identified and every single target report is allocated to an airport segment, this process computes a trajectory based on “dynamic” segments, which are based on the airport segments. A “dynamic” segment incorporates not only the basic information of the airport segment (e.g. initial node, final node, or length), but also the ones related to the kinematics of the target. This means:

- Aircraft mean ground speed if an estimate is available.
- Initial node, final node and orientation defined taking into consideration the actual heading of the aircraft.

8.4.1.3.8 Movement model identification

This process classifies each target report of the multisensor associated trajectory as belonging to a manoeuvring area and completes the remaining kinematic information. Therefore, it performs the following tasks:

- Identification of any longitudinal or transversal manoeuvre.
- If there is a transversal manoeuvre, it estimates the turn radius and the turn centre.

This process is a prerequisite for the “trajectory reconstruction” processing block.

8.4.1.3.9 Bias estimation and cancellation

SMR bias correction

The SMR bias correction relies on the airport segment assignment that is done by the “data location in the airport” process to compute the separation distance that is used for the bias estimation algorithm. The convergence of the estimated biases is shown in the following pictures.

Figure 8.29: Convergence of the estimated range bias of the SMR sensor

Figure 8.30: Convergence of the estimated azimuth bias of the SMR sensor

These results are obtained after processing the SMR plots from the three previously presented trajectories that are not located in a transversal manoeuvring area. In order to assess the preciseness of the applied correction, the following pictures provide an overview of the result.

Figure 8.31: Detail of the correction applied to the SMR plots

Figure 8.32: Detail of the correction applied to the SMR plots

Target-based bias correction

The result of the MLAT ATC antenna location bias estimation is presented hereafter:

Figure 8.33: Convergence of the ATC antenna position bias of the MLAT sensor

The estimation that is computed includes both, the difference between the ATC antenna position and the centroid of the aircraft, and any time bias due to a potential lack of synchronization between the time reference used by each of the sensors. Knowing the length of the aircraft, which is provided for all aircraft types in the scenario in ANNEX II, and the position of the ATC antenna, which is provided for an Airbus A320 in ANNEX III, it is possible to estimate the time bias. It is worth noting that in the estimation, the last two measurement differences that are processed present an issue with their covariance

matrices due to the estimated uncertainty of the SMR reconstructed values that take part in these measurement differences. This issue leads to an artificial increase of the weight that is given to both, which biases the result. This is another aspect to be considered as future work, as it deals with the initialization values of reconstructed covariance matrices.

8.4.1.3.10 Noise estimation

The output of this process is out of the scope of the prototype.

8.4.1.3.11 Time re-ordering

There is no need to reorder any data at this point of the processing.

8.4.1.3.12 Association completion

This process is out of the scope of the prototype.

8.4.1.4 Trajectory reconstruction

8.4.1.4.1 Data smoothing

This section presents the results obtained for the taxiing trajectory as it can be assumed a representative case of a manoeuvring target. The results of the backward smoother are compared with the ones from the forward filter for a monosensor reconstruction based on the SMR data which is bias corrected.

Figure 8.34: Detail of the reconstructed information

As it can be appreciated, the performance of the backward smoother is much better than the one of the IMM forward filter, not showing the typical overshooting when entering in manoeuvring areas.

Next figures show the improvement in the mode probabilities:

Figure 8.35: Detail of the correction applied to the SMR plots

As it can be appreciated, these pictures show how the backward filter better identifies the movement model that applies at any moment which is quite interesting as it can be exploited for MoSM classification purposes.

Next, a comparison between the filtered, extrapolated and reconstructed state variables is presented:

Figure 8.36: Comparison of the speed of the IMM forward and the backward smoother

The red values show an estimate per “dynamic” segment obtained from the SMR local tracking information. This estimate is computed as the mean speed value of all the SMR plots that are allocated to each of the “dynamic” segments that build up the trajectory.

Figure 8.37: Comparison of the heading of the IMM forward and the backward smoother

In the previous picture, the reference is taken from the orientation of the “dynamic” segments that build up the trajectory while, in the next one, reference steering angles are computed as the arc tangent of the aircraft wheel base divided by the mean turn radius.

Figure 8.38: Comparison of the steering angle of the IMM forward and the backward smoother

All the previous figures confirm the better performances of the backward smoother compared to the forward IMM.

Finally, the next figure represents the evolution of the separation distance (ref. 6.10.7.2.1) computed for the SMR plots as they go through the different algorithms that are applied. The first picture represents the separation distance of the raw data. As already mentioned before, it clearly shows that these plots are biased. The second picture presents in green the separation distance after bias correction. As it can be seen in the title of the subplot, the mean value of the separation distance is almost 0m, which is quite acceptable result. Please note that only the samples in non-maneuvring segments are considered while computing this indicator. These values are presented in the last picture as black circumferences. The third picture represents in blue the separation distance after the reconstruction and lastly, the remaining picture overlays all the previous plots to better visualize the evolution of the separation distance.

Figure 8.39: Evolution of the separation distance all along the data processing

8.4.1.5 Performance evaluation

The output of this processing block is considered out of the scope of this thesis.

8.5 Performance evaluation results

The results displayed in the previous section show that the allocation of each plot to an airport segment, the bias estimation architecture and the proposed trajectory reconstruction algorithm comply with the expected performance to be part of any PET. Moreover, they show that they are robust and resilient enough to cope with real data that present all kind of errors, that is, not only random ones but also systematic ones. This represents an added value which is of relevance when looking for reliable algorithms for surface trajectory reconstruction.

CHAPTER 9

CONTRIBUTIONS AND FUTURE RESEARCH DIRECTIONS

“Most people don't believe something can happen until it already has. That's not stupidity or weakness, that's just human nature.”

— Max Brooks, *World War Z: An Oral History of the Zombie War*

After having presented the full architecture of a generic PET, having described the performance evaluation process, and having shown the results of the implementation proposed by the author that includes all his contributions, it is the moment for summarizing what has been achieved along this thesis.

The exposition of the state of the art performed in CHAPTER 3 showed that there is a gap concerning the performance evaluation of surface surveillance systems which needs to be filled in. It has been shown that current tools mainly focus on the elaboration of the performance indicators and their presentation in a convenient way to the user either graphically or by means of data files. However, they expect the reference for the assessment to be provided by other external systems. These systems are not usually intended to provide an actual computation of the ideal trajectory as it is the case when using a tracking system as reference provider or when they are thought for individual test purposes, like when using a D-GPS receiver on board a ground vehicle or an aircraft.

The contribution of this thesis goes in the direction of closing the gap left by actual surveillance evaluation tools not only by providing a reference for the performance assessment but going beyond the current paradigm introducing new context information which has not been correctly exploited yet. Additionally, it also points toward interesting and relevant future research areas that have appeared while deriving and implementing the actual contributions and the prototype that instantiates them. These ones are going to be compiled aiming at further research investigations will develop them.

9.1 Current dynamizers of performance evaluation

9.1.1 Performance-based operations

As it was introduced in CHAPTER 1, the ATM domain is moving toward a performance-based paradigm in which services are provided taking into consideration the performance of the target and the performance of the ATM/CNS infrastructure. ICAO introduced the concept “Performance Based Approach” back in 2009 when they produced their document “Manual on Global Performance of the Air Navigation System” [ICAO 9883]. This concept focusses on results which are achieved through the setting of performance objectives and targets, it relies on facts and data for decision making and it states that there is a need for a periodical performance review which requires adequate performance measurement and data collection capabilities.

These considerations clearly point to the need for a PET per domain or field where performance objectives and targets need to be monitored and checked and, clearly, surveillance is one of them. Current surface surveillance tools lack the functionality of periodically generating the reference to be used for the computation of the performance indicators that will highlight whether the objectives and target have been achieved or not. Actually, there is no better and safer way of generating these references than the one based on reconstructing the trajectories from opportunity traffic, which is an information always available and which does not interfere with the activities of the operational system.

This trend is more extended in navigation, where Performance-Based Navigation (PBN) is almost fully implemented. Conversely, the same concept for surveillance, i.e. Performance-Based Surveillance (PBS), and for communication, i.e. Performance-Based Communication (PBC), are not that advanced. In fact, ICAO has recently updated the concept that fuses the last two ones, i.e. Performance-Based Communication and Surveillance (PBCS), in the second edition of the “Performance-based Communication and Surveillance (PBCS) Manual” [ICAO 9869]. This document states that “The PBCS concept provides a framework for managing communication and surveillance performance in accordance with globally accepted RCP and RSP specifications”. The more interesting aspect is that it is explicitly mentioned that this concept is also ready to accept new RCP and RSP specifications for other purposes (ref. 3.2), such as ADS-B.

9.1.2 Gate-to-gate trajectories

Another trend that is here to stay is the G2G operations and, in line with this concept, there is the G2G performance assessment or evaluation. Although in this thesis only the airport part of a surveillance performance evaluation is tackled, this can be seen as just the starting point of a journey that should end with a G2G ideal trajectory and, as a consequence, in a G2G performance evaluation of the surveillance chain.

It is true that on the air side, there are already some commercial products, like SASS-C (ref. 3.3.3.1), that cover the en-route and the TMA airspaces providing reconstructed trajectories. However, their reconstructed references do not reach the airport stands, something that is already achieved by the prototype presented in this thesis. Although it may be considered that the reconstruction on the airport domain might be easier than the one performed on the TMA and en-route airspaces, the use of additional constraints like

the airport map or some features from the aircraft might imply the appearance of new issues that need to be tackled. Just as an example, finding the path followed by a target is by far not straightforward when the sensors that detect it are either biased or not well time synchronized.

Therefore, the possibility of reaching the G2G reconstructed trajectory starting from the reconstruction performed at the airport has its advantages as there are some bias computations, which can be forwarded to the TMA first and the en-route airspace later, that can positively contribute to the alignment of the measurements in those airspaces. Two examples can be: a better initial synchronization and the possibility of estimating and correcting the difference between the height computed from barometric measurements and the one provided by geometric ones. This is what has been called the QNH correction in section 2.6.4 and, when introducing the measurements from MLAT or ADS-B systems, in sections 8.4.1.2.2 and 8.4.1.2.3, respectively.

Consequently, the integration of TMA approach and departure procedures, such as the ones presented in [Frontera 14] and [Frontera 16], will also enter in the extension of the scope of the currently described tool and will provide a possibility to link the performances of aircraft navigation capabilities with its surveillance ones not only at aircraft level, but also on the ground segment. The ground support tools for ASAS-IM-S&M operations, which are proposed in [Frontera 16], rely on the integrity, timeliness and accuracy of ADS-B data for, on the airborne side, properly achieving the spacing that the controller delegates to the aircrew and, on the ground side, monitoring the safe execution of the intended merging procedure that is issued by the controller. Consequently, if there is an interest on implementing this type of procedures, which nowadays has become more a need than a pure interest due to the lack of airport capacity, there is a clear necessity to prepare the systems not only for processing ADS-B data but also for properly assessing if these data are meeting the performance requirements (i.e. RSP) that are mandatory to proceed with this kind of operations.

Similar to the previous situation, in en-route airspace there is also the need for monitoring whether the aircraft are following the routes they declare in their flight plans or flight objects, or not. This is crucial if ATM transitions towards a free routing airspace and, even more, if a free flight airspace is foreseen in a near future. This monitoring relies on surveillance sensors and tracking systems for performing its assessments so, again, there is a need for checking whether they can achieve the enforced performance requirements to proceed with this type of operations. At the end, everything points toward an integrated management of all flight phases.

9.1.3 Evolution of the surveillance chain

The current strong presence in the airport domain of the “new” surveillance sensors, such as ADS-B and multilateration systems, justifies the need for a specially devoted tool to assess their compliance to the expected performance requirements. These sensors enlarge the available surveillance data and improve its quality, but they also introduce new challenges while fusing them with radar data. These challenges not only affect the tracking system but also the PET, which needs to be upgraded to correctly process them. This thesis contributes to this aspect, by enlarging the bias estimation strategy to cope with new context information related to these sensors.

9.1.4 User expected functionalities

Surface surveillance users are permanently exposed to regulatory aspects that can impose certain measures when new procedures are implemented. So, there is a need for an automatic way of carrying out some of these requirements and a clear presentation of the results, which should help on the decision making related to the elements of the surveillance chain that can be used and the ones that need some kind of action. This work provides a solution which is initially thought for an off-line processing of the surveillance data. However, it should not be difficult to modify some of these algorithms to work on a pseudo-online mode which can run in parallel to the tracking system and that can feed it with current bias estimations and with almost real time warnings concerning the degradations of some of these feeding sensors. Even more, the users can also have access to almost real-time figures of the performance of their surveillance chains.

Additionally, from the user side, there is always the need for obtaining performance evaluation information in the most convenient way. This is a field in which current tools are working hard to meet user needs, which are always very demanding.

9.1.5 Airport capacity optimization

Last, but not least, there is a constant need for optimizing airport capacity. One of the possible axes is by increasing runway throughput by introducing new operational procedures such as the ones mentioned in [Frontera 16], but another one is by reducing the uncertainty related to the output provided by tracking systems which relies on the accuracy of surveillance sensors. If sensor measurements are better statistically characterized and their biases are correctly estimated and cancelled, the separation of aircraft can be reduced to their minimum required values and, by achieving this, the number of ground movements can be enlarged.

Nowadays, the situation derived by the COVID19 pandemic has relieved the previous pressure on airport capacity and, by extension, on the whole ATM system. However, this situation, for sure, will be inverted and, therefore, the ATM system will be again stressed due to the lack of capacity. Consequently, some actions need to be taken before this situation comes back.

9.2 Contributions

Taking into consideration the current situation concerning the need for performance evaluation and the state-of-the-art tools that can be found in the market, the proposals derived and presented in this thesis represents a clear forward step on the knowledge, adaptation and application of bias estimations techniques and smoothing algorithms to the problem of performance evaluation based on the trajectory reconstruction from opportunity traffic for the airport domain.

The research done all along this thesis has highlighted more topics than the ones already addressed in this quite large document. The work presented in this thesis includes much more than just the search contributions derived by the author. This is a text which tries to provide the whole picture where the research investigations have taken place. I apologize for the length of this work, but due to the extension of my research activities along time,

it has become some kind of compilation document where I have stored all the knowledge that I have obtained during more than ten years of intermittent but never completely stopped research activity. But looking back, all these pieces matched and contributed to the final result, which is a quite complete document that covers most of the aspects needed for designing and implementing what I have denominated a PET. Within this overall framework, I have been introducing my proposals for solving not only some already known issues, but also the new ones that have appeared. Some of these proposals are evolutions of the work done during the time I spent at my research group, i.e. Grupo de Procesado de Datos y Simulación (GPDS) at the Universidad Politécnica de Madrid (UPM) on the air and surface tracking algorithms and some others come from my curiosity concerning newer tracking techniques, such as the UKF, and their application to the “fixed-interval” smoothing problem that is faced when reconstructing trajectories from off-line data.

Following the same order of exposition in this document, the main contributions of this thesis are:

- **PET logical architecture:** this thesis presents a complete description of a PET from a logical point of view. This description includes a functional decomposition, the description of the internal data flows between identified functions and the interaction with the external actors. This is something that cannot be found in the literature at least with the provided detail. The purpose of this contribution is to support a potential technology transfer to a commercial product of the described tools.
- **Evaluation process description:** this contribution aims at extending the data processing flow presented in [Besada 13] for an air surveillance performance analysis tool like SASS-C, to the airport domain. This description allows introducing some order in the tasks to be executed by the logical functions so that the evaluation process can be properly accomplished. In line with the previous contribution, this one also supports a potential technology transfer to a commercial product.
- **Use of context information:** another contribution of this thesis is the extensive use of context information such as the airport map or the precise physical information of the target, like the position of its GNSS or ATC antennae, the value of its wheelbase, its length or its width. This information is exploited by both, the bias estimation strategy and the trajectory reconstruction algorithm. In the current literature, the use of the airport map has been proposed in references like [Soto 05], [Mann 12], or [López-Araquistain 19] but with different purposes and scopes. However, the use of the identification information for obtaining the aircraft type and from this information accessing the physical dimension of the target has not been found neither for tracking nor for trajectory reconstruction.
- **Multi-Hypotheses Path Finder algorithm:** this algorithm is key for the subsequent SMR bias estimation, which at the same time is also the corner stone of the complete bias estimation strategy. Consequently, it has a paramount relevance in the success on the whole data processing. Although, it might seem trivial to extract the path followed by the aircraft from the target reports generated by surveillance sensors, the experience has shown that it is clearly not the case. The fact that surveillance data usually do not behave as expected contributes to

the complexity that needs to be added to the algorithm to finally achieved the expected result, which is not only the path followed but also the assignment of the right airport segment to each of the target reports associated to the trajectory. Opposite to what happened in [Soto 05], where most of the target reports had to be discarded because they were located in several airport segments while online processed, the MHPF algorithm resolves this uncertainty, so initially all the target reports might be used by the SMR bias estimation algorithm, which later can discard the ones in manoeuvring areas. This advantage clearly contributes to shorten the time needed for its convergence. The results of this algorithm are presented in section 8.4.1.3.7, where it is possible to visually assess the correct identification of the path followed by the target.

- **Complete bias estimation architecture:** this thesis proposes a complete bias estimation architecture, which compared to previous proposals like the ones presented in [Soto 05], [Besada 05] or [Besada 13] uses aircraft dependent information like the position of its GNSS and ATC antennae not only for correcting the position provided by ADS-B and MLAT measurements (if needed), but also for improving the synchronization with the sensor considered as the one providing the time reference. This information can be used at airport level, but it can also be forwarded to the air surveillance, which can initially correct those measurements that overlap with the domain of interest of the surface surveillance or even initiate its bias estimators with the values provided by the surface surveillance in case both share a similar measurement model. The already mentioned ADS-B uncompensated latency (ref. 2.3.2.3), is another piece of information that can be forwarded to upstream tracking systems. However, it is still under discussion how variable is this bias term.
- **Enhanced 3-wheel ground movement model:** this model which was first introduced in surface tracking systems by NLR in [Van Hintum 13] for the ARTAS SMS prototype is enhanced by incorporating precise aircraft information like the wheelbase. Opposite to what is done in the previous reference where a constant value is defined for all aircraft, the wheelbase value that is used in this proposal comes from the exact information concerning the target. Additionally, the model is implemented by the filtering and smoothing algorithms that are proposed for reconstructing the target trajectory in order to cope with manoeuvring segments of the target trajectory.
- **Smoothing algorithm based on manoeuvre classification and on target information:** this contribution propose a multiple model smoothing algorithm that exploits the information concerning the different manoeuvring segments that can be found along the trajectory followed by the target. This way it is possible to reduce the number and select models of the modes that are used in a IMM-like forward filtering and backward smoothing algorithm, which is based on the one presented in [Nadarajah 12]. The previous proposal is extended in this thesis to cover the “fixed-interval” smoothing problem that it is faced when reconstructing trajectories from off-line data. The set of selected ground movement models is composed of one mode for manoeuvring segments and another one for non-manoevring ones. The manoeuvring segment is selected based on the manoeuvring classification performed by the MHPF algorithm while the non-manoevring one is selected based on the mean ground speed of the target in the

airport segment where it is located. This strategy tries to combine the best of both model-matched modes assuming that the change in the target movement model happens in an unknown moment. Moreover, this contribution relies on smoothing algorithms that take advantage of the previously mentioned 3-wheel ground movement model and the knowledge of the aircraft type. The comparison with other state-of-the-art algorithms demonstrates that the proposed one significantly improves the behaviour during manoeuvres, correctly handle standing targets and reduce the noise in straight segments. These results are presented in section 8.4.1.4.

9.3 Research directions and future work

While working on this thesis, there have been some research topics that have emerged, and which have been considered as out of the scope. Some of these topics are quite close to the scope of this work while in other cases are more distant. Anyhow, all them are presented hereafter:

- **Map enhancement:** currently, the airport map that is used is the one obtained by the procedure described in section 8.4.1.1.2. This means that curves are approximated by a succession of very short segments (between 8 to 12m) which change their orientation to describe the turn. The problem appears when measurements, which come from noisy, biased, and, potentially, not synchronized sensors, are processed in order to be allocated to them while additionally imposing some movement constraints. In this case, the further work would be to implement an airport map pre-processing task that approximates the curves by other means such as the ones proposed in section 8.4.1.3.7, that is, substituting the series of short turning segments by a circular segment or by an approximating spline.
- **Association strategy:** as mentioned in section 6.7.4, data association is a topic which has been left outside of the main scope of this thesis. Therefore, it is an area that can be the focus of more in-depth research.
- **Trajectory classification:** the identification of the type of trajectory (e.g. inbound or arrival, outbound or departure, ground movement, etc.) that the target is performing can help the MHPF when identifying the segments that are visited while arriving or departing to or from an airport. This is especially relevant when departing from the stand and going towards the runway, because this type of trajectories normally starts with a push-back movement until the aircraft is aligned on the taxiway from where it will start its forward movement looking for the runway.
- **Improving ground movement models:** there are ground movements that are identified but which have not been properly modelled. This is the case, for example, of aircraft that are turning without speed or the ones that are moving backwards. These models can improve the trajectory reconstruction, even more, if combined with a trajectory classification as the one proposed in the previous bullet point.
- **Use of ADD or DAP:** nowadays, it is possible that aircraft provide some airborne information to the ground systems. This is the case when using mode S enhanced

surveillance or ADD obtained by ADS-B or multilateration systems. Similar to what it is proposed in [Soto 06], this information can be exploited for reconstruction purposes. For example, for rotating targets without speed, it can be interesting to know the true track angle or the track angle rate to correctly track the heading of the target. In line with this, there would be the need for defining a movement model that matches aircraft turning without ground speed, as it is proposed in the previous bullet point.

- **Ground bit use:** this is a bit of information (I021/040 #GBS), which has not been exploited in this thesis, that might be used, if found reliable, to identify whether an aircraft has touched down, if it is detected on the ground, or if it has already taken off. So, the future work would be to first check whether this information is reliable and, if it is found that yes, then it can be exploited by the data filtering tasks and the trajectory reconstruction algorithms, even more, if they cover approach and departure procedures.
- **SMR measurement model enhancement:** this future work tries to cope with the discrepancies found while testing the initial assumption about the bias terms and the inclusion of trajectory-based bias estimation adjustments that later can be put in common with the results from other trajectories if the airport map model allows to store this information. A good starting point for this activity would be [García 01].
- **INCS measurement models:** as it was highlighted in section 2.5.1.2 and in [Žáčik 17], there is no commonly accepted measurement model for INCS systems. So, this is a topic that may deserve further investigation as, in several years, there will be the need for coping with this type of sensors.
- **Enlarge the scope to approach and departure procedures:** another interesting future work, as a first step towards the full G2G surveillance performance analysis, can be to extend the current scope of the proposed tool to include the aircraft landing and taking off (e.g. SID and STAR procedures). Extending the trajectory reconstruction to these phases of flight opens the door to taking into consideration improved operational procedures like the ones related to ASAS applications as presented in [Frontera 16].
- **Height reconstruction:** this research topic is linked to the previous one and it represents one of the current war horses in aircraft tracking in the proximity of airports. The possibility of aligning and then fusing height information from barometric and geometric sources is relevant not only for ground surveillance but also for air surveillance. Systems like ARTAS lack of a QNH correction, which sometimes is translated in tracking issues such as split tracks or wrong association.
- **Artificial intelligence:** this is one of the hot topics in recent ATM research discussions. Introducing artificial intelligence (AI) techniques in some of the issues addressed in this thesis might be a challenging and motivating task.
- **Drone trajectory management:** this is the other trending topic together with AI. The unmanned traffic management (UTM) and drone operation will for sure introduce new challenges for the surveillance chain and for extension to its performance monitoring. The discussion is still kept at high level, so there is a

need to keep an eye on how drone trajectory management may impact surveillance sensors and tracking systems as it will be translated into modifications on the performance monitoring tools.

9.4 A Visionary Look at Performance Evaluation Tools

As it was done at the end of CHAPTER 2, the author would like to provide its view concerning the evolution of PET tools. Therefore, the future will increasingly demand the presence of tools that monitor and assess that the required performance indicators are kept within acceptable levels.

For sure, PETs will transition from off-line tools, which partially cover some aspects of a certain CNS domain, to pseudo-online or even online tools that will transversally integrate several domains and that will cover G2G operations. At the end, they will provide global performance indicators, not only local ones, that might also be particularized per trajectory or per area of interest.

The future of PET focussed on the surveillance chain will deal with new sensors, so new measurement models will have to be developed as stated in the previous section. Moreover, the tracking strategy can change based on the high update rates that actual sensors (e.g. ADS-B or multilateration systems) provide, preventing the use of tracking algorithms the way they are used today. Consequently, depending on the evolution of surface surveillance and the performance requirements that are required by the authorities, the PET will have to take one shape or another.

Another derivative that it is important not to forget is the one that introduces, in the discussion concerning the future ATM, the need for environmental improvements and sustainability. Actions in this direction can contribute to the reduction of the RF pollution and the rationalization of the currently deployed surveillance infrastructure, which needs to remain as performant as before but consuming less resources. This process will need to be properly monitored.

Depending on how the new commers, such as drones or UAVs, are integrated within the ATM service and how UTM is finally implemented, there will be a different need on the surveillance performances to be evaluated.

Finally, the regulatory framework can also alter the whole picture by mandating the carriage of more performant equipment and the removal of obsolete one. However, there will always be a transition phase in which both technologies will have to live together while the performances should be progressively increased.

The conclusion is that the performance-based approach is clearly here to stay. Consequently, operational systems should be prepared to monitor and assess such performances. The sooner the better.

FILTERING ALGORITHMS

A.1 Introduction

As already mentioned in CHAPTER 7, filtering algorithms or filters are a key part of the smoothing algorithms as they deal with the information coming from the past and present observations, while their backwards or time reverse versions deal with the information from the future backward to the present. Not all the filters admit a backward version, as it depends on the statistics of the underlying process. If the latter is time-reversible, then it is possible to find such a backward version, otherwise not.

The focus of this annex is placed on those filters that may be of interest for building smoothing algorithms or on the ones that have set the ground for future developments.

A.2 Wiener filter

The Wiener filter is an intermediate step between the least-squares method and the introduction of the Kalman filter.

It was in the beginning of the forty's when first Kolmogorov, in 1941 [Kolmogorov 41], and Wiener, the following year, developed a linear mean-square estimation technique, which can be seen as the foundation of the Kalman filter theory. Although Wiener published his work a few years later in 1949 [Wiener 49] due to the World War II, the filter has kept his name. However, the theory is known as the Wiener-Kolmogorov filter theory [Sorenson 70].

A.2.1 General description

As mentioned before, the developments of both authors are related to a linear minimum mean-square error estimator with the following appearance:

$$\hat{x}(n) = \sum_{k=-\infty}^{\infty} h(n, k)z(k) \quad (A.1)$$

Where:

- $h(n, i)$: filter components.

The target is to estimate $x(n)$, which can be a time-varying parameter (e.g. a system state variable), by means of $\hat{x}(n)$, which is obtained applying a linear operation to the noisy measurement data $\{z(0), \dots, z(n)\}$. $x(n)$ and $z(n)$ are related through the knowledge of the cross-correlation function $R_{XZ}[n_x, n_z]$.

A.2.2 Assumptions of the Wiener filter

There are some assumptions which characterize the Wiener filter that is worth presenting previous to the description of the different versions [Brown 97]:

- The filter is linear and time-invariant (LTI), so it can be defined by its impulse response $h[n]$.
- The signal and noise are stochastic processes with known statistical features such as the auto- and cross-correlation functions (R_X/R_Z and R_{XZ}).
- Both processes are stationary in wide sense (WSS). Although a block processing is also possible, that is, to process blocks of N signal samples [Vasegui 00].
- The optimality criterion is the minimum mean-square error (MMSE). Even more, it could be added “linear” to the performance metric, leading to LMMSE.
- Generally speaking, its solution is based on scalar methods.

These are the most characterizing issues concerning the Wiener filter.

A.2.3 Types of Wiener filter

There are two possible situations depending on the values of the limits of equation A.1 that are related to the length of the filter impulse response. The previous expression is now adapted to an LTI filter:

$$\hat{x}(n) = \sum_{i=-\infty}^{\infty} h[n-i]z(i) \quad (A.2)$$

A.2.3.1 Non-Causal Wiener filter

Therefore, if we assume an infinite input response, i.e. $h[i]$ could be defined for $-\infty \leq i \leq \infty$; the designed filter would be an IIR (Infinite Impulse Response), which is a non-causal filter as it takes advantage of measurement from both the past and the future of the signal. For the objective of this work, this version of the Wiener filter is the most

significant one as we are processing recorded data, so no requirements concerning causality are expected.

A.2.3.2 Causal Wiener filter

If the assumption is to have just past measurements, the filter will be causal. In this case, there are two options, one is that the number of past measurements is finite or infinite, which will lead to an IIR or FIR (Finite Impulse Response) filter, respectively.

For the sake of completeness, the way in which the causal filter is designed is going to be outlined for its two versions, that is, the IIR and the FIR ones.

Thus, the estimation $\hat{x}(n)$ is computed as a linear combination of the measurements as shown in equation A.1. The coefficient of the filter $h(n, i)$ are computed aiming at minimizing the mean-square error, that is:

$$\epsilon = E \left[(x(n) - \hat{x}(n|n))^2 \right] = E[\tilde{x}(n)^2] \quad (A.3)$$

This is reached because the estimation error is orthogonal to the space created by the measurements (i.e. $E[\tilde{x}(n)z(i)] = 0, \forall i \leq n$), which is known as satisfying the orthogonality property. With little manipulation, this leads to the Wiener-Hopf equation:

$$E[x(n)z(i)] = \sum_{j=0}^n h(n, i) E[z(j)z(i)] \quad (A.4)$$

If we assume that it is a FIR filter, which implies that the number of filter components that are non-zero is limited, the Wiener-Hopf equation can be expressed in vector-matrix form and so, the estimation can be obtained by computing the inverse matrix or by using the Levinson-Durbin algorithm [Levinson 46].

Otherwise, the causal solution must be provided by an IIR filter. This case is the one that kept the attention of both Wiener and Kolmogorov as they assumed an infinite number of observations and additionally the stationarity of the system. The expression of the filter in this case is:

$$\hat{x}(n) = \sum_{i=-\infty}^n h[n-i]z(i) \quad (A.5)$$

This system of equations was resolved by spectral factorization, which takes advance of the use of a shaping filter. For further details please refer to [Kolmogorov 41][Wiener 49].

A.2.4 Non causal Wiener filter equations

When there is no need to meet any causality condition, the problem can be treated in the frequency domain, as the limits of the summation go from $-\infty$ to ∞ .

$$\hat{x}(n) = \sum_{k=-\infty}^{\infty} h[k]z(n-k) \quad (A.6)$$

Consequently, applying the orthogonality condition ($\tilde{x}(n) \perp z(n-i) - \infty \leq i \leq \infty$), which is defined as:

$$E[\tilde{x}(n)z(n-i)] = 0 \quad (A.7)$$

One can reach the “design equation” following the next development:

$$\begin{aligned} E[\tilde{x}(n)z(n-i)] &= E[(x(n) - \hat{x}(n))z(n-i)] = \\ &= E\left[\left(x(n) - \sum_{k=-\infty}^{\infty} h[k]z(n-k)\right)z(n-i)\right] = \\ &= E[x(n)z(n-i)] - \sum_{k=-\infty}^{\infty} h[k] E[z(n-k)z(n-i)] = 0 \end{aligned} \quad (A.8)$$

Giving:

$$R_{XZ}[i] = \sum_{k=0}^n h[k] R_Z[i-k] = h[i] * R_Z[i] \quad (A.9)$$

This expression allows to solve the problem in the transformed domain. Therefore, taking the $\mathcal{Z}$ -transform on both sides, the transfer function of the filter is obtained:

$$H[z] = \frac{S_{XZ}[z]}{S_Z[z]} \quad (A.10)$$

Where:

- $E[x]$: expectation function of x .
- R_{XZ} : cross-correlation function of X and Z .
- R_Z : auto-correlation function of Z .
- $S_{XZ}[z] = \mathcal{Z}\{R_{XZ}[i]\}$: cross spectrum of X and Z .
- $S_Z[z] = \mathcal{Z}\{R_Z[i]\}$: spectrum of Z .

This is possible due to the stationary condition that is requested to both sequences.

The values of the filter components are obtained taking the inverse $\mathcal{Z}$ -transform, so:

$$h[z] = \mathcal{Z}^{-1}\{H[z]\} \quad (A.11)$$

These values are the ones that provide the optimal filter for the assumptions that have been stated.

A.2.5 Final considerations

Going back to the introduction of section 7.4 of CHAPTER 7, it can be said that this approach is in line with the classical approach as opposed to the one introduced by Kalman [Kalman 60], where the focus will be in the internals of the system, not the input-output relation.

Taking into consideration the topic that is addressed in this work, it can be said that the Wiener filter is not foreseen as one of the candidate filtering techniques to be considered when smoothing data because the stationary condition is not held when working with surveillance data. However, the rationale for the inclusion of the Wiener filter in this text, relies on the fact that it is a precursor of the Kalman filter, which is really a pillar of the state estimation theory.

A.3 Steady-state filters

The term “Steady-state” filters includes the family of Alpha-Beta and the Alpha-Beta-Gamma filters [Bar-Shalom 88]. These are one of the first approaches to the state estimation problem before the appearance of the Kalman filter. The Alpha-Beta filter considers just position and velocity (two-dimensional case), while the Alpha-Beta-Gamma filter also includes acceleration (three-dimensional case). Thus, they are associated to the nearly constant velocity (ref. 7.4.6.3.2) and acceleration (ref. 7.4.6.3.3) models, respectively. It was in mid-50's when the development of both target trackers was significant [Kalata 92].

A.3.1 General description

The Alfa-Beta and Alpha-Beta-Gamma filters are target tracking systems that can be seen under certain assumptions as the steady-state version of the Kalman filter. They use a very similar notation, although they lack the possibility of adjusting the filter gain to be applied to the innovations depending on their statistical properties. Another difference is the fact that they do not include an inherent way of computing the error covariance matrices, albeit there are explicit expressions for obtaining them [Bar-Shalom 88].

A.3.2 Assumptions of the steady-state filters

The filter gain is composed of two or three constant components depending on the dimensions of the filter. The constant gain for the position is referred to as Alpha (α), the one for the velocity as Beta (β) and finally, the one for the acceleration, in case it is taken into consideration, as Gamma (γ). The election of the values of the constant gains is most of the time based on the expertise of the system engineer or based on any rule of thumb. However, it is possible to find in the literature some optimal elections based on the error characteristics of the measurements or on the manoeuvrability level of the target [Blackman 99].

These filters are based on time-invariant models¹⁹⁸ such as the nearly constant velocity model or the nearly constant acceleration one, already mentioned above.

A.3.3 Types of steady-state filters

The main division between this category of filter is the one based on the dimensions that are considered while tracking. Thus, one can consider the two already presented types the Alfa-Beta and Alpha-Beta-Gamma filters. These are the ones that are going to be described in the next section.

A.3.4 Steady-state filter equations

The equations that are going to be presented in this section are the ones concerning both filters, first the ones referring to the two-dimensional case and then the ones for the three-dimensional case:

Process and measurement models

The process and measurement models are the ones presented in section 7.4.6.2.1 and they are shared by both filters. Thus, the system state vector evolves with time as:

$$x(k+1) = Fx(k) + Gw(k) \quad (A.12)$$

While the measurement or observation model is assumed as:

$$z(k) = Hx(k) + v(k) \quad (A.13)$$

The particularities of $w(k)$ and $v(k)$ have already been also mentioned in section 7.4.6.2.1. The projection matrix (H) is time-invariant, while the state transition matrix (F) and the noise projection matrix (G) are also time-invariant, if the measurements are provided on a periodic basis.

Prediction step

Every time instant a measurement is received; the algorithm extrapolates the state, i.e. x , to this time (k) based on the stored state calculated in the previous update time ($k-1$). The equations for the predicted state and the initial conditions of the two-dimensional case are [Bar-Shalom 88]:

$$\hat{x}(k|k-1) = \hat{x}(k-1|k-1) + T\hat{v}(k-1|k-1) \quad (A.14)$$

$$\hat{v}(k|k-1) = \hat{v}(k-1|k-1) \quad (A.15)$$

In case it is the first prediction, the initial values could be:

$$\hat{x}(0) = x_0 \quad (A.16)$$

¹⁹⁸ Time-invariant filters are the ones that are characterized by having constant coefficients in both, the state and the measurement equations.

$$\hat{v}(1) = \frac{\hat{x}(1) - x_0}{T} \quad (\text{A.17})$$

For the three-dimensional case, the predictions are:

$$\hat{x}(k|k-1) = \hat{x}(k-1|k-1) + T\hat{v}(k-1|k-1) + \frac{T^2}{2} \hat{a}(k-1|k-1) \quad (\text{A.18})$$

$$\hat{v}(k|k-1) = \hat{v}(k-1|k-1) + T\hat{a}(k-1|k-1) \quad (\text{A.19})$$

$$\hat{a}(k|k-1) = \hat{a}(k-1|k-1) \quad (\text{A.20})$$

In case it is the first prediction, the initial values could be:

$$\hat{x}(0) = x_0 \quad (\text{A.21})$$

$$\hat{v}(1) = \frac{\hat{x}(1) - x_0}{T} \quad (\text{A.22})$$

$$\hat{a}(0) = 0 \quad (\text{A.23})$$

Gain computation

Due to the steady-state features of the “steady-state” filters, the gain does not need to be constantly updated. In fact, the main research point concerning this tracking filter is the optimal value for the components of such a gain, which generally depends on the knowledge of the target manoeuvrability and measurement noise, as already mentioned in section A.3.2.

Filtering step

Finally, the measurement $z(k)$ is combined with the predicted information about the state. Hence, for the two-dimensional case it follows that:

$$\hat{x}(k|k) = \hat{x}(k|k-1) + \alpha[z(k) - \hat{x}(k|k-1)] \quad (\text{A.24})$$

$$\hat{v}(k|k) = \hat{v}(k|k-1) + \frac{\beta}{T}[z(k) - \hat{x}(k|k-1)] \quad (\text{A.25})$$

Likewise, in the three-dimensional one, the following equation is added to the two previous ones:

$$\hat{a}(k|k) = \hat{a}(k|k-1) + \frac{\gamma}{2T^2}[z(k) - \hat{x}(k|k-1)] \quad (\text{A.26})$$

A.3.5 Final considerations

The use of these filters was very extended when computational constraints were a relevant issue while designing a system. Nowadays, this is not a problem any more, at least for most of the systems, as computation power is not a limiting factor to be taken into consideration. The computation power of desktops, and even more if we talk about workstations or servers, allows them to run a Kalman filter in any case. Steady-state filter

are still of interest for embedded system or for motes, where the computation power is still a constraining factor. So, for performance evaluation system, there is no special interest in this kind of filter as they can be substituted by Kalman filters. However, they are the base of some smoothing algorithms. This is the reason why they are kept in this presentation of tracking algorithms.

A.4 Kalman filter

The Kalman filter is considered by many people as one of the most relevant discoveries in the 20th century in estimation theory and its importance in software algorithms is often compared to the one of silicon when referring to electronic devices.

The filter was named after Rudolf E. Kalman who described it in a paper called "A New Approach to Linear Filtering and Prediction Problems" [Kalman 60], which was published in 1960. Similar to what happens with the squabble between Gauss and Legendre concerning the priority of the Least-Squares method, Peter Swerling also reached a similar recursive procedure when working on orbit determination [Swerling 59], but with a more complex way of obtaining the covariance matrix of the estimation. Swerling claimed for priority of the Kalman filter equations but his complain never succeed.

A.4.1 General description

The Kalman filter is a multiple-input multiple-output recursive digital linear estimator which provides an optimal means for estimating in real-time parameters and state variables of a certain system from indirect and normally noisy outputs of such a system.

This algorithm is said to be optimal as it minimizes a certain error criterion, i.e. the mean square error, when its assumptions are fulfilled.

A.4.2 Assumptions of the Kalman filter

There are some assumptions, which characterize the Kalman filter, that need to be conveniently presented before the description of the different versions:

- The filter is linear but not time-invariant (LTI), so it cannot be defined by its impulse response $h[n]$.
- The signal is divided into two, a deterministic part which is driven by the process model and a stochastic one which is described by the mean and covariance matrix of the plant noise.
- The filter assumes that the observation model, which describes the measurements, is also linear.
- Both, the plant noise and the measurement noise, are assumed white and Gaussian.
- The output is then also expressed as Gaussian, and as such, it is fully characterized by the mean vector and its covariance matrix.

- The initial conditions are represented by Gaussian random variables with a mean of x_0 and a covariance matrix of P_0 .
 - $E[x(0)] = x_0$
 - $P(0|-1) = P_0$
- The plant noise $w(k)$ and the measurement noise $v(k)$ are white noise sequences with zero mean, and $Q(i)$ and $R(i)$ covariance matrices, respectively.
 - $E[w(k)] = 0$
 - $E[v(k)] = 0$
 - $E[w(i)w(j)] = Q(i)\delta_{ij}$
 - $E[v(i)v(j)] = R(i)\delta_{ij}$
- The initial value, the plant noise and the measurement noise are uncorrelated vectors.
 - $E[x_0w(k)] = 0$
 - $E[x_0v(k)] = 0$
 - $E[w(k)v(k)] = 0$
- The optimality criterion is the minimum mean-square error (MMSE).

These are the most characterizing issues concerning the Kalman filter.

A.4.3 Types of Kalman filter

The Kalman filter has many versions. Thus, one can find the discrete-time version or its counterpart in continuous-time. The latter is called the Kalman-Bucy filter [Kalman 61]. There are also linear and nonlinear versions. The Extended Kalman filter (EKF) is a clear linear example, however it is not the only one, as linear versions admit model extensions and the use of approximation methods that make tractable some slightly nonlinear problems. Due to the importance of the EKF, there is a specific section devoted to it. Additionally, there are other types of filters that also include the name of Kalman in their denomination, such as, the Unscented Kalman filter (UKF) [Julier 97] or the Cubature Kalman filter (CKF) [Arasaratnam 09] with a quite high acceptance nowadays. The UKF is presented in a separate section due to their importance for the work done in this thesis. In this section, the basics of the Kalman filter are going to be presented as a fundamental pillar for the algorithms that are presented later.

A.4.4 Kalman filter equations

The equations that are going to be presented in this section are the ones concerning the linear and discrete version of the Kalman filter. Due to the use of the state-space representation, these equations take a very compact representation.

Process and measurement models

The process and measurement model for the linear version are the ones presented in section 7.4.6.2.1. Hence, the system state vector evolves with time as:

$$x(k+1) = F(k)x(k) + G(k)w(k) \quad (A.27)$$

While the measurement or observation model is assumed as:

$$z(k) = H(k)x(k) + v(k) \quad (A.28)$$

The particularities of $w(k)$ and $v(k)$ have already been mentioned in section A.4.2 and the state transition matrix (i.e. $F(k)$) and the projection matrices (i.e. $G(k)$ and $H(k)$) are considered, in general, as time-variant.

Prediction step

Every time instant a new measurement is received, the algorithm extrapolates the state vector, i.e. x , and its corresponding covariance matrix, i.e. P , to this time (k) obtaining the predicted state vector, $\hat{x}(k|k-1)$, and covariance matrix, $P(k|k-1)$.

This calculation takes as starting point the information on the previously filtered state vector, $\hat{x}(k-1|k-1)$, and covariance matrix, $P(k-1|k-1)$, calculated at update time ($k-1$). The equations for the predicted state are:

$$\hat{x}(k|k-1) = F(k-1)\hat{x}(k-1|k-1) \quad (A.29)$$

And the ones for the predicted covariance matrix are:

$$P(k|k-1) = F(k-1)P(k-1|k-1)F^T(k-1) + Q(k-1) \quad (A.30)$$

In case it is the first prediction, the initial values are:

$$\hat{x}(0) = x_0 \quad (A.31)$$

$$P(1|0) = P(0) \quad (A.32)$$

Gain computation

The Kalman gain is computed as:

$$K(k) = P(k|k-1)H^T(k) [H(k)P(k|k-1)H^T(k) + R(k)]^{-1} \quad (A.33)$$

Being the term within square brackets the so-called residual or innovation covariance matrix S :

$$S(k) = H(k)P(k|k-1)H^T(k) + R(k) \quad (A.34)$$

Filtering step

Finally, the measurement $z(k)$ is combined with the predicted information about the state to obtain the updated filtered state vector:

$$\hat{x}(k|k) = \hat{x}(k|k-1) + K(k)[z(k) - H(k)\hat{x}(k|k-1)] \quad (A.35)$$

While the term within square brackets is the so-called residual or innovation v , whose expression is:

$$v(k) = z(k) - H(k)\hat{x}(k|k-1) \quad (A.36)$$

Likewise, the updated filtered covariance matrix is computed as:

$$P(k|k) = P(k|k-1) - K(k)S(k)K^T(k) \quad (A.37)$$

Therefore, eqns. A.29, A.30, A.33, A.35 and A.37 are the key equations that describe the update process of a Kalman filter whenever a new measurement is received.

A.4.5 Final considerations

Almost everything is already said about the Kalman filter as the extensive literature involving it shows. However, nowadays the Kalman filter remains as the main tracking filter either in isolation or in combination with other Kalman filters while giving shape to any of the multiple model approaches that can be found in state estimation of stochastic hybrid systems.

The next section presents one of the possible evolutions of the Kalman filter to cope with slightly nonlinear process and/or measurement models, which are quite extended in the nature. The EKF is one alternative but not the only one, as it will also be shown in the coming sections where the UKF is also presented. Each new approach tries to overcome the drawbacks of the previous one or improve their performances while introducing new concepts, approaches and techniques. The presentation of these new filters is stopped at some point knowing that there are some other variants which are not covered, but at some point, the limit has to be set.

A.5 Extended Kalman filter

The EKF is the extension of the Kalman filter ideas to the case of slightly nonlinear process and measurement models.

A.5.1 General description

Similar to the Kalman filter, the EKF is a multiple-input multiple-output recursive digital linear estimator which provides a means for estimating in real-time parameters and state variables of a certain system from indirect and normally noisy outputs of such a system. However, the EKF only provides a sub-optimal solution to the problem it addresses.

A.5.2 Assumptions of the EKF filter

The assumptions of the EKF are coincident to the one defined for the Kalman filter. Adding the fact that models are expected not to show heavy nonlinearities.

A.5.3 Types of EKF filter

There are further improvements concerning the EKF named:

- Iterated EKF which is based on repeating the linearization step with the first filtered output $\hat{x}(k|k)$.
- 2nd order EKF, which not only consider the Jacobian but also the Hessian while expanding the nonlinear model evaluated at the estimated state.

Both evolutions of the EKF are presented in-depth in [Bar-Shalom 88].

A.5.4 EKF filter equations

The equations of the EKF take a similar form as the one coming from the Kalman filter.

Process and measurement models

The process and measurement models are the ones presented in section 7.4.6.2.2. Thus, the system state vector presents a non-linear evolution with time, such as:

$$x(k+1) = f(k, x(k)) + w(k) \quad (A.38)$$

Equivalently, the measurement or observation model is described as:

$$z(k) = h(k, x(k)) + v(k) \quad (A.39)$$

The state transition matrix, i.e. $f(k, x)$, and the projection matrix, i.e. $h(k, x)$, are considered, in general, as non-linear functions.

Prediction step

Every time instant a new measurement is received, the algorithm extrapolates the state vector, i.e. x , and its corresponding covariance matrix, i.e. P , to this time (k) obtaining the predicted state vector, $\hat{x}(k|k-1)$, and covariance matrix, $P(k|k-1)$.

As before, this calculation takes as starting point the information on the previously filtered state vector, $\hat{x}(k-1|k-1)$, and covariance matrix, $P(k-1|k-1)$, calculated at update time ($k-1$). The equation for the predicted state is:

$$\hat{x}(k|k-1) = f(k-1, \hat{x}(k-1|k-1)) \quad (A.40)$$

And the one for the covariance matrix is:

$$P(k|k-1) = \left[\frac{\partial f}{\partial x} \right] P(k-1|k-1) \left[\frac{\partial f}{\partial x} \right]^T + Q(k-1) \quad (A.41)$$

As it can be seen, this expression is the same as in eqn. A.30, just changing the Jacobian of f , i.e. $\left[\frac{\partial f}{\partial x} \right]$, by the state transition matrix $F(k)$. The Jacobian of f is evaluated at $\hat{x}(k-1|k-1)$.

The same initial values as for the KF are taken for the EKF, that is:

$$\hat{x}(0) = x_0 \quad (A.42)$$

$$P(1|0) = P(0) \quad (A.43)$$

Gain computation

The Kalman gain is computed as:

$$K(k) = \hat{P}(k|k-1) \left[\frac{\partial h}{\partial x} \right]^T \left[\left[\frac{\partial h}{\partial x} \right] \hat{P}(k|k-1) \left[\frac{\partial h}{\partial x} \right]^T + R(k) \right]^{-1} \quad (A.44)$$

The residual or innovation covariance matrix S takes now the following appearance:

$$S(k) = \left[\frac{\partial h}{\partial x} \right] \hat{P}(k|k-1) \left[\frac{\partial h}{\partial x} \right]^T + R(k) \quad (A.45)$$

Similar to eqn. A.34 the residual or innovation covariance matrix presents the same form as in the linear counterpart just replacing the Jacobian of h , i.e. $\left[\frac{\partial h}{\partial x} \right]$, by the projection matrix $H(k)$. The Jacobian of h is evaluated at $\hat{x}(k-1|k-1)$.

Filtering step

Finally, the measurement $z(k)$ is combined with the predicted information about the state to obtain the updated filtered state vector

$$\hat{x}(k|k) = \hat{x}(k|k-1) + K(k)[z(k) - h(k, \hat{x}(k|k-1))] \quad (A.46)$$

The residual or innovation v takes now the following form:

$$v(k) = z(k) - h(k, \hat{x}(k|k-1)) \quad (A.47)$$

Likewise, the updated covariance matrix is computed as:

$$\hat{P}(k|k) = \hat{P}(k|k-1) - K(k)S(k)K^T(k) \quad (A.48)$$

Thus, eqns. A.40, A.41, A.44, A.46 and A.48 are the key equations that describe the update process of a 1st order EKF whenever a new measurement is received.

A.5.5 Final considerations

The EKF is a very extended version of the Kalman filter for those problems which are slightly nonlinear. When implementing an EKF, it is important to bear in mind that the initial conditions affect considerably the performance of the filter output, so some care has to be placed when choosing these values. Unlike the Kalman filter, the EKF is not optimal and it is not stationary, forcing the computation of $K(k)$ and $P(k|k)$ each time step. This fact makes it more computationally demanding than its linear counterpart.

Since it is a first-order linear approximation around the predicted trajectory, it is mandatory that these predictions are good enough otherwise the filter will show an erratic behaviour, causing a significant error for highly nonlinear systems due to the propagation of uncertainty through the nonlinear system which may possibly end in divergence. This feature provides no guarantee that the filtering error can be bounded a-priori. In order to prevent or mitigate this drawback and another related to the computation of the Jacobian, there are some evolutions, such as the UKF which is presented next.

A.6 Unscented Kalman filter

The UKF is a relatively novel approach [Julier 97][Julier 04] in the state estimation field, which tries to overcome some of the already mentioned problems that appear when the process and measurement models are clearly nonlinear or non-Gaussian and so, the assumptions of the Kalman filter are no longer valid, and the first-order Taylor series expansion performed by the EKF is not enough to approximate those models, preventing the convergence of the state estimate.

A.6.1 General description

The UKF is based on the idea of selecting several sample points around the current state estimate while taking into consideration its covariance matrix. These points are called “sigma points” and, in some fashion, they represent a sampling of the conditional state density, which is then propagated forward in time by means of propagating these “sigma points” through the nonlinear state transition function. Then, the “predicted” conditional state density is approximated by the mean and covariance matrix given by the weighted average and the weighted outer product of the transformed “sigma points”. These values are the ones that enter the update step of the UKF resulting in a more accurate estimation of the “filtered” mean state vector and its covariance matrix, every time a new measurement arrives. This is justified since the UKF captures higher order statistics than the KF and EKF under this situation.

The propagation of the “sigma points” is done taking advantage of the Unscented Transform (UT), which transforms random vectors through nonlinear functions. As it is mentioned in [Julier 97], this transformation “is founded in the intuition that it is easier to approximate a Gaussian distribution than it is to approximate an arbitrary nonlinear function or transformation”. The UT is presented in section A.6.4.

It is significant to highlight that the selection to the “sigma points” is not done at random opposite to the case of particle filters [Ristic 04], but following a specific and deterministic algorithm.

However, the UKF also comes with some drawbacks such as the need for slightly more processing time than the equivalent EKF and the inability to capture the statistics of multi-modal densities [Veth 15].

A.6.2 Assumptions of the UKF filter

No assumptions are made about the system and the measurement models, or about the noises that affect them. The main assumption of the UKF is the fact that it needs to define the “sigma points” and then, using the UT, to propagate them through nonlinearities.

A.6.3 Types of UKF filter

There are different flavours of the UKF depending on the noise source. Hence, there are special versions of the UKF for addressing non-additive noises. However, they are not presented in this document, but further information can be found in the literature, references like [Julier 04] or [Ristic 04] are two examples.

A.6.4 UKF filter equations

The structure of this section differs a little bit from the ones presented in the previous sections as now an initial step appears, which is the calculation of the “sigma points”. The new steps are presented hereafter [Veth 15].

Process and measurement models

Similar to the EKF presented in the section above, the process and measurement models are the ones presented in section 7.4.6.2.2. Consequently, the system state vector presents a nonlinear evolution with time, such as the one shown in the following expression:

$$x(k+1) = f(k, x(k)) + w(k) \quad (A.49)$$

Equivalently, the measurement or observation model is described as:

$$z(k) = h(k, x(k)) + v(k) \quad (A.50)$$

The state transition matrix, i.e. $f(k, x)$, and the projection matrix, i.e. $h(k, x)$, are considered, in general, as non-linear functions and the noises are not assumed as Gaussian.

Unscented transform

Calculation of the “sigma points”

This is the first step of the forward UT. It consists of defining the “sigma points” that will be used in the propagation through the nonlinear function that models the process, which is implemented by the UT. These points are a deterministic function of the mean state vector, $\hat{x}(k|k)$, and covariance matrix, $P(k|k)$. The number of “sigma points”, χ_j , to be defined is $2L + 1$ and depends on the dimension of the state vector (L). They are given by the following expressions:

$$\chi_0(k|k) = \hat{x}(k|k) \quad (A.51)$$

$$\chi_j(k|k) = \hat{x}(k|k) + \left[\sqrt[{}^c]{(L + \lambda)P(k|k)} \right]_j \quad \forall j \in \{1, \dots, L\} \quad (A.52)$$

$$\chi_j(k|k) = \hat{x}_j(k|k) - \left[\sqrt[{}^c]{(L + \lambda)P(k|k)} \right]_{j-L} \quad \forall j \in \{L + 1, \dots, 2L\} \quad (A.53)$$

The subscript in the square brackets represents the column number (i.e. $[P(k|k)]_j$ is the j -th column of the matrix $P(k|k)$) and $\sqrt[{}^c]{\cdot}$ is the Cholesky square root. λ is the scaling parameter, whose value is given by:

$$\lambda = \alpha^2(L + \kappa) - L \quad (A.54)$$

Where α and κ are tuning parameters.

Calculation of the “sigma points” weights

This is the second step of the forward UT and its target is the computation of the “sigma point” weights, which are given by:

$$W_0^m = \frac{\lambda}{L + \lambda} \quad (A.55)$$

$$W_0^c = \frac{\lambda}{L + \lambda} + (1 - \alpha^2 + \beta) \quad (A.56)$$

$$W_j^m = W_j^c = \frac{1}{2(L + \lambda)} \quad (A.57)$$

Where β is another tuning parameter. The default values for the tuning parameters in case of a Gaussian density are: $\alpha \approx 1 \times 10^{-3}$, $\beta = 2$ and $\kappa = 0$. These parameters control the dispersion of the “sigma points” around the mean state value.

Reverse unscented transform

This step computes back the mean and covariance matrix taking as input both, the “sigma points” and the “sigma points” weights. The equations that define this process are:

$$\bar{x}(k|k) = \sum_{j=0}^{2L} W_j^m \chi_j \quad (A.58)$$

$$\bar{P}(k|k) = \sum_{j=0}^{2L} W_j^c \left(\chi_j(k|k) - \bar{x}(k|k) \right) \left(\chi_j(k|k) - \bar{x}(k|k) \right)^T \quad (A.59)$$

The use of this transform is based on the fact that the propagation of the conditional state density keeps most of its features once passed through a nonlinear function as it is the case when propagating the state and the measurement following the process and measurement models.

Unscented Kalman Filter

The UKF algorithms exploit the UT while estimating the state of the system and its covariance matrix. It follows a structure similar to the one of the Kalman filter, that is, there are two steps, one devoted to the propagation of the state vector density through the nonlinear transition function defined in the system model and then the update of this density taking into consideration the received measurement.

Prediction step

This step is devoted to the propagation of the state vector and its covariance matrix to the time instant of the last received measurement. This assumes that the filtered state and covariance matrix at time $k - 1$ are available. Using the forward UT both are propagated along time.

The first step is to generate the “sigma points” at time $k - 1$, $\chi_j(k - 1|k - 1)$, by applying the forward UT to the state and covariance matrix at time $k - 1$.

Then, these values are transformed through the nonlinear system state transition function generating the transformed “sigma points”, which are given by:

$$\chi_j(k|k - 1) = f(k, \chi_j(k - 1|k - 1)) \quad \forall j \in \{0, \dots, 2L\} \quad (A.60)$$

The “predicted” or “a priori” mean and covariance matrix is then obtained by applying the backward unscented transform and adding the effect of the plant noise, as it is shown hereafter:

$$\bar{x}(k|k - 1) = \sum_{j=0}^{2L} W_j^m \chi_j(k|k - 1) \quad (A.61)$$

$$\bar{P}(k|k - 1) = \sum_{j=0}^{2L} W_j^c \left(\chi_j(k|k - 1) - \bar{x}(k|k - 1) \right) \left(\chi_j(k|k - 1) - \bar{x}(k|k - 1) \right)^T + Q_{k-1} \quad (A.62)$$

These expressions represent the “predicted” mean state vector and covariance matrix.

Measurement update step

This step performs a similar process to generate the transformed predicted measurements. Thus, starting with the transformed “sigma points”, $\chi_j(k|k - 1)$, the process transforms them into the measurement space. This is done by applying the measurement model stated in A.50 to the transformed “sigma points” as shown below:

$$Z_j(k|k - 1) = h(k, \chi_j(k|k - 1)) \quad \forall j \in \{0, \dots, 2L\} \quad (A.63)$$

Applying the backward UT, the “predicted” measurement mean $\bar{z}$, covariance matrix $\bar{P}_z(k|k - 1)$, and cross-correlation matrix $\bar{P}_{xz}(k|k - 1)$ are computed as:

$$\bar{z}(k|k - 1) = \sum_{j=0}^{2L} W_j^m Z_j(k|k - 1) \quad (A.64)$$

$$\bar{P}_z(k|k-1) = \sum_{j=0}^{2L} W_j^c \left(Z_j(k|k-1) - \bar{z}(k|k-1) \right) \left(Z_j(k|k-1) - \bar{z}(k|k-1) \right)^T + R_k \quad (\text{A.65})$$

$$\bar{P}_{xz}(k|k-1) = \sum_{j=0}^{2L} W_j^c \left(\chi_j(k|k-1) - \bar{x}(k|k-1) \right) \left(Z_j(k|k-1) - \bar{z}(k|k-1) \right)^T \quad (\text{A.66})$$

Filtering step

The last step consists of applying the Kalman Filter equations to the values transformed back to the state space and to the measurement space to obtain the “filtered” mean state and covariance matrix. The first thing is to compute the Kalman Gain which is given by:

$$K(k) = \bar{P}_{xz}(k|k-1) \bar{P}_z(k|k-1)^{-1} \quad (\text{A.67})$$

Once calculated, the “filtered” mean state vector and covariance matrix are given by:

$$\hat{x}(k|k) = \bar{x}(k|k-1) + K(k)[z(k) - \bar{z}(k|k-1)] \quad (\text{A.68})$$

$$P(k|k) = \bar{P}(k|k-1) - K(k) \bar{P}_z(k|k-1) K^T(k) \quad (\text{A.69})$$

A.6.5 Final considerations

The UKF shows some advantages compared to the KF and EKF, mainly related to the higher order approximation that it performs to the conditional state density that allows capturing higher order statistics than the KF and EKF. Additionally, it avoids the calculation of the Jacobian matrix which is a source of problems in many situations. On the other hand, the disadvantages are mainly related to the processing time and the fact that the UKF is not able to cope with multi-modal densities [Veth 15].

A.7 Interacting multiple model filter

Interacting multiple model (IMM) filters are one of the possible alternatives when a multiple model (MM) approach is chosen to estimate the state of a hybrid system (i.e. a manoeuvring target) [Blom 84]. They belong to a class of estimators known as Gaussian sum filters. The other common multiple-model alternatives are the Multiple Model Adaptive Estimation (MMAE) [Magill 65] and the Generalized Pseudo-Bayesian (GPB) algorithms [Ackerson 70]. Due to the better performances of the IMM filter for the same computational load, it is preferred to its equivalent in the GPB approach, that is the GPB of order 2. The IMM has shown to be one of the most cost-effective hybrid state estimation schemes [Bar-Shalom 01], which makes it one of the integral parts of any ground or air tracker that can be found in current ATC systems. Hence, there are plenty of references presenting different variants for this algorithm.

Some examples focused on the air surveillance are [Li 93], where an IMM filter based in two modes is described, one for non-maneuvring targets and the other for manoeuvring ones; [Blom 92] seems to be the foundation of what would become the ARTAS system

[ARTAS], whose IMM-based tracking algorithm is presented in [Hogendoorn 99]; or [Mathias 07], who describes the IMM filter that is implemented in the PHOENIX tracker [Wikipedia].

However, the IMM algorithm is also part of the surveillance function of A-SMGCS systems or surface surveillance trackers like in [García 05a] who describes a data fusion architecture that incorporates an IMM filter intended to the Adolfo Suarez-Madrid Barajas A-SMGCS system; [Van Hintum 13] describes the IMM-based algorithmic extension of ARTAS towards ground surveillance; or [López-Araquistain 19], who presents a further evolution of ground surface surveillance based on exploiting the airport map that also relies on an IMM.

A.7.1 General description

The IMM is a suboptimal hybrid state estimator that incorporates several linear filters (generally Kalman filters or EKF, although they can also be UKF) working in parallel, each of them matched to a possible “system mode”, i.e. $m(k)$, whose outputs (state estimate and covariance matrix) are afterwards conveniently combined in order to obtain the output of the IMM filter. This is done by means of a procedure in charge of estimating the probability of the target being in each of the possible modes. It is relevant to note that the IMM filter introduces an initial step (interacting logic) to compute the inputs that will enter each one of the parallel filters. Consequently, the inputs are obtained as a mixture of the previous stage outputs coming from the different matched filters.

A.7.2 Assumptions of the IMM filter

It is assumed that the evolution of the system mode behaves as a homogeneous Markov chain, which is characterized by its probability transition matrix, whose elements represent the probability of being in the j -th system mode given that the system mode in the previous time instant was the i -th one:

$$\pi_{ij} = P\{m_j(k+1)|m_i(k)\} \quad (\text{A.70})$$

Each system mode-matched filter will stick to the assumptions of its own filter type.

A.7.3 Types of IMM filter

Due to the good compromise between performance and complexity, the IMM is a very extended filter that has been applied to the state estimation of many different systems. In the ATC domain, it is quite often found together with the association technique that has been selected, such as IMMPDA, IMMJPDA or IMMNN, depending whether the association logic is based on Probabilistic Data Association (PDA), on Joint Probabilistic Data Association (JPDA) or the Nearest Neighbour (NN) algorithm.

It is also possible to find IMM filters named after the type of the mode-matched filters, like IMMKF, IMMEKF, IMMUKF or IMPPF, for referring to IMM filters which rely on Kalman filters, EKF, UKF or particle filters, just for naming some of them.

Another family of IMM filters is the variable structure ones (VS-IMM), which are also quite extended on the ATC domain, mainly on surface surveillance as they introduce an additional logic in order to manage the number and particularities of the “active” mode-matched filters depending on certain situations, which could integrate additional information.

Additionally, there are also multi-hypothesis IMM (MH-IMM), adaptative IMM estimators, or multisensor IMM filters. A review of the IMM filters used for target tracking can be found in [Mazor 98].

A.7.4 IMM filter equations

The equations of a generic IMM filter differ in part from the ones of the previously shown as they refer to an additional logic in charge of managing the evolution of the different filters that run in parallel. They can be found, for example, in [Li 93].

Process and measurement models

The process and measurement models might be a selection of models which can belong to any of the previously presented in sections 7.4.6.2.1 and 7.4.6.2.2. Therefore, the system state vector could present a linear or nonlinear evolution with time, such as:

$$x(k+1) = f(k, x(k), m(k)) + w(k, m(k)) \quad (A.71)$$

or

$$x(k+1) = F(k, m(k))x(k) + w(k, m(k)) \quad (A.72)$$

The term $m(k)$ refers to the system mode that is considered in a certain time instant k . Following a similar presentation, the measurement or observation model can be described as:

$$z(k) = h_j(k, x(k), m(k)) + v_j(k, m(k)) \quad (A.73)$$

or

$$z(k) = H_j(k, m(k))x(k) + v_j(k, m(k)) \quad (A.74)$$

In this case, the subindex j refers to the sensor that is providing the measurement at time instant k . Meaning this way that the filter is ready to work in a multisensor environment. Please note that for the sake of clarity, this subindex has been omitted in the previous filters although they are also capable of dealing with several sensors.

Interaction (mixing)

In the case of the IMM, the first step consists in the computation of the inputs that will enter each of the mode-matched filters, which take part in the filtering process. Thus, the mixed initial state $\hat{x}^{0j}(k-1|k-1)$ and covariance matrix $P^{0j}(k-1|k-1)$ are obtained by combining the last outputs of the matched filters calculated in the previous update time $k-1$.

The equation for the mixed initial state is:

$$\hat{x}^{0j}(k-1|k-1) = \sum_{i=1}^r \hat{x}^i(k-1|k-1) \mu_{i|j} \quad (A.75)$$

Where $\mu_{i|j}$ is the mixing probability:

$$\mu_{i|j} = \pi_{ij} \mu_i (k-1) / \mu_j^- \quad (A.76)$$

Which is computed using the value of the predicted mode probability μ_j^- :

$$\mu_j^- = \sum_i \pi_{ij} \mu_i (k-1) \quad (A.77)$$

The initial covariance matrix is:

$$P^{0j}(k-1|k-1) = \sum_{i=1}^r \mu_{i|j} (k-1|k-1) P^i(k-1|k-1) + X_j \quad (A.78)$$

Being X_j the “spread-of -the means” term, which has the following expression:

$$X_j = \sum_i [\hat{x}^i(k-1|k-1) - \hat{x}^{0j}(k-1|k-1)] [\hat{x}^i(k-1|k-1) - \hat{x}^{0j}(k-1|k-1)]^T \mu_{i|j} \quad (A.79)$$

Filtering

The next step consists in filtering each of the previously calculated initial inputs by means of its mode-matched filter. Consequently, for a generic filter "j" the equations would be:

Prediction for the j-th mode

$$\hat{x}_j(k|k-1) = F_j(k) \hat{x}_{0j}(k-1|k-1) \quad (A.80)$$

$$P_j(k|k-1) = F_j(k) P(k-1|k-1) F_j^T(k) + Q_j(k) \quad (A.81)$$

Gain computation

$$S_j(k) = H_j(k) P_j(k|k-1) H_j^T(k) + R_j(k) \quad (A.82)$$

$$K_j(k) = P_j(k|k-1) H_j^T(k) S_j^{-1}(k) \quad (A.83)$$

Filtering step

$$v_j(k) = z(k) - H_j(k) \hat{x}_j(k|k-1) \quad (A.84)$$

$$\hat{x}_j(k|k) = \hat{x}_j(k|k-1) + K_j(k) v_j(k) \quad (A.85)$$

$$P_j(k|k) = P_j(k|k-1) - K_j(k) S_j(k) K_j^T(k) \quad (A.86)$$

Mode probability update

The likelihood function takes the form:

$$\Lambda_j = |2\pi S_j|^{-1/2} e^{-\frac{1}{2} v_j^T(k) S_j^{-1}(k) v_j(k)} \quad (\text{A.87})$$

and the mode probability:

$$\mu_i(k) = \frac{\mu_j^{\bar{}} \Lambda_j}{\sum_i \mu_i^{\bar{}} \Lambda_i} \quad (\text{A.88})$$

Estimate and covariance combination

Finally, the output of the IMM filter is computed taking into consideration all the previous results. Thus, the estimate and covariance matrix look like:

$$\hat{x}(k|k) = \sum_j \hat{x}_j(k|k) \mu_i(k) \quad (\text{A.89})$$

$$P(k|k) = \sum_j P_j(k|k) \mu_i(k) - X \quad (\text{A.90})$$

Where X is:

$$X = \sum_i [\hat{x}_i(k|k) - \hat{x}(k|k)] [\hat{x}_i(k|k) - \hat{x}(k|k)]^T \mu_i(k) \quad (\text{A.91})$$

A.7.5 Final considerations

The IMM is nowadays one of the preferred options when tracking a manoeuvring target as it shows three desirable properties. The first one is that it is recursive, so there is no need to process batches of data. The second one is its modular architecture, which allows including or removing system modes in an easy way. Finally, its computational load is fixed, so once decided the number of modes that are going to be implemented the computational resources per cycle are known [Bar-Shalom 95].

AIRCRAFT INFORMATION

B.1 Target information retrieval

Thanks to the ICAO 24-bit address (i.e. AA), it is possible to retrieve information concerning the target (e.g. its type) that might be exploited by the data processing logic.

Table 27: aircraft physical information per target type

Aircraft Type	Length (m)	Wing span (m)	Height (m)	Wheelbase (m)
ATR 42-600	22,68	24,56	7,6	8,79
Airbus A310-300	46,66	43,9	15,8	15,21
Airbus A318-100	31,44	34,1	12,56	10,25
Airbus A319-100	33,84	35,8	11,76	11,04
Airbus A319-111	33,84	35,8	11,76	11,04
Airbus A319-112	33,84	35,8	11,76	11,04
Airbus A320-200	37,57	35,8	11,76	12,64
Airbus A320-214	37,57	35,8	11,76	12,64
Airbus A321-100	44,51	35,8	11,76	16,9
Airbus A321-200	44,51	35,8	11,76	16,9
Airbus A321-212(WL)	44,51	35,8	11,76	16,9
Airbus A330-200	58,82	60,3	17,39	22,18
Airbus A330-202	58,82	60,3	17,39	22,18
Airbus A340-300	63,69	60,3	16,99	25,6
Boeing 737-800	39,5	34,32	12,5	15,6
Boeing 737-8F2(WL)	39,5	35,8	12,5	15,6
Boeing 737-8AS(WL)	39,5	35,8	12,5	15,6
Boeing 767-319(ER)	54,94	47,57	15,85	22,76
Boeing 777-31H(ER)	73,9	64,8	18,5	31,22
Embraer ERJ-145EP	29,87	20,04	6,75	14,44
Embraer ERJ-195	38,65	28,72	10,55	14,64
Fokker F100	35,53	28,08	8,5	14,01

The information concerning the different aircraft types is taken from documents released by the aircraft manufacturer like [Boeing 737], from their web site as in the case for the Airbus models, like the A320neo [Airbus A320neo] or even from databases provided by organizations like the FAA [FAA database].

In this annex, some examples are presented. These are the one corresponding to the target types of the actual targets that are found in the recording from Lisbon airport that is used for the generation of results in CHAPTER 8 and that is presented in section 8.4.1.

As the ICAO 24-bit address also identifies airport vehicles, they are also modelled within this work as it is shown in the following table.

Table 28: airport vehicle physical information

Aircraft Type	Length (m)	Wing span (m)	Height (m)	Wheelbase (m)
LPPT-GND	5	2	2	3,5

The physical information concerning airport vehicles was not available, so it has been approximated by the values shown above. The wheelbase is assumed as the distance between the front wheel axis and the rear one.

ANNEX III

ANTENNAE POSITION

C.1 A320: ATC and TCAS antenna location

C.2 AIRBUS family: GPS antennae location

X value with regards to the aeronautics reference point of the aircraft.

Xo = 2540mm SA,

Xo = 6382.5mm LR,

Xo = 6382.5 WB,

Xo = 4882.5mm Beluga,

Xo = 6680.2mm A380

X' = value on the X axis
referenced to the physical
nose of the a/c (top of
radome)

MMR

	X	Y	Z
A318	D230 10005		
GPS 1	6923	0	1975
GPS 2	7271,3	0	1975
A319	D230 10004		
GPS 1	6923	0	1975
GPS 2	7271,3	0	1975
A320	D230 10008		
GPS 1	6923	0	1975
GPS 2	7271,3	0	1975
A321	D230 10011		
GPS 1	6923	0	1975
GPS 2	7271,3	0	1975
A330-200	F230 10005		
GPS 1	12650	-12	2609.67
GPS 2	12950	-12	2646.29
A330-300	F230 10006		
GPS 1	12650	-12	2609.67
GPS 2	12950	-12	2646.29
A340-200	F230 10003		
GPS 1	12650	-12	2609.67
GPS 2	12950	-12	2646.29
A340-300	F230 10004		
GPS 1	12650	-12	2609.67
GPS 2	12950	-12	2646.29
A340-500	F230 10002		
GPS 1	12650,5	-12	2609.67
GPS 2	12950,5	-12	2646.29
A340-600	F230 10001		
GPS 1	12650,5	-12	2609.67
GPS 2	12950,5	-12	2646.29

X'
4383
4731,3
4383
4731,3
4383
4731,3
4383
4731,3
6267,5
6567,5
6267,5
6567,5
6267,5
6567,5
6268
6568
6268
6568

BIBLIOGRAPHY

- [Abbud 15] ABBUD, J. J. (2015). “Localización y seguimiento de aeronaves mediante sistemas de multilateración de área extensa.” Madrid: Universidad Politécnica de Madrid (UPM).
- [Ackerson 70] ACKERSON, G. & K. FU, (1970). On state estimation in switching environments. *IEEE Transactions on Automatic Control* 15 (1): pp. 10-17.
- [AENA ADS] <http://www.aena.es/csee/Satellite/navegacion-aerea/en/Page/1043829981402//Automated-Dependent-Surveillance-ADS.html>
- [AIP LPPT] NAV PORTUGAL E.P.E. LPPT AD 2 – 1. AMDT 012-1. AIRAC 004-13. 08-JAN-2016
- [Airbus A320neo] <https://www.airbus.com/aircraft/passenger-aircraft/a320-family/a320neo.html>
- [Arasaratnam 09] ARASARATNAM, I., & S., HAYKIN, (2009). Cubature Kalman Filters. *IEEE Transactions on Automatic Control* 54 (6): pp. 1254-69.
- [Arasaratnam 11] ARASARATNAM, I., & S. HAYKIN, (2011). Cubature Kalman Smoothers. *Automatica* 47 (10): 2245–50.
- [ARTAS] <https://www.eurocontrol.int/product/artas>
- [ASTERIX history] http://www.eurocontrol.int/asterix/public/standard_page/history.html
- [ASTERIX website] http://www.eurocontrol.int/asterix/public/standard_page/documents.html
- [ASTERIX Part 1] EUROCONTROL (2007). Surveillance Data Exchange - Part 1 All Purpose Structured Eurocontrol Surveillance Information Exchange - ASTERIX (October 2016).
- [ASTERIX Cat.001] EUROCONTROL (2011). Category 001: Radar Data Exchange - Part 2a Transmission of Monoradar Data Target Reports (August 2011).
- [ASTERIX Cat.002] EUROCONTROL (1997). Category 002: Radar Data Exchange - Part 2b Transmission of Monoradar Service Messages (November 1997).
- [ASTERIX Cat.010] EUROCONTROL (2007). Category 010: Surveillance Data Exchange - Part 7 Transmission of Monosensor Surface Movement Data (March 2007).
- [ASTERIX Cat.011] EUROCONTROL (2008). Category 011: Surveillance Data Exchange - Part 8 Transmission of A-SMGCS Data (May 2008).

- [ASTERIX Cat.017 App.A] EUROCONTROL (2005). Category 017 Appendix A: Coordinate Transformation Algorithms for the Hand-Over of Targets Between POEMS Interrogators (June 2005).
- [ASTERIX Cat.019] EUROCONTROL (2010). Category 019: Surveillance Data Exchange - Part 18 Multilateration System Status Messages (December 2010).
- [ASTERIX Cat.020] EUROCONTROL (2010). Category 020: Surveillance Data Exchange - Part 4 Multilateration Target Reports (December 2010).
- [ASTERIX Cat.021] EUROCONTROL (2011). Category 021: Surveillance Data Exchange - Part 12 ADS-B Reports (June 2015).
- [ASTERIX Cat.023] EUROCONTROL (2009). Category 023: Surveillance Data Exchange - Part 16 CNS/ATM Ground Station and Service Status Reports (March 2009).
- [ASTERIX Cat.032] EUROCONTROL (2009). Application of ASTERIX to ARTAS. (January 2009).
- [ASTERIX Cat.034] EUROCONTROL (2007). Category 034: Surveillance Data Exchange - Part 2b Transmission of Monoradar Service Messages (May 2007).
- [ASTERIX Cat.048] EUROCONTROL (2012). Category 048: Surveillance Data Exchange - Part 4 Transmission of Monoradar Target Reports (July 2012).
- [ASTERIX Cat.062] EUROCONTROL (2014). Category 062: Surveillance Data Exchange - Part 9 SDPS Track Messages (December 2014).
- [ASTERIX Cat.063] EUROCONTROL (2014). Category 063: Surveillance Data Exchange - Part 9 SDPS Track Messages (December 2014).
- [ASTERIX Cat.065] EUROCONTROL (2014). Category 065: Surveillance Data Exchange - Part 15 SDPS Service Status Reports (August 2014).
- [Auer 14] AUER, J., H. KOBELBAUER, H. SCHRANZ, E. BERNDT & W. LANGHANS, (2014). ATM system integration of a nationwide wide-area multilateration system: Integrating WAM into a state of the art ATM system including an ARTAS tracker. In *2014 Tyrrhenian International Workshop on Digital Communications - Enhanced Surveillance of Aircraft and Vehicles (TIWDC/ESAV)*, pp. 96–100.
- [Aveillant] <https://www.aveillant.com/products/>
- [BADA] <https://www.eurocontrol.int/services/bada>
- [Balakrishnan 04] BALAKRISHNAN, H., I. HWANG, J. S. JANG & C. TOMLIN, (2004). Inference Methods for Autonomous Stochastic Linear Hybrid Systems. In *Hybrid Systems: Computation and Control, Lecture Notes in Computer Science*, Alur, R. & Pappas, G. J. (Eds.), pp. 64-79. Springer Berlin Heidelberg.

- [Bar-Shalom 88] BAR-SHALOM, Y., (1988). *Tracking and data association*. Boston: Academic Press.
- [Bar-Shalom 95] BAR-SHALOM, Y. & X.-R. LI, (1995). *Multitarget-multisensor tracking: principles and techniques*. Storrs, Conn.: YBS.
- [Bar Shalom 00] BAR-SHALOM, Y., (2000). *Multitarget-Multisensor Tracking : Applications and Advances*. Yaakov Bar-Shalom, ed. Vol. 3. Boston: Artech House.
- [Bar-Shalom 01] BAR-SHALOM, Y., (2001). *Estimation with applications to tracking and navigation*. New York: Wiley.
- [Bar-Shalom 07] BAR-SHALOM, Y., X.-R. LI & T. KIRUBARAJAN, (2007). *Estimation with Applications to Tracking and Navigation: Theory, Algorithms and Software*. A Wiley-Interscience Publication. New York, NY: Wiley.
- [Bartels 87] BARTELS, R. H., J. C. BEATTY & B. A. BARSKY, (1987). *An introduction to splines for use in computer graphics and geometric modeling*. Los Altos, Calif.: M. Kaufmann Publishers.
- [Barton 92] BARTON, D. K., (1992). *Modern radar system analysis software*. Version 2.0. Boston: Artech House.
- [Besada 00] BESADA, J. A., J. GARCÍA, G. DE MIGUEL, J. R. CASAR & G. GAVÍN, (2000). ADS Bias Cancellation Based on Data Fusion with Radar Measurements. In *Proceedings of the Third International Conference on Information Fusion*, Paris, France.
- [Besada 04a] BESADA, J. A., J. GARCÍA & G. DE MIGUEL, (2004). New Approach to Online Optimal Estimation of Multisensor Biases. *Radar, Sonar and Navigation, IEE Proceedings* - 151 (1): 31-40.
- [Besada 04b] BESADA, J. A., J. GARCÍA, A. SOTO & M. C. GÓMEZ, (2004). On-line sensor calibration for airport data fusion. In *Proceedings of the IEEE Radar Conference, 2004.*, pp. 175–180.
- [Besada 05] BESADA, J. A., A. SOTO, G. DE MIGUEL & J. I. PORTILLO, (2005). Design of an A-SMGCS prototype at Barajas airport: airport surveillance sensors bias estimation. In *2005 8th International Conference on Information Fusion*, vol. 2, p. 8 pp.
- [Besada 08a] BESADA, J., G. DE MIGUEL, A. SOTO, J. GARCIA, R. ALCAZAR & E. VOET, (2008). TRES: Multiradar-multisensor data processing assessment using opportunity targets. In *Radar Conference, 2008. RADAR'08. IEEE*, , pp. 1–6.
- [Besada 08b] BESADA. J. A., G. DE MIGUEL, A. SOTO, A. M. BERNARDOS & J. GARCÍA, (2008). Algorithms for Opportunity Trajectory Reconstruction. In *Proceedings of ESAV'08*. Capri, Italy.
- [Besada 09] BESADA, J. A., G. DE MIGUEL, P. TARRIO, A. M. BERNARDOS & J. GARCIA, (2009). Bias Estimation for Evaluation of ATC Surveillance Systems.

In 2009 12th *International Conference on Information Fusion*, Seattle, USA.

- [Besada 12] BESADA, J. A., G. DE MIGUEL, A. M. BERNARDOS & J. R. CASAR, (2012). Generic Multisensor Multitarget Bias Estimation Architecture. *IET Radar, Sonar Navigation* 6 (5): 365–78.
- [Besada 13] BESADA, J. A., A. SOTO, G. DE MIGUEL, J. GARCIA & E. VOET, (2013). ATC trajectory reconstruction for automated evaluation of sensor and tracker performance. *IEEE Aerospace and Electronic Systems Magazine*, 28, 2 (2013), 4–17.
- [Besada 14] BESADA, J. A., J. LÓPEZ-ARAQUISTAIN, G. DE MIGUEL & A. SOTO. (2014). Performance Requirements Specification Design for Airport Surface Surveillance. In *2014 Tyrrhenian International Workshop on Digital Communications - Enhanced Surveillance of Aircraft and Vehicles (TIWDC/ESAV)*, 23–28.
- [Best 97] BEST, R.A & J. P. NORTON, (1997). A new model and efficient tracker for a target with curvilinear motion. *IEEE Transactions on Aerospace and Electronic Systems* 33 (3): pp. 1030-37.
- [Blackman 99] BLACKMAN, S. S., (1999). *Design and analysis of modern tracking systems*. Artech House radar library. Boston: Artech House.
- [Blom 84] BLOM, H. A. P., (1984). An efficient filter for abruptly changing systems. In *The 23rd IEEE Conference on Decision and Control, 1984*, 656-58.
- [Blom 84a] BLOM, H. A. P., (1984). A sophisticated tracking algorithm for Air Traffic Control (ATC) surveillance radar data. *NASA Technical Reports* N8617334 (NLR-MP-84004-U): 7.
- [Blom 89] BLOM, H. A. P. & Y. BAR-SHALOM, (1989). Time-reversion of a hybrid state stochastic difference system. In *IEEE International Conference on Control and Applications, 1989. Proceedings. ICCON '89*, pp. 619 – 624.
- [Blom 90] BLOM, H. A. P. & Y. BAR-SHALOM, (1990). Time-reversion of a hybrid state stochastic difference system with a jump-linear smoothing application. *IEEE Transactions on Information Theory*, 36, 4 (July 1990), 836 –847.
- [Blom 92] BLOM, H. A. P., R. A. HOGENDOORN & B. A. VAN DOORN, (1992). Design of a Multisensor Tracking System for Advanced Air Traffic Control. In Y. Bar-Shalom (ed.) *Multitarget-Multisensor Tracking : Applications and Advances*, Vol. II, Artech House.
- [Blue book] EUROCONTROL. (1997). EUROCONTROL Standard Document for Radar Surveillance in En-Route Airspace and Major Terminal Areas. Edition: 1.0.

- [Boing 737] BOEING (2013). 737 Airplane Characteristics for Airport Planning. D6-58325-6 (September 2013).
- [Branicky 96] BRANICKY, M. S., (1996). General hybrid dynamical systems: Modeling, analysis, and control. In *Hybrid Systems III, Lecture Notes in Computer Science*, Alur, R., Henzinger, T. A. & Sontag, E. D. (Eds.), pp. 186–200. Springer Berlin Heidelberg.
- [Brown 97] BROWN, R. G., (1997). *Introduction to random signals and applied Kalman filtering: with MATLAB exercises and solutions*. 3rd ed. New York: Wiley.
- [Brown 12] BROWN, R. G. & P. Y. C. HWANG, (2012). *Introduction to random signals and applied Kalman filtering with MATLAB exercises*. 3rd ed. New York: Wiley.
- [B-splines] http://ict.udlap.mx/people/oleg/docencia/IMAGENES/chapter5/image_50_IS548.html
- [Calvo 07] CALVO, J. A., (2007). *Fundamentos de navegación aérea*. Cantoblanco (Madrid): Ediciones Universidad Autónoma de Madrid.
- [Cassandras 07] CASSANDRAS, C. G. & J. LYGEROS, (eds.) (2007). *Stochastic hybrid systems*. Boca Raton: CRC/Taylor & Francis.
- [CARPET 15] <https://www.tno.nl/en/focus-area/defence-safety-security/information-superiority/carpet-computer-aided-radar-performance-evaluation-tool/>
- [Chang 06] CHAN, K.-S., L.-H. HO & H. TONG, (2006). A Note on Time-Reversibility of Multivariate Linear Processes. *Biometrika*, 93, 1 (March 2006), 221–227.
- [Chen 87] CHEN, M. J. & J. P. NORTON. (1987). Estimation Technique for Tracking Rapid Parameter Changes. *International Journal of Control* 45 (4): 1387–98.
- [Cheng 99] CHENG, Q., (1999). On Time-Reversibility of Linear Processes. *Biometrika*, 86, 2 (June 1999), 483–486.
- [Coetzee 06] COETZEE, E., B. KRAUSKOPF & M. H. LOWENBERG, (2006). Nonlinear Aircraft Ground Dynamics. *International Conference on Nonlinear Problems in Aviation and Aerospace*.
- [Coetzee 11] COETZEE, E., (2011). Modelling and Nonlinear Analysis of Aircraft Ground Manoeuvres. Ph.D., University of Bristol.
- [Comer 13] COMER, D. E., (2013). *Internetworking with TCP/IP*. Upper Saddle River. N.J: Pearson Prentice Hall.
- [Cruz 92] CRUZ, E. D., A. T. ALOUANI, T. R. RICE & W. D. BLAIR, (1992). Sensor registration in multisensor systems. In *Signal and Data Processing of*

- Small Targets 1992*, 1698:382-93. International Society for Optics and Photonics.
- [CSO 03] COMSOFT, (2003). Surveillance Supervision. Company profile. www.airtrafficmanagement.net.
- [Davis 93] DAVIS, M. H. A., (1993). *Markov models and optimization*. 1st ed. London; New York: Chapman & Hall.
- [De Boor 66] DE BOOR, C. (1966). *The Method of Projections as Applied to the Numerical Solution of Two Point Boundary Value Problems Using Cubic Splines*.
- [De Boor 01] DE BOOR, C., (2001). *A practical guide to splines: with 32 figures*. New York: Springer.
- [De Boor 01a] DE BOOR, C., (2001). Spline Basics. In *CAGD Handbook*.
- [De Miguel 08] DE MIGUEL, G., J. IGLESIAS, J. A. BESADA & J. GARCÍA, (2008). Integration of ADS-B surveillance data in operative multiradar tracking processors. In *2008 11th International Conference on Information Fusion*, pp. 1–8. 10.1109/ICIF.2008.4632309.
- [Definedterm] <https://definedterm.com/>
- [Drouilhet 73] DROUILHET, P., (1973). The Development of the ATC Radar Beacon System: Past, Present, and Future. *IEEE Transactions on Communications*, 21, 5 (May 1973), 408– 421.
- [Drummond 93] DRUMMOND, O. E., (1993). Multiple Target Tracking with Multiple Frame, Probabilistic Data Association. In *Signal and Data Processing of Small Targets 1993*, 1954:394–408. International Society for Optics and Photonics.
- [Drummond 97] DRUMMOND, O. E., (1997). Target Tracking with Retrodicted Discrete Probabilities. In *Signal and Data Processing of Small Targets 1997*, 3163:249–68. International Society for Optics and Photonics.
- [EATM 2004] EATM (2004). Aeronautical Information Lexicon - EATM. (2004). <http://www.eurocontrol.int/aim/gallery/content/public/pdf/lexicon-eatm-1.pdf>.
- [ECTL 97] EUROCONTROL, (1997). Radar Sensor Performance Analysis. (June 1997).
- [ECTL 2004] EUROCONTROL, (2004), EUROCONTROL OPERATIONAL CONCEPT DOCUMENT, Vol. 1, Ed. 2.1.
- [ECTL A-SMGCS] EUROCONTROL, (2010). Definition of A-SMGCS Implementation Levels. Edition: 1.2. (June 2010).

- [ECTL A-SMGCS 20] EUROCONTROL, (2020). EUROCONTROL Specification for Advanced-Surface Movement Guidance and Control System (A-SMGCS) Services. Edition: 2.0 (April 2020).
- [Engels 2008] ENGELS, K. & R. HEIDGER, (2008). An infrastructure for online tracking quality control. In *2008 TIWDC/ESAV 2008. Tyrrhenian International Workshop on Digital Communications - Enhanced Surveillance of Aircraft and Vehicles*, pp. 1-7.
- [ERNIP P3 12] EUROCONTROL DNM/NOM/OPL, (2012). European Route Network Improvement Plan - PART 3 - Airspace Management Handbook Guidelines for Airspace Management. (October 2012).
- [ESASSP 12] EUROCONTROL, (2012). ESASSP- EUROCONTROL Specification for ATM Surveillance System Performance. Edition: 1.0.
- [ESASSP 15] EUROCONTROL, (2015). ESASSP- EUROCONTROL Specification for ATM Surveillance System Performance. Edition: 1.1.
- [EUROCAE ED102] EUROCAE, (2000). ED-102, MOPS for 1090 Mhz Automatic Dependent Surveillance-Broadcast (ADS-B). (November 2000).
- [EUROCAE ED102A] EUROCAE, (2009). ED-102A, MOPS for 1090 MHz Extended Squitter Automatic Dependent Surveillance – Broadcast (ADS-B) and Traffic Information Services – Broadcast (TIS-B). (December 2009).
- [EUROCAE ED108A] EUROCAE, (2005). ED-108A, MOPS for the Very High Frequency (VHF) Digital Link (VDL) Mode 4 Aircraft Transceiver. (October 2005).
- [EUROCAE ED117] EUROCAE-WG-41, (2003). ED-117, MOPS for Mode S multilateration systems for use in advanced surface movement guidance and control systems (A-SMGCS). (November 2003).
- [EUROCAE ED142] EUROCAE-WG-70, (2010). ED-142, Technical specification for wide area multilateration (WAM) systems. (September 2010).
- [EUROCAE ED222] EUROCAE, (2013). ED-222, Aeronautical Mobile Airport Communications System (AeroMACS) profile. (November 2013).
- [EUROCAE ED223] EUROCAE, (2013). ED-223, MOPS for the Aeronautical Mobile Airport Communication System. (October 2013).
- [EUROCONTROL] <http://www.eurocontrol.int>
- [FAA 07] FAA, (2015). Market Survey for FAA Radar Support System (RSS). (October 2015).
- [FAA AC 96] FAA, (1996). Advisory Circular - Surface Movement Guidance and Control System. (December 1996).
- [FAA database] https://www.faa.gov/airports/engineering/aircraft_char_database/

- [Fielding 87] FIELDING, J. E., (1987). *RGCalc radar range detection software and user's manual*. Norwood, MA: Artech House.
- [Fielding 88] FIELDING, J. E., (1988). *VCCalc vertical coverage calculation program: software and user's manual*. Version 1.0. Boston: Artech House.
- [Flow management] <http://www.eurocontrol.int/nm-services/184>
- [Fraser 69] FRASER, D. & J. POTTER, (1969). The optimum linear smoother as a combination of two optimum linear filters. *IEEE Transactions on Automatic Control*, 14, 4 (August 1969), 387 – 390.
- [Frontera 14] FRONTERA, G., D. J. MARTIN, J. A. BESADA, AND A. SOTO, (2013). Controller Support Tools for Merging and Monitoring. In *Digital Avionics Systems Conference (DASC), 2013 IEEE/AIAA 32nd*, pp. 1E4-1:1E4-16.
- [Frontera 16] FRONTERA, G., D. J. MARTIN, J. A. BESADA, AND A. SOTO, (2016). Advanced Controller Support Tools for Sequencing and Merging Operations. *Proceedings of the Institution of Mechanical Engineers, Part G: Journal of Aerospace Engineering*, March.
- [Furuti 2008] <http://www.progonos.com/furuti/MapProj/CartIndex/cartIndex.html>
- [Galati 05] GALATI, G., M. LEONARDI, P. MAGARO, Y V. PACIUCCI, (2005). Wide Area Surveillance Using SSR Mode S Multilateration: Advantages and Limitation. In *European Radar Conference, 2005. EURAD 2005*, pp. 225-29.
- [García 01] GARCIA, J., J. A BESADA, F. J JIMENEZ, AND J. R CASAR, (2001). Surface Movement Radar Data Processing Methods for Airport Surveillance. *IEEE Transactions on Aerospace and Electronic Systems* 37 (2): 563–85.
- [García 03] GARCÍA, J., G. DE MIGUEL, A. BERLANGA, J. M. MOLINA & A. SOTO, (2003). ASDE-radar data association for robust and efficient tracking on airport surface. In *Proceedings of International Radar Symposium 2003*, pp. 173–178. Dresden, Germany.
- [García 05] GARCÍA, J., DE MIGUEL, G., BESADA, J.A. & CASAR, J.R. (2005). Efficient blobs-to-tracks assignment for ASDE-radar data processing. *IEE Proceedings Radar, Sonar and Navigation*, 152, n.º 2, pp. 68-80.
- [García 05a] GARCIA, J., J. M. MOLINA, G. DE MIGUEL, & A. SOTO, (2005). Design of an A-SMGCS prototype at Barajas airport: data fusion algorithms. In *Proceedings of the 8th International Conference on Information Fusion, 2005*, pp. 2-8.
- [García 07] GARCIA, J., MOLINA, J. M., BESADA, J. A. & DE MIGUEL, G. (2007). Model-based trajectory reconstruction using IMM smoothing and motion

- pattern identification. In *2007 10th International Conference on Information Fusion*, pp. 1–8. IEEE.
- [García 08] GARCÍA, J., A. SOTO, G. DE MIGUEL, J. A. BESADA & P. TARRÍO (2008). Trajectory reconstruction techniques for evaluation of ATC systems. In *Proceedings of ESAV'08*,. Capri, Italy.
- [García 09] GARCÍA, J., J. A. BESADA, A. SOTO & G. DE MIGUEL, (2009). Opportunity Trajectory Reconstruction Techniques for Evaluation of ATC Systems. *International Journal of Microwave and Wireless Technologies*, 1, Special Issue 03, pp. 231–238.
- [Garcia 10] GARCIA, M. & T. GILBERT, (2010). ADS-B mops updates & impact on surveillance & broadcast services system (SBSS). In *Integrated Communications Navigation and Surveillance Conference (ICNS), 2010*, pp. L6–1:L6–10.
- [Gauss 1857] GAUSS, C. F. & C. H. DAVIS, (1857). *Theory of the motion of the heavenly bodies moving about the sun in conic sections. A translation of Gauss's "Theoria motus," with an appendix*. Boston Little, Brown.
- [Gelb 74] GELB, A., (1974). *Applied Optimal Estimation*. MIT Press.
- [GEN-SUR SPR] EUROCAE, (2020). Draft ED-261 “Safety and Performance Requirements Standard for a Generic Surveillance System - GEN-SUR SPR”.
- [GEN-SUR SPR Annex J] EUROCAE, (2020). Safety and Performance Requirements Standard for a Generic Surveillance System (GEN-SUR SPR) - Volume 2 - ‘Application of GEN-SUR to ADS-B’ Annex J. Draft Edition: 15 January 2020.
- [Ghahramani 01] GHAHRAMANI, Z., (2001). An introduction to hidden Markov models and Bayesian networks. In *International Journal of Pattern Recognition and Artificial Intelligence*, pp. 9–42. River Edge, NJ, USA: World Scientific Publishing Co., Inc.
- [Ghosh 97] GHOSH, M., A. ARAPOSTATHIS & S. MARCUS, (1997). Ergodic Control of Switching Diffusions. *SIAM Journal on Control and Optimization*, 35, 6 (November 1997), pp. 1952–1988.
- [Goodwin 00] GOODWIN, G. C., S. F. GRAEBE & M. E. SALGADO, (2000). *Control System Design*. US ed. Pearson US Imports & PHIPEs.
- [Guiar 86] GUIAR, C. N., F. L. LANSING & R. RIGGS, (1987). Antenna pointing systematic error model derivations. In *TDA Progress Report 42-88*, pp. 36-46.
- [Gustafsson 96] GUSTAFSSON, F., & A. J. ISAKSSON, (1996). Best choice of coordinate system for tracking coordinated turns. In *Proceedings of the 35th IEEE Conference on Decision and Control*, vol.3, pp. 3145-50.

- [Hansberg 08] HASBERG, C., S. HENSEL, M. WESTENKIRCHNER & K. BACH, (2008). Integrating Spline Curves in Road Constraint Object Tracking. In *2008 11th International IEEE Conference on Intelligent Transportation Systems*, pp. 1009-14.
- [Heidger 04] HEIDGER, R., T. KLENNER & R. MALLWITZ, (2004). The PHOENIX Multi-Radar Tracker System for Air Traffic Control Applications. *Air Traffic Control Quarterly* 12 (3): 193–222.
- [Heidger 07] HEIDGER, R. & H. S. NGUYEN, (2007). An Analysis Working Position for ATC Radar Data Processing Quality Control. In *ESAVS 2007 conference proceeding*. Bonn, Germany.
- [Helferty 94] HELFERTY, J. P., (1994). Improved tracking of maneuvering targets: the use of turn-rate distributions for acceleration modeling. In *Multisensor Fusion and Integration for Intelligent Systems, 1994. IEEE International Conference on MFI '94*, pp. 515–520.
- [Helmick 93] HELMICK, R. E., W. D. BLAIR & S. A. HOFFMAN, (1993). Interacting multiple-model approach to fixed-interval smoothing. In *Proceedings of the 32nd IEEE Conference on Decision and Control*, vol.4, pp. 3052–57
- [Helmick 95] HELMICK, R. E., BLAIR, W. D. & HOFFMAN, S. A., (1995). Fixed-interval smoothing for Markovian switching systems. *IEEE Transactions on Information Theory*, 41, 6 (November 1995), 1845–55.
- [Hespanha 04] HESPANHA, J. P., (2004). Stochastic Hybrid Systems: Application to Communication Networks. In *Hybrid Systems: Computation and Control, Lecture Notes in Computer Science*, Alur, R. & Pappas, G. J. (Eds.), pp. 387–401. Springer Berlin Heidelberg.
- [Hogendoorn 99] HOGENDOORN, R.A., C. REKKAS & W.H.L. NEVEN, (1999). ARTAS: An IMM-based Multisensor Tracker. In *Proceedings of the 2nd International Conference on Information Fusion Workshop & Exhibition*. Sunnyvale, California, USA.
- [Horrell 93] HORRELL, J. M. & M. R. INGGS, (1993). Satellite and airborne SAR simulator. In *Proceedings of the 1993 IEEE South African Symposium on Communications and Signal Processing*, pp. 193–198.
- [Hu 00] HU, J. & S. SASTRY, (2000). Towards a Theory of Stochastic Hybrid Systems. In *Hybrid Systems: Computation and Control, , Lecture Notes in Computer Science*, Lynch, N. & Krogh, B. H. (Eds.), pp. 160–173. Springer Berlin Heidelberg
- [Huizing 91] HUIZING, A. G. & A. THEIL, (1991). CARPET: A Versatile Program for Radar Performance Analysis. *Final Report Physics and Electronics Lab. TNO, The Hague (Netherlands)*.

- [Huizing 93] HUIZING, A. G., (1993). *CARPET (Computer-aided radar performance evaluation tool) radar performance analysis software and user's manual*. Version 1.0. Boston: Artech House.
- [Hwang 06] HWANG, I., H. BALAKRISHNAN & C. TOMLIN, (2006). State estimation for hybrid systems: applications to aircraft tracking. *IEE Proceedings of Control Theory and Applications* 153 (5): pp. 556-66.
- [IBM 2010] http://publib.boulder.ibm.com/infocenter/jviewmap/v8r7/topic/com.ibm.ilog.jviews.defense.doc/Content/Visualization/Documentation/JViews/JViews_Defense/_pubskel/ps_usrprgmaps892.html
- [ICAO Annex 3] ICAO, (2007). *Annex 3 - Meteorological Service for International Air Navigation*. International Civil Aviation Organization.
- [ICAO Annex 10 Vol 2] ICAO, (2001). *Annex 10 - Aeronautical Telecommunications Volume II Communication Procedures Including Those with PANS Status*. Sixth Edition. October 2001.
- [ICAO Annex 10 Vol 4] ICAO, (2014). *Annex 10 - Aeronautical Telecommunications Volume IV Surveillance and Collision Avoidance Systems*. Fifth Edition. July 2014.
- [ICAO Annex 11] ICAO, (2001). *Annex 11 - Air Traffic Services*. Montréal: International Civil Aviation Organization.
- [ICAO Annex 12] ICAO, (2004). *Annex 12 - Search and Rescue*. Montréal: International Civil Aviation Organization.
- [ICAO Annex 14] ICAO, (2004). *Annex 14 - Aerodromes*. Montréal: International Civil Aviation Organization.
- [ICAO Annex 15] ICAO, (2004). *Annex 15 - Aeronautical Information Services*. Montréal: International Civil Aviation Organization.
- [ICAO 4444] ICAO, (2007). *Doc 4444 - Air Traffic Management*. Montréal: International Civil Aviation Organization.
- [ICAO 9735] ICAO, (2011). *Doc 9735 - Universal safety oversight audit programme continuous monitoring manual*. Montréal: International Civil Aviation Organization.
- [ICAO 9750] ICAO (2002). *Doc 9750 - Global Air Navigation Plan for CNS/ATM Systems*. Second edition. Montréal: International Civil Aviation Organization.
- [ICAO 9830] ICAO (2004). *Advanced Surface Movement Guidance and Control Systems (A-SMGCS) Manual (Doc 9830)*. First edition.

- [ICAO 9854] ICAO (2005). *Doc 9854 - Global air traffic management operational concept*. First edition. Montréal: International Civil Aviation Organization.
- [ICAO 9869] ICAO (2017). *Doc 9869 - Performance-Based Communication and Surveillance (PBCS) Manual*. Second edition. Montréal: International Civil Aviation Organization.
- [ICAO 9871] ICAO (2007). *Doc 9871 - Technical provisions for Mode S services and extended squitter*. First edition. Montréal: International Civil Aviation Organization.
- [ICAO 9883] ICAO (2009). *Manual on Global Performance of the Air Navigation System*. First edition. Montréal: International Civil Aviation Organization.
- [IE 09] INTERSOFT ELECTRONICS (2009). Reducing Test Flights Using Simulated Targets and a Carefully Chosen Set-up. (February 2009).
- [ISO 19111] ISO/TC 211 (2002). Geographic information - Spatial referencing by coordinates.
- [Jarama 17] JARAMA, A. J., J. LÓPEZ-ARAQUISTAIN, G. DE MIGUEL & J. A. BESADA, (2017). Complete Systematic Error Model of SSR for Sensor Registration in ATC Surveillance Networks. *Sensors* 17 (10): 2171.
- [Julier 97] JULIER, S. J., & J. K. UHLMANN, (1997). A new extension of the Kalman filter to nonlinear systems. In *Signal Processing, Sensor Fusion, and Target Recognition VI*, 3068: pp. 182-93. International Society for Optics and Photonics.
- [Julier 04] JULIER, S.J., & J. K. UHLMANN, (2004). Unscented filtering and nonlinear estimation. *Proceedings of the IEEE* 92 (3): pp. 401-22.
- [Kailath 68] KAILATH, T., (1968). An innovations approach to least-squares estimation—Part I: Linear filtering in additive white noise. *IEEE Transactions on Automatic Control* 13 (6): pp. 646-55.
- [Kalata 92] KALATA, P. R., (1992). Alfa-Beta target tracking systems: a survey. In *American Control Conference*, pp. 832–836.
- [Kalman 60] KALMAN, R., (1960). A New Approach to Linear Filtering and Prediction Problems. *Transactions of the ASME – Journal of Basic Engineering*, 82 (Series D), pp. 35–45.
- [Kalman 61] KALMAN, R. E. & R. S. BUCY, (1961). New Results in Linear Filtering and Prediction Theory. *Journal of Basic Engineering*, 83, 1, 95.
- [Kay 93] Kay, S. M., (1993). *Fundamentals of statistical signal processing*. Prentice Hall signal processing series. Englewood Cliffs, N.J: Prentice-Hall PTR.

- [Klančar 17] KLANČAR, G., A. ZDEŠAR, S. BLAŽIČ & I. ŠKRJANC, (2017). *Wheeled Mobile Robotics: From Fundamentals towards Autonomous Systems*. Oxford Cambridge, MA: Butterworth-Heinemann. Elsevier.
- [Knippers 2009] <http://kartoweb.itc.nl/geometrics/index.html>
- [Koch 00] KOCH, W., (2000). Fixed-Interval Retrodiction Approach to Bayesian IMM-MHT for Maneuvering Multiple Targets. *IEEE Transactions on Aerospace and Electronic Systems* 36 (1): pp. 2–14.
- [Kolmogorov 41] KOLMOGOROV, A. N., (1941). Interpolation and extrapolation of stationary random sequences. *Izvestiia Akademii Nauk SSSR, Serii Matematicheskii*, 5, 3-14. [Translated, 1962, Memo Rm-3090-PR, Rand Corp, Santa Monica, CA.]
- [Langhans 13] LANGHANS, W., C. SCHEIFLINGER, W. WEIDNER, J. AUER, P. FITZGERALD & M. ANZALONE, (2013). Implementation of a nationwide wide-area multilateration system for Austrian Airspace. In *Integrated Communications, Navigation and Surveillance Conference (ICNS), 2013*, pp. 1–18.
- [Lengyel 12] LENGYEL, E., (2012). *Mathematics for 3D game programming and computer graphics*. Third edition. Boston, MA: Course Technology, Cengage Learning.
- [Levanon 00] LEVANON, N., (2000). Lowest GDOP in 2-D scenarios. In *IEE Proceedings Radar, Sonar and Navigation*, 147, no: 3, pp. 149-55.
- [Levinson 46] LEVINSON, N., (1946). The Wiener (Root Mean Square) Error Criterion in Filter Design and Prediction. *Journal of Mathematics and Physics* 25 (1-4): pp. 261-78.
- [Lexicon] http://www.eurocontrol.int/lexicon/lexicon/en/index.php/Main_Page
- [Li 93] LI, X.R. & Y. BAR-SHALOM, (1993). Design of an interacting multiple model algorithm for air traffic control tracking. In *IEEE Transactions on Control Systems Technology*, 1 (3): pp. 186-94.
- [Li 03] LI, X. R. & V. P. JILKOV, (2003). Survey of maneuvering target tracking. Part I. Dynamic models. *IEEE Transactions on Aerospace and Electronic Systems*, 39, 4 (October 2003), pp. 1333–64.
- [Lilly 97] LILLY, A. S., P. V. BRENNAN, H. D. GRIFFITHS & D. G. MONEY, (1997). A generic approach to radar performance modelling. In *Radar 97 (Conf. Publ. No. 449)*, pp. 662–666.
- [Lin 04] LIN, X., Y. BAR-SHALOM & T. KIRUBARAJAN, (2004). Exact Multisensor Dynamic Bias Estimation with Local Tracks. *IEEE Transactions on Aerospace and Electronic Systems* 40 (2): 576–90.

- [Lin 05] LIN, X., Y. BAR-SHALOM, AND T. KIRUBARAJAN. (2005). Multisensor Multitarget Bias Estimation for General Asynchronous Sensors. *IEEE Transactions on Aerospace and Electronic Systems* 41 (3): 899–921.
- [Llinas 01] HALL, D. L., & J. LLINAS, (2001). *Handbook of multisensor data fusion*. The Electrical engineering and applied signal processing series. Boca Raton, FL: CRC PRESS.
- [Lofting] <http://www.favourite.org.uk/lofting.html>
- [López-Araquistain 19] LÓPEZ-ARAQUISTAIN, J., A. J. JARAMA, J. A. BESADA, G. DE MIGUEL & J. R. CASAR, (2019). A New Approach to Map-Assisted Bayesian Tracking Filtering. *Information Fusion* 45, pp. 79-95.
- [Lygeros 03] LYGEROS, J., K. H. JOHANSSON, S. N. SIMIC, J. ZHANG, & S. S. SASTRY, (2003). Dynamical properties of hybrid automata. *IEEE Transactions on Automatic Control*, 48, 1 (January 2003), pp. 2–17.
- [Macfadzean 92] MACFADZEAN, R. H. M. & J. M. JOHNSON, (1992). *Surface-based air defense system analysis: software and user's guide*. Boston: Artech House.
- [Magill 65] MAGILL, D., (1965). Optimal adaptive estimation of sampled stochastic processes. *IEEE Transactions on Automatic Control* 10 (4): 434-39.
- [Mancini 97] S. MANCINI, B. LAMBIN & P. KERSTENS, (1997). Wavelet Filters in Trajectory Reconstruction from Multi-Radar Data. In *Proceedings of the IRCTR Colloquium on Surveillance Sensor Tracking*, Delft, The Netherlands.
- [Mann 11] MANN, G. W., (2011). Aircraft taxiway conformance monitoring with constrained stochastic linear hybrid systems. *Theses and Dissertations Available from ProQuest* (January 2011), pp. 1–133.
- [Mann 11a] MANN, G. W. & I. HWANG, (2011). 4D Aircraft Taxiway Conformance Monitoring with Stochastic Linear Hybrid Systems. In *AIAA Guidance, Navigation, and Control Conference*. American Institute of Aeronautics and Astronautics.
- [Mann 12] MANN, G. W. & I. HWANG, (2012). Four-Dimensional Aircraft Taxiway Conformance Monitoring with Constrained Stochastic Linear Hybrid Systems. *Journal of Guidance, Control, and Dynamics* 35 (5): pp. 1593-1604.
- [Mann 13] MANN, G. W., & I. HWANG, (2013). State estimation and fault detection and identification for constrained stochastic linear hybrid systems. *IET Control Theory Applications* 7 (1): pp. 1-15.
- [Mantilla 15] MANTILLA-GAVIRIA, I. A., M. LEONARDI, G. GALATI & J. V. BALBASTRE, (2014). Localization Algorithms for Multilateration (MLAT) Systems

- in Airport Surface Surveillance. *Signal, Image and Video Processing*, vol.9, no: 7, pp. 1549-58.
- [Mathias 07] MATHIAS, A. & R. HEIDGER, (2007). Design of an Interactive Multiple Model Kalman Filter (IMMKF) in PHOENIX. In *ESAVS 2007 conference proceeding*. Bonn, Germany.
- [Mathias 08] MATHIAS, A., (2008). ADS-B error estimation and correction in the PHOENIX tracker. In *Tyrrhenian International Workshop on Digital Communications - Enhanced Surveillance of Aircraft and Vehicles, 2008. TIWDC/ESAV 2008*, pp. 1–4.
- [Mayer 97] MAYER, E., (1997). The European Way to Harmonization of ATC. In *ATCA Conference Proceedings*.
- [Mayer 98] MAYER, E., (1998). ASTERIX-The Emerging World-Wide Standard in the Surveillance Domain. In *ATCA Conference Proceedings*.
- [Mazor 98] MAZOR, E., A. AVERBUCH, Y. BAR-SHALOM & J. DAYAN, (1998). Interacting multiple model methods in target tracking: a survey. *IEEE Transactions on Aerospace and Electronic Systems* 34 (1): 103-23.
- [Merrington 98] MERRINGTON, I. S. & G. M. GREEN, (1998). Measurement and interpretation of radar system performance in a complex environment. In *1998/221), IEE Colloquium on Specifying and Measuring Performance of Modern Radar Systems*, pp. 4/1–4/6, IET.
- [Meyer 13] MEYER, F., A. SCHROEDER & M. EDRICH, (2013). Passive Radar Performance Analysis Tool. In *NATO STO-MP-SET-187*.
- [Miyazaki 14] MIYAZAKI, H., H. SHIMADA, T. KOGA, Y. KAKUBARI & S. NIHEI, (2014). Performance evaluation of a WAM system using measured range. In *2014 Tyrrhenian International Workshop on Digital Communications - Enhanced Surveillance of Aircraft and Vehicles (TIWDC/ESAV)*, pp. 78–81.
- [Nabaa 99] NABAA, N. & R. H. BISHOP, (1999). Solution to a Multisensor Tracking Problem with Sensor Registration Errors. *IEEE Transactions on Aerospace and Electronic Systems* 35 (1): 354-63.
- [Nadarajah 12] NADARAJAH, N., R. THARMARASA, M. McDONALD & T. KIRUBARAJAN. (2012). IMM Forward Filtering and Backward Smoothing for Maneuvering Target Tracking. *IEEE Transactions on Aerospace and Electronic Systems* 48 (3): 2673–78.
- [Nandakumaran 09] NANDAKUMARAN, N., S. SUTHARSAN, R. THARMARASA, T. LANG, M. McDONALD & T. KIRUBARAJAN. (2009). Interacting Multiple Model Forward Filtering and Backward Smoothing for Maneuvering Target Tracking. In *SPIE Optical Engineering Applications*, edited by Oliver E. Drummond and Richard D. Teichgraeber, 744503. San Diego, CA.

- [NASA 1999] CASSELL, R., C. EVERS, D. HICOK & D. LEE, (1999). *NASA Low Visibility Landing and Surface Operations (LVLASO) Atlanta Demonstration: Surveillance System Performance Analysis*. Rannoch Corporation, Alexandria, Virginia.
- [Neven 94] NEVEN, W. H., H. A. P. BLOM & P. C. DE KRAKER, (1994). Jump linear model-based aircraft trajectory reconstruction. *Proceedings of SPIE*, 2235, 1 (July 1994), 540–557.
- [Next Gen] <http://www.faa.gov/nextgen/>
- [NGS 2001] http://www.ngs.noaa.gov/GEOID/geoid_def.html
- [NGS 2011] <http://www.ngs.noaa.gov/>
- [NIMA 1991] NIMA WGS 84 UPDATE COMMITTEE, (1991). *Department of Defense World Geodetic System 1984: Its Definition and Relationships with Local Geodetic Systems*. Second Edition. ADA280358.
- [NIMA 1997] NIMA WGS 84 UPDATE COMMITTEE (1997). *Department of Defense World Geodetic System 1984: Its Definition and Relationships with Local Geodetic Systems*. Third Edition.
- [NIMA 2004] NIMA WGS 84 UPDATE COMMITTEE (2004). *Department of Defense World Geodetic System 1984: Its Definition and Relationships with Local Geodetic Systems*. Third Edition. Amendment 2.
- [NOAA 2011] <http://www.noaa.gov/>
- [Nolan 11] NOLAN, M. S., (2011). *Fundamentals of air traffic control*. Fifth edition. Clifton Park, N.Y: Delmar Cengage Learning.
- [Nunn 77] NUNN, W. R. & A. M. DESIDERIO, (1977). Semi-Markov processes: an introduction. CRC 335. ADA046158.
- [O'Hern 97] O'HERN, B., J. MACK, A. PIERAMICO, D. RUGGER, & L. MOYER, (1997). The Radar Support System (RSS): A tool for siting radars and predicting their performance. In *IEEE National Radar Conference, 1997*, pp. 211–216. IEEE.
- [Oppenheim 97] OPPENHEIM, A. V., WILLSKY & S. H. NAWAB, (1997). *Signals & systems*. Upper Saddle River, N.J.: Prentice Hall.
- [ORDNANCE 2010] ORDNANCE SURVEY, (2010). A guide to coordinate systems in Great Britain. (December 2010).
- [OSI model] ISO/IEC, (1994). ISO/IEC 7498-1: Information Processing Systems - OSI Reference Model - The Basic Model.
- [OSI web] http://www2.themanualpage.org/networks/networks_osi.php3

- [Paradowski 1998] PARADOWSKI, L. R. & Z. KOWALSKI, (1998). An effective coordinates conversion algorithm for radar-controlled anti-aircraft systems. In *12th International Conference on Microwaves and Radar, 1998. MIKON '98*, vol. 3, pp. 771-775. IEEE.
- [PJ14-04-01] <https://www.sesarju.eu/sesar-solutions/surveillance-performance-monitoring>
- [Pola 03] POLA, G., M. L. BUJORIANU, J. LYGEROS, & M. D. DI BENEDETTO, (2003). Stochastic hybrid models: An overview. In *Proceedings IFAC Conference on Analysis and Design of Hybrid Systems*.
- [Polderman] POLDERMAN, J. W. & J. C. WILLEMS (n.d.). *Introduction to the Mathematical Theory of Systems and Control*.
- [Pourvoyeur 08] POURVOYEUR, K., A. STELZER & G. STELZHAMMER, (2008). Error estimation for reliable fault detection of a TDOA local positioning system. In *2008 Tyrrhenian International Workshop on Digital Communications - Enhanced Surveillance of Aircraft and Vehicles. (TIWDC/ESAV)*, pp.1-6.
- [Pourvoyeur 11] POURVOYEUR, K., A. MATHIAS & R. HEIDGER, (2011). Investigation of measurement characteristics of MLAT / WAM and ADS-B. In *2011 Tyrrhenian International Workshop on Digital Communications - Enhanced Surveillance of Aircraft and Vehicles (TIWDC/ESAV)*, pp. 203–206.
- [Pourvoyeur 14] POURVOYEUR, K. & HEIDGER, R., (2014). Secure ADS-B usage in ATC tracking. In *2014 Tyrrhenian International Workshop on Digital Communications - Enhanced Surveillance of Aircraft and Vehicles (TIWDC/ESAV)*, pp. 35–40.
- [Rabiner 86] RABINER, L. & B.-H. JUANG, (1986). An introduction to hidden Markov models. *IEEE ASSP Magazine*, 3, 1, pp. 4–16.
- [Rafati 07] RAFATI, A., B. MOSHIRI & J. REZAEI, (2007). A New Algorithm for General Asynchronous Sensor Bias Estimation in Multisensor-Multitarget Systems. In *2007 10th International Conference on Information Fusion*, pp. 1–8.
- [RAPS3] FREQUENTIS-COMSOFT, (2020). RAPS-3 Surveillance Data Support, Test & Evaluation System. 20-016-a-XVIII.
- [RASS-R] INTERSOFT ELECTRONICS, (2020). Sensor Performance Evaluation & Monitoring. RASS-R Product Catalogue. IE-CD-01074-002 RASS-R.
- [RASS-S] INTERSOFT ELECTRONICS, (2020). Radar Subsystem Evaluation Site Expert Measurement Tools. RASS-S Product Catalogue. IE-CD-00108-009 RASS-S

- [Rauch 65] RAUCH, H., C. STRIEBEL & F. TUNG, (1965). Maximum Likelihood Estimates of Linear Dynamic Systems. *Journal of the American Institute of Aeronautics and Astronautics*, 3, 8, pp. 1445–50.
- [Rekkas 14] REKKAS, C., (2014). Status of WAM, ADS-B out and ATSAW deployment in Europe. In *2014 Tyrrhenian International Workshop on Digital Communications - Enhanced Surveillance of Aircraft and Vehicles (TIWDC/ESAV)*, pp. 1–5.
- [Renes 85] RENES, J. J., P. V. DE KRAAN & C. EYMANN, (1985). Flightpath reconstruction and systematic radar error estimation from multi-radar range-azimuth measurements. In *1985 24th IEEE Conference on Decision and Control*, vol. 24, pp. 1282–85. IEEE.
- [Renes 89] RENES, J. J. & M.R. BEST, (1989). *MURATREC Multi-Radar trajectory reconstruction facility theoretical background*. National Lucht- en Ruimtevaartlaboratorium (NLR).
- [Ristic 04] RISTIC, B., S. ARULAMPALAM, & N. GORDON, (2004). *Beyond the Kalman Filter: Particle Filters for Tracking Applications*. Boston, MA: Artech House.
- [RMCDE] <https://www.eurocontrol.int/services/surveillance-message-conversion-and-distribution-equipment-rmcde>
- [RMD-V2] FREQUENTIS-COMSOFT, (2020). RMD-V2 Radar Monitoring Display. 20-020-j-XIX.
- [Roberts 94] ROBERTS, S., R. MCLEOD, M. VERMIJ, T. HEASLIP, & G. MACWILLIAM, (1994). RAP (Radar Analysis Program) An Interactive Computer Program for Radar Based Flight Path Reconstruction and Analysis, *25th International Seminar of the International Society of Air Safety Investigators*.
- [Ródenas 20] RÓDENAS, F., E. PINEDO, R. ASENJO, M. MARTÍNEZ, & J. BALBASTRE. (2020). Development of a Surveillance Performance Monitoring Tool for Performance Based Surveillance. In *2020 1st Aerospace Europe Conference*. Bordeaux, France.
- [Roth 14] ROTH, M., G. HENDEBY, & F. GUSTAFSSON, (2014). EKF/UKF Maneuvering Target Tracking Using Coordinated Turn Models with Polar/Cartesian Velocity,. In *17th International Conference on Information Fusion, 2014*, pp. 1-8.
- [RTCA DO-212] RTCA SC-170, (1992). Minimum Operational Performance Standards for Airborne Automatic Dependent Surveillance (ADS) Equipment. (October 1992).
- [RTCA DO-242B] RTCA SC-186, (2002). Minimum Aviation System Performance Standards For Automatic Dependent Surveillance Broadcast (ADS-B). (June 2002).

- [RTCA DO-260] RTCA SC-186, (2000). Minimum Operational Performance Standards for 1090 MHz Automatic Dependent Surveillance - Broadcast (ADS-B). (September 2000).
- [RTCA DO-260A] RTCA SC-186, (2003). Minimum Operational Performance Standards for 1090 MHz Extended Squitter Automatic Dependent Surveillance - Broadcast (ADS-B) and Traffic Information Services - Broadcast (TIS-B). (April 2003).
- [RTCA DO-260B] RTCA SC-186, (2009). Minimum Operational Performance Standards for 1090 MHz Extended Squitter Automatic Dependent Surveillance - Broadcast (ADS-B) and Traffic Information Services - Broadcast (TIS-B). (February 2009).
- [RTCA DO-282B] RTCA, (2011). Minimum Operational Performance Standards for Universal Access Transceiver (UAT) Automatic Dependent Surveillance - Broadcast. (December 2011).
- [RVSM] [http://www.skybrary.aero/index.php/Reduced_Vertical_Separation_Minima_\(RVSM\)](http://www.skybrary.aero/index.php/Reduced_Vertical_Separation_Minima_(RVSM))
- [SAFPA] <https://www.eurocontrol.int/function/surveillance-avionics-monitoring>
- [Saini 07] SAINI, L., F. BASTI, & H. NEUFELDT, (2007). ADS-B surveillance integrated into A-SMGCS - Architecture and implementation. In *ESAVS 2007 conference proceeding*. Bonn, Germany.
- [Salazar 08] SALAZAR HERNÁNDEZ, D. J., (2008). *Navegación Aérea, Cartografía y Cosmografía*. Sixth edition.
- [Sarkka 08] SARKKA, S., (2008). Unscented Rauch–Tung–Striebel Smoother. *IEEE Transactions on Automatic Control* 53 (3): 845–49.
- [Sarmavuori 12] SARMAVUORI, J. & S. SARKKA. (2012). Fourier-Hermite Rauch-Tung-Striebel Smoother. In *2012 Proceedings of the 20th European Signal Processing Conference (EUSIPCO)*, pp. 2109–13.
- [SASS-C] <http://www.eurocontrol.int/services/sass-c-surveillance-analysis-support-system-atc-centre>
- [Savard 2000] <http://www.quadibloc.com/maps/mapint.htm>
- [Schnell 14] SCHNELL, M., U. EPPLE, D. SHUTIN & N. SCHNECKENBURGER, (2014). LDACS: future aeronautical communications for air-traffic management. *IEEE Communications Magazine*, 52, 5 (May 2014), pp. 104–110.
- [Schoenberg 46] SCHOENBERG, I., (1946). Contributions to the problem of approximation of equidistant data by analytic functions, Part A: On the problem of smoothing of graduation, a first class of analytic approximation. *Quarterly of Applied Mathematics*, 4, pp. 45–88.

- [SDDS] <https://www.eurocontrol.int/articles/surveillance-data-distribution-system-sdds-service-based-network-work>
- [Seah 09a] SEAH, C. E. & I. HWANG, (2009). Terminal-Area Aircraft Tracking Using Hybrid Estimation. *Journal of Guidance, Control, and Dynamics* 32 (3): pp. 836-49.
- [Seah 09] SEAH, C. E. & I. HWANG, (2009). Stochastic Linear Hybrid Systems: Modeling, Estimation, and Application in Air Traffic Control. *IEEE Transactions on Control Systems Technology*, 17, 3 (May 2009), pp. 563–575.
- [SESAR] <http://www.sesarju.eu/>
- [SESAR B4.3 D09] SESAR B04.03 (2011). DEL09 Architecture of the Technical Systems Description Document for Step 1.
- [SESAR D3] SESAR CONSORTIUM (2007). The ATM Target Concept - D3. (September 2007).
- [SESAR MP] SESAR (2012). European ATM Master Plan. Edition 2. (October 2012).
- [SESAR P12.03.01 D02] SESAR P12.03.01 (2011). DEL02 Phase1-Architecture design.
- [SESAR P12.03.01 D03] SESAR P12.03.01 (2011). DEL03 Technology Study Report.
- [Simons 96] SIMONS, R. W., (1996). Guglielmo Marconi and Early Systems of Wireless Communication. *General Electric Company REVIEW*, 11, 1.
- [Simons 98] SIMONS, R. W. & J. W. SUTHERLAND, (1998). Forty Years of Marconi Radar from 1946 to 1986. *General Electric Company REVIEW*, 13, 3.
- [Singer 70] SINGER, R. A., (1970). Estimating Optimal Tracking Filter Performance for Manned Maneuvering Targets. *IEEE Transactions on Aerospace and Electronic Systems*, AES-6, 4 (July 1970), pp. 473–483.
- [Skillman 87] SKILLMAN, W. A., (1987). *SIGCLUT radar circulation software*. Boston: Artech House.
- [Skillman 91] SKILLMAN, W. A., (1991). *DETPROB probability of detection calculation software*. Boston: Artech House.
- [Skolnik 08] SKOLNIK, M., (2008). *Radar Handbook*. Third Edition. McGraw-Hill Education.
- [SJU about] <http://www.sesarju.eu/about>
- [SMD] FREQUENTIS-COMSOFT, (2020). Product Brief: SMD Sensor Monitoring Display. 26_FCO_SMD_0620.

- [Snider 09] SNIDER, J., (2009). Automatic Steering Methods for Autonomous Automobile Path Tracking. Master Thesis, Pittsburgh, PA: Carnigie Mellon University, Robotics Institute.
- [Snyder 1987] SNYDER, J. P., (1987). *Map projections - A working manual*. Geological Survey (U.S.). U.S. G.P.O.
- [Sorenson 70] SORENSON, H. W., (1970). Least-squares estimation: from Gauss to Kalman. *IEEE Spectrum*, 7, 7 (July 1970), pp.63 –68.
- [Soto 04] SOTO, A, J. A. BESADA, J. R. CASAR & J. I. PORTILLO, (2004). On-line Bias Estimation of Secondary Radar Networks for ATC. In *Proceedings of the Seventh International Conference on Information Fusion*. Stockholm, Sweden.
- [Soto 05] SOTO, A., J. A. BESADA, G. DE MIGUEL & J. R. CASAR, (2005). Airport based bias estimation of surface movement radars. In *2005 IEEE International Radar Conference*, pp. 916-21.
- [Soto 06] SOTO, A., G. DE MIGUEL, J.A. BESADA, & J. GARCIA. (2006). Robust Tracking Architecture for Mode-S Enhanced Surveillance. In *2006 9th International Conference On Information Fusion*, pp. 1–8.
- [Soto 11] SOTO, A., P. MERINO & J. VALLE, (2011). ADS-B integration in the SESAR surface surveillance architecture. In *2011 Tyrrhenian International Workshop on Digital Communications - Enhanced Surveillance of Aircraft and Vehicles (TIWDC/ESAV)*, pp. 13 –18.
- [Shrader 73] SHRADER, W., (1973). Radar Technology Applied to Air Traffic Control. *IEEE Transactions on Communications*, 21, 5 (May 1973), 591– 605.
- [Skolnik 03] SKOLNIK, M., (2003). Attributes of the ubiquitous phased array radar. In *IEEE International Symposium on Phased Array Systems and Technology*, 2003., pp.101-6.
- [SQUIRE] <https://www.thalesgroup.com/es/squire-ground-surveillance-radar>
- [Stanley 12] STANLEY, R. M., A. COLAVITO, K. RAMMELSBERG, E. K. STELZER, & D. TUOMEY, (2012). Visual Demonstrations of Performance Characteristics of Surface Surveillance. In *2012 IEEE/AIAA 31st Digital Avionics Systems Conference (DASC)*, 6D3-1:6D3-9.
- [Studer 98] STUDER, F. A., R. R. TOMA, & F. VINELLI, (1998). A modern approach to radar performance analysis and field validation. In *1998 IEE Colloquium on Specifying and Measuring Performance of Modern Radar Systems*, pp. 5/1:5/7. IET.
- [Swerling 59] SWERLING, P., (1959). First-Order Error Propagation in a Stagewise Smoothing Procedure for Satellite Observations, Rand Corp, Santa Monica, CA.

- [Tabernero 07] TABERNERO, M. M., & P. RASTRILLA, (2007). *La Vigilancia aérea del futuro: la ADS*. Madrid: Centro de Documentación y Publicaciones de Aena.
- [Tanenbaum 11] TANENBAUM, A. S., (2011). *Computer networks*. Fifth edition. Boston: Pearson Prentice Hall.
- [Theil 17] THEIL, A., (2017). Finally There: CARPET 3. TNO innovation for life.
- [Thompson 03] THOMPSON, S. D. & S. R. BUSSOLARI, (2003). Analysis of terminal separation standards and radar performance. In *ATM Seminar 5th edition*, Budapest, Hungary.
- [Thompson 06] THOMPSON, S. D. & FLAVIN, J. M. (2006). Surveillance Accuracy Requirements in Support of Separation Services. *LINCOLN LABORATORY JOURNAL*, 16, 1 (2006), 97–122.
- [Tong 05] TONG, H. & Z. ZHANG, (2005). On time-reversibility of multivariate linear processes. *Statistica Sinica*, 15, 2 (April 2005), 495–504.
- [Torrieri 84] Torrieri, D. J., (1984). Statistical Theory of Passive Location System. *IEEE Transactions on Aerospace and Electronic Systems*. Vol. AES-20, no: 2, pp. 183-98.
- [Trofimova 07] TROFIMOVA, Y. (2007). Multilateration Error Investigation and Classification. Error Estimation. *Transport and Telecommunication*, 8, 2, pp. 28–37.
- [Unser 98] UNSER, M. & J. ZERUBIA, (1998). A generalized sampling theory without band-limiting constraints. *IEEE Transactions on Circuits and Systems II: Analog and Digital Signal Processing*, 45, 8 (August 1998), 959 – 969.
- [Unser 99] UNSER, M., (1999). Splines: a perfect fit for signal and image processing. *IEEE Signal Processing Magazine*, 16, 6 (November 1999), pp. 22 – 38.
- [Van der Kraan 90] VAN DER KRAAN, P., (1990). Aircraft Trajectory Reconstitution on the Basis of Multi-Radar Plot Information. (May 1990).
- [Van Hintum 13] VAN HINTUM, J.E.A., (2013). ARTAS SMS Design Document. NLR-CR-2012-552. NLR.
- [Vasegui 00] VASEGHI, S. V., (2000). *Advanced Digital Signal Processing and Noise Reduction*. Second Edition. Chichester: John Wiley Sons Ltd
- [Veth 15] VETH, M. J., (2015). Nonlinear Estimation Techniques for Navigation. *STO-EN-SET-214*. STO – Science and Technology Organization. NATO.

- [Vidakovic 91] VIDAKOVIC, B. & P. MUELLER, (1991). *Wavelets for kids: A tutorial introduction*. Duke University.
- [Weiss 75] WEISS, G., (1975). Time-reversibility of linear stochastic processes. *J. Appl. Prob.*, 12, pp. 831–836.
- [Wiener 49] WIENER, N., (1949). *Extrapolation, interpolation, and smoothing of stationary time series, with engineering applications*. Cambridge. Technology Press of the Massachusetts Institute of Technology.
- [Wikipedia] http://en.wikipedia.org/wiki/Main_Page
- [Wood 00] WOOD, M. L., (2000). Multilateration system development history and performance at Dallas/Ft. Worth Airport. In *Digital Avionics Systems Conference, 2000. Proceedings. DASC. The 19th*, vol. 1, pp. 2E1/1:2E1/8.
- [Žáčik 17] ŽÁČIK, N., A. NOVÁK, A. SOTO & G. DRUPKA. (2017). SASS-C Processing Capability of MSPSR Data. *Transportation Research Procedia*, INAIR 2017, 28 (January): 79–88.

LIST OF ABBREVIATIONS

The following list compiles the main abbreviations and acronyms that have appeared all along this document:

ACARS	<i>Aircraft Communication Addressing and Reporting System</i>
ACAS	<i>Airborne Collision Avoidance System</i>
ACC	<i>Area Control Center</i>
ARP	<i>Airport Reference Point</i>
ADS-B	<i>Automatic Dependent Surveillance- Broadcast</i>
ADS-C	<i>Automatic Dependent Surveillance- Contract</i>
AEMET	<i>Agencia Estatal de Meteorología</i>
AENA	<i>Aeropuertos Españoles y Navegación Aérea</i>
AeroMACS	<i>Aeronautical Mobile Airport Communications System</i>
AIS	<i>Aeronautical Information Service</i>
AMS	<i>Aeronautical Mobile Service</i>
ANSP	<i>Air Navigation Service Provider (e.g. AENA, NATS, DFS, etc.)</i>
APP	<i>Approach Centre</i>
ARTAS	<i>ATM suRveillance Tracker And Server</i>
ARV	<i>Air Referenced Velocity</i>
ASAS	<i>Airborne Separation Assurance System</i>
ASDE	<i>Airport Surface Detection Equipment</i>
ASEP	<i>Airborne Separation</i>
ASPA	<i>Airborne Spacing</i>
ASR	<i>Airport Surveillance Radar</i>
ASTERIX	<i>All Purpose Structured Eurocontrol Surveillance Information Exchange</i>
ASM	<i>Airspace Management</i>
ATC	<i>Air Traffic Control</i>
ATCRBS	<i>Air Traffic Control Radar Beacon System</i>
ATFM	<i>Air Traffic Flow Management</i>
ATIS	<i>Automatic Terminal Information Service</i>
ATM	<i>Air Traffic Management</i>
ATN	<i>Aeronautical Telecommunication Network</i>
ATS	<i>Air Traffic Services</i>
ATSA	<i>Airborne Traffic Situation Awareness</i>
ATSAW	<i>Airborne Traffic Situation Awareness</i>
ATSU	<i>Air Traffic Services Unit</i>
BDS	<i>Binary Data Store or Comm-B Data Registers</i>
C	<i>Communications</i>
CAPT	<i>Coverage and Planning Tool for ADS-B and TIS-B</i>
CDM	<i>Collaborative Decision Making</i>
CFMU	<i>Control Flow Management Unit</i>
CMP	<i>Comparator</i>
CNS	<i>Communications, Navigation and Surveillance</i>

CPDLC	<i>Controller-Pilot Data Link Communications</i>
CPR	<i>Compact Position Reporting</i>
CPS	<i>Central Processing Station</i>
CRC	<i>Cyclic Redundancy Check</i>
DABS	<i>Discrete Address Beacon System</i>
DTED	<i>Digitised Terrain and Elevation Data</i>
DFS	<i>Deutsche Flugsicherung GmbH</i>
DoD	<i>Department of Defence</i>
DoT	<i>Department of Transportation</i>
ECAC	<i>European Civil Aviation Conference</i>
ECEF	<i>Earth-Centred Earth-Fixed</i>
ECI	<i>Earth-Centred Inertial</i>
ECM	<i>Electronic Counter Measures</i>
EKF	<i>Extended Kalman Filter</i>
ESASSP	<i>EUROCONTROL Specification for ATM Surveillance System Performance</i>
FAF	<i>Final Approach Fix</i>
FANS	<i>Future Air Navigation System</i>
FAT	<i>Factory Acceptance Test</i>
FB	<i>Functional Block</i>
FCU	<i>Flight Control Unit</i>
FDDI	<i>Fiber Distributed Data Interface</i>
FDP	<i>Flight Data Processing</i>
FIR	<i>Finite Impulse Response</i>
FMS	<i>Flight Management System</i>
FPPS	<i>Flight Plan Data Processing System</i>
FRUIT	<i>False Replies Unsynchronized in Time</i>
GNSS	<i>Global Navigation Satellite System</i>
GPS	<i>Global Positioning System</i>
IAF	<i>Initial Approach Fix</i>
IIR	<i>Infinite Impulse Response</i>
INCS	<i>Independent and Non-Cooperative Sensor</i>
LLF	<i>Local-Level Frame</i>
LAN	<i>Local Area Network</i>
LMMSE	<i>Linear Minimum Mean-Square Error</i>
MCP	<i>Mode Control Panel (BOEING)</i>
MET	<i>METeorological service</i>
MLT	<i>Multilateration system</i>
MLAT	<i>Multilateration (Airport domain)</i>
MMSE	<i>Minimum Mean-Square Error</i>
MRO	<i>Maintenance, Repair and Overhaul</i>
MSDF	<i>Multisensor Data Fusion</i>
MSPSR	<i>Multi-Static Primary Surveillance Radar</i>
MSSR	<i>Monopulse Secondary Surveillance Radar</i>
MURATREC	<i>Multi-Radar Trajectory Reconstruction</i>

N	<i>Navigation</i>
NAC	<i>Navigation Accuracy Category</i>
NAS	<i>National Airspace System</i>
NATS	<i>National Air Traffic Services</i>
NDA	<i>Non-disclosure agreements</i>
NGS	<i>National Geodetic Survey</i>
NIC	<i>Navigation Integrity Category</i>
NLR	<i>National Airspace Laboratory</i>
NMOC	<i>Network Manager Operations Center</i>
NOAA	<i>National Oceanic and Atmospheric Administration</i>
NUCp	<i>Navigation Uncertainty Category for Position</i>
NUCv	<i>Navigation Uncertainty Category for Velocity</i>
OTR	<i>Opportunity Traffic Reconstruction</i>
PET	<i>Performance Evaluation Tool</i>
PIC	<i>Position Integrity Category</i>
PPI	<i>Plan Position Indicator</i>
PSR	<i>Primary Surveillance Radar</i>
P/B	<i>Push back</i>
RASCAL	<i>Radar Sharing Calculation</i>
RADAR	<i>Radio Detection and Ranging</i>
RCP	<i>Required Communication Performance</i>
RMCDE	<i>Surveillance Message Conversion and Distribution Equipment</i>
RNP	<i>Required Navigation Performance</i>
RDT&E	<i>Research, Development, Test & Evaluation</i>
RSP	<i>Required Surveillance Performance</i>
RSSP	<i>Radar System Specialist Panel</i>
RTQA	<i>Radar Track Quality Assessment</i>
RVSM	<i>Reduced Vertical Separation Minima</i>
S	<i>Surveillance</i>
SALADT	<i>Screening angle and line of sight analysis from digital terrain data</i>
SAM	<i>Surface-to-Air Missile</i>
SAR	<i>Search And Rescue service</i>
SASS-C	<i>Surveillance Analysis Support System for ATC-Centre</i>
SAT	<i>Site Acceptance Test</i>
SCAS	<i>Surveillance Coverage Analysis Suite</i>
SDA	<i>System Design Assurance</i>
SDP	<i>Surveillance Data Processor</i>
SDPS	<i>Surveillance Data Processing System</i>
SDPDS	<i>Surveillance Data Processing and Distribution System</i>
SES	<i>Single European Sky initiative</i>
SESAR	<i>Single European Sky ATM Research</i>
SIL	<i>Surveillance Integrity Level</i>
SIL	<i>Source Integrity Level</i>
SLI	<i>Service Level Indicator</i>

SMART	<i>Simulator for Multi-radar Analysis from Realistic Traffic</i>
SMR	<i>Surface Movement Radar</i>
SSEP	<i>Self Separation</i>
SSR	<i>Secondary Surveillance Radar</i>
S/U	<i>Start up</i>
TBS	<i>Time Based Separation</i>
TDoA	<i>Time Difference of Arrival</i>
TOA	<i>Time of Arrival</i>
ToA	<i>Time of Applicability</i>
ToD	<i>Time of Day</i>
TRES	<i>Trajectory Reconstruction and Evaluation Suite</i>
U.S. or USA	<i>United States of America</i>
USGS	<i>United States of America Geological Survey</i>
TWR	<i>Tower Control Unit</i>
WAM	<i>Wide Area Multilateration</i>
WAN	<i>Wide Area Network</i>