APPLICATION OF ROM-BASED METHODS TO AEROENGINES
HEALTH MONITORING, PERFORMANCE, AND CONTROL

AUTHOR:
José Rodrigo Ramírez – Aeronautical Engineer

DIRECTOR:
José Luis Montañés García – PhD in Aeronautical Engineering / Professor

2021
INDEX

ACKNOWLEDGEMENT ............................................................................................................. iii
RESUMEN .................................................................................................................................. v
ABSTRACT .............................................................................................................................. vii
LIST OF FIGURES .................................................................................................................... ix
LIST OF TABLES ....................................................................................................................... xi
LIST OF ACRONYMS ............................................................................................................... xiii
LIST OF SYMBOLS ................................................................................................................ xix

1.- INTRODUCTION: MOTIVATION AND TARGETS ..................................................... 1

1.1.- Current historical context and reasons for this research ........................................... 1
1.2.- Introducing the high-bypass two-spool turbofan engine .......................................... 4
1.3.- The historic role of the turbofan engines ................................................................ 8
1.4.- A continuous search for efficiency improvements and cost reductions ..................... 12
1.5.- Applicable methodologies to evaluate engine’s health status ................................. 15
1.6.- Where the industry currently is ............................................................................... 18
1.7.- The path ahead ........................................................................................................... 23

2.- REVISION OF THE STATE OF THE KNOWLEDGE .............................................. 25

2.1.- Gas Path Analysis: Origin, concept, versions, and applicability .............................. 25
2.2.- Exploring the applicability of genetic algorithms ...................................................... 33
2.3.- Optimizing with gradient-based methods .................................................................. 37
2.4.- Managing the information coming from the sensors ............................................... 38
2.5.- Compilatory studies analyzing the available known techniques ............................... 48
2.6.- Some non-conventional approaches ......................................................................... 58
2.7.- The management of big data sets and its potential benefit ........................................ 61
2.8.- Conclusions ................................................................................................................. 64

3.- ENGINE CONFIGURATION, INSTRUMENTATION, AND MODEL ..................... 67

3.1.- Engine configuration ................................................................................................... 68
3.1.1.- The CF6 family ........................................................................................................ 68
3.1.2.- The CFM56 family .................................................................................................. 79
3.1.3.- The GE90, the GEnx, and the GE9X ...................................................................... 85
3.2.- Monitoring System and Instrumentation ................................................................. 91
3.2.1.- Preliminary concepts ............................................................................................ 91
3.2.2.- Engine Sensors ...................................................................................................... 93
3.2.3.- Engine Control System: EEC and FADEC .......................................................... 99
3.2.4.- Instrumentation considered in the model ............................................................. 102
3.3.- Performance model of the engine ............................................................................ 105
3.3.1.- The model developed in PROGIS® ....................................................................... 105
3.3.2.- On-design calculations .......................................................................................... 116
3.3.3.- Off-design calculations .......................................................................................... 124
3.3.4.- Model validation .................................................................................................... 132
3.3.5.- The interest on the cruise phase .......................................................................... 137

4.- MATHEMATICAL AND NUMERICAL METHODOLOGY ................................. 145

4.1.- Analysis of the inverse problem ................................................................................ 146
4.2.- Scaling of the data provided by the instrumentation ................................................. 151
4.3.- Problem exploration with Genetic Algorithms ......................................................... 153
4.6. Use of the SQP Method ............................................................................. 158
4.5. Calculation of Jacobian matrices .............................................................. 165
4.6. Applied Newton-like method ..................................................................... 167
4.7. Singular Value Decomposition (SVD) ....................................................... 168
4.8. Correlations between degradations .......................................................... 176
4.9. Calculations including one additional sensor ............................................ 178
4.10. Analysis of the model application, convexity, uniqueness, and convergence .............................................................................................................. 181
4.11. The use of tensors and HOSVD for the resolution of the inverse problem .............................................................................................................. 192
4.12. Conclusions .............................................................................................. 208

5. RESULTS ........................................................................................................ 211
5.1. Results obtained with GAs ......................................................................... 212
  5.1.1. Example 1 .............................................................................................. 212
  5.1.2. Example 2 .............................................................................................. 214
  5.1.3. Example 3 .............................................................................................. 215
5.2. Results obtained with SQP ......................................................................... 217
  5.2.1. Example 4 .............................................................................................. 217
  5.2.2. Example 5 .............................................................................................. 218
  5.2.3. Example 6 .............................................................................................. 219
  5.2.4. Example 7 .............................................................................................. 221
  5.2.5. Example 8 .............................................................................................. 222
5.3. Results obtained with the adapted Newton method .................................. 223
  5.3.1. Example 9 .............................................................................................. 223
  5.3.2. Example 10 ............................................................................................ 224
  5.3.3. Example 11 ............................................................................................ 225
  5.3.4. Example 12 ............................................................................................ 225
  5.3.5. Example 13 ............................................................................................ 226
5.4. Analysis of the results obtained after applying SVD .............................. 229
  5.4.1. Example 14 ............................................................................................ 229
5.5. Calculations after the upgrade by including one additional sensor .......... 231
  5.5.1. Example 15 ............................................................................................ 231
  5.5.2. Example 16 ............................................................................................ 235
  5.5.3. Example 17 ............................................................................................ 236
  5.5.4. Example 18 ............................................................................................ 238
  5.5.5. Example 19 ............................................................................................ 239
5.6. Results obtained with the ROM and the application of HOSVD .......... 240
  5.6.1. Example 20 ............................................................................................ 243
5.7. Comments on the calculations performed and conclusions .................. 250

6. APPLICABILITY AND FUTURE WORK ......................................................... 257
  6.1. Applicability of the methodology .............................................................. 257
  6.2. Business Case ............................................................................................ 264
  6.3. Future Work ............................................................................................... 276

APPENDIX A – Kalman Filters ......................................................................... 281
APPENDIX B – Non-linear Conjugate Gradient .............................................. 283
APPENDIX C – Bayesian Belief Networks ....................................................... 285
APPENDIX D – Neural Networks ..................................................................... 287
APPENDIX E – Fuzzy Logic ................................................................................ 289
APPENDIX F – Hidden Markov Models ............................................................ 291

BIBLIOGRAPHICAL REFERENCES ..................................................................... 293
ACKNOWLEDGEMENT

If I had to choose only one person to thank for the support received during the preparation of this document, I would be committing a great injustice, since my path has been enlightened by different estimated collaborators. Nevertheless, I truly have a debt of gratitude with:

**Dr. Luis Sánchez de León**, my dear colleague, because without his unwavering support, his natural born intellectual brilliance, and his complete dedication during these past long years of preparation, even under extremely difficult personal circumstances that still impress me, this work would not have been possible.

**Dr. José Manuel Vega**, master, mentor, and top-class researcher, who provided an enormous support, with his expertise, mindset, generosity, and advice during the sometimes difficult, but still rewarding, elaboration of this thesis. He showed to me the way to follow, he took me under his wings during difficult moments, being a personal and academic reference, while sharing selflessly, even after extremely worrying health circumstances, his vast knowledge regarding the different topics treated in this document.

Other brilliant colleagues at ETSIAE-UPM, like **Dr. Ángel Velázquez** and **Dr. Francisco Sastre**, have also contributed remarkably to the development of the proposed research line by showing innovative ways to improve results. They provided an unvaluable support during critical moments.

And last, but not least, I must thank to the director of this thesis, **Dr. José Luis Montañés**, master, mentor, and friend since many years, his leading and focused technical approach, his clarity, his passion on everything he endeavors, his smart vision, his continuous support by being always accessible and the pilar upon everything was elevated, and the faith he inexplicably always had in me. I wish I could deserve one day just a fraction of it. He gave me the tools to achieve all the goals I had so far in my life, and he transmitted to me the respect and consideration for the Engineering, the Science, and for our Alma Mater.

They are the model or, better said, the mirror in which I wish to contemplate myself in the future. As an engineer, and as a person.

Climbing on their shoulders, I could see beyond.

Thank you.

**Manzanares el Real, Spain, 2021.**

- To my parents and to the director -

*Requies mea pugnare*
RESUMEN

Los fabricantes de aerorreactores más importantes del mundo muestran actualmente gran interés por los sistemas de monitorización encargados de seguir el estado de degradación en sus motores. Y ello para poder controlar mejor su funcionamiento, poder determinar con precisión los intervalos entre eventos de mantenimiento, reducir los tiempos de parada programada y aumentar el retorno de inversión de sus clientes. En definitiva, para mejorar la calidad de sus productos e incrementar el valor percibido por el mercado. Los intentos por aumentar la eficiencia de estos motores han sido determinantes, en las últimas décadas, en cuanto al rumbo que ha seguido la industria aeronáutica por diversos motivos. Conocer de forma precisa el estado de degradación de estos motores es clave para alcanzar todos estos objetivos. De entre todos los motores de reacción, los motores tipo turbofán bieje son los más empleados en la aviación comercial, motivo por el que esta tesis se centrará en ellos.

En este sentido, los métodos basados en modelos de orden reducido (también conocidos como ROMs, por sus siglas en inglés) han resultado ser, en los últimos años, técnicas de una aplicabilidad muy destacable en diferentes campos técnicos. Buenos ejemplos de ello pueden encontrarse en su uso para el análisis de efectos aerolásticos durante vuelo transónico en aviones actuales, o para el estudio de efectos de transferencia de calor en flujos aislados, entre otros. Dada la importante reducción de tiempo de cálculo que puede lograrse implementando este tipo de metodologías simplificadas (entre otras potenciales ventajas que se evaluarán en la tesis) en ciertos problemas que impliquen el manejo de una gran cantidad de datos, o el uso de modelos analíticos complejos, uno de los objetivos del estudio será determinar si los ROMs pueden también emplearse ventajosamente para la realización de diagnósticos y pronósticos en motores tipo turbofán (y similares), así como en el cálculo preciso de una variable tan relevante en estos motores como es la temperatura a la salida de la cámara de combustión, parámetro que suele emplearse en la determinación del régimen de funcionamiento.

Para este propósito, se ha generado un modelo de motor, por medio del programa informático PROOSIS®, con la capacidad de simular su comportamiento, de modo que cuente con cierto grado de deterioro definible en cada uno de sus componentes. Dicho modelo se combinará con algoritmos desarrollados en MATLAB®, de optimización y de resolución de problemas inversos, con el objeto de determinar el estado de degradación del motor a partir de la información obtenida de la instrumentación montada en el mismo. Es decir, la degradación de ciertos componentes principales (fan, compresores y turbinas) se calculará a partir de datos recopilados por sensores instalados en el motor, mediante la aplicación de distintas técnicas numéricas (los sensores serán simulados).

El objetivo principal de la tesis será, por lo tanto, la creación de una metodología que permita resolver el mencionado problema inverso asociado a la determinación del estado de degradación del aerorreactor más representativo en la actualidad. La exploración de las capacidades de los ROMs, en la realización de los cálculos llevados a cabo con el modelo completo, constituirá la última parte de la tesis. Se substituirá entonces al modelo, previamente desarrollado a través de PROOSIS®, por un tensor que contenga la información mínima necesaria para llevar a cabo la monitorización del estado del motor durante su operación, con vistas a lograr menores tiempos de cálculo, así como menores costes de implementación de la metodología en los sistemas de control disponibles a bordo de la aeronave, si ello fuera posible.

Se espera que el beneficio que la metodología propuesta pueda suponer, como consecuencia del conocimiento preciso del estado de degradación de los motores, atraiga el interés de diversos operadores (e.g., aerolíneas) para implementarla en sus respectivas flotas. Dicho beneficio repercutiría en los costes de operación directos (mejora de eficiencia y ahorro de combustible), y en los costes indirectos asociados (mejoras derivadas en cuanto a mantenimiento y gestión de operación).
ABSTRACT

The most relevant jet engine manufacturers in the world are currently very interested in systems dedicated to the monitoring of the degradation in the engines they produce. The reasons behind include aiming for a better operational control, being able to accurately determine the intervals between maintenance events, reducing scheduled downtimes and, finally, improving the return on investment for their customers. In short, the main target is improving the quality of their products and to increase the value perceived by the market. Attempts to improve the efficiency of these engines have been decisive, in recent decades, in terms of the direction that the aeronautical industry has followed, for various reasons. Getting to know accurately the degradation condition in these engines is crucial to achieve all those goals. From all the jet engine models available today, the two-spool turbofan engine is the most widely used in commercial aviation, reason why this thesis will be focused on it.

In this sense, methods based on reduced order models (also known as ROMs) have become techniques of a very remarkable applicability in different technical fields in recent years. Good examples can be found in its use for the analysis of aeroelastic effects during transonic flight in modern aircraft, or for the study of heat transfer effects in isolated flows, among others. Given the significant reduction in computational time that can be achieved by implementing this type of simplified methodologies (among other potential advantages that will be evaluated in the thesis) in certain problems that involve handling of a large amount of data, or the use of complex analytical models, one of the objectives of this study will be determining if ROMs can also be used advantageously for the development of diagnoses and prognoses in turbofan engines (and similar gas turbine engine models), as well as in the accurate calculation of a variable as relevant in these engines as the temperature at the outlet of the combustion chamber is, parameter that is typically used in determining the engine’s operating regime.

To do so, an engine model has been created, by means of the PROOSIS® software, with the ability to simulate engine’s performance, so that it counts with a certain degree of definable degradation in each of its components. Such model will be combined with algorithms developed in MATLAB®, for optimization and for inverse problems solving, to determine the engine’s degradation condition with the information obtained from the instrumentation mounted on it. Thus, the degradation of certain main engine components (fan, compressors, and turbines) will be obtained from data collected by sensors installed in the engine, after the application of different numerical techniques (sensors will be simulated).

The main goal of the thesis will be, therefore, the development of a methodology that allows solving the inverse problem associated with the determination of the degradation condition in the most representative aeroengine nowadays. The exploration of the capabilities of ROM-based methods, in the solving process of the calculations carried out initially by the complete model, will constitute the last part of the thesis. The model, previously developed through PROOSIS®, will then be replaced by a tensor that contains the minimum necessary information to perform the monitoring of the engine’s degradation state during its operation, with the aim of achieving shorter calculation times, as well as shorter costs of implementation of the methodology in the control systems available on board in the aircraft, if that could be finally possible.

It is expected that the benefit the proposed methodology may mean, because of the accurate knowledge of the engine’s degradation condition, will attract the attention of different operators (e.g., airlines) to implement it in their respective fleets. This benefit would have a positive impact on their direct operating costs (with improved efficiency and fuel savings), and on their associated indirect costs (with derived improvements in terms of maintenance and operation management).
LIST OF FIGURES

FIGURE 1: IMPACT OF COVID-19 IN THE COMMERCIAL AIR TRAFFIC WORLDWIDE. .......................................................... 1
FIGURE 2: EXAMPLE OF A MODERN JET AIRCRAFT, THE BOEING 777X……………………………………………………………… 2
FIGURE 3: THE GE9X, AN EXAMPLE OF MODERN TURBOFAN ENGINE. ................................................................................. 5
FIGURE 4: FIGURE SHOWING THE TURBOMACHINERY AND MODULAR DESIGN OF A CFM56-7 DAC. .............................................. 5
FIGURE 5: SCHEMATIC SHOWING THE EVOLUTION OF PRESSURE AND TEMPERATURE ALONG THE DIFFERENT STAGES OF A TURBOFAN. .... 7
FIGURE 6: EVOLUTION OF FUEL EFFICIENCY IN DIFFERENT AIRCRAFT MODELS OVER TIME. .................................................. 10
FIGURE 7: HISTORICAL EVOLUTION OF MAIN PERFORMANCE PARAMETERS. ............................................................................. 11
FIGURE 8: HISTORICAL EVOLUTION IN RELIABILITY. .................................................................................................................. 12
FIGURE 9: DIFFERENT MRO ACTIONS PERFORMED BY SPECIALIZED PERSONNEL. ................................................................. 13
FIGURE 10: OPEN FAN CONCEPT AND ITS INTEGRATION IN DIFFERENT LOCATIONS OF CONVENTIONAL AIRCRAFT. ..................... 14
FIGURE 11: SCHEMATIC DIAGRAM SHOWING THE WAY A PERFORMANCE MODEL TYPICALLY WORKS TO SOLVE A DIRECT PROBLEM. ........ 16
FIGURE 12: SOLVING SCHEME FOR THE INVERSE PROBLEM. ................................................................................................. 17
FIGURE 13: SCHEMATIC SHOWING THE POTENTIAL IMPLEMENTATION OF A REAL-TIME MODEL-BASED METHOD. ...................... 21
FIGURE 14: SOME EXAMPLES OF SENSORS INSTALLED IN A COMMERCIAL TURBOFAN NOWADAYS. .............................................. 22
FIGURE 15: SCHEMATIC EXAMPLE OF A TYPICAL COMMERCIAL FLIGHT PROFILE, SHOWING ITS DIFFERENT STAGES. ...................... 23
FIGURE 16: TYPICAL GENERIC DEGRADATION PATH DIAGRAM. .................................................................................................. 28
FIGURE 17: EXPLORATION FOLLOWED SO FAR WITHIN THE AVAILABLE LITERATURE. ................................................................. 48
FIGURE 18: MRO COSTS IN THE AIRLINES AND POTENTIAL COST SAVINGS. ................................................................................ 49
FIGURE 19: DATA FROM THE CF6 ENGINE FAMILY. .................................................................................................................... 68
FIGURE 20: CF6-6 ENGINE CUTAWAY DRAWING ....................................................................................................................... 69
FIGURE 21: CF6-6 SCHEMATIC SHOWING THE MATERIALS MAINLY USED IN ITS MANUFACTURING. ............................................... 71
FIGURE 22: DATA AND APPLICATIONS OF THE LM2500 AERODERIVATIVE GAS TURBINE. .......................................................... 73
FIGURE 23: THE CF6-80 ENGINES DID NOT NEED A TMF MODULE AS THE CF6-50 ENGINES DID. .................................................... 74
FIGURE 24: THE CF6-80 ENGINES INTRODUCED SEVERAL IMPROVEMENTS. ................................................................................ 75
FIGURE 25: THE CF6-80C2 WAS TAKEN AS A REFERENCE FOR THE DEVELOPMENT OF THE LM6000. ...................................................... 76
FIGURE 26: ENGINE’S AIR FLOWS IN THE CF6-80C2. .................................................................................................................... 77
FIGURE 27: CFM56 FAMILY. ..................................................................................................................................................... 79
FIGURE 28: CFM56-5B SCHEMATIC SHOWING ITS ARCHITECTURE. ............................................................................................... 81
FIGURE 29: CFMI FAN SECTION EVOLUTION. ............................................................................................................................. 82
FIGURE 30: COMBUSTION CHAMBERS USED IN THE DIFFERENT SERIES OF THE CFM56 FAMILY. .................................................... 84
FIGURE 31: ARCHITECTURE OF THE MOST POWERFUL MODEL OF THE GE90 SERIES. ................................................................ 85
FIGURE 32: LM9000 AERODERIVATIVE GAS TURBINE. ................................................................................................................ 87
FIGURE 33: CUTAWAY SCHEMATIC OF A GENX. ......................................................................................................................... 88
FIGURE 34: B777-9 FRONTAL VIEW. ........................................................................................................................................... 88
FIGURE 35: CUTAWAY SCHEMATIC OF A GE9X. ........................................................................................................................... 89
FIGURE 36: EVOLUTION OF THE TAPS COMBUSTION TECHNOLOGY IN THE LAST YEARS. .......................................................... 90
FIGURE 37: GE DIGITAL’S PREDIX® PLATFORM, HIGH-LEVEL SCHEMATIC LAYOUT. ................................................................. 92
FIGURE 38: DIFFERENCE BETWEEN ACCURACY AND PRECISION IN SENSOR READINGS. ............................................................ 93
FIGURE 39: TYPICAL INSTRUMENTATION MOUNTED IN A CF6-80C2. ........................................................................................... 95
FIGURE 40: SCHEMATIC DIAGRAM OF A FADEC. ........................................................................................................................ 100
FIGURE 41: CCR CABINETS LOCATION IN A B767. ....................................................................................................................... 100
FIGURE 42: GAS PATH INSTRUMENTATION IN A CFM56-5A. ......................................................................................................... 103
FIGURE 43: HIGH-BYPASS TWO-SPODL TURBOFAN ENGINE REPRESENTATION IN PROOSIS®. .................................................. 106
FIGURE 44: BALANCE OF VARIABLES AND EQUATIONS FOR A HIGH-BYPASS UNMIXED TWO-SPODL TURBOFAN ENGINE IN PROOSIS®. 107
FIGURE 45: SCHEMATIC REPRESENTATION OF A COMPRESSOR, AND ITS ASSOCIATED MAP, IN PROOSIS®. ................................. 108
FIGURE 46: SCHEMATIC REPRESENTATION OF A FAN, AND ONE OF ITS ASSOCIATED MAPS, IN PROOSIS®. ........................................ 110
FIGURE 47: SCHEMATIC REPRESENTATION OF A TURBINE, AND ITS ASSOCIATED MAP, IN PROOSIS®. ............................................... 111
FIGURE 48: SCHEMATIC REPRESENTATION OF A TURBINE, AND ITS ASSOCIATED MAP, IN PROOSIS®. ............................................... 112
FIGURE 49: Brayton cycle for a turbofan engine like the one under study. ....................................................................................... 116
FIGURE 50: FAN MAP (CORE), AS DEFINED IN PROOSIS®. ............................................................................................................ 129
FIGURE 51: FAN MAP (BYPASS), AS DEFINED IN PROOSIS®. ................................................................................................. 129
FIGURE 52: LPC MAP (BOOSTER), AS DEFINED IN PROOSIS®. ........................................................................................................ 130
FIGURE 53: HPC MAP, AS DEFINED IN PROOSIS®. ....................................................................................................................... 130
FIGURE 54: HPT MAP, AS DEFINED IN PROOSIS®. ....................................................................................................................... 131
FIGURE 55: LPT MAP, AS DEFINED IN PROOSIS®. ....................................................................................................................... 131
FIGURE 56: T’S DIAGRAM SHOWING AN ESST APPROACH IN A COOLED TURBINE. ................................................................. 135
FIGURE 57: EXAMPLES OF SHORT-HAUL FLIGHTS INSIDE EUROPE. ............................................................................................ 138
FIGURE 58: DATA OBTAINED FROM FLIGHT DISTRIBUTIONS. ........................................................................................................ 139
FIGURE 59: EXAMPLES OF MEDIUM-HAUL FLIGHTS. .................................................................................................................... 139
FIGURE 60: EXAMPLE OF LONG-HAUL FLIGHT. ............................................................................................................................ 140
FIGURE 61: EXAMPLE OF ULTRA-LONG-HAUL FLIGHT. .................................................................................................................. 141

ix
FIGURE 62: MAIN STAGES OF A TYPICAL SHORT-HAUL FLIGHT. ............................................................ 122
FIGURE 63: MAIN STAGES OF A TYPICAL LONG-HAUL FLIGHT. ......................................................... 123
FIGURE 64: INVERSE PROBLEM SCHEMATIC. ....................................................................................... 146
FIGURE 65: CFM56-SA TURBOFAN ENGINE SCHEMATIC. ................................................................. 151
FIGURE 66: CHARTS SHOWING THE STAIR-LIKE STRUCTURE FOUND. ............................................. 160
FIGURE 67: PROCESS TO SELECT THE RIGHT VALUE FOR δ ............................................................... 162
FIGURE 68: SCHEMATIC REPRESENTATION OF A FULL SVD OF A GENERIC RECTANGULAR MATRIX X. 172
FIGURE 70: SVD, PRODUCT OF THE SINGULAR VALUES AND LEFT AND RIGHT MODES. .................. 173
FIGURE 71: GEOMETRICAL INTERPRETATION OF THE SVD, IN 3D, GIVEN A RECTANGULAR MATRIX. 174
FIGURE 72: SCHEMATIC REPRESENTATION OF THE DIFFERENT SETS AND APPLICATIONS INVOLVED IN THIS INVERSE PROBLEM. ................................................................. 181
FIGURE 73: PERFORMANCE MAPS EMPLOYED BY PROOSIS®, FOR BOTH HPC AND HPT. ............... 183
FIGURE 74: VARIATIONS IN P1 3t WHEN ANY OF THE COMPONENTS OF X INCREASES FROM THE NEW AND CLEAN STATE. 185
FIGURE 75: VARIATIONS IN P2 5t WHEN ANY OF THE COMPONENTS OF X INCREASES FROM THE NEW AND CLEAN STATE. 186
FIGURE 76: VARIATIONS IN P3 t WHEN ANY OF THE COMPONENTS OF X INCREASES FROM THE NEW AND CLEAN STATE. 186
FIGURE 77: VARIATIONS IN T25t WHEN ANY OF THE COMPONENTS OF X INCREASES FROM THE NEW AND CLEAN STATE. 187
FIGURE 78: VARIATIONS IN T3t WHEN ANY OF THE COMPONENTS OF X INCREASES FROM THE NEW AND CLEAN STATE. 187
FIGURE 79: VARIATIONS IN T45t WHEN ANY OF THE COMPONENTS OF X INCREASES FROM THE NEW AND CLEAN STATE. 188
FIGURE 80: VARIATIONS IN T5t WHEN ANY OF THE COMPONENTS OF X INCREASES FROM THE NEW AND CLEAN STATE. 188
FIGURE 81: VARIATIONS IN NL WHEN ANY OF THE COMPONENTS OF X INCREASES FROM THE NEW AND CLEAN STATE. 189
FIGURE 82: VARIATIONS IN NH WHEN ANY OF THE COMPONENTS OF X INCREASES FROM THE NEW AND CLEAN STATE. 189
FIGURE 83: VARIATIONS IN WF WHEN ANY OF THE COMPONENTS OF X INCREASES FROM THE NEW AND CLEAN STATE. 190
FIGURE 84: RESTRICTION IMPOSED TO THE TIT VALUES. THE TARGET WAS TO SOLVE UNIQUENESS’ LACKS IN THE MODEL. 191
FIGURE 85: COUPLE OF FSI ERROR CODE NUMBERS AND ASSOCIATED MESSAGES IN PROOSIS®. 191
FIGURE 86: SCHEMATIC OF THE APPLICATION OF A ROM .................................................................. 194
FIGURE 87: SEVERAL EXAMPLES OF CHARTS OBTAINED WITH TENSORLAB®. 197
FIGURE 88: EXAMPLE OF VECTORIZATION OF ONE DEGRADATION TO FIND THE RIGHT POSITION INSIDE THE GRID OF THE TENSOR. 198
FIGURE 89: DECOMPOSITION APPLIED TO A 3D SENSOR. ................................................................. 199
FIGURE 90: MLSVD APPLIED TO A 3-D SENSOR, T, WITH A SMALLER CORE TENSOR. ........................ 201
FIGURE 91: CORE TENSOR SIZE CHART BY TENSORLAB® ............................................................... 206
FIGURE 92: MULTILINEAR SINGULAR VALUES OF THE ORIGINAL TENSOR, PROVIDED BY TENSORLAB®. 207
FIGURE 93: CHARTS REPRESENTING THE RESOLUTION, BY GAS, OF A RANDOM CASE WITH DEGRADATIONS UP TO A 2%. ................................................................. 213
FIGURE 94: CHARTS REPRESENTING THE RESOLUTION, BY GAS, WITH DEGRADATIONS UP TO 2% WITH CALCULATED T4t. ................................................................. 216
FIGURE 95: CHARTS AND DATA FROM FOUR RANDOM CASES (2% DEGRADATION) WHERE THE VALUES OF T4t WERE GIVEN. SQP ................................................................. 219
FIGURE 96: CHARTS AND DATA FROM FOUR RANDOM CASES (2% DEGRADATION) WHERE VALUES OF T4t WERE CALCULATED. SQP .................................................. 220
FIGURE 97: CHARTS WITH THREE RARE CASES (UP TO A 2% DEGRADATION AND TIT’S BEING UNKNOWN). SQP ................................................................. 221
FIGURE 98: CHARTS AND DATA FROM FOUR RANDOM CASES (20% DEGRADATION) WHERE T4t WERE CALCULATED. SQP ............................................. 222
FIGURE 99: RESULTS FROM 100 RANDOM CASES, INITIAL AND FINAL DEGRADATIONS UP TO A 2%. NEWTON ................................................................. 226
FIGURE 100: RESULTS FROM 100 RANDOM CASES, INITIAL AND FINAL DEGRADATIONS UP TO A 2%, 3 iterations. NEWTON ................................................................. 227
FIGURE 101: RESULTS FROM 100 RANDOM CASES, INITIAL AND FINAL DEGRADATIONS UP TO A 2%, 2 iterations. NEWTON ................................................................. 227
FIGURE 102: RESULTS FROM 100 RANDOM CASES, INITIAL AND FINAL DEGRADATIONS UP TO A 2%, 1 iteration. NEWTON ................................................................. 227
FIGURE 103: RESULTS FROM 100 RANDOM CASES, DEGRADATIONS UP TO 1%, INCREMENTS UP TO 0.1%, 3 iterations. NEWTON ................................................................. 228
FIGURE 104: RESULTS FROM 100 RANDOM CASES, DEGRADATIONS UP TO 1%, INCREMENTS UP TO 0.1%, 2 iterations. NEWTON ................................................................. 228
FIGURE 105: RESULTS FROM 100 RANDOM CASES, DEGRADATIONS UP TO 1%, INCREMENTS UP TO 0.1%, 1 iteration. NEWTON ................................................................. 228
FIGURE 106: FIGURE SHOWING THE REPLACED SENSORS. ............................................................... 229
FIGURE 107: RESULTS FROM 100 RANDOM CASES, DEGRADATIONS UP TO 2%, 3-4 iterations. P45t included. NEWTON ................................................................. 232
FIGURE 108: RESULTS FROM 100 RANDOM CASES, DEGRADATIONS UP TO 2%, 3 iterations. P45t included. NEWTON ................................................................. 232
FIGURE 109: RESULTS FROM 100 RANDOM CASES, DEGRADATIONS UP TO 2%, 2 iterations. NEWTON P45t included ................................................................. 233
FIGURE 110: RESULTS FROM 100 RANDOM CASES, DEGRADATIONS UP TO 2%, 1 iteration. P45t included. NEWTON ................................................................. 233
FIGURE 111: RESULTS FROM 100 RANDOM CASES, DEGRADATIONS UP TO 1%, INCREMENTS UP TO 0.1%, 3 iterations. NEWTON ................................................................. 234
FIGURE 112: RESULTS FROM 100 RANDOM CASES, DEGRADATIONS UP TO 1%, INCREMENTS UP TO 0.1%, 2 iterations. NEWTON ................................................................. 234
FIGURE 113: RESULTS FROM 100 RANDOM CASES, DEGRADATIONS UP TO 1%, INCREMENTS UP TO 0.1%, 1 iteration. NEWTON ................................................................. 234
FIGURE 114: RESULTS FROM 100 RANDOM CASES, DEGRADATIONS UP TO 20%. P45t included. NEWTON ................................................................. 236
FIGURE 115: RESULTS FROM 100 RANDOM CASES, DEGRADATIONS UP TO A 50%. P45t included. NEWTON ................................................................. 237
FIGURE 116: ITERATIONS AFTER 100 CASES SOLVED, FOR UNUSUAL OR EXTREME DEGRADATIONS. P45t included ................................................................. 238
FIGURE 117: CHARTS AND DATA FROM TWO RANDOM CASES (20% DEGRADATION). T4t was calculated. SQP ................................................................. 238
FIGURE 118: CHARTS AND DATA FROM TWO RANDOM CASES (50% DEGRADATION). T4t was calculated. SQP ................................................................. 239
FIGURE 119: LOCATION OF DIFFERENT CORE TENSORS USED TO SOLVE THE PROBLEM IN THE “ERROR-COMPRESS” CHART. ROM ................................................................. 240
FIGURE 120: RESULTS FROM 100 RANDOM CASES, USING THE CORE TENSOR OF SIZE [7 18 3 3 3 3 3 3 3 3 3 3]. ROM ................................................................. 242
FIGURE 121: RESULTS FROM 100 RANDOM CASES, USING THE CORE TENSOR OF SIZE [6 15 3 3 3 3 3 3 3 3 3 3]. ROM ................................................................. 242
FIGURE 122: RESULTS FROM 100 RANDOM CASES, USING THE CORE TENSOR OF SIZE [3 4 2 2 2 2 2 2 2 2 2 2]. ROM ................................................................. 242
FIGURE 123: RESULTS FROM 100 RANDOM CASES, INITIAL AND FINAL DEGRADATIONS UP TO A 2%. ROM ................................................................. 243
FIGURE 124: RESULTS FROM 100 RANDOM CASES, INITIAL AND FINAL DEGRADATIONS UP TO A 2%, ALLOWING 3 ITERATIONS. ROM ................................................................. 245
FIGURE 125: RESULTS FROM 100 RANDOM CASES, INITIAL AND FINAL DEGRADATIONS UP TO A 2%, ALLOWING 2 ITERATIONS. ROM. - 245 -
FIGURE 126: RESULTS FROM 100 RANDOM CASES, INITIAL AND FINAL DEGRADATIONS UP TO A 2%, ALLOWING 1 ITERATION. ROM. - 245 -
FIGURE 127: RESULTS FROM 100 RANDOM CASES, DEGRADATIONS UP TO A 1%, INCREMENTS UP TO 0.1%, 3 ITERATIONS. ROM. - 246 -
FIGURE 128: RESULTS FROM 100 RANDOM CASES, DEGRADATIONS UP TO A 1%, INCREMENTS UP TO 0.1%, 2 ITERATIONS. ROM. - 246 -
FIGURE 129: RESULTS FROM 100 RANDOM CASES, DEGRADATIONS UP TO A 1%, INCREMENTS UP TO 0.1%, 1 ITERATION. ROM. - 246 -
FIGURE 130: SCHEMATIC CONCEPT OF THE LSA APPLIED TO THE TENSORS GENERATED. ROM. - 250 -
FIGURE 131: EVOLUTION OF THE DEGRADATION IN A TYPICAL TURBOFAN ENGINE. - 257 -
FIGURE 132: SCHEMATIC REPRESENTATION OF A HIERARCHICAL ALGORITHM TO CALCULATE ENGINE’S DEGRADATION. - 259 -
FIGURE 133: SUGGESTED PRESENTATION OF RESULTS IN THE LOWER SCREEN OF AN ECAM. - 259 -
FIGURE 134: SUGGESTED PRESENTATION OF THE RESULTS OBTAINED WITH THE METHODOLOGY. - 261 -
FIGURE 135: EVOLUTION WITH TIME OF ONE OF THE COMPONENTS OF THE DEGRADATION VECTOR AND RUL. - 262 -
FIGURE 136: TSFC INCREASE (CRUISE) WITH DEGRADATION IN THE MAIN ENGINE COMPONENTS, ACCORDING WITH MODEL. - 266 -
FIGURE 137: TSFC PERCENTUAL INCREASE WITH THE DEGRADATION IN THE DIFFERENT MAIN ENGINE COMPONENTS. - 267 -
FIGURE 138: MAINTENANCE PROGRAM SCHEMATIC FOR THE 1st MOH CYCLE ("HONEYMOON"), INCLUDING WASHES (CF6-80E1). - 271 -
FIGURE 139: INTEGRATION OF THE VARIABLE Wf. - 276 -
FIGURE 140: SIMPLIFIED VERSION OF A BBN. THE GRAPH SHOWS ITS MOST RELEVANT ELEMENTS: NODES AND EDGES. - 286 -
FIGURE 141: SIMPLIFIED VERSION OF A FEEDFORWARD NN. - 287 -
LIST OF TABLES

<table>
<thead>
<tr>
<th>Table</th>
<th>Description</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>Table 1</td>
<td>Financial data regarding fuel consumption costs and aircraft maintenance costs.</td>
<td>3</td>
</tr>
<tr>
<td>Table 2</td>
<td>Financial data comparing the costs associated to fuel consumption and maintenance.</td>
<td>4</td>
</tr>
<tr>
<td>Table 3</td>
<td>Engine census.</td>
<td>9</td>
</tr>
<tr>
<td>Table 4</td>
<td>Some examples of sensors installed in the engine.</td>
<td>96</td>
</tr>
<tr>
<td>Table 5</td>
<td>Some examples of sensors typically installed in the lubrication system.</td>
<td>97</td>
</tr>
<tr>
<td>Table 6</td>
<td>Some examples of accelerometers used in aircraft engines.</td>
<td>98</td>
</tr>
<tr>
<td>Table 7</td>
<td>Parameters for the calculation of the design point of a turbofan engine.</td>
<td>122</td>
</tr>
<tr>
<td>Table 8</td>
<td>Model validation: comparative between engine data and model results.</td>
<td>132</td>
</tr>
<tr>
<td>Table 9</td>
<td>Model outputs for the design point at cruise conditions.</td>
<td>133</td>
</tr>
<tr>
<td>Table 10</td>
<td>Nozzle discharge coefficient as a function of the NPR, with a nozzle angle of 10 degrees.</td>
<td>137</td>
</tr>
<tr>
<td>Table 11</td>
<td>Typical flight profile considered for a short-haul flight.</td>
<td>142</td>
</tr>
<tr>
<td>Table 12</td>
<td>Typical flight profile considered for a long-haul flight.</td>
<td>143</td>
</tr>
<tr>
<td>Table 13</td>
<td>Degradation vector components and instrumentation vector components.</td>
<td>150</td>
</tr>
<tr>
<td>Table 14</td>
<td>Scaling values obtained from the direct problem results.</td>
<td>152</td>
</tr>
<tr>
<td>Table 15</td>
<td>Associated singular values to the Jacobians.</td>
<td>176</td>
</tr>
<tr>
<td>Table 16</td>
<td>List of coefficients for the T4t values used in the previous table.</td>
<td>177</td>
</tr>
<tr>
<td>Table 17</td>
<td>Representative data of the engine (new and clean condition), simulated in a generic cruise flight.</td>
<td>185</td>
</tr>
<tr>
<td>Table 18</td>
<td>Results obtained with GAs. Degradations up to a 2% for 100 individuals.</td>
<td>212</td>
</tr>
<tr>
<td>Table 19</td>
<td>Degradation vector obtained using GAs 10 unknowns, TIT given.</td>
<td>214</td>
</tr>
<tr>
<td>Table 20</td>
<td>Degradation vector obtained using GAs. Calculated TIT.</td>
<td>215</td>
</tr>
<tr>
<td>Table 21</td>
<td>Results obtained in two cases by using SQP, with degradations up to a 2%, for given values of T4t.</td>
<td>217</td>
</tr>
<tr>
<td>Table 22</td>
<td>Results obtained by using SQP, calculating the values of T4t as unknowns.</td>
<td>218</td>
</tr>
<tr>
<td>Table 23</td>
<td>Results obtained with the adapted Newton method. T4t given.</td>
<td>223</td>
</tr>
<tr>
<td>Table 24</td>
<td>Results obtained with the adapted Newton method. T4t calculated.</td>
<td>224</td>
</tr>
<tr>
<td>Table 25</td>
<td>Results obtained with the adapted Newton method for degradations up to a 20%, T4t values given.</td>
<td>225</td>
</tr>
<tr>
<td>Table 26</td>
<td>Results obtained with the adapted Newton for degradations up to a 20%. T4t values calculated.</td>
<td>225</td>
</tr>
<tr>
<td>Table 27</td>
<td>Case a). List of coefficients for different typical cruise T4t, after replacing P25t by P45t.</td>
<td>230</td>
</tr>
<tr>
<td>Table 28</td>
<td>Case b). Analogous table, replacing NH by P45t.</td>
<td>230</td>
</tr>
<tr>
<td>Table 29</td>
<td>Case c). Analogous table, replacing T5t by P45t.</td>
<td>230</td>
</tr>
<tr>
<td>Table 30</td>
<td>Results obtained with the adapted Newton including P45t in the instrumentation. T4t calculated.</td>
<td>231</td>
</tr>
<tr>
<td>Table 31</td>
<td>Results obtained with the adapted Newton including P45t, degradations up to 20%, T4t calculated.</td>
<td>235</td>
</tr>
<tr>
<td>Table 32</td>
<td>Results obtained with the adapted Newton including P45t, degradations up to 50%, T4t calculated.</td>
<td>236</td>
</tr>
<tr>
<td>Table 33</td>
<td>Summary of the cases run with the different core tensors. ROM.</td>
<td>241</td>
</tr>
<tr>
<td>Table 34</td>
<td>Results obtained with random calculated TIT values. ROM.</td>
<td>243</td>
</tr>
<tr>
<td>Table 35</td>
<td>Instrumentation’s considered full scales and accuracies.</td>
<td>252</td>
</tr>
<tr>
<td>Table 36</td>
<td>Verification of instrumentation when capturing changes of 1% in the different H&amp;Q parameters.</td>
<td>252</td>
</tr>
<tr>
<td>Table 37</td>
<td>Verification of instrumentation when capturing changes of 0.5% in the different H&amp;Q parameters.</td>
<td>252</td>
</tr>
<tr>
<td>Table 38</td>
<td>Examples of real-time applications based on the criticality of the associated processes.</td>
<td>254</td>
</tr>
<tr>
<td>Table 39</td>
<td>Summarizing table for GAs and SQP.</td>
<td>256</td>
</tr>
<tr>
<td>Table 40</td>
<td>Summarizing table for adapted Newton used with both the full model and ROM.</td>
<td>256</td>
</tr>
<tr>
<td>Table 41</td>
<td>Summarizing table for potential business case.</td>
<td>275</td>
</tr>
</tbody>
</table>
LIST OF ACRONYMS

- ACARS: Aircraft Communications Addressing and Reporting System.
- ACCV: Active Control Clearance Valve.
- ADEM: Advanced Diagnostic and Engine Maintenance.
- AFT: Aft ward.
- AGB: Auxiliary Gearbox.
- ANN: Artificial Neural Networks.
- ASD: Aeronautical Standard Data Buses.
- ATAG: Air Transport Action Group.
- AVM: Airborne Vibration Monitoring.
- BBN: Bayesian Belief Networks.
- BFGS: Broyden-Fletcher-Goldfarb-Shanno formula/method.
- BP: Bypass.
- BP-NN: Back Propagation Neural Network.
- BPR: Bypass Ratio.
- BWB: Blended Wing Body.
- C-MAPPS: Commercial Modular Aero Propulsion System Simulation.
- CAEP: Committee on Aviation Environmental Protection.
- CBM: Condition-Based Maintenance
- CC: Combustion Chamber.
- CCCV: Core Compartment Cooling Valve.
- CDP: Compressor Discharge Pressure.
- CEA: Chemical Equilibrium with Applications by NASA.
- CEO: Chief Executive Officer.
- CIP: Component Improvement Program.
- CMC: Ceramic Matrix Composite.
- CN: Conditioning Number.
- CPU: Central Processing Unit.
- CRF: Compressor Rear Frame.
- CROR: Contra-Rotating Open Rotor.
- CS: Control System.
- CSDB: Commercial Standard Data Buses.
- CV: Constant Volume Combustor Cycle.
- DAC: Double Annular Combustor.
- DCS: Distributed Control System.
- DOD: Domestic Object Damage.
- EB: Electron Beam.
- EC: European Community.
- ECAM: Electronic Centralized Aircraft Monitor.
- ECI: Engine Component Improvement program.
- ECU: Engine Control Unit.
- EDMS: Exhaust Debris Monitoring System.
- EEC: Electronic Engine Control.
- EEE: Energy Efficient Engine program (E3).
- eFAST: Electronic Full-flight data Acquisition, Storage and Transmission.
- EGT: Exhaust Gas Temperature.
- EHM: Engine Health Management.
- EICAS: Engine-Indicating and Crew-Alerting System.
- EKF: Extended Kalman Filter.
- EMO: Engine Maintenance Optimization program.
- EMU: Engine Monitoring Unit.
- EPNb: Effective Perceived Noise (measured in decibels).
- EPR: Engine Pressure Ratio.
- ES: Evolution Strategy (GA variant) and Expert System as well.
- ESST: Equivalent Single-Stage Turbine.
- FAA: Federal Aviation Administration.
• FADEC: Full Authority Digital Engine Control.
• FAR: Federal Aviation Regulations or Fuel-to-Air Ratio.
• FARB: Fuel-to-Air Ratio Burnt.
• FARU: Fuel-to-Air Ratio Unburnt.
• FC: Flight Cycle.
• FH: Flight Hour.
• FCM: Fault Coefficient Matrix.
• FDD: Fault Detection and Diagnostic.
• FL: Fuzzy Logic.
• FMV: Fuel Measuring Valve.
• FOD: Foreign Object Damage.
• FOQA: the Flight Operations Quality Assurance.
• FPR: Fan Pressure Ratio.
• FS: Full Scale.
• FWD: Forward.
• GA: Genetic Algorithm.
• GE: General Electric.
• GMS: Generalized Minimum Squares.
• GPA: Gas Path Analysis.
• GSS: Ground Support Services in an airport.
• HMM: Hidden Markov Models.
• HMU: Hydro-Mechanical Unit.
• HOSVD: Higher Order Singular Value Decomposition.
• HOT: Higher Order Terms.
• HPC: High-Pressure Compressor.
• HPT: High Pressure Turbine.
• HPTACC: HPT Active Clearance Control.
• HPTC: High-Pressure Turbine Cooling.
• HSE: Hot Section Exchange.
• HIS: Hot Section Inspection.
• HSMM: Hidden Semi-Markov Model.
• H&Q: Health and Quality.
• IATA: International Air Transport Association.
• ICAO: International Civil Aviation Organization.
• ICM: Influence Coefficient Matrix.
• ICR: Inter-cooling regenerative cycle.
• ISA: International Standard Atmosphere.
• ISO: International Standards Organization.
• IDMS: Inlet Debris Monitoring Sensor.
• IEKF: Invariant Extended Kalman Filter.
• IFSD: In Flight Shut Down.
• IGV: Inlet Guide Vanes.
• JAR: Joint Aviation Requirements.
• KF: Kalman Filters.
• LAPACK: Linear Algebra PACKage for FORTRAN90.
• LGPA: Linearized Gas Path Analysis.
• LHV: Lower Heating Value.
• LKF: Linear Kalman Filter.
• LLP: Life Limited Parts.
• LOBEM: Linear On-Board Engine Model.
• LPC: Low-Pressure Compressor.
• LPT: Low-Pressure Turbine.
• LPTACC: LPT Active Clearance Control.
• LPTC: Low-Pressure Turbine Cooling.
• LSA: Latent Semantic Analysis.
• LVDT: Linear Variable Displacement Transducers.
• MEC: Hydro-Mechanical Controller.
• MEMS: Micro-Electro-Mechanic Sensor.
• MFT: Map Fitting Tool by NASA.
• MLS: Minimum Least Squares.
• MLSVD: Multi Linear Single Value Decomposition.
• MOH: Major Overhaul.
• MRO: Maintenance, Repair, and Overhaul.
• MS: Monitoring System.
• MSMHSMM: Multi-Sensor Mixture Hidden Semi-Markov Model.
• MW: Megawatts / Mechanical power delivered by a gas turbine (shaft).
• NASA: National Aeronautics and Space Administration.
• NGV: Nozzle Guide Vanes.
• NLGPA: Non-Linear Gas Path Analysis.
• NN: Neural Networks.
• NPR: Nozzle Pressure Ratio.
• OBEM: On-Board Engine Model.
• OEM: Original Equipment (or Engine) Manufacturer.
• OF: Objective Function.
• OPEC: Organization of the Petroleum Exporting Countries.
• OPR: Overall Pressure Ratio.
• PHM: Propulsion Health Management System.
• PIMU: Propulsion Interface Monitor Unit.
• PMA: Parts Manufacturer Approval.
• PMM: Performance Monitoring Module in PROOSIS®.
• PR: Pressure Ratio.
• PROOSIS®: Propulsion Object Oriented Simulation Software.
• PS: Plasma Spray.
• PT: Power Turbine.
• PVD: Physical Vapor Deposition.
• P&W: Pratt and Whitney.
• RAS: Random Adaptive Search.
• RCA: Root Cause Analysis.
• RISE: Revolutionary Innovation for Sustainable Engines.
• RMS: Root Mean Square.
• ROI: Return on Investment.
• ROM: Reduced Order Model.
• RPK: Revenue Passenger Kilometer.
• RR: Rolls-Royce.
• RRMS: Relative Root Mean Square.
• RTD: Resistance Temperature Detector.
• RTEDS: Real-Time Engine Diagnostic System.
• RTO: Rejected Takeoff.
• RUL: Remaining Useful Life.
• R&D: Research and Development.
• SAC: Single Annular Combustor.
• SARS-CoV-2: Severe Acute Respiratory Syndrome Coronavirus 2.
• SAS: Secondary Airflows (cooling in PROOSIS®).
• SDF: Symbolic Dynamic Filtering.
• SI: Spark Ignited, and International System (of units).
• SFC: Specific Fuel Consumption.
• SL: Sea Level.
• SOT: Stator Outlet Temperature.
• SPSO: Self-Tuning Particle Swarm Optimization.
• SQP: Sequential Quadratic Programming.
• STANAG: Standardized NATO Agreement for the transfer of data.
• STARGATE: Sensors Towards Advanced Monitoring of GT Engines program.
• SV: Shop Visit.
• SVD: Singular Value Decomposition.
• SVR: Support Vector Regression or Shop-Visit Rate.
• SW: Software.
• SWAN: Stress Wave Analysis.
• TAPS: Twin Annular Premixing Swirler combustor.
• TBC: Thermal Barrier Coatings.
• TBO: Time Between Overhauls.
• TBV: Thrust Balance Valve.
• TC: Thermocouple.
• TD: Tucker Decomposition.
• TIT: Turbine Inlet Temperature.
• TLA: Thrust Lever Angle (Thrust or Power setting determining the regime).
• TMF: Turbine Mid Frame.
• TO: Take-off.
• TOC: Top of Climb.
• TOSL: Take-Off at Sea Level.
• TOW: Time-On-Wing.
• TRF: Turbine Rear Frame.
• TSFC: Thrust Specific Fuel Consumption.
• UHC: Unburnt Hydrocarbons.
• UKF: Unscented Kalman Filter.
• USAF: United States Air Force.
• VAC: Volts AC.
• VATN: Variable Area Turbine Nozzle.
• VDC: Volts DC.
• VBV: Variable Bleed Valves.
• VSV: Variable Stator Vanes.
• VG: Variable Geometry.
• WAR: Water-to-Air Ratio.
• WHO: World Health Organization.
• WRTC: Wave Rotor Topping Cycle.
LIST OF SYMBOLS

The nomenclature used in this document is well explained in the different chapters, where it is needed. Usual mathematical operators and associated nomenclature has been kept aiming to facilitate the reading. For reference, the following list (not comprehensive) of symbols and notations applies unless specifically indicated otherwise in the text:

- $\epsilon$ is typically used to denote a termination threshold in an algorithm.
- $\eta$ stand for adiabatic efficiencies in the different modules of the engine (e.g., $\eta_{\text{FAN}}, \eta_{\text{LPT}}, \text{etc.}$).
- $\delta$ denotes an increment when computing derivatives by finite differences. It also denotes the required pressure correction factor used in certain parameters in the performance calculations.
- $\Delta$ is typically used to denote both scalar and vectorial increments (e.g., $\Delta T, \Delta \vec{X}$).
- $\Gamma_{\text{r}}$ stands for mass flow capacities in the different modules of the engine (e.g., $\Gamma_{\text{FAN}}, \Gamma_{\text{LPT}}, \text{etc.}$).
- $\theta$ denotes temperature corrections factor used in certain parameters in the performance calculations.
- Approximate matrix of a Hessian: Symmetric and, typically, positive definite, denoted as $\vec{H}$.
- Cross-sectional areas are indicated by $A_i$ (e.g., $A_{\text{a}}$).
- Engine stations will be numbered as per SAE ARP755 A (e.g., station “45" is in between HPT and LPT, etc.).
- Gradient of scalar function $F$, $\nabla F_b$, calculated in step $k$.
- Hessian matrix of $F$, calculated in step $k$: $H_k = \nabla^2 F_k$.
- Indexes go with regular letters, being their maximum indicated by a capital letter (e.g., $i = 1, 2, \ldots , I$).
- Jacobian matrix once included both TIT and $P_{45s}$ and was denoted by $J_k$.
- $M$ is usually denoting the total number of measurements of readings from sensors. $M$ also indicates mean.
- Air mass flows are typically indicated by $W$.
- Matrices are indicated by bold capital letters (e.g., $\mathbf{A}, \mathbf{D}, \mathbf{H}, \mathbf{I}$, etc.). Meaning $\mathbf{D} = D_{ij}$.
- $N_{\text{deg}}$ is for the different components of the degradation vector.
- $N_{\text{sen}}$, for the different components of the instrumentation vector.
- $N$-mode product of $\vec{S}$, with a size $(l_1 \times l_2 \times \ldots \times l_n)$, by a matrix $U$, of size $(l_m \times l_n)$, denoted as $\vec{S} \times_{l_m} U$.
- Operator $\| \cdot \|_2$ stands for the usual Euclidean norm.
- Operator $\| \cdot \|_F$ stands for the usual Frobenius norm.
- Outer product of two vectors, $\vec{u}$ and $\vec{v}$, delivers a matrix, $(\vec{u} \otimes \vec{v})_{ij} = u_i v_j$.
- Pseudo-inverse matrix is denoted with the "$^\dagger$" exponent, or $\mathbf{A}^{\text{pseudo}}$.
- Quadratic approximation of function $F$ goes with angular top line (e.g., $\tilde{F}$).
- Rank-$Q$ matrix, $\mathbf{A}^{\text{approx}}$, approximation of $A$.
- Ranks used in the thesis are typically denoted by the "rank" operator. Specific versions for tensors are indicated in the respective paragraphs.
- Reference values (e.g., ambient, conditions in maps, etc.) are indicated with "ref".
- Scalar magnitudes are indicated by both lowercase and capital letters (e.g., $x, X, T, \delta$, etc.).
- Sets of points in a particular vectorial space are given by italic capital letters (e.g., Set $X\text{etc}$).
- Static magnitudes (stagnation) at the different engine stations have a "s" in the suffix (e.g., $P_{45s}$ etc.). Not to be confused with the magnitudes after an isentropic evolution in a T-s diagram that will also be denoted with a "s" (i.e., $P_{45s} \rightarrow P_{\text{is}}$, meaning isentropic expansion, $s$ = constant, from station 45 to station 5).
- $T_{45}^1$ and $T_{5s}^0$, two reference values of $T_{45}$, low and high, respectively, during cruise phases.
- Sensors are indicated by bold capital letters with double top line (e.g., $\mathbf{T}, \mathbf{S}$, etc.).
- Tensor of third order is expressed the following way by its respective suffixes: $\mathbf{A} = A_{ijk}$. Suffixes are generalized to higher dimensions as usual, in a natural way (as indicated in Chapter 4).
- The approximate solution of a vector $\vec{Y}$, will be represented as $\hat{\vec{Y}}$, for the sake of clarity.
- Total magnitudes (stagnation) at the different engine stations have a "T" in the suffix (e.g., $P_{45T}, T_{13T}$, etc.).
- Usual function inverse is denoted with the "$^{-1}$" exponent. Similar with rest of applications and matrices.
- Usual matrix transpose is denoted with the "T" exponent.
- Usual n-dimensional vectorial spaces: $\mathbb{R}^n$.
- Usual real functions of real variables are denoted with regular letters (e.g., $g(T)$).
- Usual scalar product with vectors is typically indicated with the operator " : " (e.g., $\Delta X_k^T \cdot \Delta \vec{g}_k$).
- Usual scalar product with tensors is typically represented by the (, ) operator.
- Vector $\vec{X}$ contains the 10 initial components of degradation vector plus TIT, once incorporated $P_{45s}$.
- Vectorial components can be denoted by parenthesis or by suffixes (e.g., $\hat{\vec{X}}(j), X_i, X(j)$).
- Vectorial magnitudes are typically indicated with a top line (e.g., $\hat{\vec{X}}, \vec{Y}, \vec{Z}$ etc.).
- Vectorial functions, application between vectorial spaces, are normally denoted with bold letters (e.g., $f(\vec{X})$).
- $\vec{X}$ will be typically the components' parameters vector, or degradation vector.
- $\vec{Y}$ typically represents the condition or instrumentation vector.
- $\vec{Y}_{\text{meas}}$ vector with measurements.
- $Y_{\text{scal}}(i)$ is the resultant variable after the scaling process.
1.- INTRODUCTION: MOTIVATION AND TARGETS

1.1.- Current historical context and reasons for this research

The modern air transportation constitutes, not only an engineering prodigy, but also one of the main pillars of the economy in our society. Even after the devastating worldwide shock that the SARS-CoV-2 virus outbreak meant for every human activity in the first quarter of 2020 (leading to the COVID-19 pandemic, as it was named by the World Health Organization), disaster that has caused millions of deaths and has spread the uncertainty globally in the air transport sector as a result of the necessary cease of travelling to control the expansion of the disease, the commercial aviation has remained as one of the principal instruments, for both private companies and governments, to continue developing their economic activity. This situation is not totally new, as past catastrophic events have already compromised the stability of the international commercial aviation (see Brauer et al., 2012, [43]), without success fortunately. Moreover, the role of the air transport will be of paramount importance in the future recovery of the global economy to pre-COVID levels, the same way it happened before when other calamities, such as armed conflicts or natural disasters, impacted dramatically the normal course of history (ATAG, 2020, [1]). Contributing to improve this essential activity, by getting a better knowledge on the condition of the equipment necessary for its development, is the main and last aim of this work.

Figure 1: Impact of COVID-19 in the commercial air traffic worldwide, resulting in a 50% reduction of flights’ intensity between 2019 (upper) and 2020 (lower). The image was obtained from ICAO’s website [143].
The air transport mostly relies, from a technical point of view, in the modern jet aircraft, which are powered predominantly by gas turbine engines, and more precisely, by high-bypass ratio turbofans. This dominance is a consequence of the high fuel efficiency levels they can reach and the remarkable thrust values these machines can afford by accelerating large masses of air to relatively modest jet velocities. This advance makes possible the transportation of hundreds of passengers, together with the movement of heavy payloads, throughout increasingly longer routes, in just one flight. The performance of these mechanic marvels is being improved continuously and their technical capabilities’ envelope is frequently expanded, by incorporating new materials and designs to its production, in what constitutes an outstanding tour de force of the whole aerospace industry (Rolls-Royce, 2015, [244]).

![Figure 2: Example of a modern jet aircraft, the Boeing 777X, powered by two GE9X turbofan engines. The image was obtained from Boeing’s website [37].](image)

A better knowledge on the health condition (or degradation status) of these engines will eventually allow their operators (i.e., airlines) to optimize its use as much as technically feasible, and to make informed decisions regarding the required maintenance they will inevitably need, sooner or later. It is typically preferred to apply such maintenance the latest possible, keeping the engine operative for the longest time, to increase the return on investment (ROI) of the financial operating asset that each engine of an airline’s aircraft fleet eventually is. Any cost reduction that might be obtained with such information could mean the difference between the survival and the financial collapse of the airlines, given the difficult environment, economically speaking, in which they work. Just to illustrate this statement with a real example, during 2008 there was a global sudden peak in the prices of crude oil that impacted seriously to the financing of the main airlines worldwide during several months. Cathay Pacific, the flag carrier of Hong Kong, reckoned an estimated loss of more than 160 millions of US dollars (USD) for every cent of increase in fuel price (Morrell, 2013, [203]). And this is only one example of the multiple obstacles, such as the increasingly stricter environmental regulations, or the current aggressive ticket pricing strategies, existing in the hostile environment in which the airlines operate today.

These facts lead naturally to the main motivation for this thesis, which is no other that the efficient determination of the health condition of this kind of engines (and some others, as it will be explained later), in a fast and reliable way. For an airline, getting to know the degree of deterioration of an aircraft engine, which practically equals to getting to know for how long that asset can be safely used, and
for how much cost, is an extremely relevant matter today considering the extreme financial pressure that all the commercial aviation industry is suffering, very especially since the last dramatic year, added to the persistently growing competitiveness in this strategic market.

Before the pandemic irrupted into the course of world economy, the airlines were managing cost breakdowns like the ones shown in Table 1 for some relevant companies. About a third of the total costs in an airline came from the fuel consumed by the engines, which maintenance meant about the 9.4%, according to the estimations provided by the International Air Transport Association (IATA). These data serve to foresee the economic scenario for the airlines that will hypothetically find once pre-COVID economic activity levels could be restored. The table provides costs in millions of USD (considering 2021 price levels):

<table>
<thead>
<tr>
<th>Airlines</th>
<th>IAG</th>
<th>Air France - KLM</th>
<th>Lufthansa Group</th>
<th>Turkish Airlines</th>
<th>Air Asia</th>
<th>American Airlines</th>
<th>Delta Airlines</th>
</tr>
</thead>
<tbody>
<tr>
<td>Fuel Costs</td>
<td>7,165</td>
<td>6,558</td>
<td>7,991</td>
<td>3,873</td>
<td>1,009</td>
<td>7,526</td>
<td>8,519</td>
</tr>
<tr>
<td>Maintenance Costs (Aircraft)</td>
<td>2,490</td>
<td>3,127</td>
<td>2,274</td>
<td>791</td>
<td>319</td>
<td>2,380</td>
<td>1,751</td>
</tr>
<tr>
<td>Maintenance Costs (Engines)</td>
<td>1,021</td>
<td>1,282</td>
<td>932</td>
<td>324</td>
<td>131</td>
<td>976</td>
<td>718</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Airlines</th>
<th>Etihad</th>
<th>Emirates</th>
<th>THAI</th>
<th>JAL</th>
<th>Air New Zealand</th>
<th>LATAM</th>
<th>Aero Mexico</th>
</tr>
</thead>
<tbody>
<tr>
<td>Fuel Costs</td>
<td>4,900</td>
<td>8,307</td>
<td>1,695</td>
<td>2,237</td>
<td>902</td>
<td>2,929</td>
<td>979</td>
</tr>
<tr>
<td>Maintenance Costs (Aircraft)</td>
<td>1,244</td>
<td>652</td>
<td>599</td>
<td>689</td>
<td>284</td>
<td>445</td>
<td>231</td>
</tr>
<tr>
<td>Maintenance Costs (Engines)</td>
<td>510</td>
<td>267</td>
<td>246</td>
<td>282</td>
<td>116</td>
<td>182</td>
<td>95</td>
</tr>
</tbody>
</table>

Table 1: Financial data regarding fuel consumption costs and aircraft maintenance costs, in the fiscal year relative to 2019, before the virus outbreak, for some relevant airlines. Footnotes were consulted from several IATA reports. For further reference, see [94], [136], and [137].

More deteriorated engines mean higher losses of efficiency, higher fuel consumptions, and higher costs of maintenance that compromise the financial balances of the airlines. In the currently challenging economic environment, illustrated in Table 2 with official data from few airlines during the first quarter of 2020 and 2021, respectively, the impact of COVID-19 results evident by the abrupt cut of budget dedicated to fuel, meaning a dramatic reduction of activity. Every company operating inside this market would be highly motivated to try to minimize its costs under such difficult circumstances. This could be achieved by maximizing the efficiency and by avoiding excessive degradation and wear in the engines, while trying to keep them in operation as much time as technically possible and financially reasonable, optimizing this way the incomes from passengers and freight. An aircraft stopped in the ground is not a productive asset, but a liability.
A percentage point improvement in maintenance and efficiency will mean, according to previous tables, reductions of costs in the order of tens of millions of USD per year for every airline. This is an appealing target, however the previous step to improve both is determining, fast and accurately, the status of every engine in the fleet, ideally on real-time basis, which constitutes a challenging problem. If one engine was detected to be dirty, a water wash could be scheduled in the next airport according to previous tables, which constitutes a challenging problem.

The complexity of this problem, especially for the engine operators and manufacturers, will be introduced now in some more detail, but still preliminarily in this first chapter, guiding progressively the reader into the subject of the analysis. It is necessary to clarify beforehand that the thesis will deal primarily with turbofan engines, given their predominance nowadays, but the usefulness associated to this research could be potentially benefitting to any other gas turbine engine model, not necessarily a turbofan (e.g., turboprop, aeroderivative gas turbine, etc.), no matter if such gas turbine engine is employed in aviation, marine or industrial applications. This circumstance will certainly redound in a greater interest on the study.

### 1.2.- Introducing the high-bypass two-spool turbofan engine

Regarding the technical features of turbofan engines employed in aviation, they are sophisticated air-breathing, not isothermal, mechanical systems that convert chemical energy from a fuel, by means of a combustion process, into mechanical energy, making use of different internal rotary and static parts installed in different modules (see Figure 3). Such mechanical energy is applied to the air surrounding the engine’s components, forcing it to move in one direction and leading eventually, because of Newton’s third law of motion, to the propulsion of the system (and any other part mechanically linked to it) in the opposite direction.

---

Table 2: Financial data comparing the costs associated to fuel consumption and maintenance, in the first quarters (Q1) of 2021 and 2020, respectively, for some relevant airlines. The effect of the lockdowns in the sector is evident. Data from the websites of the companies and IATA (go to [94], [136], and [137] for further details).

<table>
<thead>
<tr>
<th>Airlines</th>
<th>IAG</th>
<th>Air France - KLM</th>
<th>Lufthansa Group</th>
<th>Turkish Airlines</th>
<th>Air Asia</th>
<th>American Airlines</th>
<th>Delta Airlines</th>
</tr>
</thead>
<tbody>
<tr>
<td>Financial Period</td>
<td>2021 Q1</td>
<td>2020 Q1</td>
<td>2021 Q1</td>
<td>2020 Q1</td>
<td>2021 Q1</td>
<td>2020 Q1</td>
<td>2021 Q1</td>
</tr>
<tr>
<td>Fuel Costs</td>
<td>288</td>
<td>1,209</td>
<td>551</td>
<td>1,410</td>
<td>327</td>
<td>1,460</td>
<td>418</td>
</tr>
<tr>
<td>Maintenance Costs (Aircraft)</td>
<td>207</td>
<td>504</td>
<td>411</td>
<td>731</td>
<td>257</td>
<td>571</td>
<td>124</td>
</tr>
<tr>
<td>Maintenance Costs (Engines)</td>
<td>85</td>
<td>207</td>
<td>169</td>
<td>300</td>
<td>105</td>
<td>234</td>
<td>51</td>
</tr>
</tbody>
</table>

**Notes:**
- All costs are in millions of USD (2021).
- By February 2020, the first air travel restrictions were applied.
- By March 2020, the World Health Organization (WHO) declared COVID-19 a pandemic.
- By April 2020, half of world’s population was under some form of lockdown.
- According to IATA, aircraft’s maintenance costs mean a 9.4% of the total airlines’ cost structure, in average.
- A 41% of that maintenance cost goes to the engines.
The design of these engines is modular, as shown in Figure 4. The main engine modules are, listing from the inlet to the exhaust (forward looking aft): The Fan, the Low-Pressure Compressor (LPC, also called traditionally “booster”), the High-Pressure Compressor (HPC), the combustion chamber (CC, also known as combustor), the High-Pressure Turbine (HPT), and the Low-Pressure Turbine (LPT). The set formed by HPC, CC, and HPT is also known as Core (or Core Engine).
Such modular configuration greatly facilitates the different maintenance tasks performed over these engines, as it is possible to replace the CC or the rotatory part of the HPT without affecting to the Fan of the other compressors. Such design solution was developed to minimize the outage time caused by reparation.

The engines are typically defined, in addition to their size and thrust level, or power delivered, by several design parameters like: The Bypass Ratio (BPR), which is the ratio between the air mass flow going through the Fan (called also secondary or bypass flow) and the air mass flow going through the combustion chamber (called also primary flow); the Fan Compression Ratio (FCR), which is the ratio of total pressures before and after the Fan; the Overall Compression Ratio (OPR), accounting for the pressure ratio existing between the end of the HPC and the engine’s inlet, which is the total compressing capability of the engine; and the exhaust gas temperature (EGT), which is directly related with some other parameters like $T_{4t}$, or turbine inlet temperature (TIT). These parameters will be used and commented with more detail in the third chapter of the thesis. Engine stations will be numbered as per SAE ARP755 A.

The two-spool turbofans count with two concentric different shafts, rotating at different speeds, mainly because of Fan blades. A high rotational speed in the Fan would induce excessive centrifugal loads in the root of the blades. Those loads grow linearly with the distance to the rotational axis and with the square of the rotational speed, so the longer the blade and the higher the rotational speed, the more intense the efforts that parts will suffer, penalizing thus the life of the material.

Nowadays, the turbofans used for commercial aviation are mostly high-bypass engines (BPR > 5). New designs are leading to ultra-high-bypass engines (BPR > 10). In the engines considered for this thesis (high-bypass), the air entering through the inlet is compressed along different consecutive axial compressor stages until reaching high enough pressure values, and an adequate density, being then mixed with a liquid fuel (i.e., kerosene like JET-A1, JET-A, JP4, etc., see Chevron [55] for further details about these fuels) in the CC. The combustion process generates very hot and highly pressurized gasses, because of the exothermic reaction that takes place, that will be expanded in consecutive axial turbine stages until reaching a rear nozzle where the final expansion will take place providing part of the desired thrust. Only a part of the total thrust delivered will come from the primary flow of the engine because, in a turbofan engine, not all the air going through the inlet is heated in the CC. In fact, most of the air is slightly compressed by the big fan, being moderately accelerated afterwards in the exhaust of the bypass duct. The relatively low speed reached by the large amount of air moved in the Fan (yet greater than the flight speed) contributes both to improve the propulsive efficiency of the engine and to abate the high noise levels produced by the jets generated from the hot gases expanded in the main engine nozzle. Those exhaust gases reach high speeds and provoke elevated noise levels when mixing with the calmed atmospheric air (the exhaust stream tears the more calmed air around, generating vortexes that rotate at different frequencies, producing the intense noise perceived nearby these engines).

Compressor stages and turbine stages are mechanically coupled by means of rotating shafts and spools, as it is indicated in Figure 5, so the rotation produced in turbine stages by the hot pressurized gases in their way out to the nozzle is used to compress the air that gets into the CC. The Fan is also rotated this way (by the LPT, same as booster). The engine follows a Brayton cycle (see Chapter 3) where a continuous combustion process, once stabilized, is self-maintained in the CC.
From a fluid-dynamic perspective, having different spools allows to accommodate better the air going through the entire engine, in terms of pressure, temperature, speed and density, properties that are highly coupled by the laws of compressible flows. These properties are always changing in the Brayton cycle.

Turbines do not need of a high expansion ratio to drive the compressors, so the remanent pressure after transferring power to the compressors will be used to produce thrust. Nevertheless from a thermodynamic perspective, it is convenient to reach the highest temperature possible in the CC, and the highest rotational speed and total pressure in the Core of the engine to optimize the work developed per mass of fuel burnt in the combustor (this means improving the thermal efficiency of the cycle, which is easy to realize when representing the Brayton cycle of the turbofan in a Temperature-Entropy diagram, given the divergent shape of the isobar lines on it, as it is done in Chapter 3). The components in the Core Engine are prepared to hold such demanding aerothermodynamic conditions. For instance, the surfaces in CC and HPT are typically protected with Thermal Barrier Coatings (TBC) made with ceramic materials, and the use of super-alloys (mostly based on the use of Nickel and/or Cobalt) is extensive in these modules. On the other hand, such demanding conditions would be harmful for the components mechanically linked to the low-speed spool, like Fan, booster, or LPT. Therefore, this bi-spool configuration makes possible an operation more closely to the ideal operating conditions for each compressing or expanding stage of the engine.
The engines that will be analyzed and used as reference in this study are typical two-spool turbofan engines, used in civil transportation, such as the iconic CF6-80C2, the best-selling CFM56 (with all its different versions), the great GE90, or the brand new GE9X, which is its youngest and more powerful brother, just to give few well-known examples. There are some big turbofan engines that count with three rotating spools (e.g., Rolls-Royce Trent Series), but they are not so common, there are other strategies to manage the flow when the power range is high enough, such as variable geometry systems to accommodate properly the airflow in the engine, and one of the leading design trends today is trying to reduce complexity by incorporating new materials, with the expectation of cutting maintenance costs.

Regarding performance and degradation status, the turbofan engines’ health condition can be characterized by different health and quality (H&Q) parameters relative to their main components. First, mass flow capacity related parameters of different rotatory components, CC, and/or the applicable ones for the engine’s internal ducts, \( \Gamma_i \), which help to define the applicable effective cross-sectional areas for the internal gas flow. A total of 5 will be used to represent their variation in the most relevant modules. The other H&Q parameters will be efficiencies (both adiabatic and polytropic could be used, but the former ones, \( \eta_i \), were preferred) of the same different engine components (another 5 parameters, making a total of 10). In this sense, along the explored literature on the topic, mass flow capacities and adiabatic efficiencies are considered the paramount H&Q parameters for this kind of engines, representing their effective degradation status. So, an important number of parameters and variables will be involved in the determination of the engine’s performance level, and this means a considerable computational power will be needed to solve any associated problem in a reasonable amount of time. The aerothermodynamics of the engine model will be briefly commented in Chapter 3.

The hot flue gases generated by the combustion process are used in marine and industrial applications, where the propulsion is not typically a desired outcome, to generate steam, and the rotation of the gas turbine engines’ spools is typically used to produce electrical power by means of an electrical generator mechanically coupled to them. They are also used to move, for example, gas compressors or high-speed axial flow waterjets, like the ones used in ferries. Some more details about these engines will be provided as well, given its relevance nowadays.

1.3. The historic role of the turbofan engines

No other prime mover has seen such a rapid rate of commercialization as turbofan engines in aviation (see Smil, 2010, [269]). They clearly dominate the air transport worldwide in the speed range around Mach 0.8. Comparatively, reciprocating engines incur in much heavier weights at high speeds and they are very limited in performance at high altitudes, not to mention their associated maintenance costs.

Turboprops are more efficient at flight speeds around Mach 0.6, but their propulsive efficiency drops clearly at higher speeds because of compressibility effects in the tip of the propeller’s blades (Hill-Peterson, 1992, [133]). More than 4.5 billion people were transported every year, right before COVID-19, with turbofan-powered aircrafts. And the expectation is to recover similar levels by not later than 2024 (see related article in [40]). Table 3 provides an engine census, by commercial type series and Original Engine Manufacturer (OEM), published in May 2019:
Table 3: Engine census published by FlightGlobal (see more in [95]). By 2019 around 53,000 engines were into service, and about 20,000 had been formally ordered. Together with AECC, that will produce engines in China, there were 12 turbofan active manufacturers. There were more than 26,000 commercial airplanes in service powered by these engines, and the expectation in 2019 was to double that number by 2035 (Boeing, 2019 [39]).

The globalization process has been dynamized by the continuous improvement and expansion of these engines, making possible that a round-trip economy-class ticket for the busiest trans-Atlantic crossing, between London and New York, had dropped from above 2,000 USD in 1985 to 550 USD in 2019 (keeping constant 2019 money value). The latest generation of large turbofans that powers wide-body airliners on intercontinental flights (many lasting more than twelve hours) clearly represents the most complex and yet most reliable machines in widespread use. From a historic point of view, the rapid post-1960 expansion of intercontinental jet travels, driven by a growing middle class that could afford cheaper flight tickets, required powerful engines, and this demand was met by the first generation of turbofan engines. The wide-body aircraft, and the high-bypass turbofans that power them, have clearly popularized the transoceanic travels.
The first turbofan engines, commercialized and intensively employed for aviation in the 1970s, were already advanced and large, comparing with standard turbojets, but they were still too inefficient, noisy, and not powerful enough for some new types of airplanes with higher passengers’ capacities. An evolution of the fuel efficiency of the aircraft is shown in Figure 6 (IATA, 2019, [139]). The need for higher efficiencies became evident few years after these new engines went into commercial service. OPEC’s first round of oil-price rallies multiplied, between 1972 and 1974, by more than five times the price of oil. The next round of price increases continued until 1981. Higher efficiency and enhanced reliability in the engines became also critical given the sustained air traffic volume growth during the next decades. In this sense, some technical improvements, developed in the military industry, have been traditionally incorporated for the commercial aviation rapidly, like the different cooling systems for high-pressure turbine blades and nozzle vanes.

![Figure 6: Evolution of fuel efficiency in different aircraft models over time. Improvements are indicated using the pioneering Comet 4 aircraft as benchmark (100%).](image)

Some of the key variables for civilian aviation, like the revenue passenger kilometers (known as RPK) and the introduction of new routes (both regional and intercontinental), together with the volume of freight air transportation, have increased annually worldwide in the last 30 years (before COVID-19) despite the more expensive fuel and the higher cost of airplanes. In particular, the RPK had maintained a growth rate of a 5% in average since 1945, until 2020 (ICAO, June 2021, [141]). Simultaneously, new airplane designs have demanded more powerful engines, even when the increased volume of traffic led to the requirement for lower noise levels. Several studies indicate that just the Core Engine of any gas turbine model is about 20 times more powerful than a piston engine of the same size (Rolls-Royce, 2015, [244]). Figure 7 gives more information on the technological evolution of turbofan engines, particularly since 1970 when its use became generalized, throughout the improvement of their main performance parameters:
Figure 7: Historical evolution of main performance parameters. Thrust, specific fuel consumption (SFC), overall compression ratio (OPR), and turbine inlet temperature (TIT), respectively, showing a sustained improvement in these four performance parameters. Background figures obtained from D. Ballal and J. Zelina, 2003, [27].

As it will be shown in Chapter 3, the need for higher thrust levels that could afford the transportation of a maximum number of passengers per flight, was a constant during the last decades, and that is reflected in the first chart in Figure 7. The continuous search for higher efficiencies in the new engines that appeared since the production of the first turbofans was initiated results evident in the other three charts, where higher OPR and TIT levels went together with lower Specific Fuel Consumptions (SFC). And the trends in the charts will continue. Even when some of the values achieved by the GE90-115B seem to be very high, the new GE9X already overcame some of them (OPR of 60, for the first time in aviation’s history).

The sometimes-opposing targets, high power and noise abatement have been progressively reached by the constant technological improvement in the turbofans as well. Three trends have been unequivocal in the last decades: larger turbofans setting new thrust records have been developed; the overall efficiency, reliability, and ease of maintenance of commercial aviation engines have been steadily improved; and, finally, their noise generation has been abated. The applicable sets of noise and pollutants restrictions, as well as the recommendations established by the Environmental Authorities are reviewed periodically to make sure the best available technology is systematically applied. The following set of charts in Figure 8 shows these trends more clearly. It is particularly striking the improvement in reliability measured as the number of engine’s shutdowns during a flight, which was quickly reduced in several orders of magnitudes, and the higher durability of the engines. In this sense, some more examples will be given in Chapter 3.
Figure 8: Historical evolution in reliability (measured as In-Flight Shutdowns), engine’s life on wing (accounted in hours), noise abatement (measured in Effective Perceived Noise in decibels, as per Part 36 of the US FARs), and smoke number (exhaust smoke emissions measured as per SAE ARP 1179), respectively. Background figures obtained from D. Ballal and J. Zelina, 2003, [27].

The historic and economic impact of the turbofans is out of discussion, the continuous effort from the industry to improve their performance has been its distinctive mark, deserving to be studied. But, also from a social perspective, just to illustrate the degree of relevance of these machines in our society since their first appearance in 1941, a survey published by Bloomberg ([289], 2014), conducted by different experts in diverse fields, showed that no other invention of the past eighty-five years had a greater impact on the world. The list of candidate inventions and advancements included the microchip, the fast food, or the Internet, among them.

**1.4. A continuous search for efficiency improvements and cost reductions**

Regarding the operational life of these machines, as it happens with every mechanical system, the gas turbine engines are subject to a continuous deterioration while in operation, caused by different factors such as the erosion in the hot parts of the engine, the potential accumulation of dirt transported by the surrounding polluted air into the internal surfaces of the machine, or by the possible damages originated because of impacts if a solid body hits any internal part (i.e., ice, sand, etc.), just to mention few relevant ones. That progressive deterioration, sometimes suddenly increased by punctual events, implies a cost for the airlines, associated to the quantifiable maintenance to be done in the engine (typically by the OEM, or by any other approved service supplier) to restore its condition to the expected levels, in terms of performance and operational safety.
But that cost is not only limited to maintenance, repair, and overhaul (MRO), meaning the associated effort to keep or restore the mechanical integrity of the different components of the engine. That deterioration implies a twofold effect because of the continuous loss of efficiency associated, situation that can be translated into an increase in the amount of energy needed, or fuel, to generate the same propulsion level, or power output, that the engine had when it was delivered, new and clean, from the factory where it was manufactured, assembled, and tested.

Obviously, when a deteriorating event is severe enough, it could lead to a catastrophic failure in the entire machine. For example, a bird strike could imply the complete loss of the engine, not to mention the potential human cost associated. These accidental events will be left out of the scope of the present study given its random nature and the remarkable theoretical complexity they imply when analyzed in detail. Statistically speaking, they are not common, fortunately (Boeing, 2021, [39]). From an economic perspective, the cumulative effect of the small but continuous deterioration during the regular operation of the engine ends up being considerably higher, in the long term, when evaluating a whole fleet of engines.

The biggest MRO cost in an aircraft comes from the engines that power it (see IATA [137], 2019, or Seemann et al., 2011, [259]). The associated mechanical complexity, and the highly resistant materials used in their components, increase the cost of labor (see Figure 9) and replacement of worn parts comparing with the rest of aircraft components. Additionally, not all the engine parts suffer the same deterioration. The parts associated to the hot section of the Core Engine, given the hotter temperatures, higher pressures, and rotational speeds on it, will experiment a faster degradation and will typically need for more frequent repairs or parts replacements. This means different degradation levels to be managed in one engine.

![Figure 9: Different MRO actions performed by specialized personnel. The scheduled maintenance in the workshop constitutes one of the main milestones in the service life of one engine. Images from MTU, [205].](image)

The commercial aviation industry counts with a highly competitive nature and the prices of the flight tickets have decreased on average a 2% annually over the past 20 years (see related article in [253]). Oil price rallies, wars among countries, low-cost carriers, and several punctual events, like the terrorist attacks in New York City, on September 2001, are some historic causes that have affected negatively to the commercial aviation. Furthermore, the COVID-19 has provoked a dramatic and unexpected fall in the industry revenues of a 40% in 2020, compared with the previous year, with an overall reduction of a 60% in the number of passengers (ICAO [141]). Costs reduction has become therefore a matter of survival for the airlines.
New and better technologies, allowing for increasingly efficient operations, have led to higher flight frequencies, aircraft with higher capacities (in terms of both number of passengers and cargo volumes), and have helped to drive down the costs of running an airline in the last years. This trend is expected to continue in the future.

From a technical point of view, the need for an increase in efficiency in these engines, and for the reduction of maintenance costs associated to their use have motivated different upgrades, re-designs, and even conceptual revolutions that have occurred in the gas turbine industry during the last decades. An evident example is the continuous enlargement in the fan diameter, leading to higher bypass ratios (i.e., the GE9X counts with a BPR of 10, doubling the values in the early turbofans developed in the 1970s), to the resulting decrease in the Fan Pressure Ratio, and to the increment in the specific impulse (i.e., thrust per unit of air mass flow going into engine’s inlet, as indicated by Kerrebrock, 1992, [154]). Other relevant upgrades aligned with those targets are the use of additive manufacturing, the improvement of materials used in the hot section parts of the engines to increase TIT, such as the ceramic matrix composites (CMCs). Their modular design, aiming to facilitate the disassembly and re-assembly of the engine’s main components, is just another example. Finally, more strict environmental requirements, including noise limitations, and NO\textsubscript{X}/CO/CO\textsubscript{2} restrictions, have also contributed to modify engine designs, but not improving necessarily in terms of efficiency or maintenance costs.

In fact, same factors are conducting to similar changes today, even with a more ambitious approach. The recently confirmed launch, by CFM International, of the Open Rotor Fan engines program (RISE: Revolutionary Innovation for Sustainable Engines) to enter service around mid-2030s, which foresees a 20% reduction in fuel consumption and CO\textsubscript{2} emissions compared with current engines (see AWN, [212], for further details) is a clear evidence. The concept is not new, as a similar program was launched in the decade of 1980 (known as CROR: Contra-Rotating Open Rotor, also Unducted Fan or Propfan, program led jointly by GE and NASA), but it has been refined to a simpler and more integrable version. The Open Fan (see Figure 10) will increase the BPR by eliminating the fan duct and it will gain efficiency by reducing the Fan Pressure Ratio and eliminating the aerodynamic drag associated to the fan case. In the same RISE program, new combustor designs will be tested to ensure their compatibility with alternative fuels like HVO (Hydrotreated Vegetable Oils, known as renewable diesels, Neste [210]) and even with hydrogen.

![Figure 10: Open Fan concept and its integration in different locations of conventional aircraft. The images were obtained from the website of CFM International [54].](image-url)
To calculate the degree of deterioration of a gas turbine engine, it is necessary to count with valid analytic tools to compare the real performance level of the machine, at every representative operational condition, with the performance level that should have, according to the theory. In other words, the real machine will be confronted with its equivalent model by means of analytic tools that must be valid in a triple variant: regarding fidelity to the real system; in terms of accuracy in the results obtained; and, finally, valid in terms of required computational time as well.

Few comments are needed to emphasize the relevance of the fidelity and the accuracy in the methodology but, regarding computational time, an analytic tool that needs a time longer than a flight to deliver the outcome, no matter how accurate it may be, would not provide a clear improvement, basically because that task could be done today, with the available knowledge on the topic, just by post-processing the data retrieved from a flight once the aircraft is landed. The target now is getting results as fast as possible and, ideally, on real-time basis (or almost real-time). A possible definition for real-time applications will be provided later in the study but, by now, it could be taken as the time needed to avoid delays in a whole process.

The results from the analytic tool (engine model) will provide the health condition of the real system by means of data taken from the instrumentation, which monitors continuously the status of different engine components. From that cooperation between analytics and measured reality, different decisions could be made accordingly to minimize maintenance costs and to try to increase, as much as possible, the efficiency in the operation, given a certain degree of deterioration.

Here is where the motivation of this thesis appears more explicitly: This document explores several different techniques to provide valid analytic tools that could be used with gas turbine engines, and more specifically with turbofans, aiming to determine the engine’s degree of deterioration with the data provided by the sensors. Fidelity, accuracy, and speed of response are the three features considered as essential to have the capability to make a documented, sensible, and diligent decision considering operation and maintenance relative approaches.

Among the different techniques used in this study, as it will be detailed in Chapter 4, it can be mentioned the well-known genetic algorithms, the sequential quadratic programming, Newton-like techniques, or methods based in reduced order models. All these techniques make use of a theoretical model of the engine, previously configured, and validated, such as the one prepared specifically for this research by means of the software PROOSIS®.

In this sense, methods based on reduced order models (or ROMs, as they are named in the literature) have shown a high degree of versatility in several problems, from different technical fields, of immediate applicability. These simplified models allow to obtain results, still accurate enough to capture changes in the system under analysis, within reduced computational times, in practical problems characterized by a strong non-linear behavior of the involved variables, and the presence of many related parameters, which is precisely the situation when dealing with performance evaluations in gas turbine engines. In fact, one of the most attractive benefits of the methodology has to do with the resulting calculation effort, in terms of computational time, being typically shortened when comparing it against other usual techniques (one good example of the use of this methodology in different fields can be consulted in M. Bergmann et al., 2014, [34]).
The use of ROMs and some other techniques make possible a tradeoff between accuracy, speed, and scalability. This way, these methods mean a potential resource for solving problems with real-time requirements, such as decision support systems by exploring different alternatives. Furthermore, in certain complex problems, it could be totally impossible to meet real-time requirements following other standard numerical and mathematical methodologies. In addition, ROMs could be used to remove undesirable information from the problem, such as noise from instrumentation. All these features are of great interest for this study.

On the other hand, the development of reliable ROM-based methods for optimization, control, and forecasting purposes is not trivial. There is no guarantee that the ROMs will effectively model the system, as it should be, in the application under consideration, given the reduction of data that ROMs mean. For the starters, the reviewed literature on the subject indicates that some ROM-based models have shown numerical instabilities when dealing with unsteady models. Also, when the configuration of the problem changes there is no reason to think that the same algorithm will properly approximate the solution (Bergmann et al., 2014, [34]). Thus, some cautions must be taken when using this tool and some ad hoc corrections may be eventually needed by the model depending on the scenario, which would have to be analyzed on a case-by-case basis.

The model containing the physical laws, the one used to determine the regime, health condition, and even some potential control optimization alternative in a gas turbine engine, should be as much autonomous, self-contained, fast, and robust as possible. It must be fully trustworthy, and it should use as few external data as possible to get a valid solution. And here, the ROM-based methods applied instead the engine specific model (in the case of the proposed application in this thesis, used against the information coming from the engine monitoring system), could be also advantageously employed. In this sense, a ROM-based method could replace the complete theoretical model of the engine with a simplified version of it, if the appropriate hypotheses and operational conditions are met.

Given the increasing degree of sophistication of the new turbofan engine models produced, driven by the pursuit of better performance (meaning normally reaching higher efficiency levels), it is expected a future increase in the complexity of the models needed to virtually replicate these real systems, like the one schematically represented in Figure 11. That could negatively affect to the computational times required to obtain solutions with standard methodologies.

![Figure 11: Schematic diagram showing the way a performance model typically works to solve a direct problem, providing the main performance figures of the engine given flight conditions, status of the engine, and regime chosen by the operator/pilot.](image)
And this circumstance, linked to the current limited available computational capability on board, calls for the potential development of adequate model reduction strategies. Namely, it calls for more computationally affordable and efficient algorithms that still could accurately capture the most relevant features of the system under consideration (A. Quarteroni et al. [234], 2014). This way, ROMs would allow for the reduction of computational cost, for instance, in the calculation of the inverse problems similar to the one schematically shown in Figure 12, where it must be determined, among others, the effect that, on the health and quality parameters of the engine (typically efficiencies and mass flow capacities), have the change detected by diverse sensors installed in the engine (such as temperature and pressure probes, or rotational speeds, at certain engine stations). Working with tensors, instead of managing complex solvers, is a strategy that will be explored to find out if some improvement is possible in the determination of the health condition of a turbofan. Certainly, this will be one of the main targets of the thesis.

![Figure 12: Solving scheme for the inverse problem where the inputs are, among others, the measurements taken from sensors. The health and quality parameters, together with the engine’s real regime, are finally obtained. The proposed methodology could be embedded in the engine’s control system, working finally in a similar way.](image)

The idea behind the use of ROM-based methods is selecting the right information of the problem, the one that is really needed to get the required results. The whole problem will not be solved, only a portion of it, but without losing too much accuracy in the process, and trying to be faster in contrast. As far as the author of this thesis is concerned, no ROM-based methodology has been used so far with this kind of turbofan engines the way is described in this work. This is where part of the novelty of this study resides, as it will be detailed. A good example of the potential of ROM-based methods, applied to thermal engines, can be found in [33] where N. Benito et al. (2011), inside a team formed by ETSIAE-UPM researchers, used this method in a Spark-Ignited (SI) engine. In [33], the information from the full model of the engine, programmed into a simulator, was truncated with a Higher Order Singular Value Decomposition (HOSVD), leading to a reduced model that could obtain the required output in a fast and sufficiently accurate way for engineering purposes. A SI engine typically counts with less sensors (so less variables will be implicated in the calculations as well) than a gas turbine engine, thus the use of ROMs could be even more justified now. A detailed summary of the explored literature on the topic is available in Chapter 2.
1.6. Where the industry currently is

It seems that the current preference in the industry for data mining and data-based methods, dealing with big data sets, responds to the perception that complex systems will clearly follow patterns that can be somehow easily identified, isolated, and understood without the need of a sophisticated model behind, built by many relationships between potentially many variables. It is true that, in several technical problems, predictions obtained by data-based methods can be really accurate. In fact, big technological companies (e.g., GE Digital [109], Siemens Power Diagnostics, P&W Digital Engine Services, etc.) have invested in these methods, at the light of such good results, for instance when optimizing combustion processes using Neural Networks. The problem with these data-based methods resides in the need of a large amount of previous information to get trained, so it can be incorporated the past abundant available information to the next estimation process. That makes the development of real-time procedures difficult, because the required training phase needs of an important amount of computational power to manage the massive quantity of information involved. Optimizing the combustion process in a gas turbine engine could imply several weeks of previous development for the required Neural Networks, in standard industrial dedicated computers, before they could be used confidently in a real machine given the complexity of the process.

Filtering would be certainly a potentially valid strategy to try to reduce the amount of information that will be used by a data-based model. Still, some intelligence would have to be implemented in the filtering process, thus implicating more computational time required by the process and potentially introducing some bias in the data selection process. In addition, absolutely all the available values would have to pass through that filtering algorithm and, inevitably, there will be always a measurable risk of ignoring relevant data.

Model-based methods, like the ones used in the present study, will be surely more agile instead and will be fitted to the application under analysis because they do not need, in theory, of a previous long training to provide accurate results. These methods do not need to manage big data sets in advance to produce results once some characteristic pattern is found with enough confidence margin, statistically speaking. There is no need to find and extract any pattern from data if the specific physical laws and theoretical models that apply to the system under consideration are already known. And, fortunately, this is the case with aircraft engines. This advantage makes them more adequate for on wing applications, given the knowledge already stored on the topic during the past decades.

Nevertheless, every methodology implies a cost, and having a truly accurate model of a particular complex physical system, such as an aircraft engine, is not an immediate task. Every OEM counts with detailed and sophisticated models, in which certain methodologies and engine-specific tools are applied (e.g., usage of specific component performance maps, advanced gas path analysis, etc.). Determining accurately components’ maps and models implies a long and rigorous theoretical and empirical process, with multiple experts involved from different disciplines, and eventually hundreds of costly testing hours in test cells before satisfactory results could be obtained. OEMs’ engine models are eventually confronted and validated with data obtained from test cells and flight tests for calibrating purposes. The knowledge of that information has a strategic value for each OEM, commercially speaking, and that is why so scarce relative data is publicly available.
And this means, even when a considerably minor quantity of data is required compared with data-based models, it is still necessary to gather some minimum amount of information from the real operation of the engine to get a representative model of it, especially when dealing with the aging process of the machine because its health and quality parameters will vary along the time. In other words, a model calibration must be done, by knowing the current condition of the system, before initiating the calculations. Even when the use of external data will be always avoided, this intention is most of times futile, as some external information to the theoretical model will be finally needed. That leads to a mixture of data-based and model-based methods, but much more polarized to the last type.

The different OEMs advertise in their respective websites, that the experience from the fleet is being incorporated to each and every of their new engines for their customers’ advantage (GE Aviation, in [109], mentions more than 46,000 years of accumulated flight data, counting all the different models produced so far by the company). Obviously, storing a large amount of information and experience does not necessarily mean it will be successfully used to manage an existing engine. Not even a fraction of it would mean it. Computational costs and required data storage will be certainly excessive for a standard engine control system unless some pre-processing is done. A modern standard desktop computer carrying an Intel-Core™ I7 processor (10th generation, as per indicated in [146]) is capable to process logical orders at a ratio of less than 5 GHz, this is less than 5 billions of cycles per second, and not continuously, only during peak periods. This computational capability could seem sufficient, but the reality is modern Control Systems (CS) managing gas turbine engines count with less powerful processors, yet more reliable, and must deal with a large amount of information, on real-time basis, just for the operation of the engine and for the diagnostic of any problem in the machine during its functioning. A Full Authority Digital Engine Control (FADEC, namely the full digital CS of a modern aircraft engine) must perform, in addition (and primarily), the rest of critical logic operations, such as the fuel control inside safety limits, variable geometry laws’ adjustments, monitoring, diagnostics, etc.

The tensor of data generated during the preparation of this thesis (as it will be explained in Chapter 4), counted with 12 dimensions, and contained tens of millions of values that had to be processed to get the solution of an inverse problem, feeding the algorithm with sensor readings (real performance), to obtain the value of the current health parameters (real degradation). The real-time capabilities in the available hardware are not so evident. And this is just to get a solution for the inverse problem based on the equations of the thermodynamic cycle of a two-spool turbofan, aiming to determine the health condition of the engine under study, given the information from the instrumentation, and a set of specific flight conditions. The CS will have to do many more operations on real time. Obtaining a quick solution for the inverse problem mentioned above will mean a further exploration into alternative strategies, less demanding from computational power perspective. Techniques like the ones based on the use of ROMs could be, in this sense, well positioned to succeed, if they can liberate the CPU of the CS of some workload, which is one point still to be clarified. As can be seen, the study implies some considerations on CS hardware capabilities (and, potentially, on airworthiness if the proposed methodology is implemented finally in commercial aviation).

In regards with the information generated by the engine’s CS, it is necessary to highlight that the modern commercial turbofan engines count with a considerable
instrumentation distributed along the engine (this characteristic of the turbofans will be illustrated as well in Chapter 3). Those sets of sensors generate a great amount of information, however not all of it, obtained through the readings continuously retrieved from the machine, is equally valuable or useful. The instrumentation and monitoring systems installed in the engines log the condition of the whole machine during its operation, but this could perfectly mean a worthless knowledge in terms of performance, diagnostics, prognostics, or control if they do not contribute properly to explain what happened to the engine.

Today, the CS managing commercial turbofans provide mainly a permanent support to the operator to effectively handle the machine while performing a diagnostic analysis, which is identifying technical problems and notifying them to operators, based on data supplied by sensors. This help is essential to understand if the engine is working inside safe limits. However, in recent years, the concept of prognostic analysis, namely the process of predicting, inside certain margins of confidence, the future state of the engine based on the current condition, has drawn also great attention, because it is directly related to the risk analysis that airlines develop for their engine fleet. As it is exposed in [186], by Amin et al. (2014), a good risk assessment system will contribute to prevent system failures and its implementation will have a clear impact in reparation costs caused by unexpected issues that may occur during the operation of the engine. And this is, obviously, applicable not only to aviation. An intelligent prognostic system could execute on real-time basis a good part of that risk assessment.

Prognostics do aim to forecast system’s condition by measuring its current state, considering potential future workloads and operational environments (scenarios). Also predicting, through simulation and by using the accumulated experience, the remaining useful life (RUL) of the system (see [91], by Farrar et al., 2005, for more details). Prognostics is a field where the accumulated knowledge from the fleet clearly shows its relevance, and it highlights the convenience of reduction model strategies. Some of the economic effects that an embedded prognostics system could mean, from a maintenance point of view (and this is valid for both civil and military sectors), were already glimpsed by Greitzer et al. (2001) in [126], focusing on military applications, and could be summarized as follows:

- Reduction in the maintenance workload, especially if unnecessary revisions can be avoided because of a better understanding of the engine condition along operational time.
- Minimization of reparation times, as big damages could be avoided in a timely manner, inside realistic confidence margins.
- Reduction of life-cycle costs, by optimizing stocks and logistics.

The new information that prognostics can supply, of a different nature from the one that diagnostics provide, will be of paramount importance for every OEM and their potential customers in the years to come. Making those prognostics faster and more accurate, considering different possible scenarios, will redound in a quicker decision process during engine operation. Managing detected damages on real-time basis, optimizing maintenance, reducing fleet costs, increasing return on investments and, eventually, preventing catastrophic accidents in aviation, marine, or industrial applications are some of the potential benefits that this promising set of techniques could mean. The targets of this thesis will also include the potential
development of strategies to estimate the evolution of RUL in one engine with the minimum information possible. Current trends show that the theoretical models of the engines will have to produce a higher amount of information in the future to support these new appealing features.

In this sense, the logic implemented in the CS of future gas turbine engines could be redesigned to work like it is schematically depicted in Figure 13, aiming to provide results in terms of not only diagnostics, but also in terms of maintenance, performance, and operational optimization. Such kind of logic would be compatible with the proposed methodology in this thesis. That logic, and the algorithms embedded on it, could be improved in the future by incorporating information from the fleet, or by using any of the techniques explored in this study, such as ROMs.

![Figure 13: Schematic showing the potential implementation of a real-time model-based method in the logic of the CS of an engine to provide decision criteria over diagnostics, prognostics, and even control.](image)

Regarding the instrumentation itself, a direct source of information about engine’s condition comes from sensors relative to pressure and temperature at the different engine’s gas path stations. They are normally located in between engine’s main modules. The aircraft engines count with probes to monitor the gas path typically the inlet and the outlet of compressors, combustor, and turbines. Some other relevant sensors take care of the rotational speed in the different spools, the vibration levels at the casings of those main components and last, but not least, the temperature of the lubricating oil, which is directly related with bearings’ condition (ball and roller type bearings are mounted in this kind of engines).

Pressure and temperature sensors, like the ones showed in Figure 14, together with rotational speed probes and fuel flowmeters, are the ones that provide today a better picture of the gas path. Currently, readings sent to the engine’s CS are filtered, and pre-processed, to detect faults regarding any potential issue affecting
to the instrumentation’s integrity. To find a fault in one engine, or a potential risk for its safe operation, the information must be analyzed in a fast and reliable way to make decisions during operation. ROM-based methods (or any other model-based technique explored in this study) could help with this task, by performing a fast confrontation of the readings taken by the sensors with the expected values that the theoretical model, programmed in the engine’s CS, will provide (acting as a “third sensor”, in the usual nomenclature, given the typical dual physical redundancy of sensors in gas turbine engines), but streamlined by the ROMs. There are multiple possibilities once the engine model is available and validated. However, the main target in this study will be the determination of the H&Q parameters and TIT (i.e., engine’s health condition, assuming instrumentation is in good shape).

A conventional commercial aircraft itself currently produces between 3 and 10 megabytes (MB) of information per flight-hour (AWN, 2016, [263]). The engines installed on the same aircraft generate a larger amount of data, typically about 25 MB per completed flight hour. That amount of information is continuously increasing, year after year, because the new engines are equipped with more complete instrumentation and with more powerful monitoring and control systems.

This means a growing amount of data to be managed, not only for each machine, but especially at a fleet level (scale factor). Similar situations are experienced in other sectors related to energy production or marine transportation.
Nevertheless, not all registered and processed information will be equally relevant when trying to anticipate a solution for certain problems. In fact, some of that information could be misleading or erroneous, if there is a miscalibration in the instrumentation.

Adding to this circumstance the fact that transmitting 1 MB of information through the traditional Aircraft Communications Addressing and Reporting System (ACARS) costs about 100 USD (see [263] in this sense, as some companies are currently working to reduce such costs), it can be inferred that it would be very convenient to develop a method able to use only information that is relevant to the resolution of the main problems associated with operation, diagnosis, and prognosis of the engine, quickly and efficiently, from the CS itself, without requiring (initially, at least) of any connectivity whatsoever. A method that could manage data this way, and on a real-time basis, would certainly optimize the available computational resources, avoiding a massive and expensive handling of information.

1.7. The path ahead

So maybe the principal question to be answered in this thesis is if proposed techniques like the adapted Newton or the ROMs could be successfully applied to modern two-spool turbofan engines and, by extension, to land and marine gas turbine engines, or any other kind of modern jet engine, for its performance evaluation, diagnostic, prognostic, and eventually for its control optimization if possible, when the number of relevant variables and parameters is higher than in the case of reciprocating SI engines, like the one studied in [33].

To answer this question, it is necessary to deepen in the problem, evaluating several factors that may have remained hidden until this point. Just to mention one, the techniques studied in this thesis, could imply calculations considering information from several different flight moments (even from different flight stages) during a standard commercial flight, like the one schematically indicated in Figure 15. When not enough information is retrieved from one sample of data gathered in a specific moment, or if the instrumentation is not optimized to calculate all the H&Q parameters and TIT, then several samples taken at different times could be needed, circumstance that could hinder the application of the methodology.

Figure 15: Schematic example of a typical commercial flight profile, showing its different stages (flight phases are not scaled, just indicated, as cruise stages are typically the longest ones in commercial routes).
This kind of circumstances contribute to complicate the problem because it would be necessary to evaluate if the results obtained in different flight conditions where the characteristics of power lever (namely the regulation of engine’s regime by adjusting TIT), speed, and ambient conditions play an important role, are compatible. Considering the H&Q parameters constant in between two consecutive flight stages is a condition that could be assumed, but such hypothesis could mean an abuse to the model depending on the engine and flight conditions, leading to unrealistic results. Some research was required to understand how quickly this kind of engines would degrade under normal operational conditions. In these situations, it is where the complexity of the study about sophisticated machines like the two-spool turbofans, in which several interrelated parameters are involved, is more palpable. However, here is also where the multiple applications of analytical tools, such as the adapted Newton or the ROMs, as well as the rest of techniques that will be detailed in the thesis, could be more helpful. In these cases, the available knowledge and expertise on the topic could contribute to find a solution as well. Fortunately, the gas turbine engines constitute a mature technology, in continuous development though, with a wide literature to explore the experience shared by other researchers in this field.

Just to finalize this section, it is worthy thinking about the relevant issue for engine operators that could arise in the future when managing big sets of data (bigger every day), needing for data-driven models training, typically accurate but not agile, during long periods of time, while dedicating a great computational power, to finally decide, based on the obtained results, about the maintenance of an engine and its future operation. Maybe during an emergency. This author thinks it would be beneficial counting with model-based methods to provide real-time (or almost) results, avoiding bulky data sets for diagnostic, prognostic, and regime determination in gas turbine engines. The aim of this study is exploring the available experience relative to the evaluation of two-spool turbofan engines’ health condition (and gas turbine engines in general), linking that experience to the use of new methods, keeping in mind the goal of improving results found in the literature.

So, after having seen the technical panorama, after having enumerated the targets, and after having commented on the novelty of the study, it is time to deepen in the different aspects of the problem. The rest of the work is organized as follows:

- A detailed summary of the vast literature consulted for the development of this study is provided in the second chapter.
- Engines usual configuration, the current instrumentation mounted on them, and the performance model utilized will be treated in the third chapter.
- The different techniques that have been used to evaluate the degradation of a two-spool turbofan engine are described in the fourth chapter.
- Completing the previous chapter, the fifth one gathers the most significant results obtained with each technique and method.
- Applicability and future work are discussed in the final chapter.
- Right after, several appendixes provide a simplified introduction to some of the secondary mathematical concepts commented along the thesis, to help on the understanding of certain parts of the document.
- Finally, the bibliography at the end lists the references used in the different chapters of this work.
2.- REVISION OF THE STATE OF THE KNOWLEDGE

Before detailing the techniques used in this study, as well as their validity to solve non-linear inverse problems relative to two-spool turbofan engines’ performance, it is convenient to summarize the consulted references from the existing knowledge in this field. Different methodologies were investigated along the thesis, and valuable previous work from different authors was reviewed to select the best approach possible. In this chapter, reader will find an overall description of the explored literature on Engine Health Management (EHM), diagnostics, and prognostics applied to different kinds of gas turbine engines. The following list of papers and works has ended up being, more than just a simple reference during the whole study, a lighthouse showing the route to navigate to a safe harbor. The consulted authors provided an important set of results and conclusions that have been determinant for the line of investigation finally followed in this thesis.

2.1.- Gas Path Analysis: Origin, concept, versions, and applicability

From a historical point of view, the gas turbine’s gas path analysis (GPA) is a veteran well-known methodology that his pioneer, L. A. Urban, developed in 1967. He continued popularizing it along the decade of 1970 with consecutive papers detailing the potential relevance of this technique. For instance, in 1973, this author used the GPA method to obtain the health condition of a Pratt & Whitney's FT4 gas generator (see [287]). The purpose of the GPA has been, since its first apparition, to detect, isolate, and evaluate gas path components’ faults that have observable effects on measurable variables, in a cost-effective way. The presence of a problem in the gas path of a gas turbine can be detected because it induces a change, or changes, in some components’ characteristics, such as efficiencies or mass flow capacities, and that provokes a deviation of measurable variables when comparing them with their respective theoretical values at certain baseline conditions. With the equations’ system that defines the behavior of a gas turbine or, in other words, with its performance model, the GPA detects faults when a discrepancy is found from the expected performance results, obtained from the model at certain ambient and operational conditions. This will be one of the main targets as well in this study.

Urban obtained in [287] a degree of success of 88% when detecting artificially implanted malfunctions and degraded parts when combining both GPA and vibration analysis for a more integral approach. The reason for this data fusion was trying to gain some effectiveness, as different partial indications could mistakenly lead to the same fault. The author highlighted that any change in one health or quality parameter (H&Q) of one specific component of the engine it is not necessarily an indicative of a specific fault in that very same component. The most likely situation will be that multiple deviations, detected in different parts of the engine, will lead to a single distinctive fault indication. Therefore, having as much knowledge of the whole engine being analyzed as technically feasible will improve the quality of the results obtained by GPA. And that is also the reason why relying only on usual instrumentation such as pressure and temperature sensors would not be typically sufficient, in certain cases, to identify the root cause of a fault.
Urban clearly stated that determining valid baselines (at different flight conditions) to compare results with is essential, as well as fixing criteria to determine when a fault has occurred (i.e., a list of fault thresholds).

As this GPA version only considered variations from those baselines, a wrong reference would certainly lead to a bad diagnostic. Inaccurate thresholds would have a similar effect. On the other hand, working with variations, and not with absolute values as Urban did, makes data accuracy from sensors not so critical. What is important then is the level of repeatability reached by the instrumentation on board. From a numerical standpoint, Urban works with linearized models which applicability is limited to very particular operational conditions, circumstance perfectly understandable in the decade of 1970, given the limited available computational capabilities by that time. In fact, the progress with the GPA was abandoned for years, with no further improvements, until the computers were powerful enough to afford the complexity required to describe the condition of a gas turbine engine. Neglecting small uncertainties in the formal mathematical approach to the problem, an engine operational point can be formally expressed this way:

\[ \bar{Y} = h(\bar{X}) \rightarrow \bar{X} = h^{-1}(\bar{Y}) \]  

(2.1)

Where \( \bar{Y} \) represents the condition vector (i.e., sensors’ readings), \( \bar{X} \) is the components’ parameters vector (i.e., the set of H&Q parameters) and, finally, \( h \) is a vector-valued function representing the expected behavior of the engine. The value of \( \bar{Y} \), given \( \bar{X} \), is obtained by solving the so-called direct problem, meanwhile the calculation of \( \bar{X} \) from the available sensor readings in \( \bar{Y} \) is known as solving the associated engine’s inverse problem. This last problem is the most interesting one for this thesis. The equations’ system that constitutes the function \( h \) is provided by the performance model of the engine, and it will include typically components’ maps and non-linear algebraic expressions, making the accurate determination of \( \bar{X} \) a complex task, numerically speaking. The success applying the GPA methodology depends greatly on the properties of this equations’ system.

In theory, the components of the condition vector should coincide with the actual measurements taken by the instrumentation on board. That will not normally happen in the practice because of different reasons, to be determined, not contemplated in the model of the engine. This circumstance will naturally lead to the reasons behind the discrepancy and, therefore, to the identification of a potential fault, or faults, that would be affecting to the machine.

The equations’ system of the non-linear problem can be expanded in Taylor series and linearized by neglecting higher order terms (HOT) when the expected changes in the engine’s sensors, represented now by \( \bar{Y} \), are small. If only the linear terms of the expansion are kept, the resulting method is the so-called Linearized GPA (LGPA, the one introduced by Urban). Then the previous complete system can be re-written, once linearized, by using the appropriate matrix approach as follows:

\[ \bar{Y} = H \cdot \bar{X} \rightarrow \bar{X} \cong H^{-1} \cdot \bar{Y} \]  

(2.2)

Matrix \( H \) will represent the influence coefficient matrix (ICM) of the system and its inverse will be the fault coefficient matrix (FCM), as they are usually known in the literature. The condition vector obtained with \( H \) will be a linear estimation of the real value. ICM inversion to provide FCM is dependent on the dimension of the
condition vector, which initially must be higher or equal to the dimension of the components’ parameters vector, and this condition is essential because the LGPA method relies on the invertibility of $H$ (i.e., in its rank). This formal requirement would imply an adequate use of the available information by conveniently getting inverse matrices when dimensions do not match, as it will be explained later.

LGPA is a useful methodology, widely exploited in the literature, when analyzing the health condition of a gas turbine engine that works exposed to usual deteriorating factors but, as it happens with every linearization method, it counts with a severe limitation when the order of magnitude of the error introduced by the hypothesis of linear behavior is like the error of the faults being analyzed. Thus, only relatively small effects can be adequately analyzed by LGPA (low percentual variations in the H&Q parameters). Urban applied typically just one iteration when solving the system associated to the inverse problem, which certainly will expedite the solving process, but will count with accuracy limitations.

The overlap in between different modes of degradation, or the existence of remarkable differences in between small and large deviations, are factors that may lead to mispredictions (for further details on the application of GPA, see Kong et al., 2014 [160], Stamatis et al., 1988 [271], and Ogaji et al., 2002 [217], in this sense). Additionally, LGPA are very sensitive to uncertainties in readings from sensors that could be caused by noise, bias, or lack of calibration in the instrumentation channels.

For many years, the limited available computational power made the use of fully non-linear GPA (making use of iterative Newton-based methods) impractical, given its numerical complexity. As it has been previously indicated, solving the full inverse problem means obtaining the H&Q parameters of the engine (i.e., efficiencies and flow capacities), by using not only the measurements obtained from the instrumentation, but also using detailed information from the engine design, such as the turbomachinery maps of the specific engine under consideration. These maps contain the information on the actual performance expected in the engine’s components at different flight or operational conditions. The complexity introduced using maps, of a clearly non-linear nature, made the linearization of the engine’s aerothermodynamic model, in the surroundings of the nominal point, the main strategy in the early works developed by Urban to solve the inverse problem. Many other authors have followed the same direction since then. However, in this thesis, the full non-linear model will be initially retained, avoiding its linearization. This is a more convenient approach today when comparing with other past studies.

A similar dilemma is exposed in the works by Stamatis et al. There are several similarities between the adaptive modeling developed by these researchers and the determination of the health condition of a gas turbine. Once more again, the computational cost associated with the method detailed in [272] by the authors was unaffordable at the time the paper was released (1990), so the same team suggested a pair of years later, in [274], the alternative strategy of the linearization expanding in Taylor series. In this new paper, the use of Singular Value Decomposition (SVD) was proposed to accelerate the calculations, technique that has shown a clear validity in the treatment of signals from sensors, among other advantages. SVD was also used in this research, as it will be explained in Chapter 4.

The different versions of GPA have led to different computational timescales, depending on the slower (non-linear) or faster (linear) technique used to solve the inverse problem that provides the engine’s H&Q parameters (leading to the engine diagnostics), with higher and lower accuracy, respectively.
When the target is performing not only diagnostics but also prognostics, a more complete GPA version should be prepared to store meaningful data for Remaining Useful Life (RUL) forecasts. RUL estimations are based on the trends observed in the signals coming from the instrumentation on board and they have found a clear validity in Condition Based Maintenance (CBM), as it was highlighted by Lei et al. in [166] (2018) and Li et al. in [168] (2009), where the goal was avoiding unnecessary inspections, increasing thus the ROI of the machines under study.

In these two works there was a clear exposition of the calculations behind RUL estimations. When the engine does not suffer sudden faults or failures, the degradation of its components will follow a slightly increasing rate (which is considered as a “healthy” or normal condition, ideal for LGPA) until a point in which the degradation rate will grow exponentially, reaching then quickly an unhealthy or faulty condition, if no maintenance is performed beforehand, leading even to catastrophic failures if no decision whatsoever is made to avoid it. That is the common experience when dealing with gas turbine engines, so the maintenance schedules are organized accordingly to prevent that last rapid deterioration phase.

**Figure 16** shows the typical degradation rate change in a gas turbine engine. Initially, the progression is relatively flat, being the assumption relative to the application of the LGPA valid. Since the operation is initiated, a continuous deterioration is reducing the performance capabilities of the engine. A precursor of performance degradation would be typically detected, as a first sign of abnormal condition. During the operating life of the asset, several anomalies could be detected as well, not necessarily meaning a technical problem. The presence of anomalies will depend on the capabilities of the monitoring system available. The precursor is for sure an indication of the sensitivity of detection of that monitoring system, and the sensitivity level will depend on the instrumentation and modeling capabilities.

As the degradation progresses, the deterioration rate will increase (LGPA not valid then), and a condition will be reached in which criteria are met to flag an alarm for the operator about the situation. This condition would be called a “fault”, and the faults are always associated with a technical root cause (the reason behind a potential failure that should be identified). Faults are less severe than a failure, and the engine could be working with one or more faults present during operation, but they are indications showing the engine is working with some sort of issue (Jaw and Mattingly, 2009, [150]). With this explanation the relationship between diagnostics and prognostics becomes more evident. And the GPA could contribute to both.

![Figure 16: Typical generic degradation path diagram (from [150]).](image-url)
The problem is a fault could develop into a failure, in a relatively short time, reaching then the end of service life for the engine. Every OEM will recommend stopping the operation of the engine before that moment comes. The derivative of the performance function observed to monitor the degradation of the engine with the time (typically its trend) increases continuously, and this circumstance could be used to establish a criterion (additional threshold) to decide when the engine must go to the workshop. The evaluation of the RUL in one engine is made based on this evolution, so some previous information is required to complete the prognostic.

In this sense, solving the inverse problem with a GPA-based method and storing the health condition parameters of the engine after each flight could potentially be advantageous when estimating the RUL because stored values of H&Q parameters are, in principle, independent of the operating condition of the engine, and just representative of its current health status. Translating the reading from the instrumentation into H&Q parameters, by means of GPA, would eventually allow to maintenance modelers knowing how much operational life can be expected in the engine when operated at certain conditions. The more complete is the GPA version used, the more accurate and trustable the estimations will be.

As it was indicated by Tumer and Bajwa in [286] (1999) these calculations could be performed in between flights, while refueling in the airport, and therefore this procedure will not be especially demanding in terms of computational time, and certainly, this circumstance was taken into consideration for this study. The time between flights could be useful to finalize calculations that may be pending, or to produce preliminary results required before the next flight. Accuracy and reliability in the results are obviously of the highest interest, particularly in aviation, and multiple sets of data will be available for EHM computations, as several snapshots could be logged during a flight for post-processing purposes, so they could be used to validate results with several calculations at different moments, if needed. In this sense, there are interesting initiatives regarding connectivity improvements for EHM continuous analytics during a flight (See AWN, 2016, [263]). These new technologies will certainly contribute for real-time EHM enhancements in commercial aviation making possible, for instance, redundant calculations and double checks of the diagnostics and prognostics obtained by the CS on board. However, ideally the target of this study would be reaching enough real-time capabilities on board with the methodologies that will be detailed in Chapter 4. If the real-time feature is essential, then computations must be completed on board (without post-processing once the aircraft has landed). Another target when solving the inverse problem would be knowing the regime in which the gas turbine is being operated. GPA-based methods should be able to work out both degradation state and engine theoretical operating TIT from just data coming from sensors (as much autonomously as possible). TIT obtention is not typically chased in the literature.

Dealing with degradations that exceed the linearization criteria typically appropriate for LGPA methods, as mentioned before, it is necessary to use Non-Linear GPA approaches. GPA was initially conceived for small degradations, performing just one iteration. The non-linear deviations in the component parameter vector, \( \Delta \bar{X} \), can be iteratively estimated as follows:

\[
\Delta \bar{X} \approx H^{-1} \cdot \Delta \bar{Y} \quad (2.3)
\]

\[
\bar{X}_{k+1} = \bar{X}_k + \Delta \bar{X} \quad (2.4)
\]
Then, an iterative solver based on the Newton method, could be used to obtain both degradation and TIT, until reaching convergence with one additional great advantage: With a Newton-based method, the changes in $\tilde{X}$ do not have to be necessarily small. $H$ is a Jacobian matrix that will have to be updated in each iteration. This method was deeply explored for this study as it will be explained in Chapter 4. Convergence in Newton-like methods will be reached when the following criterium (there are more different criteria that can be used instead) is met:

$$\sum_{j=1}^{M} |\tilde{Y}(j)_{\text{meas}} - \tilde{Y}(j)_{\text{calc}}| \leq \varepsilon$$  \hspace{1cm} (2.5)

Here, $M$ is the number of available measurements, and $\varepsilon$ the acceptable error allowed in between measured and calculated parameters for the condition vector. The smaller the value of $\varepsilon$, the higher the accuracy degree achieved but also more time consuming the solving process will be. The selection of the acceptable error when working with PROOSIS® will be described with more detailed in Chapter 4.

A typical assumption in GPA-based methods is that measurements taken are noise free, or previously treated, which is not always a realistic approach, given the nature of the signals from the instrumentation in a real engine. However, GPA will be one of the fundamental pillars of the following chapters and the starting point to evaluate the H&Q values from the instrumentation of a particular engine.

Regarding [217], which is one of the most relevant works consulted for this thesis, it must be highlighted that Ogaji et al. (2002) indicated that it is necessary to count with specific sensor readings to assess distinctive failures in one aeroengine. According to these authors, the principal issues affecting the gas path engine related components and the performance in gas turbine engines are the following:

- **Fouling**: Caused by the presence of dirt in the internal surfaces of the turbomachinery because of the accumulation of deposits over blades and vanes, modifying their geometry, changing angles of attack, narrowing airfoil-throat apertures and, as a result, decreasing the mass flow through the components that compound the engine's gas path.
- **Tip clearance**: Meaning the physical separation between rotor blade tips and casings in rotatory parts of the engine, causing even a bigger impact in efficiency than fouling, which can be accentuated because of hard landings and casing and shaft distortion, among other reasons.
- **Erosion**: Caused by abrasive removal of material from the gas path components by hard particles suspended in the air, affecting more intensively to rear stages of compressors than to the initial ones, given the much higher pressure there. Caused by losses of thermal coating as well.
- **Corrosion**: Induced by chemical reactions between gas path components and pollutants that are ported by the air, fuel or water ingested by the engine, eventually causing material losses, and being more common at the end of the hot section of the engine, such as the HPT (hot corrosion).
- **Foreign/Domestic Object Damages (FODs/DODs)**: Which is the result of the hitting of a body, with an external or internal origin, respectively, against the internal parts of the gas path components, resulting sometimes in an identical effect, in terms of performance, to that of fouling and leading potentially to a safety issue if the damage evolves unfavorably.
Some of these issues are considered typically to be part of the aging process in the engines, especially when dealing with the expected deterioration in the hot section parts of the Core Engine. That deterioration in hot section parts includes effects like thermal coating material losses, increase of tip clearance, etc., and it is true that these issues can be detected (or intuited) from the model, but there will be several cases in which other techniques will have to be employed, beyond GPA.

For instance, cracks, sudden punctual damages, or local cooling holes blockage, will not be typically foreseen by the GPA-based methods. Most of faults in the gas path usually have different and distinctive signatures, but some faults may have identical signatures and would require other engine-monitoring techniques, such as modern vibration analyses or debris analysis, to identify them, being this aligned with some of Urban’s conclusions. The CC was not included in the paper because its deterioration does not mean relevant variations in the engine’s overall performance. Furthermore, it was indicated in the paper that Power Turbines (PT) or LPT normally suffer issues that could not be easily analyzed with the GPA methodology employed in their study and, obviously, this could mean an important limitation to consider when using GPA-based methods.

The authors provided in [217] some estimations on the impact that the previous types of engine deteriorations could mean to the overall decrease in system’s efficiency, but they did not mention some other potential physical faults that could appear in the gas path, such as:

- Air leakages throughout the compressor casings split lines.
- Leakages because of damaged air seals.
- Plugged fuel nozzles.
- Burned or warped turbine stator vanes or rotor blades.
- Non-calibrated variable geometry systems, such as bleed valves.

The document finally mentioned the impact in maintenance hours that phenomena such as creep or thermal fatigue will mean in the gas turbine engines under consideration in their study. A higher power output demand implies a higher temperature in the combustor and a shorter life of the engine (typically, an increase of $\Delta T \approx 15^\circ C$ equals to a twofold decrease in engine’s life). With adequate data gathered from the machine, it is possible to improve the maintenance plan, increasing this way not only the reliability, but the availability of the system as well.

In this sense, Ogaji et al. remarked that installing just enough sensors, optimally selected, and wisely located, would be as effective as over-instrumenting the engine, which would be a more expensive and complex solution from maintenance and control standpoint. In other words, it is possible the optimization of the instrumentation in an aeroengine by selecting properly the set of sensors, with the NLGPA (or SVD) techniques for instance, resulting unnecessary and even inconvenient the traditional redundancy of instrumentation, which is a widely extended practice in the industry (for operational safety reasons). Having a clear idea on the recommended instrumentation necessary in a gas turbine engine is a crucial step to obtain all the necessary information when performing diagnostics, control optimization, performance evaluation or prognostics.

Nevertheless, it is not immediate finding out which combination of sensors will be capable to detect specific failures and what instrumentation system will be
optimum for a specific engine model. Sometimes, engine’s external configuration (i.e., pipes, brackets, cables, etc.) could be a problem to install a required sensor.

In this sense, a very direct but still insufficient method to proceed, commented in [217], was by provoking failures artificially in the whole engine, and taking notes on which sensors are providing good feedback about the new operating conditions. Just to illustrate the inefficiency of the previous idea, combustors typically get deteriorated keeping the performance level practically constant, so any damage on them will remain hidden until it is too late to react. It would be more sensible to pay special attention to rotating components, like compressors and turbines, in which the deterioration will be more clearly and quickly perceived. These engine components will provide generally more useful information. But, even so, this method would not be generic, and it will only focus on some relevant faults, potentially leaving without detection some other modes of failure. A set of sensors will be sufficient, as it will be shown later, if it allows to obtain accurately the value of the main H&Q parameters of the engine without spurious correlations. This will be the criterium followed in the thesis, and it will be proved that it is possible to optimize the instrumentation in an aeroengine by means of its application.

Another interesting conclusion from this paper was that, depending on which parameter was used to fix the power output, the measurements needed to diagnose failures in an engine could vary. Namely, both TIT and MW, which corresponds with the temperature reached after the CC, and the mechanical power developed (measured in megawatts) in the rotational spool of the machine, respectively. Both parameters were used as there are different kind of gas turbine engines considered in this paper, susceptible to be used in different applications, not necessarily for aviation only. This conclusion is not surprising, as the determination of thrust (critical parameter, but only in aviation) will need some readings in specific stages of the engine, and similarly happen with the calculation of the power output (crucial for gas turbines used in industrial and marine applications, turboshafts in helicopters, etc.). MW and thrust values are not obtained with the same parameters.

From a mathematical point of view, authors employed linearized expressions (like Urban did), with matrix systems, discarding small terms function of flight conditions and TLA position (i.e., Thrust Lever Angle, or power setting, meaning TIT or MW setpoints). The NLGPA was the technique used to find out the best combination of instruments needed. Another remarkable result obtained in the paper was that high values in the resolution Matrix of the linearized system could hide a bad election of sensors. The great relevance of the Condition Number of the problem is something that will be highlighted in Chapter 4.

With the method used in [217], each iteration provides a baseline to compare with, and different baselines are generated until the final convergence. This is a convenient approach because it avoids storing a large amount of information to obtain the actual condition of the engine. Just working with the new baseline will suffice to evaluate the aeroengine’s status. The stages of the process were:

a) Engine performance model development.
b) Measurement evaluation.
c) Determination of the best sensors’ combination (optimum sensor’s set).
In this sense, the lower Root Mean Square (RMS) error value calculated from the residuals obtained with a predefined objective function (OF), the better capability of the selected instrumentation to detect failures.

Over-instrumenting meant no difference in terms of results when comparing with an optimized set of sensors. This means the over-instrumentation of an aeroengine should be avoided and it would be only justified because of safety and airworthiness reasons. Authors were also aware of the potential risk of obtaining identical sensor signatures for different engine faults, recommending the use of additional or different instrumentation (again, avoiding over-instrumenting) to complete the results reached by their NLGPA methodology. Some sensors were crucial irrespective of the control parameter selected and some others were dependent on the power setting parameter in the paper. Authors developed a very detailed sensitivity analysis after monitoring different perturbations caused by different potential failures in the engine. Such analysis was performed to be in a better position when selecting the best sensors’ combination. That analysis was carefully reviewed but there are more efficient ways to determine an optimum set of sensors. Vibration sensors and lube oil temperature sensors would complete the results obtained by GPA, needing some sort of data fusion.

According to the work done by Ogaji et al., and based on their experience, by the time the paper was published (2002), the best existing method when assessing failures in engines employing the limited information coming from the available instrumentation, was the NLGPA. By considering together the different elements of the engine to match flow, speed, and power, a robust inverse problem solver could be always developed with this methodology. In this sense, and illustrating the opinion of the authors on the topic, the importance of the different methods based on GPA could be noted by the extensive use of it that the main OEMs did since their first apparition [160], implementing such methodology into relevant software used in the industry (see Doel, 1990, [69]): Rolls Royce (RR) developed COMPASS diagnostic system in 1987, Pratt & Whitney (P&W) did something similar with SHERLOCK diagnostic system in 1991 and finally General Electric (GE) released TEMPER diagnostic system in 1994, and GEM few years before.

The approach, the conclusions and, very in particular, the different recommended possible instrumentation sets for isolating turbine faults in a gas turbine engine were evaluated during the development of this thesis. This paper, as well as some others from the same authors, was considered as a key reference on the topic, and these same researchers will be mentioned again in this chapter given their active implication with this topic.

2.2.- Exploring the applicability of genetic algorithms

Some studies, like the one performed by Kong et al. in 2013 [160], among others such as [213] by Ntantis et al., in 2015, showed how certain modern methodologies based on the use of genetic algorithms (GAs) could be a good alternative to the solving process followed in the different GPA versions, especially when considering effects like sensor noise and bias in the instrumentation of two-spool turbofan engines (AE3007H was the engine studied in [160]).

The GAs are designed to simulate processes in natural systems necessary for evolution (processes that follow the Darwinian principle of “Survival of the Fittest”),
and they are applied as an effective optimization tool, not so efficient from CPU time standpoint (as it will be explained in Chapter 5), to get a set of components’ H&Q parameters that generate another set of predicted dependent parameters, through an algorithmic process that leads to predictions that best match the measurements.

The solution is obtained when an objective function (OF, i.e., the difference between predicted and measured parameters) achieves a minimum value, ideally the global minimum in the solution space, when it exists and it is unique, based on model configuration. The algorithm uses a population of abstract representations, which are vectors containing engine health parameters as components in this case, that will compete during several iterations (so-called generations, following the Darwinian analogy) in which mechanisms of selection, crossover, and mutation (among others) take place. The accuracy of the solution can be improved with more individuals per generation, and by running more generations. But accuracy will mean longer computational times, so it is necessary a commitment between accuracy and speed. Another handicap is that evolution does not necessarily advance towards a good solution, it evolves away from bad circumstances instead, so there is always a risk of finding a suboptimal solution.

As it normally happens with studies regarding GAs, the authors in [160] checked how the increase in population size, in probability of mutation, and in probability of crossover could led to a variation in the maximum fitness achieved. And the reason is exploratory methods like the ones based on GAs still mean a considerable computing time comparing with other more direct techniques, existing chances to slightly improve the convergence (however, improvement margin is limited with constrained GAs). On the other hand, no derivatives are required when using GAs which simplifies the calculations, constraints can be imposed with ad hoc penalty functions contributing to improve the convergence of the algorithm.

Global search is typically used to avoid getting stuck in a local minimum, and probabilistic instead of deterministic transition rules can be used to create the consecutive generations, so there are different alternatives to try to improve the GA used. However, even so, the expected convergence times will be high with this technique (more information on the application of GAs can be found in Chapter 4).

A variation of the typical GA methodology is shown by Ogaji et al. (2004) who used in [219] a technique so called ES (Evolution Strategy), which apparently resulted in an efficient tool to diminish the necessary time to reach convergence when solving the equations’ system of the inverse problem established in the paper. GPA was the method selected by default, but it was found to be potentially limited by the relative relevance, number and accuracy of the sensors’ readings taken from the engine’s instrumentation. Also, sensor bias was a factor more carefully considered this time. This method, as usual, needed several sensor readings, at least equal to the number of parameters considered, which would be again the efficiency and mass flow parameters of each main component of the aeroengine. Getting the right information from the engine could be complicated in the practice, for different reasons, if some previous cautions are not taken regarding the condition of the instrumentation system. The main problems encountered by the authors in this sense were the following:

- The non-linear behavior of the components.
- Noise in measurements.
- Deviations from the sensors (potential miscalibration).
Obviously, this is also a common challenge for other techniques and methodologies, but GAs typically behave well with them. So, the main idea to keep from the paper was the way to overcome certain difficulties encountered when applying GAs to the analysis of multiple operational points, such as:

- Complexity of the optimization process, numerically speaking.
- Number and choice of relevant operational points for the solution.
- Number of sensors to use, as discussed in a previous paper of these authors.
- GA parameter selection (i.e., generations, mutations, crossovers, etc.).

Basically, the novelty of this document, compared with others based on GA, like [160], was the inclusion of parameters considered "strategic" that became a part of the calculations implemented in the algorithm finally developed. It is a variation of a typical GA which tries to improve the robustness and convergence of the typical overall algorithm, applying additional intelligence based on the available experience, by emphasizing some effects of the system with new parameters:

\[ g = g((X_1, ..., X_n), (S_1, ..., S_n)) \]  

(2.6)

So, in the proposed new GA method (ES), represented by \( g \), there will be a strategic parameter \( S_j \), for each engine health and quality parameter \( X_j \), modifying the mutation and crossover processes, eventually acting similarly to what a weighted parameter would do. The conclusion of the paper was that the inclusion of these new parameters would help to achieve a faster convergence and a better accuracy of the results (but no time records were given).

The new parameters evolved by means of a new heuristic function called “\( \alpha \)”, especially developed to improve calculations in the genetic algorithm. This function introduces some extra complexity to the method regarding its definition.

Authors defined an objective function (OF) based on a norm that had to be minimized, which considered all the measurements taken and compared them with the calculated values. The expression, in this case, was as follows:

\[ F(\bar{X}, \bar{U}) = \sum_{j} \frac{|Y_{ODj}(\bar{U}) - g_j(X_j, S_j)|}{\sigma_j} \]  

(2.7)

In Eq. (2.7), \( F \) must be minimized globally, task not always easy as it could contain several local minima. Here \( \sigma_j \) is the standard deviation associated to the \( j^{th} \) measurement, \( Y_{ODj} \) represents the \( j^{th} \) off-design undeteriorated measurement and \( \bar{U} \) is the vector which contains the information relative to the engine's regime. The conclusions of the paper resulted somewhat short, but authors showed a possible way to follow inside GA-based techniques to improve the results obtained so far.

It seemed initially that this approach could have been useful for the present research and, therefore, the inclusion of other new weighed parameters was considered initially valid to try to get higher precision in the results, and faster convergence, in general, when applying GAs. However, it was not clearly mentioned in the paper how those new “strategic” parameters were obtained and, given the much lower rate of convergence of the GAs when comparing them against other more effective methods, the GA were quickly substituted by those techniques in this
study, avoiding thus long optimization processes regarding mutation, crossovers, and the rest of factors that determine the configuration of each GA.

In this sense, Silva et al. (2005) used, in reference [264], an advanced type of genetic algorithm known as "StudGA" when dealing with some engine performance-related problems. This time, it consisted in taking the best member, in each generation, to share its genetic information with other individuals that formed the mating pool that would produce the new offspring. The speed of convergence was improved, except where there was little to improve. And that happened because the StudGA did not use stochastic selection of individuals as standard GAs do. Just the best individual was chosen. Authors used Simulink® in their models when looking for optimal solutions. They suggested in their paper three applicable performance optimization scenarios where to use the new GA variant:

I. Minimization of fuel consumption, maintaining nominal thrust.
II. Maximization of thrust, keeping constant the fuel consumption.
III. Reduction of temperature in the hot section of the engine.

In each scenario, the model of the engine in Simulink® was added by a new layer to the control logic to introduce one of those optimizing criteria. It is highlighted in the paper the relevant role of the OF, like the one used in [219] because, even though the GA used to find an optimal solution could be really effective, a bad selection of the OF could penalize the whole analysis, increasing the computational effort required, leading eventually to inaccurate solutions.

Authors in [264] also indicated that premature convergence and local optima should be avoided, in case of exist. These are also common issues, when using GA, reported in other papers. Typically, one strategy to avoid such situations is playing with the rate of mutations in the algorithm. Unfortunately, that option is considerably less effective when dealing with constrained data (if it can be applied with any benefit), like happened in this study.

Authors did not look explicitly for fault diagnostics or prognostics, but just for the performance optimization in gas turbine engines. This is a field in which a small improvement could provide a great value for potential customers, and it is something that can be also attempted by some other methods, providing thus a quicker complete solution, including control rules for operational optimization and health management of the machine during its functioning.

Regarding the three different scenarios evaluated in the paper, the first two were directly related with control and performance, as the main objective of the work was to establish control schedules, for the available control inputs (like Variable Geometry, VG, for instance), to reach an optimal operation. The last scenario, intended to mitigate the effects of fatigue and creep, was the one that had more to do with the maintenance and health management, as it would lead to an improvement in the RUL of the components in the hot section parts before its replacement during Hot Section Exchanges (HSE), or Major Overhauls (MOH).

The engine used as reference was a Rolls-Royce SPEY, like the EJ200 in terms of physiognomy and associated expected relative performance levels. VG devices such as the Inlet Guide Vanes (IGVs), or the exhaust nozzle area \((A_8)\) were kept in open loop control initially, but the whole loop was finally closed to arrive to the definitive and coupled performance model during normal operation.
The most remarkable feature in the work from Silva et al. was the clear aim to reduce computational time, critical element when delivering real-time results, like the ones searched in this thesis. In some cases, the reduction was about 3.5 times the number of evaluations, which is a considerable reduction for a GA-based method, yet far from the solving speed in other methods. This paper shows how a highly effective optimization of a GA would not be typically enough to deliver results on real-time basis. GAs are useful methods when exploring open solution spaces or noisy problems, but they are not definitely the ones with fastest convergence rates.

2.3. **Optimizing with gradient-based methods**

In [206], Najjar et al. (2006) worked with the topic of performance optimization in aeroengines, as many other researchers have done in the last decades, getting to a classical problem to solve. In this case, authors considered on-design performance only (this is selecting the most appealing design point, but without working with already designed engines over their operational line), paying special attention to usual key design variables such as BPR, FPR, OPR or the maximum achievable TIT. This approach has an indirect applicability for this thesis because this study deals mostly with off-design problems, however the solving process followed was found to be interesting, as a valid alternative technique, improving GAs performance.

A gradient-based method (more specifically based on a conjugate gradient technique), was the one used to find the optimal solution to the problem. The gradient can be calculated once the OF is available and well defined so the vectorial function can be consistently obtained over the surface associated to it. Depending on the dimension of the problem, given the number of variables involved, it may be necessary to work with hyper-surfaces instead. The SFC function was the OF.

The authors indicated that, if no care is taken, the associated algorithm could zigzag even if the optimum solution is nearby. To avoid this situation when using the gradient-based method, the problem was divided in two different parts:

1. Determining a suitable search direction, as per the conjugate method.
2. Taking the right step size in that direction.

Authors applied a method that optimized in one direction without interfering or ruining previous optimization steps. Some of these difficulties were already glimpsed by Kurzke (1999) who established in his already classic work [163], that there are typically two ways to follow when looking for optimal values:

1. Running parametric studies: Not useful when the number of parameters is high, like in this case, becoming a time-consuming method, but still valuable when a previous sensitivity analysis must be done anyway.
2. Doing a numerical optimization by using gradient-based methods: Providing faster unnoticed solutions that could be overlooked by parametric studies. On the other hand, when the neighborhood of the optimum found is not well known, the path to global optima of the OF could be overlooked (metaphor of the mountaineer adapting the steps walking down the hill, not finding the steepest descend in his surroundings).
Kurzke did not go into deeper details and developed his results with the help of GasTurb®, as a proved and well-known performance software for aircraft engines, lacking typically the flexibility of PROOSIS®, though. The way in which GasTurb® looks for optimal results is the Random Adaptive Search (RAS), which philosophy is commented in [163]. The starting point in the algorithm is critical to achieve the best optimum possible which, unless the OF is very well known, and counts with a good behavior (convexity), will not be in general the global optimum. To find the global optimum in OF, several tries (shots) could be needed, iterating thus with the optimization algorithm, and consuming time. Further information on the topic of convexity and convergence is provided in Chapter 4.

2.4. Managing the information coming from the sensors

The different techniques to solve the inverse problem should be capable to perform fine adjustment of the engine models installed on board to reliably reflect the progressive deterioration that engines suffer along operation with enough level of confidence. Some relevant research regarding the refreshment of the current health condition with time is available in Lu et al. [179], Li [169], and Volponi et al. [294]. All these works constitute different approaches, not necessarily based on the use of GPA, to update engine models, in which the data obtained from sensors is crucial.

Lu et al. [179] developed in 2013 an interesting method based on non-linear calculations, performed on board, to determine potential sensors' faults. Sensors installed in gas turbine engines are exposed to harsh conditions, including high temperatures and vibration levels. They are susceptible to failure and any undetected sensor fault may lead to undesirable conditions for the whole engine. That circumstance is accentuated by the constant degradation of the engine through its aging process, being this factor one of the main justifications for their study.

Authors worked with available readings from installed sensors and imposed several usual simplifying assumptions in the models to alleviate somehow the associated computational burden. They used several techniques based on Kalman filters this time to obtain real-time updates of the applicable fault thresholds needed to complete the diagnosis of the different instrumentation installed in the engine.

In this work, it was suggested a combination of the traditional data-based methods and model-based methods for sensor diagnostics, leveraging the advantages of each one. Model-based methods keep a better physical understanding of the system and are less sensitive to measurements than the data-based approaches but, according with authors, they cannot be always taken as the baseline because the engines degrade, which is an opinion also shared in this thesis. Some refreshment to the models is necessary and, ideally, it should be done continuously by knowing which is the health condition of the engine on real-time.

Their approach led to an integrated control architecture for sensor fault diagnostics that used engine dual-non-linear models, working in parallel:

- One of them was a non-linear, adaptive, real-time performance for the engine model with an Extended Kalman filter (EKF), to avoid big lacks in accuracy.
- The other one was a non-linear model of the on-board baseline. It provided the updated online reference for the sensors’ output.
The most interesting statement of the work was that the initial engine theoretical model should not be used as a reference for the baseline forever, as the machine gets deteriorated along time. This would mean the diagnostic system will be experiencing a continuous effectiveness’ decrease along consecutive operational cycles. Which is true but, if the engine model is properly refreshed every time a change in the H&Q parameters occurs, that circumstance should not be a major issue. In fact, the sensors’ failure thresholds, employed in the sensors’ failure diagnostic logic, was cyclically refreshed in the model presented by the authors.

This relevant novelty which was the establishment of a new strategy to set up failure thresholds for sensors, as well as for noise levels, could be also very useful for prognostics on the whole engine (to be incorporated to GPA-based methods, for instance), but that possibility was not explored in the paper. The main idea is that a sensor was considered to fail when the residual between measurement and theoretical model calculations exceeded a particular threshold. That threshold would need of a periodical update with the engine deterioration, so authors included in their software architecture a specific module to produce that update based on statistical characteristics of each sensor. Initially, a similar approach could be used for the whole engine, keeping some past values from the previous flight as reference, or from the fleet if available, and watching its future evolution during operation.

Consistently, as an a priori condition set up in the paper, the H&Q parameters that were allowed to suffer most during the regular operation of the engine under consideration, counting with more relaxed thresholds, were the ones relative to the high-pressure compressor (HPC). This component counted with a far longer drifting range than the rest of the components, following all a quadratic evolution. As it also happens in the real world, the parts in the hot section of the engine were the ones exposed to harsher conditions in the calculation presented in the paper.

A relevant feature of the proposed method, that will be also incorporated to this thesis (to be used systematically), is the inclusion of the different engine components’ characteristics maps, circumstance that allowed to particularize the study to a very specific engine, gaining thus in accuracy.

The inertias of the different rotors of the engine model, necessary data when analyzing transient processes, were also included in the engine model. These data provided some more realism to the model even when, in the practice, most of cases analyzed in the paper were steady. Same consideration will be assumed in the thesis, because normally a big portion of the deteriorations suffered by one aeroengine usually occur during steady operation time intervals, given the long periods of time a commercial aircraft typically flies inside cruise stages. Take-offs and climbs are certainly more demanding flight phases, but their duration is considerably lower, as it will be explained in Chapter 3. Maintenance schedules in commercial aviation are designed taking into consideration both flight cycles and flight hours. When cruise phases in a route are long enough, flight hours become the leading factor to decide when an engine must go to the shop for the required maintenance.

The authors considered some usual simplifications as negligible influence of Reynolds number or lack of combustion delays, together with some assumptions for the starting conditions in the EKF algorithm. That made their model certainly more affordable but, obviously, not more realistic, or accurate. The engine model equations were solved with a Newton-based method.

No computational time estimations are given, only the capabilities of the processor used, so the paper does not clarify if such complete methodology could be
applicable for real-time applications. The paper focused mainly on the integrity of the sensors, which is critical to keep the engine operation reliable.

Such approach can be considered to complete the diagnostics of the whole system, including the instrumentation. Certainly, the most relevant conclusion of the work is that having an accurate model for the engine and developing a good technique to establish operational thresholds for the instrumentation are both necessary conditions, but not sufficient, to obtain reliable sensor fault diagnostics. Authors needed to combine several methods, by means of an integrated system, to get acceptable results with the computational cost and complexity this implied. The obtention of those thresholds is a topic of the highest relevance in diagnostics, as they eventually decide what is a fault and what not. A similar approach to diagnose aeroengine components was presented by Lu et al. in [180] (2014).

Li shown in [169] (2015) a new method destined to develop calculations regarding performance of an aeroengine, to predict its behavior and to control it. The target was to be able to adjust an engine fleet model to represent each machine within the whole fleet, by means of some model adaption. Something remarkable in this paper is the inclusion of noisy signals. Surprisingly, the associated increase in prediction mistakes were usually low. Additionally, something that deserves to be mentioned in this work is the time reported to obtain the required predictions: around 30 seconds per each point. Finding these time values is something unusual in the literature. Author suggested this method did not need of maps of components, which reduced calculation times, accordingly with the degree of accuracy reached. However, it took 30 seconds per prediction point. Still a long time for what is expected in real-time applications. The method used a Newton-based technique with an influence matrix (it was used an ICM, to solve the inverse problem with the associated FCM matrix later). The increase found in prediction errors was typically low (around 0.3%), excepting for SFC (2.0%) and BPR (9.5%).

Author indicated that, in case of lack of coincidence in between the number of measurements (M) and H&Q parameters (N), a pseudo-inverse matrix could be generated ($H^#$) in the best least squares sense, when system’s rank obligates to do so (and no considering multiple samples at different times), as follows:

$$H^# = H^T(HH^T)^{-1} \quad (N>M)$$

$$H^# = (HH^T)^{-1}H^T \quad (M>N)$$

In the work, it was carried out an interesting study of sensitivities, and an analysis of impact on the deterioration in the engine. The author based the exercise on an EJ200, a military turbofan engine. For the simulations, Pythia® was the software used. Therefore, this document constitutes a good exercise in calculating performances that has been taken as a valuable reference for the thesis.

The two strategies previously mentioned were applied by Volponi et al. in [294] (2016), by combining some layers of Kalman filters to an adaptive model dedicated to the diagnostics of two “twin engines” installed in the same aircraft. Cross-wing information was employed to find potential differences between the operating conditions of both machines, leading to early detection of sudden changes or remarkable variations in the degradation level of the two engines. This is a very interesting contribution to be considered in the future because the totality of commercial jet aircrafts counts with more than one engine. So, one of the machines on board could help to diagnose a problem in another machine, and vice versa. The
authors used data from the Flight Operations Quality Assurance (FOQA) program to calibrate their model. Again, it seems that model-based approaches used to solve inverse problems are the best chance for engine control and performance optimization (even if the models are so complex to include variable geometry schedules, engine air bleed charts, etc.) when dealing with real-time on-board requirements, given the high solving speeds demanded by these applications, and the large amount of data to be processed by counterpart data-based methods.

Lyantsev et al. (2003) went in this direction indicating in [184] that modern engine control systems often use traditional control laws, with good margin for improvement and optimization. Certainly, this would be an interesting research line aiming to improve the performance in gas turbine engines. Contrarily to full data-driven approaches, in this paper it is available an interesting example on how a good analytic knowledge of the engine model could help to improve computational times. The authors studied, from an impeccable analytical perspective, a relationship between mass flows, effective passing areas, SFC and thrust, so that it became clear what was needed to reduce SFC to the minimum for each thrust value. In other words, it made easier finding the global optimum for the problem. Authors performed a thorough analytical work with the aim of obtaining an online optimization tool. The problem was finally linearized, as usual, and optimized, reaching a very simple expression for the SFC optimum. Most of results were obtained analytically. Variable Geometry (VG) actuators' laws and associated engine limitations, for control purposes, were modelled through well-known integral expressions, from Control System theory, which would need some sort of calibration to refer to a specific engine. Then the problem was solved by standard least squares, an algorithm that is solved very fast by the conventional computers currently available. This way, the available computing power by the time the paper was released was not excessive. This is obviously a very appealing approach. Authors showed how, by means of a deep investigation and knowledge of the case, they could reach remarkable convergences in a very short time, repeating the algorithms every eight seconds. In flight, after 30 seconds, they could reduce the SFC in a two-spool turbofan engine by 0.70%.

It is still to be clarified what would occur in real cases where the analytical determination of the optimum is not so simple, real noisy instrumentation is involved, and some more flexibility for the algorithms would be required to fit into different operational conditions. Nevertheless, from this document, it is important to emphasize how the relationships between the different performance parameters were obtained to optimize the system. It must be understood how, and when, a problem can be linearized, which is the common practice in many of the works reviewed, to switch to another more sophisticated technique later, if needed.

With strong simplifications, other different authors have been capable of providing results on real-time basis (few seconds), as it has been verified in the explored literature. However, non-linear models have been used in the past too, as it was done by Silva et al. (2005) in [264], where authors used an improved version of genetic algorithms (GAs) this time, tailored to minimize times, when carrying out the optimization of several VG schedules using the readings from instrumentation mounted in the engine.

Something to recall at this point is that, in modern aeroengines, some of the most relevant performance parameters for the engine cannot be directly measured and must be obtained by indirect measurements. For example, the EPR is considered
as a valid indicator for the engine net thrust. Another example is the temperature at the exhaust of the engine, or EGT, which is considered as the closest indicator for the actual turbine inlet temperature, or TIT, and a critical factor in maintenance. This means that, by using readings from a reduced sub-set of the available sensors on-board, these directly unmeasurable parameters could be indirectly obtained. Selecting wisely that sub-set of sensors would help to get all the necessary information from the engine’s status. On the other hand, degradation will occur both in engine components and sensors, and the relations that bind the measurable quantities with the unmeasurable performance parameters will eventually vary, depending on the technology of the instrumentation installed on board. Sensors will be treated in Chapter 3 but, for clarification’s sake, in the future, they will be less directly immersed in the flow being measured, will be more protected from harsh conditions and, therefore, their reliability will improve. The selection of the required set of sensors for the resolution of inverse problems will be discussed in Chapter 4.

Assuming that the inverse problem can be solved in the appropriate timescale, GPA-based methods provide a robust framework to obtain the actual performance parameters based on the whole set of available sensor readings, even when components’ degradation and some sensor faults may happen. These parameters could be also used then either as targets or as constraints, for optimization purposes. For example, a target could be minimizing the SFC, subject to a certain net thrust value, as it was done by Lyantsev et al. [184] and Silva et al. [264]. In this sense, every single time a performance model SW is run to obtain the value of the thermodynamic variables in the different stations of the cycle associated to one engine, the main performance results are calculated as well (i.e., thrust, SFC, etc.), so it would be inexpensive to use them inside any methodology, if required, to optimize the operation of the engine, when possible. The application of GPA-based methods for performance optimization, as it happens with engine’s diagnostics and prognostics, depends on the model thresholds’ update.

Such control optimization problem would count essentially with the same requirements than the diagnostics and prognostics problem because the same information will be needed, and same calculations will be made. Obtaining the TIT that the theoretical model predicts for certain engine regime, under the applicable flight conditions, will lead to finding out the desired operation for the engine.

About the use of meaningful available information coming from the complete EHM systems, Volponi et al. (2004) described in [293] a data fusion architecture, so-called propulsion health management system (PHM), applied in their paper to an C17-T1 aircraft, powered by P&W F117 turbofan engines, inside a NASA research program, which was utilized to integrate information coming from different sources, not necessarily instrumentation only, typically available today, such as:

- Engine gas path measurements (standard engine instrumentation).
- Oil and fuel systems measurements (standard engine instrumentation).
- Vibration monitoring readings (standard engine instrumentation).
- Structural assessment sensors (relatively novel engine instrumentation).
- Control System codes (signal integrity condition and fidelity checks).
- On-board engine models (virtual sensors' software embedded in the CS).
- Maintenance and analysis historical logs (a priori data from the operator).
- Companion engine data (comparatives on multi-engine applications).
- Negative information (conditions assuming a fault exists).
And this data fusion, used to handle, reduce, analyze, and interpret the information collected at different levels (sensor level, feature level, decision level, etc.), was developed to maximize the meaningful information extracted from disparate data sources obtaining comprehensive diagnostics, prognostics, and knowledge about the engines. Different instrumentation will be available depending typically on the application (commercial vs military). Advanced instrumentation was traditionally used mostly in the military sector, but this trend is progressively changing, and the newest commercial turbofans mount advanced sensors like electro-static Inlet Debris Monitoring Sensors (i.e., IDMS, installed in the new GE9X).

The more amount of data available is integrated, the more accurate and informed answer will be provided, increasing the system’s reliability, improving the diagnostics’ capability, and increasing eventually the coverage against failure while decreasing the risk of false alarms. That bigger amount of information will obviously also imply a higher degree of complexity, and potentially longer computing times.

Ideally, a data fusion system should also allow for future expandability, being easily adaptable to new data sources. The methodology showed in the paper included a wide range of engine sensors covering high frequency (mainly vibration and structural sensors) and low frequency (mainly gas path, lubrication, and fuel systems) bandwidth signals. The structure of the proposed fusion system allowed for synchronization to align the data into a common timeframe.

A very interesting input to consider in this work, justifying the data fusion strategy, is that GPA-based systems typically depend on a defined set of gas path measurements (pressures, temperatures, fuel flow, and rotational speeds) taken together to assess changes in a set of performance parameters. This analysis requires of theoretical models to establish the reference from which a comparison will be made when estimating changes in performance. The developed algorithms had such a structure that a detected fault would directly impact the system’s ability to accurately assess the fault level. If the instrumentation counted with another sensor, detecting a separate fault, that circumstance would affect the diagnostic.

For example, an IDMS showing a positive (debris ingestion) value would indicate a FOD event that would lead to pay more attention to those components most likely affected. Something similar would happen with vibration sensors or lubrication temperature sensors. This example illustrates the potential interdependency between available data/information and diagnostics’ algorithms. It also suggests the benefit of counting with different data sources. Once more again, the information coming from the engine defines the knowledge achievable on it. A wider variety of sensors available on board would increase the chances to better represent the machine from a performance and diagnostic perspective.

Authors employed different type of techniques and algorithms to model the gas path instrumentation (Kalman filters), lubrication system (simple linear time regressions), etc. The final fusion was performed by using Bayesian belief networks (BBN). This implies a higher level of complexity in the overall system. The real challenge is finding the balance in between data management and computational effort (not mentioned in the paper, but still considerable for the goals of the study) to obtain diagnostic and prognostic results regarding the health condition of the engines under consideration. It results evident that the available data, from a quantitative and qualitative standpoint, will determine the analysis, the algorithms,
and the engine modelling. However, each available strategy should be explored to avoid resorting to complex architectures like the ones in this paper.

A new valid question that arises is how to make sure that the selected H&Q parameters, and not the sensors, for gas turbine engine health status estimation, are rigorously enough, because only some indications on sensors’ selection have been reviewed in [217] so far, assuming always the set of status parameters was enough, according with the literature on the topic. This new question goes in the opposite direction to what has been explored so far. When a given set of measurements is available from the instrumentation, the optimum set of necessary H&Q parameters should be known beforehand, to understand if such instrumentation will be enough to provide a full picture of engine’s health condition given by those H&Q parameters.

In this sense, Stamatis et al. (1991) presented in [273] such a rigorous approach, mathematically speaking, defining how the H&Q parameters’ selection should be done, guaranteeing the operational safety when monitoring a gas turbine engine. They also dealt with the instrumentation required. The key factors in their methodology were the engine configuration (and its associated theoretical model), the available instrumentation, and the required accuracy. In this work, a selective procedure was utilized based on the SVD technique. The main objective of the study was providing a health monitoring expert system and giving a value of uncertainty for the results obtained with it, so this method could be used to make decisions as well. Authors clearly indicated that the performance models used for gas turbine engines needed to include specific performance data from each engine component, as well as compatibility of conditions and constraints (usually called matching conditions), and the right application of the principles of aerothermodynamics.

Then, with the available data from each component, it is feasible to elaborate a formal study to evaluate the impact generated when one component changes in terms of efficiency, mass flow capacity, etc., like the GPA-based methods do. This way, it is also possible to find out which component is failing when a sudden perturbation occurs. Several status “indexes” are defined in the paper which could be compound by one or several health or quality parameters, and they are examined when crossing various operational thresholds.

According to the authors, there are certain criteria that must be followed when selecting health or quality indexes, and when applying them to a specific engine. These status indexes must:

- Allow the isolation of each specific main component of the engine, as they must ensure fault isolation at a component level.
- Be sensitive to changes in the measurements obtained.
- To count with a clear physical meaning, aspect not always taken into consideration in maintenance as there are convenient and traditional parameters that have been used already for a long time (e.g., EGT is used to decide about engine’s remaining life, but it does not improve the information provided by efficiencies or flows, regarding components’ condition).
- Allow for the isolation in the effects of the measurements coming from the instrumentation to minimize the interdependence between sensors.

These conditions are met typically by the usual H&Q parameters mentioned (and extensively used, \( \eta_i \) and \( \Gamma_i \)) so far. Authors also established a set of rules, defined mathematically, to address the selection of measures and H&Q indexes:
- For each combination chosen of health or quality indexes, it is necessary to select them \( (X_k) \) in such a way the value of “m” in the sensitivity norm \( S \) given below is maximized:

\[
S(X_k) = \left( \frac{1}{m} \sum_j (\Delta X_k(j))^2 \right)^{1/2}
\tag{2.10}
\]

If \( r \) denotes the baseline for a healthy engine and \( j \) the status when the \( j^{th} \) index deviates from that baseline, the sensitivity of each measured parameter \( (k = 1, \ldots, n) \) under variations of the different health and quality indexes \( (j = 1, \ldots, m) \) is expressed as follows:

\[
\Delta X_k(j) = \frac{X_k(j) - X_k(r)}{X_k(r)}
\tag{2.11}
\]

- To select the right health indexes, the mathematical criteria must be rigorously applied because it may happen that the number of measurements could be lower than the number of health or quality indexes in the method. So, several rules are followed to choose the optimal combination:

  o **Rule 1**: Minimum uncertainty in the method, by employing a different norm, which will have to be minimized this time. Now, \( v \) denotes the health or quality indexes (given by the components of \( \bar{X} \)) that vary, as not all of them would have to suffer changes necessarily:

\[
S(X_{v,k}) = \left( \sum_i (\Delta X_{v,i})^2 \right)^{1/2}
\tag{2.12}
\]

  o **Rule 2**: The indexes should identify failures in accordance with its frequency of appearance.

  o **Rule 3**: Propagation error, caused by non-estimated indexes, should be minimum.

The application of each rule will lead to a different classification of the set of indexes selected, according with the degree of accomplishment with those rules. The rules just constitute a formal and rational way to justify the selection of an adequate set of parameters to avoid algebraic issues during the ongoing operations, such as matrix inversions. A quick procedure to select the necessary indexes comes by linearizing the sensitivity of the effects (given by the components of \( \bar{Y} \)) with a Taylor’s expansion by the H\&Q indexes parameters (thus the vector \( \bar{X} \)) as follows:

\[
\frac{Y_k - Y_k(r)}{Y_k(r)} = \left[ \frac{\partial \ln \bar{F}}{\partial \bar{X}} \right] \cdot \{ \bar{X} - \bar{X}(r) \} \bigg|_k + \text{HOT}
\tag{2.13}
\]

In this sense, \( \Delta \bar{Y} = \mathbf{J} \cdot \Delta \bar{X} \) would represent the linearization of the model (i.e., \( \bar{F} \)), characterized by the Jacobian matrix. In \( [\partial \ln \bar{F}/\partial \bar{X}] \), it must be considered the logarithmic function. It was left out of consideration higher order terms (HOT).
From Linear Algebra, the status indexes with more important contributions or with highest projections in the direction of the system’s single values (SVD) will be the most relevant ones for the final solution and they will be, eventually, the chosen ones. This criterion has been followed in the thesis, but when choosing the adequate set of sensors, as it will explained in Chapter 4.

The engine used in the paper as reference for a practical example was the Tornado gas turbine, a well-known machine in industrial applications. According to the results of the paper, it is necessary to use two indexes for each component. In the case of the Tornado, those main components are: HPC, Combustor (CC), HPT and Power Turbine (PT). So, eight indexes and eight parameters to solve the system when inverting the matrixes that will be produced. And it must be said that this was the common selection of indexes in most of the papers reviewed so far. In the case of the Tornado, only seven parameters could be measured, so only seven indexes could be determined, following the rules exposed before. By considering more than one operational point and calculating the resulting inverse matrix it could be possible to try to overcome this issue. With those rules and by using SVD, efficiency in CC was not considered finally. All these results underpin, for researchers’ peace of mind, the consistency of all the “building of knowledge” on the topic.

Sarkar et al. in [249] (2009), and [251] (2012), seemed to have followed a very similar procedure when selecting a proper engine model for a data-driven fault detection method as their engine theoretical simulator counted also with two parameters per main component. The degree of detail on the engine's theoretical model was even improved by adding, to the usual set of sensors, a set of actuators for variable geometry control, such as Variable Bleed Valves (VBVs), and Variable Stator Vanes (VSVs). The fuel supply regulation was also included in the model. In this case, the Commercial Modular Aero Propulsion System Simulation (also known as C-MAPSS), developed by NASA, was used as test bed (dynamic gas path model) to produce simulated data.

None the less, that is probably the only similarity with the methods seen so far. The methodology used now to perform a diagnostic of the physical system was based on statistical estimation of fault conditions given an experimental data set. This means some data would be needed a priori or, at least, recommended, to gain in convergence speed, as it usually happens with data-driven methods like this one. Previous training and testing processes were required.

Some complex mathematical developments, caused by the desired data fusion, had to be incorporated as well. It is true that the system made possible the analysis of multiple faults, but the subsequent penalty in mathematical complexity was remarkable. In fact, an important number of issues was admitted and highlighted in the work when applying this method for real operations. Authors used a nonlinear feature extraction technique called symbolic dynamic filtering (SDF) for detection of anomalies in dynamical systems, proven with time series and validated even for real-time execution, under certain conditions, in different technical applications. Authors also suggested the use of SVD for data fusion.

The mathematical and numerical details of the SDF technique are quite elaborated, being considerably more complex than similar methods such as GAs, as it implied the establishment of different time scales (fast vs. slow) for feature extraction, the management of symbolic or semantic sequences of a specific length representing the information from different sensors, the application of a product space of different states, and the definition of mathematical models, ad hoc, for the
problem under consideration. Then, the semantic framework for multi-sensor data interpretation and fusion generated was used in two different scenarios:

- **Scenario 1**: Natural deterioration versus faults in a single component. The two types of engine changes considered were the progressive deterioration (typically happens at a relatively slow rate, due to aging and usage) and sudden failures or rapid faults (relatively more rapid change rate due to anticipated harmful events), both to be applied to a single engine component. Authors needed to distinguish in between both cases to avoid the apparition of false alarms, which does not redound in more robustness for the method.

- **Scenario 2**: Multiple faults occurring simultaneously in different components. Authors selected HPC and HPT as faulty components for the simulations to introduce more complexity to the method, when isolating faults, given the existing coupling in both components.

Finally, in [251], it was established a hierarchical selection criterion (so-called “classifier”) based on the patterns of appearance of the different type of failures (so-called “fault classes”). Authors highlighted in the paper the problem of handling time series of data, due to its volume and computational complexity.

A system to filter, reduce or compress the available data would be advisable to fulfill that task without losing valuable information. And this is another call for the use of ROM-based methods in the future, as the amount of available data from the gas turbine engines tends to increase. This complex approach, clearly valid for electronic circuit analysis and fatigue damage monitoring, could be excessively time consuming when applied to a gas turbine engine application.

In fact, once more again, no mentions on computational effort were done, and no time estimations whatsoever were given. So, the authors did not provide information in relation with the time needed to reach a valid solution. In the paper, it was combined multivariate Gauss functions, symbolic dynamic filtering (SDF), and Bayesian techniques, which are not immediate and require a deep mathematical and numerical knowledge to be successfully applied. However, even when this method needed some polish, after a confrontation with C-MAPSS, results were not far from the initial goals established by the authors. The paper showed another valid way to evaluate the information coming from different types of sensors, which is always challenging, by measuring different physical magnitudes along time. It illustrated the variety of potential valid alternatives, mentioned in the literature, to address the same problem relative to diagnostics (prognostics and control-related features would come potentially together). The previous results are interesting for the goals pursued in this thesis; similar approaches will be evaluated to try to cover as many fault cases as possible and to evaluate if the proposed methodology suffers of any weakness given a particular fault scenario. The SDF will be stored also for the future in this research line just in case it could be needed.

This way, with the previous set of references schematically organized in Figure 17, it has been already covered several relevant examples, available in the classic literature on the subject (including the extremes in the methodologies’ spectrum), of model-driven and data-driven techniques relative to gas turbine engine performance and EHM analysis that will be stored for the future. However, there were more recent studies reviewed and incorporated into the “Building of Knowledge” for this thesis, as it will be detailed later.
The list of reviewed documentation in this matter has been continuously growing during the last five years, and the expectation is it will keep similar growth rate. The next section will cover some relevant references that served to confirm that the most relevant techniques and methodologies had been already considered, before initiating the calculations. Some few unconventional techniques will be introduced then and, finally, some recent works on the use of ROM-based methods will be commented, before closing this chapter with few concluding remarks.

![Building of Knowledge from Literature](image)

**Figure 17:** Exploration followed so far within the available literature.

### 2.5. Compilatory studies analyzing the available known techniques

In order to compile and summarize the different techniques applicable to optimization, diagnostics and prognostics, in a systematic way, to avoid getting lost in the profuse literature, Marinai et al. (2004) revised and compared, in their reference [187], the main different methodologies existing by the time the paper was released, from the ones based on GPA to more recent ones based on GAs or Neural Networks (NNs), attending to different economic justifications, in a more detailed way than Ntantis et al. did in [213]. To understand the interest from the research community on the topic in the last decades, sufficient is to say that, in today’s commercial aviation, the operational life of a conventional aircraft is getting longer and estimated in more than 25 years (see [6] in this regard), low-cost airlines have recently emerged and prospered in response to the increased commercial competition, well-established airlines are driven to improve efficiencies in their fleets already counting with mature aircraft (as much as technically and financially feasible), etc. Circumstances that eventually mean important business opportunities, and a promising field for future research.
As mentioned in Chapter 1, the search for operational cost reductions, and efficiency (i.e., fuel management) improvements is becoming critical for many companies operating gas turbine engines these days, like the commercial airlines. In this sense, a big portion of the pressure, in terms of responsibility for the operability of the aircraft and survival of the airlines, has been transferred to the respective OEMs because airlines demand high quality fleet management inside a CBM framework, and such technical support should come from the OEMs initially. It must be understood that an engine is a costly asset financed during the period in which it is in operation. No airline can afford keeping the engines stopped and aircrafts grounded for a long time. A similar situation happens with gas turbine engines inside cogeneration plants, industrial combined cycles, or in marine applications, for both civil and military sectors, as Orsagh et al. (2002) commented in [222]. Because of this circumstance, all the different fault diagnostic and health condition techniques have been considered of great help when making informed decisions about MRO of gas turbine engines, given the great flows of capital dedicated to it every year worldwide. Methods are useful depending on the payoff they mean in terms of ROI. The time it takes to recover the initial investment cost (the pay-back period) will be acceptable, or not, depending on the economic benefit for the customer after implementing such method [222]. Obviously, both the uncertainty associated, and the potential cost savings (like the ones in Figure 18) must be considered to evaluate the net benefits and effectiveness of any new EHM system.

![Figure 18: MRO costs in the airlines and potential cost savings. Data obtained from IATA [137].](image)

Coming back to [187], the authors analyzed the different techniques appeared sequentially since 1970, meaning a continuous increment in complexity and a higher degree of combination with other methods, that had been evaluated at the light of the challenging scenario in the gas turbine maintenance industry:
- Technique 1: LGPA with inversion of the ICM matrix, which means a linearization of the equations of the model.
- Technique 2: NLGPA with inversion of ICM, employing Newton-based methods to obtain the optimum of the inverse problem.
- Technique 3: Kalman filters (KF), based on GPA with linear approximation and Minimum Least Squares (MLS). It needs some previous work and the use of KF, together with the linearization, introduced some additional errors. The MLS are sometimes used with specific weights to improve the convergence (leading to the so-called Generalized Minimum Squares, GMS).
- Technique 4: GPA-based on different Extended KF methods: EKF and IEKF.
- Technique 5: Estimation of an optimum by means of a non-linear model employing Genetic Algorithms (GAs). This was done in this study. It is again necessary to find out the minimum of a specific OF. A big computational power is necessary to get convergence in a reduced amount of time. This technique is typically limited to few parameters experiencing simultaneous deteriorations and it has been improved along the years. GAs count with a remarkable level of accuracy, even when their degree of convergency is lower than in other methods, leading to large computational times.
- Technique 6: Artificial Neural Network (ANNs / NNs). These methods need previous training for the NNs, and that process consumes a considerable time. The interested reader counts with the work of Sampath and Singh (2004, [246]) for further details. In that work, authors used a NN, as a pre-processor, to reduce the number of cases to explore.
- Technique 7: Bayesian Belief Networks (BBN). They need a long time of calculation given the complex model implementation.
- Technique 8: Expert Systems (ES). Limited in capability of providing valid results and normally applied to cases where only a qualitative answer is needed, achieved by the consecutive application of different rules.
- Technique 9: Fuzzy Logic based diagnostics (FL). Very popular in the last years as this technique gets good results when identifying failures. It is necessary to highlight that this technique does not provides accurate results.

This list of techniques probably includes the most relevant ones from a historical perspective but is not comprehensive, as it is constantly updated and a good number of similar procedures (i.e., variations, simplified versions, etc.) do not appear specifically on it. For example, some authors, like Gomez et al. (2000) in [115] or Ganguli et al. (2004) in [98], used parametric estimation approaches combined with likelihood tests and least squares, respectively, to identify and isolate several potential failures in gas turbine engines. This approach has been recently overcome by more advanced and powerful techniques in terms of effectiveness, accuracy, robustness, and in economy of computational resources, when dealing with nonlinear models and decoupling the different fault modes. Even when that is probably the main reason why they did not prove enough relevance to be mentioned in [187], they constituted interesting attempts that deserved, at least, to be mentioned. The issues found in the different available techniques have motivated the apparition of new ones. Some clarifications should be made to give content to the techniques’ list exposed above, and to illustrate the reasons why the methodology followed in this thesis (based on the use of GAs, SQP, adapted Newton,
and ROMs), mainly dependent on the performance model of the engine, was finally chosen over others, also potentially valid.

According with [213], one of the biggest issues of the GPA-based methods is the lack of precision caused when only few measures are available and because of the errors inherent to those measures (uncertainty, noise, bias, etc.). The first issue was identified during this study as it will be shown in Chapter 5.

However, GPA-based methods also count with a clear advantage over some other data-based methods which is the endowing of the problem variables with a clear physical meaning. The output values from any GPA-based method count with an easy interpretation, intuitive and accurate, especially when representing the level of degradation that the engine has experienced over time. In some works detailing the use of data-based methods, such as [305] by Zhou et al. (2018), it was necessary to define independent "health indicators" by means of sensor fusion, at each flight condition, as if no physical relation would link those conditions, marking the engine as “faulty” if one of those indicators would become true, independently of what the state of the other indicators might be. That kind of approach complicates somehow the interpretation of results, independently of the accuracy level achieved.

Contrarily to the method shown in [305], GPA-based methods could use the redundancy given by data obtained from the same engine but at different operating conditions, to increase the robustness and accuracy in the predictions on the health condition of that engine, without causing confusion during the process. And there is no need to define any independent indicator, ad hoc, or any artificial parameter when using GPA-based methods. The way the problem is built leads naturally to an immediate interpretation of the results.

In this sense, data-driven techniques rely absolutely on the previous knowledge of the possible degradation that could have occurred in the engine under consideration and are not typically capable to detect faults for which they have not been specifically trained. GPA-based methods, on the contrary, do not try to classify faulty conditions as per pre-established patterns, strategy that will clearly restrict their detection capability. They just quantify the separation between the operating condition of an engine component and its nominal “healthy” state, being able thus to detect any potential degradation, even if no previous data about that performance pattern had been collected before. This circumstance contributes to make the use of GPA-methods a more robust and reliable strategy, although at a considerable online computational effort. In data-driven methods, most of the computational load is usually handled offline, instead.

Regarding Kalman filters (KFs), it is necessary to mention they are massively used today because they are probably the best optimal estimators (predictor-corrector kind, minimizing estimated error covariance) for a considerable number of problems. KFs are optimal recursive data processing algorithms that have been successfully used for gas turbine engines' health estimation in presence of measurement noise and sensor bias. Besides this undoubtable practical reason, it is not necessary a deep mathematical background to implement them and to follow the associated reasoning, corollaries, and relative calculations, which makes this technique appealing for relatively profane researchers. It is also a robust method that provides right results even in cases when the preliminary working hypothesis are not totally satisfied. As indicated in [213], a KF-based system processes all available measurement data regardless of their precision, together with a prior knowledge about the system and measuring devices, to generate an estimation of
the desired variables in such a way the error of estimation is statistically minimized. The linear algorithm of a KF is based in two mechanisms:

1. **Prediction**: Used to propagate the internal state of the system. The internal state of the system is predicted, in the next time step, by the filter.
2. **Correction**: Used for the fine tuning of the prediction mechanism under the influence of external observations. When a real measurement of the next time step is available, the filter corrects itself on the prediction error, minimizing error covariance.

D.L. Simon (2008) indicated in [266] that the most common approach for the estimation of the different states of linear dynamic systems was still the KF. For non-linear systems, more elaborated variants of the KF should be used such as LKF (linearized), UKF (unscented), EKF (extended) or IEKF (invariant extended). With LKF, EKF and IEKF it is necessary to obtain Jacobian matrixes that must be properly refreshed, and this operation needs of a relevant (cumulative) computational effort, the higher effort the more variables (both sensors and H&Q parameters) are involved, as it was verified during the realization of this thesis (see Chapter 4 for further details on the calculation of Jacobian matrices).

In UKF, there are no Jacobians to calculate, but the need of several calculations per step in the dynamic system under consideration tends to increase the computational effort. In [266], the author showed that the computational effort in UKF is one order of magnitude higher than in EKF. And in EKF one order of magnitude higher than in LKF. The conclusion is EKF is the preferred KF-based method (even when needing computational times in the order of $\sim 10^3$ seconds) for aircraft engine health estimations but, it must be taken into consideration that EKF is difficult to tune, and it often provides unreliable estimates when nonlinearities are severe (because EKF depends on linearization to propagate the mean and covariance of the state). The author used C-MAPPS, as a software package (written using MATLAB-Simulink®), for the simulation of a turbofan engine. Noise terms and health parameter degradations are not modelled initially by C-MAPSS, so these effects must be added to the method. Once again, the same number of observations than health parameters were needed to achieve system’s “observability”, so to estimate “n” health parameters it was needed, at least, “n” measurements.

KF-based methods still constitute one of the most common systems employed so far for EHM purposes. Nevertheless, these methods typically consider slow deterioration effects only. In case some other sudden effects may appear during engine operation (such as the ones caused by FOD, DOD, unexpected sensor faults, etc.), and they had to be quickly taken into consideration, then additional logic should be implemented to identify and analyze properly those faults.

Kobayashi (2003), and Kobayashi together with Simon (2003) applied in [156] and [157], respectively, a bank of KF for engine fault diagnostics (detection and isolation) to overcome this inherent issue of the method. Authors used a KF of the complete bank per sensor being monitored and they adapted the method to take care of engine actuators as well. So, the filters used for the different sensors and actuators were integrated using a fault isolation logic afterwards.

The reality is this technique successfully detected and isolated faults in sensors and actuators but, on the other hand, it found some problems of robustness when it was also considered the effects of degradation in the engine.
Another work in this direction is [172], where Liu et al. (2017) proposed a new method to update the on-line health reference baseline of the On-Board Engine Model (OBEM) incorporating banks of Hybrid KF (HKF) to the overall control logic. HKF are variations of regular LKF that are composed of non-linear OBEMs and linear models that include, among other additional elements, Kalman gain matrices. The HKF replaces the performance baseline of a linear KF with the outputs of the OBEM. The HKF consequently does not need data training in advance, and it counts with a simple algorithmic structure. Nevertheless, when simulating sudden changes in a gas turbine engine, the method needed reaction times in the order of magnitude of one second to follow the real variables. Results were found enough accurate, but the accumulated reaction times were longer than expected for the targets in the paper.

Additionally, in the KF methods, the a priori estimate is normally a reasonable guess and must be close to the true value to get a high degree of accuracy, that’s why the HKF is fed by the output from an On-Board Engine Model (OBEM) in [172]. Kobayashi et al. (2008) used in [158] a linear OBEM version (LOBEM), which is composed of piecewise linear models, generated at multiple operating conditions for on-line sensor fault diagnostics. The LOBEM acted in this paper, again, as the third channel (a “third sensor”), meaning as a baseline system, when dual-channel measurements are compared. This third channel worked, obviously, as a “referee” in the decision making when an issue in a dual-channel measurement was found. LOBEM worked better when the nonlinearities found were moderate, as usual.

In this sense, several authors have particularized their analysis to the important topic of the instrumentation’s reliability installed in the engine. Lu et al. (2012) analyzed in [178] the reliability of the instrumentation and affirmed it is not possible to know which of the channels arriving to the engine control system, namely FADEC, is failing. The processors inside must discriminate which one, in a dual-channeled sensor, is causing problems (topic also treated in [158] by Kobayashi et al.). In those situations, it was simply taken the measure that is considered correct according to some safety criteria (as it is also done in the industry). This circumstance was explored for the T700s, very common gas turbine engine model (turbo-shaft engine, used normally in helicopters), in which the associated CS was not capable to detect soft failures in certain sensors. Authors revised in their work available techniques based on mathematical methods, previous knowledge, and signal processing techniques. The Kalman gain matrix was calculated iteratively and a particular FDD (Fault Detection and Diagnostics) logic, based on a simplified OBEM, was finally applied. The FDD was a triplex logic theoretically capable to determine the sensor failure cause. Working with a simplified model of the gas generator and power turbine, and assuming limited changes to the rotational speed of the Core Engine, results could improve. The simplified model was used as a third sensor to corroborate the measure in the other two sensors. Authors in [178] did not use measures from sensors directly, they first removed noise and outliers. The method was based in the comparison of residuals with threshold values. This meant eventually a purely statistical procedure. If one of the sensors counted with a residual value that exceeded a stablished threshold, it would have been considered as a failing sensor. If the two mounted sensors were simultaneously exceeding the threshold, then it was supposed to be an anomaly in the system of instrumentation.

There were some more scenarios depending on the combination of potential failures but that was, in short, the criterion followed along the paper. To choose the
applicable threshold was again essential, and the authors selected the applicable thresholds depending on the statistical characteristics of the sensors: noise, bias, possible modes of failure, etc. What was not mentioned to be done on real-time basis was the development of an analysis of confidence intervals to find out if one of the sensors was failing with enough certainty, statistically speaking. Two main types of failure in the sensors were considered in this work: step failure and drift. The algorithm so called SPSO-SVR (Support Vector Regression Trained by Self-Tuning Particle Swarm Optimization) was employed to diagnose failures in sensors. The results obtained, when correct, were enough accurate to be used for predictive purposes. The problem was the system was not robust, and not fast either, therefore these circumstances would make difficult its use on real-time applications like EHM during real flights or gas turbine operational cycles.

In addition to all the previous considerations on the use of KFs, the methods based on this technique could become unstable numerically if the requirements in terms of accuracy are too demanding. It is also difficult to overcome systematic errors with KF methods. “Smearing” effects could appear when intermediate quantities of large size sweep important data (loss of information) and cause the final response to be the product of numerical cancellations. All these reasons advised initially against the use of KF-based techniques for the purpose of the thesis.

Regarding another of the reviewed techniques, the Neural Networks or NNs (also ANNs, as they are “artificially” generated), it must be mentioned that they are non-linear estimators that attempt to simulate the brain learning process, making them effective at performance pattern recognition. The mathematical structure behind distributes input data into different interconnected simple units or cells (called “artificial neurons”, following the analogy). Then, the data is processed in parallel by each neuron of the neuronal system previously defined. The functionality of the neuronal system depends on the network structure and connection strengths.

Because of the parallelism and high connectivity achievable with this system in between neurons, the NNs are capable to link, in a non-linear way, a multidimensional input space with a multidimensional output space, thus allowing for a very high computational speed. This method stores and adapt experimental knowledge in the network during the training phase (unavoidable with NNs), constituting this a remarkable weakness when considering its application to real-time, on-board, EHM systems. The training phase ends when an OF is minimized (it could be, for instance, the RMS error of the set of residuals generated in between measured and predicted values). That function uses the whole training data set and establishes a comparison between the target and the outputs (especially appropriate for faulty sensor detection).

The preliminary phase could take an important amount of computational effort until its conclusion and this preliminary phase is needed every single time a change in engine’s hardware is performed or detected (i.e., a sudden event could count as “change” or modification to the engine’s configuration). Also, the quality of the data needed for the training phase is high (stable and representative) and, thus, difficult to obtain. On the other hand, NNs can be tolerant to the noise in the available data, and this is a very positive feature of the method (even when that does not mean dealing with noisy data is easy when using NNs). Additionally, once the training phase is completed, NNs use to be fast. However, some NNs modeling complex systems involving combustion processes could take weeks to be fully trained.
When applying the NNs to engine health monitoring systems, it is common in the literature the separation of the fault diagnosis problem into two phases: identifying the faulty component first, and quantifying the fault later, which could be a sensible strategy to avoid unnecessary delays.

About the use of NNs on performance optimization, diagnostics and prognostics, Ogaji et al. (2002), in [216], dealt with situations in which these techniques could be useful: possible failures in sensors that generate false alarms and provide wrong indications from healthy components in a specific engine (this time, one Rolls-Royce Avon was selected, thus a single shaft gas turbine engine). In the paper, three different architectures for the neural network were used: the first one differentiated the events in between “faults” and “no faults”, the second one classified the faults into either a sensor fault or a component fault (such as VG actuators), and the third one quantified the magnitude of each fault, isolating first any single or dual faulty component. This gives an idea of the complexity of the approach. The non-linearity of gas turbine performance models is mentioned to be one of the main reasons to explore the NNs’ capabilities. Authors managed a maximum of six sensors, which is maybe enough for such a simple engine model, but still to be determined how it would be for a more complex modern turbofan. A random noise distribution was also considered to gain robustness in the results.

In [218], Ogaji et al. (2003) highlighted the importance and utility as well of the NNs by performing diagnostics in gas turbines (models employed in industrial applications like the RR-Avon and the GE-LM2500) for maintenance purposes. The authors found, by that time, very limited the capability of simple linear techniques based on GPA when detecting, isolating, and assessing faults, so they established a hierarchy of algorithms to order the effects in the different quality variables when changes in the measures occur. That means the use of several decentralized networks to handle different tasks. NNs were used in [218] when filtering the information containing noise and bias from sensors, allowing thus the detection of failures in one sensor or in several of them simultaneously:

- First, the optimal set of necessary instruments to diagnose multiple failures in the gas path rotatory components (i.e., mainly compressors and turbines, combustors were excluded as its efficiency is relatively stable with time and they do not provide sufficient information) was obtained by means of simulation.
- Then, a hierarchical system of NNs was used. The first ones detected the presence of a failure in the engine and the next ones discriminated the potential failure from other possible failures.

It is also clearly stated in the paper that the determination of a good baseline is again essential. The NNs selected for the study could be set up in less than 2 minutes when valid data was available and this, obviously, does not help to develop a real-time method (using a Pentium III processor at 933 MHz, so with modern computers the required time should be lower, but this is still to be confirmed for a more complex engine). In any case, this is a technique to keep in mind as it counts with some benefits such as expansion capability and possibility of data fusion.

Another set of techniques often found in the literature are the Bayesian belief networks, or BBNs. These are powerful tools for fault component identification in gas turbine engines. They are based on formal and advanced probability theory. In
these techniques, it is integrated both test measurements and performance analysis with information regarding operational history and direct physical observation, which is helpful when developing cost-effective diagnosis tools, and performing “value of information” estimations. The BBNs are graphical representations of a probability distribution which encodes the cause-and-effect relationships between variables represented as strings and nodes, respectively, in the system layout. Each node of the network represents an observation or a fault that contains the conditional probability that describes the relationship between that node (effect) and its parents (causes). The H&Q parameters could occupy the parent nodes meanwhile the measurements should be in the child nodes, or vice versa, depending on if the scheme is thought to solve a direct or an inverse engine problem.

These methods need of substantial computational effort and time to gather the information required for setting up a good data base. It is not easy to perform any modification in the networks as certain expertise in the mathematical procedures is needed to do it properly. When the BBNs are combined with any performance analysis method, then the drawbacks of such method will be inevitably incorporated to the combined method. Additionally, BBNs cannot deal typically with sensor bias. A good overview on the topic, with the aim of using BBNs in aeronautical maintenance applications, is presented in [93] by Ferreiro et al. (2012).

In relation with Bayesian belief networks, it is necessary to mention that another popular technique, missing in the list given in [187], is the hidden Semi-Markov model-based methods or HSMMs. Both diagnostics and prognostics could be typically addressed by means of this methodology, but its complexity makes its use something hard and almost restricted to experts in the matter. Besides its inherent difficult mathematical approach, as it probably happens with methods that try to cover in a very generic way (model-based) processes which characterization is dependent on experience (data-based), results are not outstanding if they are not somehow calibrated with real data. Several assumptions in the model could lead to remarkable lack of accuracy in the results. Nonetheless, when the understanding of the system’s degradation improves, the HSMM-based methods can be adapted to increase its accuracy. Regarding computational efficiency, some of the structures and statistical algorithms used can be really time consuming. The training process improves considerably the effectiveness of the overall system. So, the HSMMs have margin to improve, and this is an interesting way for future investigations.

To understand this methodology, it is necessary to define first the so-called “Markov Chains”. These “chains” are sequence of events (called also “states”) which probability of happening depends only on the immediately preceding event. This logic is suitable for deterioration processes like the ones analyzed within diagnostics and prognostics. The hidden Markov models (HMMs, one the simplest form of dynamic Bayesian networks relating variables to each other over adjacent time steps) represent stochastic sequences, as Markov Chains, where the states are not directly observed but are associated to a probability function. The HMMs are defined by the number of health states, time units, number of observations per state (i.e., measurements), state transition probability distributions, observation probability distributions, and the initial state probability distributions. These probabilities need of initial estimations. In the simplest Markov models, the state is directly visible to the observer, so the state transition probabilities are the only parameters to consider.
In the HMMs, the state is not directly visible (is hidden), but the output (in the form of data of the following states), which is dependent on the state, is visible. Each state has a probability distribution over the possible output data.

Therefore, the sequence of data generated by HMM gives some information about the sequence of states. It is important to recall that the adjective “hidden” refers to the state sequence through which the model passes, not to the parameters of the model; the model is still referred to as a hidden Markov model even if these parameters are known exactly because the states are hidden from direct observation (states manifest via some probabilistic behavior). HSMMs are a kind of statistical model with almost an identical structure than HMMs, excepting the unobservable process, which is semi-Markov rather than Markov (i.e., the probability of change in the hidden state depends on the time that has passed since its entry into the current state). This is the crucial difference comparing with HMMs, where there is a constant probability of changing states.

In [70] and [71], Dong et al. (both in 2007) developed a new statistical framework and methodology for multi-sensor equipment health diagnosis and prognosis. In these papers, it was indicated that the state duration of HMMs followed an exponential distribution. HMMs did not provide adequate representation of temporal structure for diagnosis and prognosis. That was why authors built a HSMM by adding a temporal component into a well-defined HMM structure. In this framework, health states were used to represent the health status of a component, always going to a worse health state. Even when HMM was applied to a multi-sensor equipment, it was analyzed each sensor separately. To train the resulting HSMM, authors estimated state duration probabilities from training data. HSMM was characterized in these papers by several parameters such as:

- Initial state distribution.
- Transition model.
- State duration distribution.
- Observation model.

Several algorithms were used to complete each piece of the method to solve the problems of evaluation, decoding (i.e., finding the best state sequence associated with a given observation sequence), and learning, associated to the HSMM.

For instance, a modified algorithm (known as Viterbi’s algorithm) was used to find the most likely state sequence that generated a given observation. Also, a modified forward-backward algorithm was used to re-evaluate all the variables implied in the problem, etc. This can give an idea of the overall degree of complexity of the method. It was confronted in the papers with simple rotatory equipment (vibration in pumps) and the diagnostic results accuracy in those cases was promising (reaching levels around a 96%). A complete aircraft engine, with its all complexity associated, will surely challenge them.

A similar approach could be found in [226] where Peng et al. (2011) used an age dependent HSMM-based prognostics method to predict equipment’s health. In this work, the failure rate for prognostics depended on the equipment age and equipment conditions. The method was also tested with simple rotatory components and, as their deterioration increased, the prediction of the RUL became more imprecise. Again, tuning the value of some estimators with data-based knowledge, it was possible to relatively improve the results.
An interesting variation of the previous approaches is available in [46] where Cheng et al. (2012) extended the classical HSMM models by modelling the observation as a linear mixture of non-Gaussian multi-sensor signals.

Authors developed a Multi-Sensor Mixture HSMM (MSMHSMM) which was a generalization of HSMM, modifying several parts of the algorithms followed so far but also allowing for multi-sensor fusion. This made sense, as the information provided by individual sensors could have synergies with the information provided by other sensors of the same equipment (especially dealing with vibration sensors installed in a same machine). In fact, MSMHSMM improved the results obtained by the basic HSMM.

In [297], Wang et al. (2014) developed a duration dependent HSMM, tuning failure prediction rates, improving thus the realism of the model. The estimation of RUL improved in accuracy with that decision when comparing with previous results.

Continuing with the review in [187], Expert Systems are sophisticated computer programs designed to simulate an engineer’s ability to solve or advice. They were first used for medical diagnosis. These programs use vast stores of organized knowledge, concerning a definite area of expertise. They justify its own line of reasoning and act on deductions as human experts would do. The level of expertise required make these systems difficult to implement for EHM purposes.

Finally, Fuzzy Logic (FL) systems are methods to formalize the human ability to reason approximately, and judge, under uncertain conditions. The main benefit of this method is its capability to work with approximate performance when analytical functions or numerical relations are not available. That means FL systems have the capability of dealing with complex systems that have not been tested yet or do not count with a vast amount of data available. This method needs to perform a “fuzzification” of the available crisp data, apply then several rules of decision, and perform a final “defuzzification” of the results. At the end, the achieved accuracy will be a commitment between the computational speed in producing the required output and the effort expended by the designer in formulating the rules (which can be complex and numerous). Historically, FL systems have been used to identify and isolate faulty components rather than determine the degree of deterioration in one engine, so they will not be considered either for future calculations.

2.6. - Some non-conventional approaches

Considering the plethora of techniques and methodologies that are available for performance estimation, optimization, fault diagnostics, and prognostics, it is not strange that different authors had tried in the last years to find out novel techniques (including all its possible combinations, modalities, and versions) different from all the wide available variety (partially exposed in this introduction), that could be more reliable, by evaluating pros and cons, and accounting for the associated effort.

In this sense, a non-conventional diagnostic method for gas turbine engines was presented in [304] by Zhang (2010). The author used a Back-Propagation Neural Network algorithm (BP) to select failure diagnostic parameters and it seems there was a gain in accuracy (up to a 20%) comparing with regular NN-based methods. The BP employed a propagation-adaptation cycle in two phases. Once a pattern had been applied to the input of the network, it propagated from the first layer throughout the network until generating an output. The output signal was
compared with the desired output, and an error signal was calculated for each of the outputs. The error outputs were propagated backwards, starting from the output layer to all the neurons in the hidden layer that contributed directly to the output.

However, the neurons of the hidden layer received a fraction of the total signal of error, based approximately on the portion each neuron had contributed to the original output. This process was repeated, layer by layer, until all the neurons in the network had received an error signal that described their relative contribution to the total error. As the network was trained, the neurons of the intermediate layers organized themselves in such a way that the different neurons learnt to recognize different characteristics of the input space. After training, when presented with an arbitrary input pattern that contained noise, the neurons in the hidden layer of the network could respond with an active output if the new input contained a pattern that resembled the characteristic that the individual neurons had learned to recognize during training, providing robustness to the method. Unfortunately, no references were given on the computational time required.

A kind of reliability analysis was attempted by Loboda et al. (2006) in [175]. This time Bayesian Methods, Euclidean Distance Methods and NNs were tuned and compared in terms of trustworthiness when conducting gas turbine diagnostics, resulting less susceptible to give wrong diagnostics the first and the last one. A potential combination of methods is something to be taken into consideration to leverage the strengths of each one. And one of the justifications of such combination is achieving real-time operation with the method.

In this sense, in [35] Bickford et al. (2002) commented the development of a real-time turbine engine diagnostic system to increase the safety, reliability and availability of US Air Force’s engines. Authors employed the RTEDS software (Real-Time Engine Diagnostic System) that monitors for and classifies the source and type of sensor, component, and engine faults. RTEDS employed a model of the engine to provide an accurate estimate for a real signal coming from the engine (i.e., the usual third sensor, or third channel, concept). It provided the means to integrate multiple parameter estimation methods and event detection algorithms within a common software framework. The diagnostic was compound by the combination of:

- An aerothermodynamic engine model.
- A statistical fault detection method (excess of residual error).
- A diagnostic decision manager (based on BBNs, which needed to define a network, the nodes, and the associated probability tables).

The results obtained, according with the authors, could justify the use of the method for real-time diagnostics (however, errors at detection were up to a 2% with bounded probability margins of detection, especially when multiple components were faulty), but CPU times were around 15 ms on average per observation processed (1 GHz Pentium III), by the time the paper was released, and the whole procedure relayed on the use of a standard software for the engine model.

No detailed information was given about the software architecture or about the algorithms implemented on it. In any case, the diagnostic procedure needed for a training phase to characterize the engine using historical data or data from a high-fidelity simulator, as usual. The simulations used for model training and testing were about 100-second time long and the data counted with a 0.5% of random noise.
On the topic of the operational data available, Tumer et al. (1999) explained in [286] how a recurrent problem nowadays with the Engine Health Monitoring Systems is the transferring of a huge amount of information, after a single flight (or operational cycle), to the ground personnel.

This situation is even more challenging with industrial gas turbine engines which could be working, non-stop 24/7, theoretically until the next HSE or MOH, obviously if no unexpected outages happen before. That means months of operational information (typically 16,000 to 25,000 running hours for HSE, depending on the fuel system, and to 50,000 to 70,000 running hours in between MOH). And considering the capabilities of the modern CS, just a data log of few seconds could contain several gigabytes of detailed information, not to mention if complete trends covering whole days or weeks were available.

Several commercial programs, based on expert software systems, have been started in the past years to manage this growing amount of data. The main problems that the authors mentioned in [286] were the big concentration of false alarms, the few valuable data obtained, and the expensive systems to manage complete flight data logs. Further advances have been reached by the time the paper was released, in the field of monitoring, with helicopters and even with space shuttles, employing well-known techniques such as NNs or FL. In commercial aircraft engines and industrial gas turbines, the applicability of these methods will finally depend on the ROI perceived by the end customers. Results leading to longer in-between major maintenance intervals, longer life in components, controlled deterioration of the materials, or more careful operational strategies, will surely justify the application of these methods.

To deal with big data, some authors have created original systems, like V. Verda (2004) did in [291], who introduced the concept of cost factor to determine, in almost a purely economic way, which anomaly should be fixed first. The idea behind this strategy was that anomalies that produce the same degradation in the performance of different components do not necessarily produce the same overall impact, and this could be a criterium to select the necessary information to feed the model. The suggested thermoeconomic-based approach (for diagnosing efficiency losses, i.e., aging, or progressive deterioration effects) consisted in a progressive removal of disturbances that impeded identifying where the anomalies had been originated. Several practical cases were evaluated in the paper.

Beyond original strategies like the one in [291], it seems that managing big sets of data will be less expensive every day and this will surely lead to more realistic, accurate and reliable methodologies in the next years to come. Very likely, it will be the most informed way to assess the condition of a gas turbine engine.

In the limit, when the capability of managing big sets of data was enough powerful, even physics-based approaches and mechanistic models, like the one presented by Kumar et al. (2010) in [162], could be considered for diagnostics and prognostics. These authors have evaluated the phenomenology that appears in the materials used in Hot Section parts (specifically, microcrack nucleation in Thermal Barrier Coatings protecting combustor liners, HPT first stage nozzle vanes, different turbine stages, etc.).

As expected, these authors did not include any numerical result in their work and the necessary algorithm to engage with prognostics is not clearly exposed, not in vain the studies and experiments on materials are expensive and take a long time to be completed. Anyway, it is true that with enough experimental knowledge and
data fusion capability, the resulting method could be extremely accurate, even with components which deterioration normally remains hidden, such as the CC. With enough computational power, even some sort of simplified structural stress analysis could complete the information on real-time in the future.

Finally, regarding the cost of obtaining valid data from real physical systems, when no experimental or raw data are available, it must be indicated that synthetic data could be created, as an alternative, if enough statistical parameters of the data are known. A possible way to create big databases is characterizing statistical distributions with attributes from real data (carefully when replicating a noisy distribution of data). That was attempted by Eklund (2006) in [83], where author produced a big set of synthetic data that closely matched the characteristics of the raw data, reducing this way the cost in measurements, instrumentation, or experiments. This paper proved its relevance given the scarcity of real data.

### 2.7. The management of big data sets and its potential benefit

Before the time comes in which the available computing capabilities made the methods to handle big data sets irrelevant, the techniques to filter, compress or reduce big data sets will have still margin to improve. And here is where the research process gets to ROM-based methods. The topic was presented by De Lathauwer et al. (2000) in the way it is typically used today [66], and it has been treated since then, with tensors containing millions of values, by many researchers.

In this sense, inside ETSIAE-UPM, several studies have been published using this technique and obtaining remarkable results when trying to reduce the computational time needed for solving different technical problems. As an example, De Lucas et al. (2013) developed in [67] a method to create aerodynamic data bases, inside a surrogate model to predict viscous drag in, apparently, reduced CPU times. Authors employed the method to model bi-dimensional profiles and it resulted robust enough to deal with shock waves and wide flow separation areas. The surrogate model was based on the multilinear SVD (HOSVD) of a multidimensional tensor which contained a limited amount of information (in form of snapshots taken from the model, which counted with a determined fidelity to the experimental reality) as a function of 5 design parameters. The modes of the HOSVD, in this case, could be obtained as the eigenvectors of symmetric matrixes associated to non-null eigenvalues.

For the sake of simplicity in the initial explanation of the concept, it is considered only a third order tensor $\bar{A} = A_{ijk}$, so it can be expressed (once the HOSVD has been applied to it) as the product of a new reduced tensor, $\bar{\sigma} = \sigma_{pqr}$, and three vector families known as modes of the decomposition, forming bases:

$$A_{ijk} = \sum_{p,q,r} \sigma_{pqr} U_i^p V_j^q W_k^r$$  \hspace{1cm} (2.14)

The three vector families are defined as the orthonormal eigenvectors associated with the positive eigenvalues of the positive definite, symmetric matrixes ($B^1, B^2, B^3$) defined like:

$$B^{li}_{ik} = \sum_{jk} A_{ijk} \cdot A_{ijk}$$  \hspace{1cm} (2.15)
\[ B_{jl}^2 = \sum_{lk} A_{ijk} \cdot A_{ilk} \]  
(2.16)

\[ B_{kl}^3 = \sum_{lk} A_{ijk} \cdot A_{ijl} \]  
(2.17)

Once the HOSVD modes have been calculated (and, again, they are mutually orthonormal), then the reduced tensor can be obtained this way:

\[ \sigma_{pqrs} = \sum_{i} \sum_{j} \sum_{k} A_{ijk} U_i^P V_j^q W_k^r \]  
(2.18)

This is what theory says. Now, if elements in the matrix \( A_{ijk} \) show redundancies along the different dimensions (maybe because of the physical laws governing the process) then an appropriate truncation of the expansion of \( A_{ijk} \), and a further optimization of the truncated tensor could provide a good approximation. There are some differences between MLSVD, PCA (Principal Components Analysis), meaning pure tensorial algebra as a generalization of the SVD and PCA used with matrices, and the HOSVD as it is usually known today, which implies an additional optimization process, that is introduced here, and it will be done with more detail in Chapter 4. HOSVD is the most common terminology used when dealing with these problems, generically used, but MLSVD and HOSVD are terms that will be kept differentiated in this thesis. By means of a previous top limit estimated, with the following error expression containing the high order singular values of the decomposition (\( \alpha_p, \beta_q, \gamma_r \)), the accuracy of the approximation would be (being \( s_1 < N_1, s_2 < N_2, s_3 < N_3 \), the reduced dimensions of the new compressed tensor):

\[ |\varepsilon| \leq \sqrt{\frac{\sum_{p=s_1}^{N_1} \alpha^2_p + \sum_{q=s_2+1}^{N_2} \beta^2_q + \sum_{r=s_3+1}^{N_3} \gamma^2_r}{\sum_{p=1}^{N_1} \alpha^2_p + \sum_{q=1}^{N_2} \beta^2_q + \sum_{r=1}^{N_3} \gamma^2_r}} \]  
(2.19)

Given that error value, several different combinations of modes can meet that condition. The modes are selected in an iterative way, being finally chosen the ones that show the maximum singular value. This means the truncated MLSVD provides a method to compress databases, thus, truncated MLSVD allows for storing an approximation of the original elements of the tensor. The compression increases exponentially as the dimension of the tensor increases. On the other hand, accuracy will be lower. As the new compressed tensor is not unique (this can be proved), an optimization method can be applied to obtain the best possible, for a given size (and results will be shown later in Chapter 5).

In the case of a gas turbine engine, a performance model tensor will typically count with non-less than 10 dimensions, as it will be explained later. The existence of clear single values, and the presence of redundancies is crucial to obtain an effective compression. If no parameter dominates, the method loses effectiveness. Some of the same authors developed in [33] a real-time performance improvement of engineering control units via MLSVD and applied the obtained results to SI reciprocating engine. In that work, model equations were replaced by a ROM-based on HOSVD method that provided simplified global descriptions of multidimensional databases with enough accuracy for engineering applications. This technique could be applied to two-spool turbofan engines this time to try to improve calculation times (comparing to GAs, for instance) when performing performance diagnostics and prognostics, in the sense Ribot et al. (2009) indicated in [242], where a generic
study on what exactly means both diagnostics and prognostics was done (providing an idea on how a prognostic is set up by knowing the ageing laws of components).

The applicability of the HOSVD-based methods to different technical problems was also illustrated by [176] Lorente et al. (2008), [13] Alonso et al. (2009), [12] Alonso et al. (2009), [177] Lorente et al. (2010), [22] Bache et al. (2010), [23] Bache et al. (2012), and [99] García-Magariño et al. (2016), among others. With this background in mind, it could be interesting to establish the necessary methodology to pave the way for a new GPA-based method combining NLGPA with HOSVD for modern aeroengines diagnostics, prognostics, and control.

Finally, it is necessary to clarify that the interest of the study done in this thesis is not only in regards of current conventional aircraft propulsion purposes with two-spool turbofan engines. The methodology would be focused on that application given its relevance, but it would not be limited to that in the future.

In [61] and [60], Corchero et al. (2008 and 2005, respectively) showed different and innovative types of aerothermodynamic cycles and engine modalities, for aircraft propulsion applications, where the diagnosis and optimization techniques reviewed in this chapter, either more classical or by means of the more recent ROM-based techniques, could be applied as well. Corchero et al. exposed interesting information on the following topics:

a) ICR: Inter-cooling regenerative cycle.
b) WRTC: Wave Rotor Topping Cycle.
c) CV: Constant Volume Combustor Cycle.

The impact of changes in different parameters for each cycle was there evaluated. In their work, the emphasis was put over SFC and emission levels (objective function of the parametric study). Turbomach® and Hermes® were the software tools used for the modeling and simulation. If the applicable engine thermodynamic cycle model is known (and validated), the procedure explained in the next chapters of this thesis could be applied to it. It will be just a matter of getting real readings from the instrumentation installed in the machine itself and having enough details to create the model in PROOSIS®. And this circumstance opens a promising set of future possibilities for its implementation in commercial software packages for multiple potential Customers.

In this sense, the performance, diagnostics, and prognostics are concepts that are experiencing an impressive growth in terms of interest inside the aeroderivative gas turbines world and inside Heavy-Duty gas turbines forums. For instance, Ogbonnaya et al. (2011) dealt in [220] with fault detection and condition monitoring in Heavy-Duty engines, incorporating vibrational analysis.

All these proposed solutions may be of interest for different companies like GE Digital, which is the division specifically created by General Electric to address the demands from operators and end customers regarding the processing of real time data coming from their assets (in aviation, marine, and industrial applications). The different techniques showed in this chapter could help to optimize the operation of Customers’ equipment, making sensible e informed decisions that could eventually prevent unscheduled outages or catastrophic failures. The developments in big companies like that, will be certainly a good reference in the future to know where the industry is and what can be expected.
2.8.- Conclusions

Ongoing research efforts based on the explored literature about aeroengine’s health condition topic, of which this thesis is part of, suggest that ROM-based methods, built upon the use of HOSVD, could help to improve results obtained with certain techniques by decreasing computational times without a suffering of a dramatic drop in accuracy, and by filtering (denoising) incoming signals from sensors. Nevertheless, before progressing with the HOSVD, some questions should be answered about the viability of the proposed methodology in this thesis. Questions that, as the consulted literature shows, would need of some consideration, such as:

- Is it possible to develop a method to evaluate both the regime and the degradation of a gas turbine engine, with enough accuracy, leveraging all the data produced during operation (at different flight stages), without making use of a massive a priori information, within reasonable time intervals (ideally on real time)?
- Could it be possible to improve that computational time in the method if the information associated to a previous degradation state and operating condition (close enough to the solution of the problem) was available?
- How applicable could be such method, installed in standard computers, regarding the time needed to solve the associated inverse problem?
- What level of accuracy could be achieved when solving the associated inverse problem with the data available?

Answering these questions, induced by the research that has been done during the last decades by different authors, will be part of the target for this work. The answers will help to build a solid foundation with which establishing a reference to compare further improvements in new potential techniques. Some key elements in the methodology that will be presented in the next chapters come from the careful review of the issues found by others, for instance:

- Properly scaling of both sensor measurements and H&Q parameters in the inverse problem will be of paramount importance. This basic principle in numerical analysis was not always applied in the different works reviewed. When applied, measurements are scaled by dividing with mean values only. In this work, the standard deviation of the data from each variable will be used for the scaling as well.
- Evaluation of the number of data samples simultaneously used, as required, that will help to find a solution to the engine’s inverse problem, in case the accuracy was not as good as expected (i.e., in case the associated problem’s conditioning was elevated, see Chapter 4).
- Application of fully non-linear models, including turbomachinery components’ maps, instead of just linearizing the model.
- Application of the methodology to the engine’s inverse problem with the minimum amount of a priori information from the engine (model-driven).
- Calculating not only engine degradations, also its regime, as part of the associated inverse problem.
- Potential improvements by optimizing the sensors mounted in the engine.
- Potential acceleration of the calculations, in general.
It will be necessary to understand how the management of tensors could be compatible with real-time applications in aviation, when possible. However, after the revision of the works available in the literature, it is also necessary to end up highlighting that the aim of the present study is to explore methods to improve the calculations when solving fully non-linear inverse problems relative to two-spool turbofans’ health condition, with the ultimate intention of reducing the required amount of computational time on EHM systems for real time applications. ROMs will complete such scope and will be incorporated if they finally result of any help.

According with Marinai et al. in [187] and Ntantis et al. in [213], the desired characteristics in an EHM tool, like the one to be presented in this thesis based in the application of new methods, would be:

- Being based on the most accurate and realistic non-linear available models.
- Being capable to detect small performance changes with enough accuracy.
- Being able to deal with measurement noise and sensor bias.
- Diagnosing with high accuracy, by using less readings than the number of health and quality parameters.
- Being capable for single or multiple fault isolation.
- Avoiding any smearing effect, focusing on the actual fault or faults.
- Being needless of training or tuning for the setting up parameters.
- Allowing data fusion and being able to incorporate expert knowledge.
- Reducing computational time to the point it can be executed on board.
- Being free from a lack of comprehensibility due to a “black box” behavior.
- Being suitable for on wing applications.

In principle, the proposed methodology, based on the use of GAs, SQP, adapted Newton, and ROMs could address every target in the list, including the data fusion and expert knowledge, when performing the proper combination of and techniques. To fully comply with this list, two requirements still seem to be necessary, but their analysis will fall out of the present study:

a) A proper method to process raw signals from the engine sensors must ensure that measurement uncertainties are appropriately dealt with.

b) Modern hardware to solve the inverse problem must be considered to speed up the convergence beyond what has been reported in the literature so far, as much as feasible.

The previous detailed list will be kept in mind throughout the study, to make sure the proposed methodology meets the expectations. However, in this thesis, the next techniques will be the finally chosen to address the turbofan inverse problem:

I. GAs: Given their exploratory nature, the GAs can provide valuable information about the solution space, particularly with problematic cases (i.e., noise, great degradations, etc.). Their null convergence rate will be the main issue to implement them in a real-time application.

II. Gradient-based methods: The quadratic programming will be used to speed up the performance rate achieved by the GAs. Methods based in the use of gradients will count with a super-linear rate of convergence that will contribute to reduce CPU times. However, the success of this technique
depends greatly on the shape of the OF chosen. Without some degree of convexity, the solution will be not guaranteed. Some valuable information of the surface of the OF will be obtained with this technique as the gradient is calculated for points contained in the OF’s surface (or hyper-surface).

III. Adapted Newton methods: The previous techniques will be applied to the optimization problem generated by the OF, meanwhile this one will attack directly to the inverse problem. GAs are very dependent on the population chosen to represent the solution space, and the Gradient-based methods are very dependent on the shape of the OF. Newton-like methods will be more direct, avoiding convergence traps that could affect to the gradient. It should be considerable faster if convergence conditions are met.

IV. Tensors replacing to the complete model: When the degradations expected are small, the full model can be replaced by a reduced part of it. And even the tensors generated to replace the model could be optimized by HOSVD.

After having explored the available literature on the EHM topic for aeroengines and more generic gas turbines, it is now time to deepen more on the architecture of the two-spool turbofans, the instrumentation that will be typically found installed on them, and the associated model that will be systematically used to represent their performance, in the next chapter.
3.- ENGINE CONFIGURATION, INSTRUMENTATION, AND MODEL

The first part of this third chapter will be dedicated to the description of the configuration of several existing representative two-spool high-bypass turbofan engines, like the ones introduced in the first chapter. The selected models are very popular and widely used by different airlines around the world since decades. Their architecture will be explained to reach a better understanding on the capabilities that can be expected from them, and how they can be effectively modelled. This knowledge will also contribute to assess, more precisely, which parts of these machines are typically submitted to harsher conditions, the reasons behind certain design decisions, or why the maintenance on them is more intensively focused on some specific modules. The reader will be redirected to appropriate references for further details as there is a profuse amount of available information on the topic.

The second part of the chapter will deepen into the Monitoring Systems used with these machines. Given the critical role that sensors play in the determination of the engines’ health condition, this second part will be necessary to get a full picture of the problem under consideration. An accurate, trustable, and healthy instrumentation system is essential, not only to control the engine, but also to evaluate the real deterioration on it, which is the main goal of this study. The different sensors installed in the engine do not represent an expensive part of it. The internal parts of the gas turbine engines are certainly far more costly, but the information the sensors provide is crucial to determine if the machine can be operated safely or if there is some problem limiting its expected level of performance, among other circumstances. If the readings provided by the sensors are not good enough in terms of accuracy, or speed of response, then the CS will end up working with an incorrect representation of the real physical system. The model of the engine will be fed with values provided by these sensors, so it is in the benefit of the methodology followed in this thesis counting with the most precise and reliable instrumentation system as possible when applying it to an existing aeroengine. The trends regarding the technology behind these devices will be briefly explored to illustrate the fact that the amount of information retrieved by the EHM systems will be growing apparently in the future. This circumstance will surely help to improve the degree of certainty and trustworthiness on the health condition of these machines, but that advantage comes together with the challenge of a big amount of data associated that must be managed and analyzed.

Finally, once achieved a better knowledge on the real systems, which health condition will be analyzed in the next chapter, and after having a clearer idea on the capabilities of the instrumentation that will inform on the real status of the machine, the last part of this chapter will be dedicated to the model developed in PROOSIS® to replicate (virtually) the behavior of the type of engines described in the first part of it. The assumptions, hypotheses, and equations holding that model will be exposed as in detail as possible. It is important to know that PROOSIS® is a powerful simulating software, and its TURBO® library manages around 800 equations when simulating turbofan engines, so only the most relevant relationships in it will be commented, for the sake of simplicity (enough references will be provided though). This model constitutes the principal tool used in this research. The degree of fidelity of the model to the real system will determine the quality of the results. Its validation and fit to the selected model of engine will be conveniently explained as well.
3.1. Engine configuration

The way chosen to illustrate the configuration of the type of machine under study has been recurring to existing, well-known, examples in the industry that could be used as reference for the concepts that will be explained later. In this first part of the chapter, the following engines will be briefly reviewed, as representative of the past, the present, and the future of the commercial aircraft engine industry:

a) Engines of the CF6 family.
b) Engines of the CFM56 family.
c) The GE90, the GEnx, and the GE9X.

3.1.1. The CF6 family

Regarding the first turbofan model of the list, GE launched in 1967 a large-engine program (St. Peter, 1999, [270]), that ended up becoming one of the most successful and longest-running family of engines ever developed (see Figure 19): the CF6.

![Evolution in the Series](image)

**Figure 19:** Data from the CF6 engine family, from its first series to the last one. Images and data were obtained from GE Aviation’s website [110].

The first model in the CF6 family, the CF6-6, has recently made its final flight, in 2021, after 50 years of revenue service (see related article in [167]). The CF6-6 helped to boost GE Aviation’s commercial engines division and became its first successful widebody aircraft engine program, contributing to popularize the international travels for more passengers around the world since its apparition in 1971. During its last three years of service, it was used primarily for cargo purposes.
Considering all the models in the family, the CF6 program has reached an operational record of 460 millions of flight hours since its debut (GE Aviation, [109]). The reliability of these engines can be documented by some units that have exceeded 92,000 flight hours and 34,000 flight cycles in total since their entrance to service. Another fact that will help to illustrate this point on engine's reliability with some more detail has to do with several long commercial routes covered by aircraft powered precisely with CF6 engines. In particular, the Qantas flight QFA8, from Dallas (Texas, US) to Brisbane (Australia) is one of the current world's longest non-stop commercial flights, and engines from the CF6 family are used to cover it. B747-400 aircraft are used in this route, powered by CF6 engines. The flight typically lasts over 15 hours and travels nearly 13,680 km. Certainly, the technological heritage left by the different series promoted the development of new and better engines.

The components of the first variants of CF6 (which manufacturing was initiated circa 1969), and very in particular those installed in the hot section part of the engine, were subject to demanding factory-testing programs designed to simulate more intensely take-offs, landings, and other flight phases in which there could be rapid thrust lever variations. The hot section became a major feature of the project because this part was designed, tested, redesigned, and refined at such a level that it was years ahead of the rest of engines developed by the competitors. Regarding the smoke (pollution) produced by this machine, it was considerably reduced comparing with some other models by the time the CF6-6 began to operate.

Analyzing the main components of the engine, as they are shown in Figure 20, the following considerations on its architecture could be made:
The Fan section consisted of a 38-bladed titanium single-stage fan with part-span shrouds, and with stator vanes designed for reduced noise during normal operation. The big Fan stage 1 disk was manufactured in titanium alloy (typically premium quality triple-vacuum-melted Ti-6Al-4V, see FAA [87]) meanwhile the Fan forward stator case (known as blade containing ring) was manufactured in stainless steel (find more information in Figure 21), circumstance that means a heavier setup comparing with more modern engines in which composite materials are being used for that component.

The booster (LPC) counted only with a single axial stage, but this circumstance changed in the next variants of the family, reaching finally up to 4 stages. This modification was the result of the need to optimize the performance (i.e., thrust and TSFC) to the flight conditions (required Mach number). Lower flight speeds implied more compression stages during the design phase, as well as the pursue for higher OPR did. Typically, same titanium alloy employed in the Fan was used in the booster as well.

Initially, the HPC counted with 16 stages in the CF6-6, and they were mainly manufactured in titanium-based alloys, steel, and nickel-based alloys (e.g., IN718 for the last stages in HPC given the higher temperatures reached). The design led to 14 stages finally, because of the additional stages that were added to the booster in the following variants.

The single annular combustor (SAC), counting with 30 duplex-type fuel nozzles and swirler cups, was designed to be virtually smokeless. Cobalt-based and nickel-based superalloys were used in the CC section of these engines, including the first stage of nozzle vanes in the HPT. The cobalt-based superalloys are not as resistant to mechanical stresses than the nickel-based alloys, but their thermal behavior at high temperatures for static components is better, thus being typically used in the combustor liners. These liners were protected with TBC (i.e., Thermal Barrier Coatings, built with chromium, aluminum, and yttrium, together with nickel or cobalt as strengtheners).

The 2-stage HPT featured advanced film convection cooling for blades and vanes. Rene’ 80 (R80) and other nickel-based superalloys were used in other hot section parts of the engine exposed at temperatures up to 650 °C (923 K). A 34% of finished components’ weight in the CF6 was manufactured in IN718 (El-Bagoury et al., 2007, [84]).

The LPT counted with 5 stages manufactured in nickel-based superalloys.

The maximum rotational speed was around 9,925 rpm in the high-speed spool (Core Engine), and around 3,810 rpm in the low-speed spool. The trend in the Core Engine of the future variants, once the materials in the hot section of the engines allowed to do so, was to increase the rotational speed to improve the HPC compression ratio and the overall thermal efficiency.

Engine’s accessories and controls were installed in the fan frame for better access during maintenance, facilitating their troubleshooting.

The preliminary designs of the CF6, with 142 kN of thrust, were considered not enough powerful for a twin-engine application in a large airliner. However, those engine versions could fit well in a three-engine configuration, such as the McDonell Douglas DC-10 or the Lockheed L-1011. By 1968, American Airlines and United Airlines had chosen the DC-10 and the CF6 engine. Also, by that time the most recent variants of the CF6 had increased the engine’s thrust up to 178 kN.
The search for higher thrust was permanent in the late 1960s and this was one of the aims in the next developments inside the CF6-6 series. The first run of the CF6-6D variant was performed in 1968. This engine was capable to reach 204 kN of thrust in a test run, at maximum power. The engine was released for production in 1969 and passed the certification test in 1970, completing its first flight on a DC-10 that same year. Several major changes had to be done few months before the scheduled deadline. The CF6-6D was the variant that officially entered airline service with the DC-10 in August 1971. This was certainly one of the main milestones for the engines’ family, but the developments did not stop there. One growth version, the CF6-6D1, was certified by the FAA (Federal Aviation Administration in the US) also in 1971 with a thrust rating of 182 kN. And right after, another version, 191 kN capable, was released: The CF6-6G.

That quick growth in terms of power brought several problems with the combustor liners. The life of those components averaged less than 4,000 hours until 1974, when it was introduced a new combustion module that included all the changes required, in the CC, to meet technical requirements up to the CF6-6H model. That action solved most of the CC problems, other than the expected ones relative to the normal growth in the next variants of the series. As usual, the demand perceived from the airlines’ market drove the required technical developments to come, which was basically pushing for more passengers, and therefore more thrust.

The CF6-6G represented the first major change in the CF6 series initial design. Developed under the engine Component Improvement Program (CIP), funded by the US Air Force, a new HPT and a new CC were produced.
These new components would enable to operate the engine at turbine inlet temperatures (TIT or $T_{4t}$) over 1,640 K while using about 15% less cooling air through the 2 HPT air-cooled stages. This change was accomplished thanks to major modifications in the surfaces of the blades and vanes of the turbine, consequence of a deeper knowledge on engine cooling technology.

Just as an example, the new turbines incorporated techniques to make cooling holes in the surfaces of HPT blades and vanes, which improved the heat transfer between the cooling air and the walls of the different parts. Designers also added ribs in the internal labyrinth paths of HPT blades and vanes to create turbulence in the boundary layer of within the coolant passage, increasing considerably the efficiency of the heat transfer (see Han, Dutta, Ekkad, 2013, [130]). The new turbine module was designed to be interchangeable with the existing CF6-6D turbine modules, and this meant a smart designing decision that contributed to minimize manufacturing and maintenance costs.

The CF6-6G also incorporated a new type of annular combustor, which featured a new cooling pattern to keep combustor metal temperatures inside acceptable limits, and a new type of fuel nozzle designed to inject more fuel in the CC, contributing thus to increase the operating temperatures, reaching higher thrust levels, and helping to improve in thermodynamic efficiency.

The CIP program was not the only improvement program in which the CF6 was involved. It should be also mentioned some others such as the Engine Component Improvement (ECI) program, developed together with NASA, during the late 1970s and early 1980s, aiming to get higher fuel savings and some other performance improvements (see [15], [225], [161], and [92] for further details).

New versions, accomplishing new requirements from the airlines, were progressively introduced in the series. Again, the new models followed the demand in the market, and the CF6-6K version of the engine, which featured decreased TSFC and improved performance retention and reliability, was certified in 1981. By 1984, the different versions of the CF6-6 engine had logged already over 10 millions of operational hours altogether.

In addition, the design of the CF6-6 was the reference (together with the TF39, its military counterpart) for the development of the aeroderivative gas turbine LM2500, which became the most popular gas turbine for marine applications, and one of the most widely used engine for industrial applications. It counted (and still does) with a single high-speed rotor which was aerodynamically coupled to a highly efficient power turbine (see Figure 22 for further details). The base model counted with a HPC of 16 axial stages, a SAC with 30 individually replaceable fuel nozzles, HPT of 2 stages, and a 6-stage PT. The base model delivers nowadays up to 25 MW, but the most recent models are capable to deliver up to 37 MW of power, reaching a net efficiency over 38%.

The technical capabilities of the CF6 were soon exported worldwide because, while GE was trying to get the CF6 established with the major airlines inside US, simultaneously some interest was detected from the European airline markets. The European airline consortium known as KSSU, which was compound by KLM, Swissair, SAS, and UTA airlines, was interested in an airplane that could fly from Copenhagen to New York during the summertime. The engineers decided that a 222 kN thrust capable engine would have to be used for such application to meet KSSU's requirements. The CF6 was grown once more, and KSSU agreed to purchase the McDonell Douglas DC-10-30 with the new engine variant.
That growth plan led to the CF6-50 series, another major milestone in the whole family. The new engine was designed to meet present and future airlines’ requirements, so the path for several new different model variants was initiated. The CF6-50A engine model was designed for a takeoff thrust of 218 kN at 30 °C of ambient temperature, although the engine went up to 240 kN of thrust during the first 9 hours of testing. Reduction in bypass ratio from 5.9 to 4.4 increased airflow through the Core Engine, from the CF6-6 engine’s 81 kg/s to 132 kg/s while slightly decreasing TIT.

A major change in the CF6-50 series was the introduction of two additional stages behind the single-stage booster, with no change in the turbofan’s external dimension. The engine was certified by the FAA in 1972, and the CF6-50A engine entered service in the DC-10-30 aircraft that same year. Further variants of the CF6-50 series were the 222 kN thrust capable CF6-50B, the 227 kN thrust capable CF6-50C (certified in November 1973), the CF6-50C1/E1 rated at 234 kN thrust, and the CF6-50C2/E2, certified in 1978, for the US Air Force’s KC-10 and the Boeing E-4. By 1984, the CF6-50 engine series was so extended that it had logged over 22 million operational hours in service.

The new engine variant proved its wide applicability for different airlines, and a second European consortium, ATLAS, consisting of Alitalia, Air France, Lufthansa, and Sabena, accepted to buy the DC-10-30 powered with CF6-50 engines, decision that contributed to expand the presence of this new version worldwide. This sequence of events proved the commercial success of the engine and led to the selection by Aerospatiale of the CF6-50 engine to power the new Airbus A300. This engine was also selected to power the B747.
In December 1977, GE launched a major new variant of the CF6 series, the CF6-80, and some new relevant improvements and redesigns were introduced with it. The engine reduced the weight and overall length of the CF6-50 by eliminating the turbine midframe (TMF), as shown in Figure 23, installing the bearings that were formerly inside the TMF inside the Compressor Rear Frame (CRF) instead. It was also reduced the length of the combustor and post-CDP diffuser.

![Figure 23](image)

**Figure 23**: The CF6-80 engines did not need a TMF module as the CF6-50 engines did, meaning this modification a relevant reduction in length and weight. A shorter HPT module separated the Core engine from the LPT module in the CF6-80 series instead (drawings were obtained from GE Aviation, [105], and [282], respectively).

The new engine series was designed to provide an average 4% improvement in TSFC comparing with the previous series. All the new engines had the same Fan section, which featured airfoil changes and an aftward movement of the blades’ part span shrouds but retaining the same number of blades. Titanium-based alloys used in previous models were used also now. Some improvements in the HPC included:

- Optimization of variable stator vanes (VSVs) schedules, allowing for a more accurate change of the angular settings in the first 6 HPC stages, as a function of the flight conditions and regime.
- Optimization of tip clearances in between rotor blades and stator casings, leading to higher global efficiencies.
- Reduction of air leakages in the HPC casings by a better control of mechanical tolerances during manufacturing.
- More aerodynamically efficient shapes in blades and vanes.
- Improved materials in blades, vanes, and shrouds.

These series of engines included a new combustor featuring a machined ring in place of the previous sheet-metal model. In Figure 24 these improvements are indicated for better reference. The CF6-80-A/A1 model was rated at 214 kN of thrust and the CF6-80A2/A3 reached 222 kN. In 1980, it was launched the CF6-80C program, a higher-thrust and reduced-weight design. Leveraging the design used in the CF6-50 as the baseline for a new model that would improve the TSFC, would reduce maintenance, and increase thrust above 267 kN.
The CF6-80C represented a remarkable advance in terms of reduced length and weight, improved TSFC (11 to 13% improvement over the CF6-50, and 5 to 8% over the CF6-80A), and enhanced performance retention along the time. The front fan of the engine was replaced by a new 2,362-mm-diameter fan. The engine counted with a redesigned 4-stage LPC, and with a LPT of 5 stages. In 1982 the engine reached thrust values above 276 kN in a 75-hour test. This made the CF6-80C the most powerful civil aeroengine ever produced by that time. It was flight-tested in 1984 and was finally certified by FAA in 1985.

In 1987, GE announced a new variant of the CF6-80C series which meant a defining point in creating the leadership position which has since attached, the CF6-80C2, rated at 285 kN of thrust. This engine incorporated new VSVs and Variable Bleed Valves systems (VBVs, located in between booster and HPC, used mainly to adjust the air flow in the HPC) to improve the efficiency of the engine during its different flight stages while preventing stalls by increasing the stall margin in the HPC Map when open (see examples of HPC maps in the last section of this chapter). To optimize the TSFC, parts in LPT were cooled with air coming from the Fan, controlled by valves managed from the engine control system (CS). The CF6-80C2B1F variant (powering the B747-400) was the first engine in the CF6 family using a FADEC instead of the MEC (hydro-mechanical controller) mounted so far. This upgrade implied a remarkable change in the way these engines would be managed (and potentially optimized), and in the way how the information supplied by the instrumentation would be used in the future. With such electronic control system, it was possible to install additional sensors in the engine, allowing for a more accurate control, and better performance as a result. The inherent limitations in MEC-based CS were left behind and upgrades like the ones suggested in this thesis became possible (once enough computational power was available, obviously).
The Fan counted with a single stage of 38 blades, followed by a 4-stage booster with blades and vanes mounted orthogonally to the flow (see Figure 26). The Fan module was made mainly in titanium, excepting for the steel used in the mid-Fan shaft, the aluminum used in the spinner, and the blade-containment shroud made of several layers of Kevlar around an aluminum casing. The exhaust section of the Fan counted with 80 composite exit guide vanes to improve its aerodynamic efficiency, reducing potential pressure drops in the Fan. The HPC had 14 stages, IGVs, and VSVs. All the vanes in the HPC were manufactured in steel, meanwhile blades in stages 1 to 5 were made in titanium. The rest of blades were in steel, as well as the casings, which were made in steel and thermally insulated nearby the CC. That CC counted with a film cooling system, and the latest models were totally capable to meet ICAO standards in terms of emissions by that time. The HPT counted with 2 expansion stages. This module was designed with both active and passive clearance control and René 88 was used in its production. Finally, the LPT mounted 5 stages with cambered struts in the rear frame (TRF) to decrease the flow swirl (this was known as the rear half-stage). Go to Jane's 2016, [145], for further details.

The CF6-80C2 was also the base for the LM6000 family of aeroderivative gas turbines. For this new application, the Fan was replaced by one additional LPC stage making a total of 5 axial stages, keeping the rest of the rotatory components with a very similar design. These gas turbines provide nowadays in between 45 and 58 MW of electrical power with a net efficiency above the 40% (see Figure 25).

Another notable feature introduced in the CF6-80C2 series was the commonality concept. Each variant was interchangeable between the aircraft types it powered and higher thrust ratings could be achieved by turning the engine at a faster rate to increase air flow (changing the rating plug in the Electronic Engine Control, or EEC, of the FADEC for each engine). MEC-based engines could not be modified this way before (see Lironi's article, 2006, [171] for further details).
Technologies from a variety of research and development programs (including the GE/NASA Energy Efficient Engine, or E³, program initiated in 1974, see [117]) were incorporated into the CF6-80C2 design, such as advanced cooling techniques to improve overall efficiency, the active tip clearance control, and several aerodynamic modifications in the gas path (see Figure 26).

Figure 26: Engine’s air flows in the CF6-80C2 (upper), and some features included in this new variant, such as upgraded VSVs and VBVs systems, cooling system for clearance control in the LPT, FADEC for the first time, etc. (lower). Images in the back were obtained from GE Aviation [107], and [282], respectively.
The continuous technological development in the CF6 family, given by improvements such as the low-emissions CC or the advanced materials used in the HPT module, contributed to maximize customers’ perceived value during years. In this sense, the CF6-80C2 consistently demonstrated counting with the lowest TSFC of any large commercial transport engine inside its thrust range because of the incorporation of such technological heritage from the CF6 series. It also received the FAA 180-minute Extended Range Operations approval (ETOPS or Extended Twin Engine Operations Performance Standards, intended to guarantee the flight of the aircraft, with only one engine operative, for more than one hour from an alternative airport) for the A300, A310, B747, and B767, offering increased route structuring flexibility, among other economic benefits. This engine, finally installed in the new Airbus A330-300, was the first to receive joint US FAA/European Joint Aviation Administration certification in 1993.

Commercially speaking, by 1991 GE had gained a considerable portion of the commercial large-engine market thanks to the CF6 engine family. The US Air Force also used the CF6 family for their E-4A, KC-10, and Boeing 7C-14 AMST prototype, so the presence of GE in both civil and military sectors were granted for years. The 1980s and 1990s were years of rapid development of consecutive variants of aircraft models and aircraft engines, with two main companies, Pratt & Whitney (P&W) and GE, dominating the international aircraft engine market. Rolls-Royce (RR) was a distant third with only about 13% of all orders in the large civil aircraft engine market. P&W was producing about 420 JT9D units per year while GE was producing about 300 CF6 units per year. Both companies continued to follow their own Core Engine development philosophies since 1950s. P&W successfully continued deepening in the twin-spool concept for the commercial market (leading to advanced technical solutions like the geared turbofan years later) while GE continued the development of their advanced VSV technology.

Regarding the CF6-80E1 (see [145] and [170]), this new variant constituted the ultimate series of the CF6 family. No more evolutions of the CF6 will be developed after this model. Even when the engine used as reference was the CF6-80C2, several major modifications led to a considerable increase in thrust:

- The entire Core Engine structure was strengthened with improved materials and manufacturing.
- The Fan section was redesigned, increasing its diameter, but reducing the number of titanium blades with mid-span shrouds from 38 to 34, while the chord of the blades was slightly increased. The BPR of the engine ended up around values of 5.3, ingesting air flows up to 873.6 kg/s.
- The booster was also redesigned, allowing for an increase in the air flow of a 9% and a remarkable improvement of a 12% in its pressure ratio.
- The HPC was prepared in its last stage for higher temperatures. The overall OPR of the engine reached values of 34.8 : 1.
- New high-temperature alloys (including René 88) were used in the HPT, while the cooling system was improved.
- It was also improved the cooling system in the LPT, new materials were introduced, and the airfoils counted with modified aerodynamics.
- The engine was equipped with the second generation of FADEC (called FADEC II). The so-called “thrust bumps” (additional power available in short periods of time) were introduced in these engines.
3.1.2. The CFM56 family

CFM International (CFMI) is an equally owned joint company, formally created in 1974 by Snecma Moteurs (French company that today is part of the Safran group, since 2005) and GE, that keeps a single interface with customers. This fruitful cooperation has remained during the last 50 years, and it has been extended by the two partners until 2050. By the time the company was formed, there were several serious issues regarding tariffs in between the European Community (EC) and the US that were solved in 1973 with the signature of a treaty prohibiting any tariff against US aircraft imports into EC territory. This was a historic event that shaped the aerospace industry during the next decades when the CFM56 family (shown in Figure 27) became the best-selling set of civil jet engines ever manufactured [145].

The vision that René Ravaud (Snecma’s CEO by that time) and Gerhard Neumann (GE Executive, and engineering icon of the company) had for the future of aviation almost became a commercial failure in 1979, when the first engine was certified, given the lack of interest from customers. Despite this hard beginning, 20 years later the 10,000th unit was handed over with a ceremony in Paris.

![Evolution in the Series](image)

**Figure 27:** CFM56 family, from its first series to the last one. Images and data obtained from CFMI (see [48]) and EASA (see [76], [79], and [78]), respectively.

In this common effort, Snecma took ownership of the design and production of the Fan and booster module, the LPT, and the accessory gearbox. The final installation of components was also managed by the French partner. GE was responsible for the Core Engine, fuel control and the design for the components’ integration. This workload distribution has remained almost untouched until today. There are assembly lines in both countries to deliver units to Boeing, sited in Washington, and Airbus, sited in Hamburg and Toulouse.
The pioneering series of the family, the CFM56-2, obtained the certification for its first engine model in 1979 under both FAR and JAR. The high level of robustness and resistance reached by these first engines of the family can be documented by some units exceeding the 60,000 operating hours and the 25,000 cycles, without visiting the workshop even after more than 10,000 hours. The first models of this series were dedicated almost entirely to military applications (i.e., tanking, reconnaissance, etc.), excepting for the DC-8 aircraft, which was powered with CFM56-2C engines, used for cargo purposes. In this sense, the USAF remains as the largest CFMI's customer today.

The next series, the CFM56-3, developed with a smaller fan and improved payload and range capabilities, was specifically conceived to be mounted in the B737-300, and flew for the first time in 1982 (see CFMI's training material [50] for further details on this engine). The commercial success of the aircraft helped greatly to boost the expansion of the CFM56 family during those years. All the models in this series received the ETOPS qualification, helping to optimize commercial routes, avoiding unnecessary fuel consumption, and certifying the reliability of the engines for transoceanic flights. The reliability reached new records within this series with some CFM56-3 units that were operated without any visit to the maintenance shop after 30,000 hours, and very few serious incidents reported. Those results contributed to the good image, in terms of safety and reliability, that Customers have kept about CFMI.

In 1984 the next series in the family, the CFM56-5, was launched with the clear target of powering the family of single-seated aircraft Airbus A320, obtaining certification in 1987. The thrust ratings are varied, from 96 kN to 151 kN, as this series was used not only in the A320, but also in the A321, A319, and A318. The size of the fan outer case came back to the dimension of the first series, improving the aerodynamic design in all the rotatory components of the engine. The efficiency was improved by a better control of the clearance in between rotor blades and casings thanks to a new FADEC (see Lufthansa, [181]), which was used in this series for the first time. The high reliability of these engines granted them the qualification for 120-minute ETOPS. From an emissions standpoint, the CFM56-5B incorporated the Double Annular Combustor (DAC), as an option (see CFMI, [52]), aiming to abate the NOx emissions levels by more than a 45% comparing with the equivalent SAC versions. Unfortunately, this change implied the apparition of cracks in the TRF of the units, so a new frame had to be designed and installed since 1997. Even so, CFMI had in the CFM56-5B an engine that counted with less hot section components than the competitors (1 HPT stage versus the usual 2 HPT stages in other models), with a subsequent shorter Shop-Visit Rate (SVR), and a cost per visit considerably lower. The efficiency of these engines was not as good as the one in engines with more expansion stages, but CFMI dedicated resources to improve the performance retention (with better coatings in the HPT), and the engine was shorter and lighter than the competitors', decisive factors to reduce TSFC. The Core Engine nozzle known as “chevron” (i.e., with sawtooth pattern, Mohamed Hussein, 2016, [199]) was used in this series to abate noise levels 10 EPNdB below values stablished by ICAO in Chapter 3 Noise Standard contained in its Annex 16 (standards also known as Stage 3, active since 1977 to 2006, while currently is the Stage 5 the one active, described in the Chapter 14 [142], for which the CFM56-5 engines would be also still compliant, as per EASA [75]). The continuous improvement was a constant in this series. The architecture of the CFM56-5B is schematically shown in Figure 28.
Finally, the CFM56-7 series was launched (see [206] for more details on this engine), being tested on a flight in 1996, and jointly certified by FAA and EASA in 2006. The engine was configured with a new kind of snubber-less wide-chord Fan blades (i.e., without part span shrouds) counting with forward-swept solid titanium blades, with a strengthened containing ring, keeping the same Core Engine than the immediately previous engine versions had, and with a new FADEC version (FADEC III). The changes introduced improved the performance comparing with previous series. TAPS combustor technology was used in these engines. The TSFC was decreased an 8% and the EGT margin increased around 50 K comparing with the CFM56-3 series models. FAA assessed the 180-minute ETOPS for the B737 aircraft powered by these engines. The natural successor of this series would be the LEAP engine, designed to power the A320neo, COMAC C919, and the B737 MAX. Figure 29 shows the evolution of the blades in the Fan section, from the use of solid titanium part-span shrouded blades in the first series to the 3D composite blades with protected leading edge (by a metallic sheath) installed in the LEAP. The chord of the successive blade models has been increased in length. The use of composites (thermoplastic resins) has meant less weight in the Fan disk, given the reduction of stress associated to the use of lighter materials. The weight savings cascade through other parts of the engine for a total of 454 kg weight reduction per aircraft. The LEAP blade’s swept geometry contributes also to mitigate the engine noise helping thus to meet ICAO standards in this regard [100].

Figure 28: CFM56-5B schematic showing its architecture. Images and data obtained from CFMI (see [181] and [52] for further details).
During the last five decades, several remarkable technological advances have been introduced in the CFM56 family contributing to the continuous improvement in performance, maintenance, and reliability of these engines (see [145], and [51] for further reference):

- The continuous developments carried out by Snecma in the Fan section installing, before the LEAP, wide-chord forward-swept hollowed titanium blades without part span shrouds (or snubbers, also in titanium with tungsten-carbide coating) proved to increase the air flow through the Fan in a 2% while keeping same efficiency levels during the most power-demanding stages of flight. The test results showed an improvement in thrust and TSFC. There has been a continuous decrease in the number of blades in this section, from 44 blades in the CFM56-2, to 24 in the CFM56-7 (the LEAP ended up with 18 blades). Blades in the Fan section count with anti-friction plasma coating in the root faces of Cu-Ni-In, and a layer of cured molybdenum-base film varnish acting as lubricant. The booster,
manufactured in titanium, was modified as well with several redesigns of 3 and 4 stages of 76 blades each. The Fan shaft was made in a steel alloy forging. The CFMI LEAP incorporated finally 3D woven RTM technology, in blades and casing, reducing weight in the Fan section.

- The HPC module has been redesigned by GE, introducing forward-swept rotors, bowed vanes, the “blisk” technology (blades and disk of one rotor stage manufactured as one single piece) in the first 2 stages of the rotor, and manufacturing the casings with a new surface treatment. There were 3 stages manufactured in titanium alloys, while the other 6 were made in steel. The first 4 stages counted with VSVs. The HPC rotor shaft was made in titanium alloy. The stage 1-3 spool was manufactured in titanium alloy, meanwhile the stage 4-9 spool was made in nickel alloy given the high temperature in that zone. The tips of the blades counted with coatings of tungsten-carbide to prevent erosion. The roots of the blades had an Al-Br coating on the mating faces. The rear stator casings were machined from Zn-Ni-Co alloys forging.

- The CC section, counting with 20 fuel nozzles, was upgraded with the installation of twin-annular pre-swirl combustion technology (TAPS). Previous versions of the CC section in the family (DAC, see Figure 30) were capable to improve the initial NOX emission levels in the family (but worsening simultaneously the CO produced, as the combustion was leaner, keeping low flame temperatures) with the incorporation of the “staging” technology, allowing to get an accurate control over the combustion by adding extra valves to the fuel system. TAPS technology was capable to abate all pollutants produced by the engine down to levels inside ICAO’s Committee on Aviation Environmental Protection (CAEP) standards, established in 2004 (i.e., NOX was reduced in an outstanding 62% comparing with the first models of the CFM56-5). More restrictive standards have motivated the development of more sophisticated lean combustion systems (e.g., TAPS II, with an improved premixing system).

- The single-stage HPT was improved to avoid aerodynamic interference losses in between HPT and LPT. It was suggested to develop counter-rotating spools to improve the combined efficiency in between HPT and LPT, but this upgrade was not finally incorporated to the family. Several different nickel-based alloys were used in this section. The CFMI LEAP incorporated Ceramic Matrix Composites (CMC) for some components (turbine shrouds [275]), and 1 additional stage to the HPT.

- Several aerodynamic improvements were done in the LPT section. The intention was to reduce the number of airfoils more than a 20% in the future. The LPT section of these engines counted with four stages, excepting the CFM56-5C counting with five, manufactured with shrouded tips. The LPT casings were made of Ni-Cr alloy. The LPT shaft was made of steel alloy. The CFMI LEAP added extra expansion stages to the LPT.

- The rotational speed ranges in the low-speed spool have varied somehow in the different series (between 4,800 rpm and 5,380 rpm, decreasing to 3,894 rpm in the LEAP), meanwhile in the high-speed spool (15,183 rpm) it has remained constant until the LEAP (high-speed spool at 19,400-21,100 rpm). The OPR grew from 30 : 1 to more than 40 : 1 in the LEAP.
Figure 30: Combustion Chambers used in the different series of the CFM56 family. Emission abatement levels are referred to CAEP/6 standards established in 2004 by ICAO. Images and data obtained from CFMI’s website [49], [238], [193], [88], and [201]. A good review on the low emissions combustion subject is available in [174].

From efficiency and maintenance improvement perspective [145], CFMI has launched several retrofitting programs based on the developments in the most recent models like the Time-On-Wing (TOW) program, in 2002, intended to introduce important improvements in the Core Engine, such as new HPC blades designed with 3D technology, new materials in the HPT blades, vanes and shrouds, new protective coatings, and cooling systems. As a result, the engines equipped with these upgrades were capable to be more efficient, reducing the TSFC in a 1%. The airlines operating CFMI engines immediately bought these upgrades through deals reaching values of 300 million USD with companies like Southwest Airlines. This illustrates the interest of the airlines in improving the efficiency of their engines. In 2007, similar upgrades were offered to retrofit the CFM56-3 models, with already 2 decades of existence in the industry, improving the EGT margin (i.e., margin between the EGT reached during operation and the maximum EGT value allowed, so the wider is this margin, the more service life is saved in the engine) and contributing to reduce the TSFC in a 1.6%. For the CFM56-5 models, the Tech Insertion Program was launched in 2004, including improvements in the HPC, a redesigned CC and new blades for HPT and LPT. With these upgrades the TSFC could be reduced as much as a 0.8%, extending the average time on wing of the engines. With the CFM56-7 series, the Engine Management Optimization (EMO) program was offered to customers that operated B737NG aircraft for combining engines between airplanes to reduce the visits to the shop. CFMI also restores thermal barrier coatings (TBC) in turbine blades and vanes [114], with Electron Beam Physical Vapor Deposition (EB-PVD) and Plasma Spray (PS) (see [256] and [281]).
3.1.3. The GE90, the GEnx, and the GE9X

One of the most iconic aircraft engines ever produced is, without any doubt, the GE90. In the early 1990s, GE developed the GE90 turbofan engine to power the large twin-engine Boeing 777 with the clear intention of setting new standards of high thrust (for wide-body aircrafts in the range between 333.6 kN and 445 kN), reduced TSFC (10% of reduction comparing with some previous models) and environmental acceptance (remarkable noise reductions and around 33% lower emissions produced comparing with previous engines) [145]. The first tests took place in 1993, the baseline engine was certified for the B777 (there was an exclusivity deal for this aircraft between GE and Boeing) in February of 1995, and finally the first commercial flight with British Airways happened in November of that same year. The GE90 was designed paying special attention to maintainability (Szecskay, 1994, [279]) and overall system reliability. Many of the components installed in the engine were derived from proven previous designs (see GE Aviation [108]), providing this derivative heritage a sound foundation for the whole machine. Snecma, Avio Aero, and IHI Corporation were revenue sharing partners of the project.

![Architecture of the GE90-94B](image)

**Figure 31:** Architecture of the most powerful model of the GE90 series, until the arrival of the GE90-115. The images and data were obtained from GE Aviation’s website and EASA (see [108] and [80]).

The evolution to the GE90-115B meant a substantial engineering effort to meet the propulsion requirements for the B777-200LR and the B777-300ER. The targeted thrust was 512 kN, the obtained thrust during testing phase was 568.95 kN, stablishing a new World Record by that time. The GE90 counts with the architecture that is shown in Figure 31, which modules are described below together with some of the modifications that were required in the new variant to achieve higher thrust levels:
• The Fan section, the largest in service until the arrival of the GE9X, has a single stage and a diameter of 3,124 mm. The Fan stage mounts 22 shroudless wide-chord blades, made of composite material (graphite and epoxy resin, with a polyurethane coating in the concave side) protected with a replaceable titanium sheath in the leading edge. The total weight of the Fan blades is 338.8 kg. The design of the whole Fan module was intended to achieve good performance retention. The confining ring is made in aluminum. During the take-off, the Fan section ingests around 1,360 kg/s of air, reaching BPR in between 8.1 and 8.7. The evolution to the GE90-115B implied a larger Fan section (3,256 mm of diameter), and forward-swept blades with a greater chord in their mid-section (blades were 6 kg heavier), manufactured in IM7/8551-7 composite material (see Kedward, 1997, [153]). The new Fan section ingests 1,641 kg/s of air, while rotating at a maximum speed of 2,550 rpm. The BPR is 7.1.

• The booster to increase Core Engine pressure ratios counts with 3 stages right after the Fan section. The booster was modified in the growth process to the GE90-115B, increasing the number of stages up to 4.

• The HPC was in part developed within the program E² (Energy Efficient Engine, between GE and NASA). The resulting module counted with 10 stages initially, although it was considered for future versions to remove one stage obtaining a shorter and lighter engine. The HPC has VSVs system in the first five vane stages. The OPR reached by the engine varies between 34 : 1 and 40 : 1, depending on the specific variant. The growth to GE90-115B implied removing the VSVs system in the stage 4. The maximum rotational speed is 10,850 rpm and the OPR reached is 42 : 1.

• The CC has a DAC configuration aiming to get a lean combustion to abate the NOx levels and to reduce TSFC. The fuel is injected through 30 nozzles, following a staging system. During the take-off and cruise, the Pilot and Main annulus of fuel nozzles inject fuel, but at low-power operations just the Pilot ring injects fuel. This accurate control over the fuel flow is possible thanks to the different staging valves that are managed by the advanced CS of the engine.

• The HPT is configured with 2 expansion stages. The cast mono-crystal blades are made with N5 superalloy. The cooling holes are laser-drilled. Blade disks are made with René R88DT. The casings in the HPT are cooled with active clearance control to mitigate the loss of efficiency associated to the tip clearance (not only the casings in the LPT were cooled). The growth to GE90-115B meant a redesign of the aerodynamics in the HPT.

• The LPT module counts with 6 stages of increasing diameter. The rotor blades, made in titanium alloys, counts with transpiration cooling technology. The casings are also cooled with active clearance control, as the HPT casings. Avio Aero took ownership of most of the components in the LPT module. The GE90-115B counts with LPT stages of lower solidity (i.e., the ratio of the chord of the airfoils in one stage and the space between two consecutive airfoils of that same stage so, the larger the solidity, the higher “metal density” is in that stage). The shaft transferring power from the LPT to the Fan had to be strengthened to handle the additional torque required.
The GE90-115B also counts with an aeroderivative gas turbine version for industrial applications: the LM9000. This gas turbine produces up to 73.5 MW of electrical power and is one of the most efficient models available today, reaching a 44% in simple cycle and an 80% in combined cycle (See Figure 32).

![LM9000: Aeroderivative Gas Turbine from GE90-115B](image)

- Long maintenance intervals and modular design for a 24-hour engine replacement.
- Fan stage replaced by an additional stage in the LPC (booster).
- Clearance control system.
- Dry Low Emissions (DLE) technology available for NOx/CO emissions control.
- GE Aviation heritage.

<table>
<thead>
<tr>
<th>Maximum Power Output</th>
<th>73.5 MW</th>
</tr>
</thead>
<tbody>
<tr>
<td>Thermal Efficiency</td>
<td>44%</td>
</tr>
<tr>
<td>Exhaust Temperature</td>
<td>495 °C</td>
</tr>
</tbody>
</table>

Values at ISO conditions

Figure 32: LM9000 aeroderivative gas turbine. Image and data were obtained from Baker Hughes (see [26]).

The next engine to be commented, the GEnx, is an engine derived from the GE90-94B specifically conceived to power the Boeing B787 Dreamliner and the B747-8 (this last one developed for cargo purposes [145]). The designing phase of the program started in 2003, being the engine tested in 2006, and flying finally for the first time in a commercial airliner in 2010. Excepting the Fan section, which was completely new, the engine kept initially many of the ideas and components from its GE90 series predecessor. Recent versions have incorporated the use of CMC in both CC and HPT. The LPT blades were also improved by using TiAl (i.e., titanium aluminide) in the last 2 stages, produced with additive manufacturing. Several companies like Avio Aero, IHI Corporation, GKN, and Techspace Aero (part of the Safran group nowadays) partnered with GE Aviation for the development of the program. The thrust of the different versions has varied typically from the 236.6 kN in the GEnx-1B54 to the 333.0 kN in the GEnx-1A75. This last engine variant was developed for the Airbus A350XWB-800 aircraft. Recently, the GEnx-1B78/P2 was rated at 357-6 kN [77]. The main components of the engine are commented below (see Figure 33):

- The Fan section counts with 18 wide-chord blades made with carbon-fiber composite material. The leading edges are protected with a titanium cover. The main Fan case is also made in composite material (first time in history for a commercial aircraft engine [113]), meanwhile the aft case is made in aluminum alloy. The diameter of the Fan is 2,819.4 mm long and the BPR in some variants clearly reaches values of 9.0 during the take-off stage. The GEnx-1B54 variant reaches 9.6 in that same flight stage.
- The booster counts with 4 stages. Some variant model counts with only 3 low compression stages.
- The HPC kept the 10 stages of the less powerful versions of the GE90, but with smaller airfoils. There are three blisk rotor stages and there are VSVs
only in the first 3 stages. The OPR varies typically, depending on the model between 36 : 1 and 47 : 1 during take-off.

- The CC makes use of the TAPS technology.
- The HPT of 2 stages is manufactured using powder-metallurgy in the turbine disks [239].
- The LPT counts with 7 stages, excepting some variant with only 6 stages of increasing diameter. The low-speed spool rotates (at a maximum of 2,778 rpm) in the opposite direction to the high-speed spool (rotating at a maximum of 13,400 rpm) to avoid swirl losses.
- The control system is a FADEC III.

Figure 33: Cutaway schematic of a GE9x. Images and data were obtained from GE Aviation’s website (see [113]).

The successor of the GE90-115B as the most powerful civil aircraft engine ever manufactured is the GE9X, which was developed to power the B777-8 and B777-9 aircraft. The program of this engine was initiated in 2008, its first flight took place in 2018, and it was finally certified in 2020. It is expected the first B777-9 powered with this new engine model will be delivered in 2023 [21]. The Figure 34 is a frontal view of the aircraft showing the relative size of the GE9X:

Figure 34: B777-9 frontal view. Image and data were obtained from Boeing’s website (see [37]).
As it happened with the GE90, there are some companies contributing to its production together with GE Aviation (i.e., Safran, IHI Corporation, MTU and Avio Aero). This model counts with several remarkable improvements such as the incorporation of approximately 300 additively manufactured parts, OPR of 60 : 1, the use of CMC in 5 different components of the engine (such as the combustor inner and outer liners, the HPT first stage and second stage nozzle vanes, or the HPT first stage shrouds, and this use is expected to be even more generalized in future versions), or the implementation of the TAPS III combustion technology, among others [20]. Figure 35 summarizes these features:

Figure 35: Cutaway schematic of a GE9X, indicating some of the main features of the engine. The images and data were obtained from GE Aviation’s website (see [111]).

One doubt that naturally arises at the light of the new technologies and materials used in new engine models is how these implementations will affect to the future engine diagnostics, prognostics, and control optimization systems. The development of new features in the components that will be part of the engines and the design of improved engine systems will imply the need of understanding how they will evolve during regular operation cycles. Certainly:

- The CMC will not degrade the same way a Ni-Co superalloy would do.
- Rotational speeds are increasing in the Core Engine, and HPT-LPT counter-rotating configurations are now more common than years ago.
- The modern Fan blades count with new metallic sheaths and plastic coatings to protect them against potential harsh ambient conditions, so their degradation under specific operations is still to be fully evaluated.
• The next combustion systems will work at higher pressure ratios (Renyu et al., 2011, [240]), with lower cooling flows bled from the HPC, and with sophisticated staged fuel tuning systems working at different flight phases, optimizing thus the TFC and the emissions (see Figure 36).

• The CS managing the engines are being progressively faster and capable of controlling more complex engine systems simultaneously, including clearance control, variable geometry, accurate fuel supply, etc. More different operational modes will be implemented in the logic uploaded in these systems to optimize the operation at each flight phase (more idle modes, additional cruise modes, different thrust ratings, etc.).

• Similar considerations would be applicable to the aeroderivative gas turbines, which designs typically leverage the ones already proved in aviation, being subject to specific requirements to be met from different environmental authorities, electrical grid regulators, etc.

Figure 36: Evolution of the TAPS combustion technology in the last years. All these upgrades will have to be accurately included into the different engine health condition methodologies (images and data from [193], [258], [49], and [103], for further details).

And these are just few examples of the implications that these improvements could mean for the future engine theoretical models and health condition methodologies, which will have to evolve accordingly, by incorporating the knowledge associated to all these new upgrades, aiming to provide the most faithful and realistic representation of the sophisticated physical system that an aircraft engine eventually is. This leads to the conclusion that there is still a long way to go in this field and the need of adapting, refining, or even redesigning these methodologies will continue in the future.
3.2.- Monitoring System and Instrumentation

3.2.1.- Preliminary concepts

In the present work, it is considered that the two-spool turbofan engines under study count with a monitoring system (MS), integrated in the FADEC, that is formed by the instrumentation installed in the machine, and by the EEC. Its mission is to receive data from sensors, to process it in the EEC, and finally to provide information to the operator about engine’s main parameters (pressures, temperatures, etc.). The EEC manages electronically the engine, completing the control loop “decision-action-feedback”. Some authors call MS to some separate CS module, counting not only with monitoring features but also with prognostics or optimization capabilities, that could add functionalities to the CS. That nomenclature will not be followed here, and the main function considered for MSs will be measuring engine parameters.

Sensors installed in gas turbine engines are devices capable to convert the physical properties they detect in the gas path, in the lubricating oil, or from the mechanical behavior of the machine (i.e., vibrations, rotational speed, etc.), into measurable magnitudes such as electrical current or voltage. The design, manufacturing, and location of these devices are based on the physical properties they will have to detect and convert into some manageable electrical magnitude. They are distributed throughout the main engine modules, they are typically exposed to harsh conditions, sometimes immersed in the hot and pressurized flow of the gas path, and they are also the only source of information, on real-time during operation, coming directly from the engine. The total number and location of sensors are model-specific characteristics. In this sense, the optimum amount and type of such instruments to be installed in an engine is a topic profusely discussed in the specialized literature during the last decades, as indicated in Chapter 2, and it will be subject of new research, and re-evaluation, for the new engine models, given the new technologies incorporated to the instrumentation (examples of new developments on engine sensors are given in NASA [301], Parker [42], etc.).

The other main component of the MS, the EEC, counts with electronic hardware powerful enough to log constantly data, getting samples from the machine during its operation through readings coming from the sensors. It uses that information to inform about the condition of the engine, detecting simultaneously potential faults, when applicable. The newest engine models installed in the latest-generation of aircraft count with FADECs which produce data in the order of ~25 MB, per flight-hour, and per engine, meaning around 4 millions of data strings to be allocated in memory storage modules, per engine, and per revenue service flight.

A part of the “snapshots” captured from sensors are transmitted during a flight [19]. In this sense, the associated costs of transmitting data are still high enough to be preferable waiting until the next landing to download the information retrieved for its post-processing. OEMs manage both real-time and post-flight collections of data nowadays. Big data sets, from an increasing number of parameters, are managed in modern aircraft, at higher frequencies (128 snapshots per second [202]), and using wider transmission lines (increasing capabilities in the bus infrastructures like ARINC 429 and ARINC 717, with more than 4,000 data points logged in parallel per engine). The logged data can be used for other purposes different to control, or diagnostics, if the CS on board counts with additional features, coded in its logic, to manage the information.
That extra capability will depend on the processing power of the CS. However, in aviation (and in other critical applications) it is usually sacrificed the computational power to count with high reliability levels and solid trustworthiness in the hardware, for operational safety reasons. The integrity of the CS is critical, so the OEMs equip their engines with highly reliable processors, aiming to provide, in time and manner, trustable data to pilots, operators, and MRO technicians. Reliability rates are in the order of less than $1 \cdot 10^{-6}$ computing failures per operating hour for EECs, according with JAR 25, see Schwamm, 1997, [257], and Hjelmgren, 1998, [134]. FADECs are therefore subject to the strictest requirements from the Aviation Authorities, which are common for all OEMs. It will not be installed hardware in one aircraft engine that could lead to low reliability levels or to a more problematic operation (or maintenance), not accepting lack of testing or lack of accumulated experience. Even so, still a 50% of the causes for inflight shutdowns (IFSD) and rejected takeoffs (RTO) suffered in CF6 and CFM56 engines were in the past related to issues with elements of their external configuration, such as sensors and associated electrical wirings (see statistics from 1994 in [279]), so both sensors and EEC are critical to guarantee the operational safety of the machine.

Although the computational capabilities from standard desktop computers typically overtake those available in EECs, the sustained progress achieved with microprocessors installed in aircraft engines during the last decades have allowed MS to retrieve more data every day. Such amount of information that OEMs work currently with “digital twins”, which are faithful digital versions of real engines that combine specific analytics, physics, and data science in SW platforms like Predix® [118], a cloud-based application, and operating system, launched in 2014 by GE Digital, that is utilized to create those digital replicas and to manage the information associated to them (see Figure 37). Predix® is fed constantly with diverse data from the engines that are being monitored, and results obtained from subsequent performed analysis are supplied to operators, MRO technicians, or parts’ brokers, aiming to speed up troubleshootings, to minimize the time an engine remains stopped, or even to optimize its operation during a flight. P&W or RR, count with similar SW platforms (i.e., P&W developed ADEM and eFAST for advanced diagnostics and data acquisition). From a historical perspective, NASA was who pioneered the use of digital twins, applying them for space exploration [128].

---

**Figure 37:** GE Digital’s Predix® platform, high-level schematic layout, showing data flows, from different inputs to the results and analysis presented in applications to end customers (images and data from GE Digital, see [118]).
So, the current trend in MSs consists in managing more data retrieved from the engines, applying new technologies to existing sensors, and to the virtual replication of systems that work interconnected into the so-called Internet of Things (IoT). However, the appealing scenario of counting with more data from more sensors also implies a challenge when analyzing such amount of information. The connectivity to digital platforms in the IoT is also a desirable feature, but it cannot be taken for granted during a full flight, depending on the route. Now, the two main components in the MS will be commented: Sensors and electronic control.

3.2.2. - Engine Sensors

Two of the main desired characteristics in sensors, beyond weight and size considerations in aviation, are probably accuracy and reliability. It is paramount to count with a sufficient degree of accuracy, maintained along the time, in the measurements taken from the engine, ideally over the full measurement range of the different sensors (this last point is not always possible, given the inherent properties and limitations of the materials used in their manufacturing). The ability to determine the evolution of engine health status almost totally depends on it, as the accuracy is the quality relative to providing readings close to the real value. It will make no sense looking for small performance changes, with any methodology, if the instrumentation on board is just providing rough readings during operation. Precision is also a very desirable quality in the case of gas turbine engines’ instrumentation (see Figure 38), because these machines are supposed to operate during extended periods of time under similar conditions (i.e., cruise or base load), experiencing a slight degradation with time, so very similar readings will be observed. The lowest dispersion, the best traceability of such slow degradation.

Measurement errors are usually compound by two distinct terms, one of which is random, meanwhile the other one is constant [62]. The random error considers the difference in between repeated measurements of the same item. This can be described as instrument non-repeatability or precision error and can be of same order of magnitude as changes induced by a real engine fault. The fixed error is called the sensor bias. Sensor failures can be categorized as either catastrophic failures (easy to detect) or soft failures (uneasy to detect). Soft failures typically will not degrade the engine performance for some time but, if it is left indefinitely, then it can eventually cause catastrophic results. Undetected errors always mean a risk. Sensors are also often affected by some degree of noise, bias, or lack of calibration, masking the true condition of the engine, and leading to incorrect computations. This may lead to sensors’ reliability being lower than components’ reliability.

![Figure 38: Difference between accuracy and precision in sensor readings.](image-url)
In this sense, it will be assumed in this work that the instrumentation installed in the engine under study is in good condition, properly calibrated, and counting with signal conditioning technologies incorporated to mitigate the possible noise that could affect to the measurements taken during operation. The target of the thesis has to do with the management of the data obtained from the engine.

Sensors are parts of the engine, physically, so they are also exposed to the same harsh conditions during operation [31]. That is the reason why the thermocouples (TC) used to measure temperatures in the hot section of the engine are typically manufactured using superalloys like René N5, Haynes 188, or Inconel-600, among others [287], that are typically used as well in the manufacturing of the HPT and LPT modules. Some sensors are integrated with other components, like the position feedback sensors installed in the VG actuators (i.e., VBVs and VSVs Systems) or in the flow control valves (i.e., active clearance control valves), which count with Linear Variable Displacement Transducers (LVDTs) and with position switches. These last two types of sensors are essential to control the air flow in the engine, preventing unsafe operating conditions, like stalls. Sensor signals managed by the CS are usually relative to thermodynamic variables, such as pressures and temperatures at different parts of the gas path, which depend on flight conditions and operating regime. So, any value measured by a given sensor is meaningless, unless flight conditions and regime of the engine are provided with it.

The aforementioned new technologies applied to the sensors will modify the design, manufacturing, and maintenance of these components, looking for higher reliability under harsh conditions, and more affordable production. New physical principles (e.g., optics-related) applied to the instrumentation could avoid the immersion of sensors in the gas path [295], contributing thus to improve their survivability, reducing distortions in the fluid, and getting faster responses from transducers that will not be affected by the TBC protecting its surface. New optical sensors, like the ones suggested to measure tip clearances in between rotor blades and casings in turbines [72], could be of applicability in the future. Additionally, the use of additive manufacturing for sensor housings, has facilitated the design of parts with unique geometries that were impossible to create using traditional machining. Additive manufactured components reduce part count by replacing assemblies with single parts, which are also lighter than previous designs, increasing engine's efficiency. The GE90-94B was the first civil aircraft engine incorporating parts produced this way (in 1995 [5]). The relevance of this topic in the industry can be documented by international initiatives, promoted by different Administrations, like the European STARGATE project (Sensors Towards Advanced Monitoring and Control of Gas Turbine Engines) which has established a common framework, together with OEMs, to develop sensors that could overcome current limitations in terms of temperature, accuracy, stability or degradation [85].

In this study, the intention is to use the information coming from the existing standard MS, to solve the inverse problem of the engine health condition through the model-based methodology that will be detailed in Chapter 4. The evaluation of the computed degradations, with the data retrieved from the instrumentation and the results obtained from the numerical methodology, it will be possible to complete the diagnostic analysis of the engine, as well as the subsequent prognostic estimation and regime determination (i.e., TIT value). Diagnostics and performance are directly interrelated. Prognostics will come later once the time enters in the analysis of the health condition of the machine. Sensors will be essential in any case.
The following list summarizes the main types of sensors used in aeroengines like the ones under study (see Figure 39), aiming to illustrate the current available instrumentation capabilities. Not all of them will be included into the model of the engine (sensors’ sets can be optimized), given the structure of the inverse problem that will be solved, and the amount of information required to calculate the representative H&Q parameters of the engine. So, there is still margin for integration and data fusion, adding for instance the vibration sensors in the future, that certainly will contribute to have a clearer picture of the health condition of the engine:

- **Engine gas path instrumentation (see Table 4 for better reference):** This is the subset of sensors covering the main gas path’s pressures and temperatures (at the main different turbomachinery stations of the engine), shaft speeds in both spools, and the fuel flow supplied to the CC. Depending on the manufacturing year and engine’s architecture, this subset can range typically from few transducers to as many as 11 or 12. The readings provided by these sensors are complemented by measurements from the ambient and flight conditions, such as inlet temperature, ambient pressure, or Mach number, which will define the current flight condition, and will help to normalize the gas path parameters, when required. These ambient measurements are obtained by a different MS installed in the aircraft. This full group of sensors will be the ones considered in Chapter 4 for the different calculations performed.
<table>
<thead>
<tr>
<th>Sensor Type</th>
<th>Practical Examples</th>
<th>Typical Dimensions and Location</th>
<th>Typical Technical Specifications</th>
</tr>
</thead>
<tbody>
<tr>
<td>Temperature Sensor (TC)</td>
<td>- Type K thermocouple (TC).</td>
<td>- Sensors used typically to measure TIT or EGT.</td>
<td>- To measure T3 similar TC sensors are used.</td>
</tr>
<tr>
<td></td>
<td>- Total Length: 180 mm.</td>
<td>- Temperatures up to 1,400 °C.</td>
<td>- Body built in Ni-based alloys with CMC sheath.</td>
</tr>
<tr>
<td></td>
<td>- Probe Width: 7 mm.</td>
<td>- Wires made in Chromel – Alumel.</td>
<td>- Chromel (90% Ni, 10% Cr).</td>
</tr>
<tr>
<td></td>
<td>- Probe length: 85 mm.</td>
<td>- Alumel (95% Ni, 26Mo, 2% Al, 1% Si).</td>
<td>-</td>
</tr>
<tr>
<td></td>
<td>- Cylindrical body.</td>
<td>- Several installed in a same stage.</td>
<td>-</td>
</tr>
<tr>
<td>Temperature Sensor (TC Harness)</td>
<td>- Type K thermocouple (TC).</td>
<td>- Sensors used typically to measure EGT.</td>
<td>-</td>
</tr>
<tr>
<td></td>
<td>- Harnesses are engine specific.</td>
<td>- Temperatures range: -65 °C to 1,149 °C.</td>
<td>-</td>
</tr>
<tr>
<td></td>
<td>- Probes have different lengths.</td>
<td>- Body built in Ni-based alloys.</td>
<td>-</td>
</tr>
<tr>
<td></td>
<td>- Probes installed in a same stage.</td>
<td>- Accuracy: 0.4% Full Scale (FS) above 260 °C.</td>
<td>-</td>
</tr>
<tr>
<td></td>
<td>- Typically, 1.2 harnesses per stage.</td>
<td>- Time response higher than 0.1 s.</td>
<td>-</td>
</tr>
<tr>
<td></td>
<td></td>
<td>- Simpler but with worse maintenance.</td>
<td>-</td>
</tr>
<tr>
<td>Temperature Sensor (RTD)</td>
<td>- Resistance Temperature Detector.</td>
<td>- Sensors used typically before the HPC section.</td>
<td>-</td>
</tr>
<tr>
<td></td>
<td>- Type Pt-100.</td>
<td>- Resistance 100 ohms at 0 °C, wire made in Pt.</td>
<td>-</td>
</tr>
<tr>
<td></td>
<td>- Total Length: 165 mm.</td>
<td>- Temperatures range: -53 °C to 260 °C.</td>
<td>-</td>
</tr>
<tr>
<td></td>
<td>- Probe Width: 12.7 mm.</td>
<td>- Accuracy: ±1 °C.</td>
<td>-</td>
</tr>
<tr>
<td></td>
<td>- Slim body.</td>
<td>- Response time: 8.5 s over FS.</td>
<td>-</td>
</tr>
<tr>
<td></td>
<td>- Typically, one per stage.</td>
<td>- Vibration resistance up to 40 g’s.</td>
<td>-</td>
</tr>
<tr>
<td>Liquid Fuel Flow Meter</td>
<td>- Turbine dual-rotor type.</td>
<td>- For fuel flows up to 19,090 kg / hr.</td>
<td>-</td>
</tr>
<tr>
<td></td>
<td>- Two magnetic pickups (MUP).</td>
<td>- Fuel temperatures from -54 °C to 177 °C.</td>
<td>-</td>
</tr>
<tr>
<td></td>
<td>- Dry Weight: 1.55 kg.</td>
<td>- Time of response of 4.0 s over full scale.</td>
<td>-</td>
</tr>
<tr>
<td></td>
<td>- Length: 185 mm.</td>
<td>- Accuracy of 0.5% over extended cruises.</td>
<td>-</td>
</tr>
<tr>
<td></td>
<td>- Diameter: 89 mm.</td>
<td>- Signal is a sinusoidal pulse.</td>
<td>-</td>
</tr>
<tr>
<td></td>
<td>- Cylindrical body.</td>
<td>- Period proportional to the flow rate.</td>
<td>-</td>
</tr>
<tr>
<td></td>
<td>- Installed in the fuel supply line.</td>
<td>- Different selectable scaling factors.</td>
<td>-</td>
</tr>
<tr>
<td></td>
<td></td>
<td>- No input power required.</td>
<td>-</td>
</tr>
<tr>
<td>Rotational Speed Sensor (N1)</td>
<td>- Variable Reluctance type sensor.</td>
<td>- High reliability and accuracy.</td>
<td>-</td>
</tr>
<tr>
<td></td>
<td>- Cylindrical probe with coils.</td>
<td>- Dual coil per redundancy purposes.</td>
<td>-</td>
</tr>
<tr>
<td></td>
<td>- Installed in Fan - LPT modules.</td>
<td>- The output voltage is higher with the speed.</td>
<td>-</td>
</tr>
<tr>
<td></td>
<td>- The tip detect gear movement.</td>
<td>- Accurate zero speed detection.</td>
<td>-</td>
</tr>
<tr>
<td></td>
<td>- Some detect blade movement.</td>
<td>- Detection of magnetic field variations.</td>
<td>-</td>
</tr>
<tr>
<td></td>
<td></td>
<td>- Gap between sensor tip and gear up to 1 mm.</td>
<td>-</td>
</tr>
<tr>
<td></td>
<td></td>
<td>- Typically made in stainless steel, Ti or inconel.</td>
<td>-</td>
</tr>
<tr>
<td></td>
<td></td>
<td>- Usual temperatures range: -150 °C to 520 °C.</td>
<td>-</td>
</tr>
<tr>
<td></td>
<td></td>
<td>- Tolerate vibration levels of 40 g’s.</td>
<td>-</td>
</tr>
<tr>
<td>Rotational Speed Sensor (N2)</td>
<td>- Variable Reluctance type sensor.</td>
<td>- High reliability and accuracy.</td>
<td>-</td>
</tr>
<tr>
<td></td>
<td>- Cylindrical probe with coils.</td>
<td>- Dual coil per redundancy purposes.</td>
<td>-</td>
</tr>
<tr>
<td></td>
<td>- Located in 4GB or front frame.</td>
<td>- The output voltage is higher with the speed.</td>
<td>-</td>
</tr>
<tr>
<td></td>
<td>- The tip detect gear movement.</td>
<td>- Accurate zero speed detection.</td>
<td>-</td>
</tr>
<tr>
<td></td>
<td></td>
<td>- Detection of magnetic field variations.</td>
<td>-</td>
</tr>
<tr>
<td></td>
<td></td>
<td>- Gap between sensor tip and gear up to 1 mm.</td>
<td>-</td>
</tr>
<tr>
<td></td>
<td></td>
<td>- Typically made in stainless steel or Ti.</td>
<td>-</td>
</tr>
<tr>
<td></td>
<td></td>
<td>- Usual temperatures range: -150 °C to 520 °C.</td>
<td>-</td>
</tr>
<tr>
<td></td>
<td></td>
<td>- Tolerate vibration levels of 40 g’s.</td>
<td>-</td>
</tr>
<tr>
<td>Pressure Transducer</td>
<td>- Micro-Electro-Mechanic (MEMS).</td>
<td>- One transducer per pressure line.</td>
<td>-</td>
</tr>
<tr>
<td></td>
<td>- Silicon pressure technology.</td>
<td>- Pressure range from 6.9 kPa to 69 Bar.</td>
<td>-</td>
</tr>
<tr>
<td></td>
<td>- FADEC mounted.</td>
<td>- Usual temperatures range: -55 °C to 125 °C.</td>
<td>-</td>
</tr>
<tr>
<td></td>
<td>- Length: 57 mm.</td>
<td>- Accuracy ±0.05% FS over temperature range.</td>
<td>-</td>
</tr>
<tr>
<td></td>
<td>- Width: 51 mm.</td>
<td>- 20 Hz frequency response.</td>
<td>-</td>
</tr>
<tr>
<td></td>
<td>- Height: 15 mm.</td>
<td>- 9 ms of latency (time to get the reading).</td>
<td>-</td>
</tr>
<tr>
<td></td>
<td></td>
<td>- Power supply needed: 0.4 W, at 5-16 VDC.</td>
<td>-</td>
</tr>
</tbody>
</table>

Table 4: Some examples of sensors installed in the engine to monitor variables relative to the gas path, such as temperatures, pressures, rotational speeds, and fuel supply. Data and images were obtained from the product description in the respective websites of the suppliers: Ametek, Meggit, HarcoSemco and Honeywell [231].

- Lubrication System and Fuel System sensors (see Table 5): A very important part of the MS covers the lubrication system, aiming to provide a clear picture on the health condition of the different engine bearings. The engine shafts rest upon those bearings, which support the great loads (both radial and axial) produced during operation. A sudden increase in any of the temperatures in the return oil lines (known as scavenge lines) from the bearing sumps, meaning any step in their trends, would indicate a
mechanical problem with any of the bearings that would need imminent intervention in the workshop. In addition, if a differential pressure sensor monitoring the condition of an oil filter shows high values, comparing with the situation when the filter was installed, new and clean, then that component will be closer to be clogged, therefore it will need to be replaced by a new filter cartridge. The fuel system needs also of certain instrumentation to control the fuel supply to the engine. It is particularly important to verify that the fuel supply is continuously flowing to the engine during operation. Equally important is understanding under which conditions the fuel is being supplied, regarding particularly the mass flow. A restriction in the fuel supply, indicated by a low-pressure value in a point of the line, would certainly explain a potential loss of power. Fuel-oil heat exchangers are typically used to cool down the oil and to warm up the fuel, increasing the efficiency of the whole machine. An abnormal indication in these sensors could mean a problem with the fuel-oil heat exchanger.

<table>
<thead>
<tr>
<th>Sensor Type</th>
<th>Practical Examples</th>
<th>Typical Dimensions and Location</th>
<th>Typical Technical Specifications</th>
</tr>
</thead>
</table>
| Temperature Sensor (RTD)     | ![Image]           | • Resistance Temperature Detector.  
• Installed in different points.  
• Type Pt-100, Pt-200, Pt-500, etc.  
• Total Length: 87 mm.  
• Probe Width: 3.2 mm.  
• Weight: 36 g. | • Wire made in Pt with ceramic substrate.  
• Temperatures range: -54 °C to 200 °C.  
• Accuracy: ±1 °C at 100 °C.  
• Response time: 5 seconds over 5%.  
• Vibration resistance up to 30 g's.  
• Configurations possible: 2 or 3 wires. |
| Differential Pressure Sensor (Diaphragm) | ![Image] | • Installed nearby the filter module.  
• Diaphragm type sensor.  
• Total Length: 83 mm.  
• Width: 25.4 mm.  
• Weight: 200 g. | • The pressure difference deforms a diaphragm.  
• The deformation is detected by a transducer.  
• Alarm appears with high values.  
• Pressure range up to 17 Bar.  
• Temperatures range: -55 °C to 150 °C.  
• Body built in stainless steel.  
• Accuracy: ±3% Full Scale (FS) during flight. |
| Magnetic Chip Detector        | ![Image]           | • Resistance Variation Detector.  
• Installed in oil scavenge lines.  
• Ideally, one detector per sump.  
• Typically, in the lube pump.  
• Different types, different locations. | • Metallic wear and deterioration means particles in the oil system.  
• When enough material is trapped by the magnet, the circuit is closed.  
• Ferromagnetic chips are trapped by magnet.  
• Almost immediate response.  
• Some detectors count with strainers to catch non-ferromagnetic chips. |

Table 5: Some examples of sensors typically installed in the lubrication system oil lines and fuel supply system lines, to monitor different aspects of the operation of the engine. Data and images were obtained from the product description in the respective websites of the suppliers: Ametek, Kulite, Allen Aircraft [232].

- Vibration Monitoring Sensors (see Table 6): Every machine vibrates during operation, and the higher the vibration level is the more compromised its remaining useful life will be. There are usually two accelerometers (sometimes three, and the trend is to install more in the future) attached to the engine casings detecting vibrations at different frequencies. One of the sensors is normally installed around the HPC (known as CRF accelerometer in GE engines), and the second one is installed in the Fan module or in the LPT module (in this last case, it is known as TRF accelerometer). These sensors are based on piezoelectric principles, so a seismic mass inside the sensor compresses a sensitive piece of barium titanate (BaTiO₃) typically, generating weak electric charges that can be converted into current or voltage signals. Those signals are filtered at the frequencies of the rotational speeds in the shafts in separated specialized modules outside the FADEC (so
called Airborne Vibration Monitoring modules or AVM [303]). The OEM know, based on experience, the usual vibration levels that can be expected in their engines under different operational profiles. When a sudden increase in the vibration levels is detected, that could mean a potential mechanical issue which nature will be analyzed with the information retrieved from the accelerometers. The problems detected range from damages in the bearings to loosen pipes in the external configuration of the engine, or oil contained in the shafts because of a leak, just to mention some of the most common issues. Vibrations filtered at increasing integer multiples of the frequencies given by the rotational speeds, the so-called harmonic components, are analyzed when there is suspicion of a potential problem with a bearing to try to find unusual vibration patterns. Some noise detected in the broadband could be related to problems like loosen parts in the engine. Several companies offer expertise in this field (i.e., diagnostics in turbomachinery mechanical problems), and some even develop their own hardware and software to analyze the signals coming from the vibration sensors. The charts provided by this software (such as bode diagrams, polar or cascade broad band charts, and rotor orbit trajectories [1]) are of great help in mechanical troubleshooting. This information could be also integrated in the analysis that will be explained in Chapter 4. For instance, a FOD that implied severe deterioration in the compressors, resulting in several blades torn or broken, could be confirmed not only by the degradation in the associated H&Q parameters of the gas path analysis, but also by a potential unbalance detected in the vibration monitoring system or a sudden change in trends.

<table>
<thead>
<tr>
<th>Sensor Type</th>
<th>Practical Examples</th>
<th>Typical Dimensions and Location</th>
<th>Typical Technical Specifications</th>
</tr>
</thead>
<tbody>
<tr>
<td>High Temperature Accelerometer</td>
<td>• Piezoelectric sensor.</td>
<td>• Electrical charges created by compression.</td>
<td>• It needs a voltage converter to amplify signal.</td>
</tr>
<tr>
<td></td>
<td>• Located in hot section areas (LPT).</td>
<td>• It contains a seismic mass applying force.</td>
<td>• Cables are integrated.</td>
</tr>
<tr>
<td></td>
<td>• Sensor weight (no cables): 120 g.</td>
<td>• A piece of BaTiO₃ produces the charges.</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Housing In Inconel.</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Charge range from 2 pc/g to 100 pc/g.</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>• For the vibration in the low-speed shaft.</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Temperatures range: -253 °C to 780 °C</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>• It needs a voltage converter to amplify signal.</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Cables are integrated.</td>
<td></td>
</tr>
<tr>
<td>Medium Temperature Accelerometer</td>
<td>• Piezoelectric sensor.</td>
<td>• Charge range from 2 pc/g to 100 pc/g.</td>
<td></td>
</tr>
<tr>
<td></td>
<td>• Located around the HPC.</td>
<td>• For the vibration in the high-speed shaft.</td>
<td></td>
</tr>
<tr>
<td></td>
<td>• Height: 40 mm.</td>
<td>• Housing In Inconel.</td>
<td></td>
</tr>
<tr>
<td></td>
<td>• Base diameter: 38 mm.</td>
<td>• Temperatures range: -253 °C to 500 °C</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>• It needs a voltage converter to amplify signal.</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Cables are not necessarily integrated.</td>
<td></td>
</tr>
</tbody>
</table>

Table 6: Some examples of accelerometers used in aircraft engines. Data and images were obtained from the product description on the website of the selected supplier: Meggitt [233].

- Structural Assessment Sensors: New engine models will incorporate instrumentation, coming in certain cases from the accumulated experience with advanced military aircraft engines [230], to aid in assessing the structural integrity of the engine. Some examples are inlet and/or exhaust debris monitors, or high bandwidth vibration sensors. The Inlet Debris Monitoring Sensor (IDMS) is installed in the Fan module and monitors the electrostatic charge associated with debris ingested at the engine inlet. The sensor is designed to evaluate size, quantity, velocity, and even composition
of debris entering the inlet (assessing if the debris will be damaging, like detached lock-wire, or non-damaging, like salt water). The use of composite materials and the increase of rotational speeds and OPR have motivated the incorporation of this technology. The GE9X will count with a debris rejection system, which not only will detect potentially harmful particles, but it will also effectively extract air with part of those debris to avoid damages in the Core Engine. The Exhaust Debris Monitoring System (EDMS) measures the electrostatic charge of debris exiting the engine, which is likely to have been produced by engine distress (e.g., loss of material, rubs, erosion, etc.). A healthy engine operation results in a small erosion, of different components, that shows up as fine debris. Changes in the nature or quantity of this exhaust debris could be an early indication of excessive wear or incipient failures. This new kind of sensors will certainly improve the prognostic capabilities in the future by estimating the amount and origin of the liberated material when detected in the exhaust section. Another example of new technology that will contribute to boost the CBM and engine prognostics is the Stress Wave Analysis sensor (known as SWAN) which is a piezoelectric transducer that monitors structurally borne ultrasonic sound vibrations to measure the energy associated to different shock or friction events. By measuring those events, the SWAN can detect wear and damage at the earliest stages and is able to track the progression of a defect. This is possible because, as the damage progresses, the energy content of friction and shock events increases [36]. This energy is logged and compared against normal machine operating conditions. This technology has been applied mainly to the early detection of damages and wear in gears of the AGB, but it could be used in other modules.

The previous list is representative but not comprehensive, because there are variants for the instrumentation installed in the different turbofan engine models. Same situation occurs with the rest of gas turbine engines. Varied designs, geometries, materials, and locations can be found easily among the OEMs. And the trend on the number of sensors is to be progressively increased (see [209], [101], and [102] on more intelligent gas turbines and the interest in new instrumentation).

3.2.3.- **Engine Control System: EEC and FADEC**

The FADEC (Full Authority Digital Engine Control) consists of the Engine Control Unit (ECU, also known as Electronic Engine Control, or EEC), the Hydromechanical Unit (HMU), and its peripheral components and sensors used for control and monitoring (see Figure 40). As it was just exposed, there are sensors covering gas path, lubrication system, fuel supply system, vibration monitoring system, and position feedbacks of all the variable geometry actuators and valves managed by the FADEC. These sensors, together with the ECU, constitute the MS. Vibration signals from accelerometers need of separate treatment (amplifying, filtering, and indication), so they are sent to independent AVM modules which show post-processed information directly in the EICAS (Engine-Indicating and Crew-Alerting System). The engines and the rest of the aircraft, count with computers exchanging information continuously through different data buses (following standard protocols for data transmissions). ARINC 429, ARINC 629, ARINC 791, CSDB or ASDB, STANAG 3910 are some of the most common nowadays [241].
The ECU sends information on the condition of the power plant and the aircraft provides atmospheric and flight data through those data buses. There are typically two Propulsion Interface Monitor Units (PIMU) where the information from the engines is processed before being shown in the EICAS to the crew.

Figure 40: Schematic diagram of a FADEC (back images from BAE Systems [24], and Woodward [302]).

The implementation of a new diagnostic, prognostic, and performance methodology, like the one proposed in this work, would likely need of separate processing modules, like the PIMU installed in the Common Computer Resource Cabinets (CCR), that could coordinate with the existing computers through the same data buses (see Figure 41). It could be a similar concept to the Engine Monitoring Unit (EMU) concept developed for the GEnx [237].

Figure 41: CCR Cabinets location in a B767 (images and information from NASA [208] and GE Aviation [107]).
An ineffective MS could impact very negatively to the reliability of the whole CS. The design and maintenance decisions made for the MS will affect to the whole FADEC and vice versa. Any event during operation in another FADEC component, such as the electrical alternator, would immediately affect to the MS. For each turbofan engine model, there are multiple operational and maintenance considerations regarding the MS that need to be carefully evaluated during its design phase and future retrofits. The following list summarizes some of them:

I. Regarding the acquisition of data:

   a. There will be different type of signals coming from the instrumentation. Some sensor signals will be analog (e.g., temperature, speed, pressure, or flow sensors, transmitting information by variations of resistance, frequency, voltage or current) and some others will be discrete (e.g., switches, or end position sensors, opening or closing circuits). Some signals will also need to be amplified. The calibration of the different sensors will be defined by this circumstance. In addition, the inherent noise in the interconnecting wiring will not affect equally to the different signals. So, there will be different signal conditioning and processing required, depending on each kind of sensor, which is based in specific physical principles (e.g., RTD temperature sensors are based on the variation of electrical resistance of a known material, meanwhile speed sensors are based on the Hall effect) and, as a result, the signals they provide are variated and must be treated differently.

   b. Signal sampling rate is a parameter that will likely increase in the next years with more powerful processors on board and improved connectivity capabilities, contributing to have a more accurate and granular information on the engine condition. With more samples, it will be easier to detect sudden spurious “peaks” in sensor lines that could lead to false alarms. Some other quick events will be certainly captured more easily with more frequent samplings. However, it will mean a considerable increase in the amount of data to be managed. Today that rate is in the order of 128 samples per second, typically.

   c. The signal synchronization across the subsets of sensors will have to be properly coordinated. Not all the sensors count with the same response times and not all of them will work at the same frequencies when supplying data to the MS (e.g., rotational speed sensors in opposition to TC or RTD), so different frequency scales will have to co-exist and the MS will have to coordinate all of them. New technologies in new sensors will mean further data fusion efforts.

II. Regarding the management of data:

   a. There will be data processed on-board and there will some data downloaded once the aircraft had landed [45]. Ideally, the MS should manage the information directly during the flight. In the future, it is expected that the connectivity capabilities of the aircraft will be considerably improved inside a IoT framework [300]. Nevertheless,
in case that connectivity would fail, there must be enough resources on-board to manage the information generated until some support could be received, either through remote connection or once landed. This means there must be enough data storage capacity in the aircraft and a quick downloading capability by the GSS in the airport.

b. The installed hardware counts with certain capabilities and with constraints that must be taken into consideration as well. The accuracy of the instrumentation is typically limited to specific ranges. Meanwhile new technology (e.g., wireless sensors [261] or networked FADEC [63]) does not replace existing transducers, the future instrumentation capabilities will remain similar. Also, some problems relative to phenomena taking place in the hot section of the engine (e.g., thermal fluence in turbine blades and disks, fatigue crack propagation, etc.), are still not covered by current MS. That situation will change, and it will come with new maintenance challenges.

c. The processing speed and data bandwidths given by the capabilities of the processors installed in the ECU is limited. These capabilities are progressively growing though. In 1978, the microprocessors used in ECU counted with 1.7 MHz clock rate [28], meanwhile modern FADEC models used in engines like the CFM56-7B series have installed proven and well-known microprocessors since 1990s, like the MPC603, working at 80 MHz [214]. There are suppliers [283] providing nowadays microprocessors at higher clock rates (120 MHz) for FADEC applications, so the trend is clearly to increase processing speeds in controllers. The airworthiness’ authorities will determine when new microprocessors can be installed in the future ECU models.

d. Finally, the outcome from the MS will lead to some condition report and fault annunciation, when applicable. Not only for the pilots or engine operators, but also for ground personnel, MRO technicians, and parts stocks. The way the indications are shown and transmitted will also have implications in the design and maintenance of the MS.

3.2.4.- Instrumentation considered in the model

Once the topic of the gas turbine instrumentation has been explored, it will be introduced the set of sensors that will provide data to the engine model, which are typically available in a generic turbofan, like the CFM56-5A [181]. Initially, only the following 10 gas path sensors will be considered for the model:

- \( P_{13b} \), corresponding with the total pressure in the duct after the Fan.
- \( T_{25b} \), which is the total temperature right before the HPC.
- \( P_{25b} \), total pressure in the same station, after the booster.
- \( T_{3b} \), for the total pressure in the CDP, after the HPC.
- \( P_{3t} \), total pressure also in the CDP.
- \( T_{45t} \), corresponding with the total temperature right before the LPT.
- \( T_{5b} \), corresponding with the EGT.
- \( N_H \), for the rotational speed in the Core Engine (N2).
- \( N_L \), for the rotational speed in the Fan, booster, and LPT (N1).
- \( W_f \), which provides the fuel flow supplied to the engine.
When analyzing this list of sensors, comparing with the actual ones available in a CFM56-5A (engine used as reference for the instrumentation layout, see Figure 42), there are some considerations to do before going on with the model:

- The list of sensors mounted in the CFM56-5A does not include a pressure sensor in the rear of the engine (e.g., between HPT and LPT). Some other engines include a pressure sensor in the turbine section, and some authors consider it necessary for performance estimation and diagnostics [267]. This is a circumstance that will lead to interesting results in the next chapters of this work. Eventually, that sensor will be added to the model to evaluate the impact of its inclusion, so the total list of sensors in the model will finally count with 11 signals, as it will be explained later.

- The pressure and temperature used in the model will be total, this means counting both their static and dynamic factors. Typically, the pressure probes immersed in the gas path decelerate the flow before reaching to the electronic transducer, so those readings could be considered as total pressures, by definition, similarly to what a Pitot probe would do. Something similar happens with the temperature probes immersed into the gas path. The incoming flow decelerates before touching the surface of the probe, which is a stagnation surface. However, some pressure sensors measure only static pressures because they detract a little portion of the flow in the gas path perpendicularly to the stream. That circumstance typically happens with the pressure in the CDP section. In this sense, total magnitudes will be considered for the calculations.

- The CFM56-5A counts with some sensors that will not be included to the model, like the $P_{12}$, which information can be obtained from the flight conditions, such as Mach number, altitude, temperature, and pressure, by adding some pressure loss factor to consider the friction in the inlet mouth. Also, the temperature sensor in the LPT casing for active clearance control will not be considered in the model, as it will be assumed that the engine is being operated inside a steady regime, close to its design point.

- The CFM56-5A counts with temperature values (so called $T_{49.5}$) in the second stage of nozzle vanes of the LPT. The model will consider that temperature to be taken in between HPT and LPT, so a bit before in the gas path. That change is consistent with some other engine models.

![Figure 42: Gas path instrumentation in a CFM56-5A and the initial list of sensors selected for the model, background picture obtained from Lufthansa [181].](image-url)
Regarding the accuracies of the sensors installed in the engine, typical values from the industry will be considered in this study. The sensors must comply with specifications established by the OEMs (based on certificates from regulatory authorities), so the suppliers produce components versatile enough to be mounted in different kind of engines, aiming to gain in product diversification. Similar sensors will be found in engines from the main different OEMs. Based on the information provided by the main sensors’ suppliers, the following list of accuracies can be taken as representative references by now (full scales will be indicated later in Chapter 5):

- \( P_{13b} \), pressure transducer with an accuracy of 0.1% over the Full Scale (FS).
- \( T_{25b} \), RTD sensor with an accuracy of 0.6% over the FS.
- \( P_{25b} \), pressure transducer with an accuracy of 0.1% over the FS.
- \( T_{3b} \), thermocouple type-K with an accuracy of 0.4%, at the usual high operational temperature values. The TC tend to be less accurate at lower values of temperature, closer to ambient conditions.
- \( P_{3b} \), pressure transducer with an accuracy of 0.1% over the FS. The FS in the CDP sensor is considerably wider than in the rest of pressure transducers. The accuracy is particularly important in this sensor, given its relevance detecting incipient stalls.
- \( T_{45b} \), thermocouple type-K with an accuracy of 0.4%, at the usual temperature values. Similar consideration with it than in the case of \( T_{3b} \).
- \( T_{5b} \), thermocouple type-K with an accuracy of 0.4%, at the usual temperature values.
- \( N_{H} \), variable reluctance sensor with an accuracy of 0.5% over the FS. The range of operation of this sensor will double the one for the rotational speed in the low-speed shaft. The speed sensors use to be very accurate and stable, so they do not typically need of recalibrations.
- \( N_{L} \), variable reluctance sensor with an accuracy of 0.5% over the FS. Similar considerations regarding stability than in the case of \( N_{H} \).
- \( W_{f} \), turbine fuel flow meter, with an accuracy of 0.5% over the FS.

It will be assumed in the next chapters that the Full Scale of each sensor is the one needed to cover all the operational conditions evaluated in the model. Further considerations, regarding accuracy in the different zones of the scale covered by the sensors, will depend on the type of instrumentation installed in each engine and they can be treated in case-by-case basis. The results obtained by the model should count with higher, or at least equal, accuracy levels than the ones shown in the previous list. Unfortunately, numerical accuracies above the levels established by the instrumentation will not be perceived in a real application, because the model will be fed with data already limited in terms of that numerical accuracy. However, the degree of accuracy of the instrumentation installed in the future engines will eventually tend to improve once new and advanced technologies are implemented in the sensors, so this is a motivation to get as much exact results as technically feasible with the numerical methodology. On the other hand, obtaining extremely accurate results could penalize the computational capabilities demanded by the algorithms, and this would lead to time delays when solving the inverse problems that will be detailed in the next chapter. This means that a commitment between time and accuracy would have to be determined to meet all the requirements in the problem inside acceptable time limits.
3.3.- **Performance model of the engine**

3.3.1.- **The model developed in PROOSIS®**

A numerical model of a high-bypass two-spool turbofan was developed, by means of a commercial software so called PROOSIS®, and it was aligned against publicly available data of a typical modern turbofan engine at its design point, to repeatedly run the necessary performance calculations that will be described in the next chapter, the quickest way possible. In this case, a CF6-80E1A2 was the engine selected for the validation of the numerical model (for validation data, see [144]).

The model of the engine will represent a turbofan with unmixed flows in the exhaust, like in the real engine, so the flow from the Core Engine and the flow from the Fan bypass duct remain separated. Some turbofan engine designs, mainly used for supersonic applications, mix both flows to get slight higher thrust levels, to provide fresh air to the afterburner, or to abate noise levels (see [215] and [260]), at the expense of a more complex exhaust and a heavier configuration. The unmixed model is representative of most of civil commercial turbofan models nowadays.

PROOSIS® (Propulsion Object Oriented Simulation Software, v.6.2.0, 2020) is a sophisticated, well-known, and flexible software (SW) platform, based on EcosimPro®, for 1D advanced modeling and simulation of gas turbine engines [73]. Different versions of this platform are used to model other complex systems, such as liquid propellant rocket engines or nuclear reactors. To virtually reproduce the engine of this study it was used the TURBO® library, which includes a complete catalogue of components (i.e., compressors, turbines, fans, propellers, auxiliary gear boxes, exhaust nozzles, combustors, ducts, mixers, etc.) for the modeling and simulation of, potentially, any kind of gas turbine engine [74]. This means the exercise described in the thesis could be done for any other gas turbine model, just by configuring its model into the SW. This feature opens the door to new studies useful, for instance, in marine or industrial applications (e.g., by simulating the aeroderivative version of the CF6-80E1: the LM6000).

Additionally, both the PROOSIS® platform and the TURBO® library have been previously used and validated in several works in which it was necessary to calculate the performance of different types of engines, from common civil turbofans to more special counter-rotating open-rotor engines (the details are available in [44] and [32]). Obviously, the reliability of the results obtained with any gas turbine modeling software tool depends on the capability of the model to accurately predict the performance of the real system over a wide range of operational conditions. In this sense, PROOSIS® allows to fine tune the model, for instance, allowing for several air bleeds, from the HPC to both HPT and LPT, for cooling purposes. It is only necessary to count with enough data to confront the model, and certain skills managing the SW, to adjust the settings for a realistic virtual version of the engine. Finding validation data is typically the most time-consuming part of the modeling process given the scarce information from real engines publicly available from the different OEMs and engine operators. The complete schematic diagram is shown in Figure 43.

So, in summary, these reasons (flexibility and successful experience), together with the accuracy and solving speed that PROOSIS® affords, were the ones that decided in favor of the use of this SW among the different tools available for engine performance (such as GasTurb®, GSP®, or GTPsim®).
Figure 43: High-bypass two-spool turbofan engine representation in PROOSIS®.
In the previous schematic, the main engine modules are represented by blue geometric shapes, indicating the different compressors, turbines, ducts, etc. Those modules are interconnected in different ways:

- The main flow interconnections are indicated by **thick blue lines**. These lines show the main gas path flows, primary (throughout the Core Engine) or secondary (throughout the Fan bypass duct).
- The bleed flows (e.g., for cooling purposes) are indicated by **soft blue lines**. The required cooling flows by the turbines and by the aircraft (e.g., cabin ambient conditioning) are represented in the model this way in the SW.
- The mechanical interconnections, regarding the different engine modules that are part of the low-speed shaft and the high-speed shaft, respectively, are represented by **brown straight lines**.
- The value of specific component variables (typically mass flows values and its associated kinetic energy values) at certain stations of the engine, during its simulation, is transferred to the Performance Monitoring Module (PMM) to be logged. This transmission of information is represented by **grey lines**.
- The value of relevant flow variables (i.e., total temperature, total pressure, fuel-air ratio, and water-air ratio) is also transferred to the PMM to be logged, and this is indicated by green lines (different green tones for static and rotatory components of the engine). The SW obtains for each station of the engine (in fact, for each stage in rotatory components) the mass flow rate, total and static pressures and temperatures, effective area of the section in that stage, Mach number, fluid velocity, fuel-air ratio, and water-air ratio.

Providing the full details of the aerothermodynamic model implemented in PROOSIS® would not possibly fit within the extension of this work. It counts with more than 800 equations and needs more than 500 parameters to work, as it is summarized in Figure 44. Nevertheless, the main relationships, assumptions and hypotheses will be briefly introduced, describing thus the modeling approach followed. For the more specific details on the settings of the model and how it is configured, the reader will be redirected to [10]. The SW allows to use turbomachinery maps, instead of algebraic expressions, for off-design calculations. The difference between on-design and off-design analysis will be explained later.

![Figure 44: Balance of variables and equations for a high-bypass unmixed two-spool turbofan engine in PROOSIS®.](image-url)
The TURBO® library's fluid model counts with a set of implemented functions to obtain the main fluid thermodynamic properties (i.e., enthalpy and entropy, among others) by interpolation of 3-D tables, or by polynomial functions for given temperature, fuel-air ratio (FAR) and water-air ratio (WAR) values [10]. The main components of the schematic shown in Figure 43 will be briefly described to get a better understanding of the way PROOSIS® works when it is executed.

The compressors can be modelled in the SW by algebraic expressions or using turbomachinery maps like the one shown in Figure 45. This last option is typically preferred, when possible, to gain in accuracy, specificity, and fidelity to the real system. However, the use of maps implies managing their associated tables of data and performing iterative interpolations, circumstance that could redound in longer computational times. Maps are qualitatively similar (not exactly equal though) for every axial compressor, like the one mounted in the engine under study. That circumstance facilitates the use of maps with PROOSIS® because the SW provides reference maps that can be re-scaled, for appropriate size and typical ingested air flow, to be used for the engine that will be simulated. The surge line (compressor instability limit, for more details on the topic, go to [126]) is indicated in red, and the SW calculates the stall margin for each operational point in the map. The lack of linearity in the problem results evident just having a look to these maps.

Some notes and comments, regarding the mathematical modeling of the compressors within the SW (see station numbers in previous figure) must be made:

- **Mass flow conservation**: It must be considered both cases, with and without air bleeds (e.g., for cooling), and the applicable flow capacity parameters, \( \Gamma_i \).

\[
W_2 = W_1 ; \text{ (with no bleeds)} \tag{3.1}
\]

\[
W_2 = W_1 - \sum \Gamma_i W_b ; \text{ (with bleeds)} \tag{3.2}
\]

\[
\text{WAR}_2 = \text{WAR}_1 ; \text{ (same humidity)} \tag{3.3}
\]
• **Energy conservation:** The SW calculates the mechanical power balances in the compressor. The air bleeds will affect that mechanical balance.

\[
pwr = -W_1 \cdot (h_{t_2} - h_{t_1}); \text{ (with no bleeds)} \quad (3.4)
\]

\[
pwr = -W_1 \cdot (h_{t_2} - h_{t_1}) - \left( \sum_{b} W_b \cdot (h_{t_2} - h_{t_{bld}}) \right); \text{ (with bleeds)} \quad (3.5)
\]

• **Pressure Ratio:** The bleeds in the compressor must be set up providing the amount of air flow bled and the stage where the bleed is done, so the SW can establish the proper ratios for the calculation of the pressure finally delivered. Stators can be configured with variable geometry as well (simulating VSVs is not critical at steady states such as cruise flights). Only the case with no bleed and fixed geometry is indicated here for the sake of simplicity.

\[
PR = \frac{P_{2t}}{P_{1t}}; \text{ (with no bleed)} \quad (3.6)
\]

• **Efficiency:** Similar consideration regarding bleeds. The SW needs certain settings to be configured before establishing the definitive efficiency achieved in the compressor when bleeding air is simulated, as in our study (standard values, common in the industry, were used here).

\[
\eta = \frac{(h_{t_{2,ls}} - h_{t_1})}{(h_{t_2} - h_{t_1})}; \text{ (with no bleed)} \quad (3.7)
\]

• **Two kinds of maps are available:** BETA maps or MFT maps (coming from the Map Fitting Tool conceived by GE/NASA [262]) can be used for performance analysis. BETA maps were finally chosen, as they are most frequently used.

• **Different physical effects can be simulated:** Speed, torque, and inertia are adjustable settings. However, no transitory state was contemplated in this study (focused on cruise flight phases), therefore those parameters were not configured, and default values were assumed as valid. The effect on the variation of Reynolds number (effect of viscosity) in the flight envelope is accounted for with several corrections applied, as it will be explained later. Heat transfers through the compressor surfaces are configurable (heat soakage effects were not considered in this study as it is focused on steady states). Tip clearance effects are configurable, but they were taken as negligible (steady state achieved in the baseline engine under simulation).

• **Calculations with maps:** In the compressor map, it is provided the PR achieved in the component, for a given corrected mass flow parameter \((W_c)\), isentropic efficiency \((\eta)\), and relative corrected speed \((N_c)\). The corrections are done with given map values for \(T\) and \(P\), considered as reference values, and with the corrected speed at design point. The efficiency is represented by concentric iso-\(\eta\) lines and several constant rotational speed lines, ending up in the surge line, are also typically indicated. A set of auxiliary lines, called \(\beta\)-lines, are defined by the SW for interpolating purposes inside the map tables. The value assigned to the \(\beta\)-lines ranges between 0 and 1. It can be chosen a value of 0.5 for the design point as it was done in this study. The interpolation process inside the map, as well as the different corrections and adders that are used by the SW, are described with considerable detail in [10], which was used as reference during the modelling phase of the study.
The Fan module was modelled following similar considerations to the ones followed for the compressors. This component can be analyzed for performance studies in two different ways: Either by using two different maps for the airflow going through the Core Engine (primary) and for the airflow going through the bypass duct (secondary); or by using only one map that needs of some adjustment to refer correctly to both flows. The first option was finally chosen, assuming initially similar degradations for both, for simplicity’s sake. Fans could be also represented by algebraic expressions instead of using maps (2 maps in this case) like the one shown in Figure 46. The main difference comparing with compressors would be the inclusion of a new variable, the Bypass Ratio (BPR), which will determine the amount of air going to the Core Engine or being bypassed through the Fan.

\[ \text{BPR} = \frac{W_{BP}}{W_{core}} ; \text{ (without bleed)} \] 
\[ W_{BP} = W_1 \cdot \text{BPR}/(1 + \text{BPR}) \] 
\[ W_{core} = W_1/(1 + \text{BPR}) \] 
\[ W_{core,bld} = W_{core} - W_{FAN,bld} ; \text{ (if bleed is considered)} \] 
\[ \text{WAR}_2 = \text{WAR}_1 ; \text{ (same humidity)} \]

**Figure 46:** Schematic representation of a Fan, and one of its associated maps, in PROOSIS®.

Some notes and comments, regarding the mathematical modeling of the Fan module within the SW (see station numbering in the previous figure) must be made:

- **Mass flow conservation:** Same air bleed can be considered for aircraft handling purposes. Certain aeroengines use air bled from the fan for cooling purposes in the turbines as well (active clearance control). When the bleed is done, it is done from the primary airflow.

- **Energy conservation:** The following expressions provide the mechanical power needed by the Fan. When bleeds are considered in the Fan, proper
ratios must be established. So, it was considered no bleed in the next expressions for the sake of simplicity.

\[
pwr = pwr_{core} + pwr_{BP}
\]

\[
pwr = -W_{core} \cdot (h_{tcore} - h_{t1}) - W_{BP} \cdot (h_{tBP} - h_{t1})
\]  

(3.13)  

(3.14)

- **Pressure Ratio:** \( PR_{core} \) and \( PR_{BP} \) come from the different Fan maps.

- **Efficiency:** \( \eta_{core} \) and \( \eta_{BP} \) come from the different Fan maps.

Regarding the turbines in the model, different kind of maps are used now for performance studies, like the one in the Figure 47, where the expansion ratio of the turbine (PQ) is given by the product \( N_{c} \cdot W_{c} \), and \( \eta \) or \( N_{c} \). ZETA or MFT (NASA) performance maps can be selected for the turbines. The first kind was chosen for this study, as they are the most frequently used. It is easily identifiable the point where the turbine is choked for each value of rotational speed (it happens for each iso-\( N_{c} \) line when it becomes totally vertical). In this case, the \( \beta \)-lines are substituted by \( \zeta \)-lines to perform iterative interpolations during the calculations performed by PROOSIS®. The cooling flow from HPC (Secondary Airflows or SAS) will contribute to the power developed by the turbine and it will decrease the temperature of the rotor. A fraction of the SAS will be injected before (66%) the rotor of the stage where the cooling flow is needed. The rest of the cooling flow will be injected after that rotor stage. This approach is called the “equivalent single stage” model, as it will be explained later, and it is configurable in the SW. It is possible to define the angle of rotation of the flow exiting the turbine. Heat transfers (always after expansion in the stages where the cooling flow is injected) can be configured as well.

Some notes and comments, regarding the mathematical modeling of the Fan module within the SW (see station numbering in the previous figure) must be made:

- **Mass flow conservation:** The cooling from SAS is accounted in this module. Therefore, the FAR and WAR will be affected by those SAS. Mass fractions are calculated accordingly by the SW.
\[ W_2 = W_1 + W_{SAS}; \text{ (with cooling)} \quad (3.15) \]

- **Energy conservation:** The cooling flows affect the energy and temperature distribution in the different stages of the turbine, as it will be explained later.

\[ \text{pwr} = W_1 \cdot (h_{t1} - h_{t2}) + W_{SAS} \cdot (h_{tSAS} - h_{t2}) \quad (3.16) \]

- **Expansion Ratio:** Just the global expansion ratio is given for the sake of simplicity, but the equivalent ratios are calculated by the SW for the SAS used in the cooling of the turbine.

\[ PQ = \frac{p_{t1}}{p_{t2}}; \text{ (global expansión ratio)} \quad (3.17) \]

- **Efficiency:** Ratio between real and isentropic power. The effect of SAS is accounted by the SW (the incoming cooling flow is accelerated when it is injected into the mainstream flow of the turbine).

The static modules, represented in the [Figure 48](#), are also modelled by the SW using algebraic expressions to determine the behavior of these components (see station numbering in the figure):

The inlet selected in the model counts with a subsonic configuration. The SW provides as outcome its thrust, the mass flow through it, and its associated kinetic energy (KE). The pressure loss in it is an adjustable parameter (the SW provides different options for its calculation). The flow is considered adiabatic, and no addition of flow is done in this component. Ambient conditions are calculated from the International Standard Atmosphere model (ISA, as per ISO2533 specifications [147]), assuming no humidity and uniform flow at the inlet (i.e., with no distortion). This information is contained in the Ambient Conditions module of the model, represented in [Figure 43](#), right before the inlet of the engine. The SW allows to include the required deviations from the standard atmosphere, in case of need. A constant pressure loss of 0.5% was imposed for the engine inlet, independently of the regime of the engine, to gain in realism comparing with the default settings in PROOSIS®, which consider null pressure loss in the inlet for subsonic flights, in accordance with the MIL-E-5008B Standard, which is typically used for supersonic...
inlet diffusers [190]. Calculations were performed for the steady state operation of the engine (i.e., cruise) and, therefore, certain effects that could appear in the inlet during transitory phases of flight were neglected in the calculations, treating the main working fluid, effectively, as an adiabatic stream of gases.

- **Mass flow conservation:**

  \[ W_2 = W_1 \]  
  \[ \text{WAR}_2 = \text{WAR}_1 \text{ ; (same humidity)} \]  

- **Energy conservation:**

  \[ h_{t2} = h_{t1} \]  
  \[ T_{t2} = T_{t1} \]  

- **Pressure Ratio:**

  \[ \text{PR} = \frac{P_{2t}}{P_{1t}} \]  
  \[ \text{PR} = 1 - \Delta P_{\text{loss}} \text{ ; (including losses)} \]

The ducts in the model keep the total mass flow through them, excepting two cases: the Compressor Discharge Pressure (CDP) duct (station 3) which bleeds air for cooling, and the duct in the Fan rear section (bypass duct) that receives air from the Fan. Bled mass flows are adjustable also here. Pressure losses in this kind of component are configurable and the SW gives several options for its estimation. The heat transfer (gas-metal heat transfer, Q) can be configured. P and T in bled flows are calculated with the proper ratios (using mass flows and enthalpies).

- **Mass flow conservation:**

  \[ W_2 = W_1 \text{ ; (with no bleed)} \]  
  \[ W_2 = W_1 \pm \sum W_{bld} \text{ ; (with bleed)} \]  
  \[ \text{WAR}_2 = \text{WAR}_1 \]  
  \[ \text{FAR}_2 = \text{FAR}_1 \]  

- **Energy conservation:**

  \[ h_{t2} = h_{t1} - \frac{Q}{W_1} \text{ ; (with no bleed)} \]  
  \[ (h_{t1} \cdot W_1) = (h_{t2} \cdot W_2) + \left( \sum W_{bld} \cdot h_{bld} \right) + Q \text{ ; (with bleed)} \]  

- **Pressure Ratio:**

  \[ \text{PR} = \frac{P_{2t}}{P_{1t}} \]
\[ PR = 1 - \Delta P_{\text{loss}}; \quad \text{(with losses)} \quad (3.31) \]

In the combustor, the difference between burnt or unburnt fuel is established. Both FARB (burnt fuel-air ratio) and FARU (unburnt fuel-air ratio) are considered in the calculations (this difference will not be shown in the expressions below for the sake of simplicity). Different methods and correlations to calculate the efficiency of this component are available in the SW. Typically, the efficiency depends on the combustor load (pressure). Several kinds of fuels can be used, just by initializing the required data in the SW, such as composition or Lower Heating Value (LHV). Jet-A was the fuel selected for the model (some others are available, like JP4, diesel, natural gas, or even H2), and its combustion products are taken by PROOSIS® as working fluid. The tables with the thermodynamic properties of the gases come from NASA’s Chemical Equilibrium with Applications (CEA) code (reader can go to [191] and [192] for further reference). The effects of dissociation are neglected, and the combustion products are supposed to comprise CO₂, H₂O, O₂, N₂, and Ar only. In this sense, the SW allows to include additional kind of fluid models, just by adding the appropriate 3D tables to the model, if needed. This could be the case in future simulations if the production of pollutants was included because of environmental protection requirements. Nevertheless, recent SW versions estimate the emission levels of the engine (i.e., NOₓ, CO, and UHC), if required. Different kind of correlations are used for it. In the manual it is indicated how the different composition ratios could be calculated and the expressions for the correlations. These basic thermodynamic settings, that will be used by the model during the simulations, are gathered in the General Model Data module, represented in Figure 43 right before the inlet of the engine. Fuel injection temperature and pressure are adjustable. Heat transfers (again, gas-metal) can be configured. Pressure losses in the combustor can be estimated by different available methods as well.

- **Mass flow conservation:** In this case, the FAR will be null at the inlet.

\[ W_2 = W_1 + W_F \quad (3.32) \]

\[ W_{\text{AIR}} = W_1/(1 + \text{FAR}_1 + \text{WAR}_1) \quad (3.33) \]

\[ \text{FAR}_2 = \eta_{\text{CC}} \cdot \left( \frac{W_F}{W_{\text{AIR}}} \right) + \text{FAR}_1 \quad (3.34) \]

\[ \text{WAR}_2 = \text{WAR}_1 ; \quad \text{(model assumption)} \quad (3.35) \]

- **Energy conservation:** Reference (ref) conditions are used for the chemical process of fuel oxidation. A simple expression for the adiabatic total outlet specific enthalpy is indicated here:

\[ W_2 \cdot (h_{t2\text{AD}} - h_{t2\text{ref}}) = W_1 \cdot (h_{t1} - h_{t1\text{ref}}) + W_F \cdot (\eta_{\text{CC}} \cdot \text{LHV} + h_{tF} - h_{tF\text{ref}}) \quad (3.36) \]

After considering heat transfers:

\[ h_{t2} = h_{t2\text{AD}} - Q/W_2 \quad (3.37) \]
**Pressure Ratio:**

\[ \frac{P_{t2}}{P_{t1}} = 1 - \Delta P_{\text{loss}} \; ; \text{(losses are estimated)} \]  \hspace{1cm} (3.38)

**Efficiency:** The efficiency \( \eta_{\text{CC}} \) is specified or calculated. Calculations employ logarithmic correlations.

Finally, the nozzles were selected with convergent configuration, as this is the one used in most of the commercial airliners nowadays. The Convergent-Divergent configuration, more typical in supersonic aircraft, is an option available in the SW. The discharge coefficient \( C_d \) is used to correlate the geometric exhaust area (A) and the effective exhaust area (AE) in terms of exhaust flow. The coefficient will be given by the nozzle pressure ratio and the nozzle angle \( \alpha \) with different correlating modes available in the SW. Similar consideration is done for the thrust coefficient of the nozzle \( C_x \). The pressure loss is an adjustable parameter (the SW provides several options). The flow is considered adiabatic and isentropic.

**Mass flow conservation:**

\[ W_{\text{out}} = \frac{(P_{\text{Sout}} \cdot V_{\text{out}} \cdot \text{AE})}{(R \cdot T_{\text{Sout}})} \]  \hspace{1cm} (3.39)

\[ \text{AE} = C_d \cdot A \]  \hspace{1cm} (3.40)

**Energy conservation:**

\[ h_t = \text{Constant} \]  \hspace{1cm} (3.41)

\[ V_{\text{out}} = \sqrt{2 \cdot (h_t - h_{\text{Sout}})} \]  \hspace{1cm} (3.42)

**Nozzle Pressure Ratio (NPR):** NPR is obtained in critical condition, with \( P_{\text{Scrit}} \).

\[ \text{NPR} = \frac{P_t}{P_{\text{amb}}} \]  \hspace{1cm} (3.43)

\[ P_{\text{Sout}} = P_{\text{amb}} \text{ when NPR} < \text{NPR}_{\text{crit}} \]  \hspace{1cm} (3.44)

\[ P_{\text{Sout}} = P_{\text{Scrit}} \text{ when NPR} \geq \text{NPR}_{\text{crit}} \]  \hspace{1cm} (3.45)

Once the model has been very briefly introduced, and the different components that take part on it have been commented, the next step is establishing the design point of the engine (on-design conditions of the engine). PROOSIS® needs results from the determination of the design point to dimension the engine and to establish its performance capabilities. Turbomachinery maps are scaled during this phase. After the on-design analysis was completed, using design data from similar engines, and once the configuration of the engine was frozen and its geometry fixed, the off-design study could be performed. Off-design studies takes the engine outside the conditions for which it was designed, and where the optimum of its performance is reached, to evaluate what can be expected of that specific engine at different regimes and operational conditions where performance optimum is not met. On-design phase builds the engine, and off-design phase tests it.
### 3.3.2. On-design calculations

The first step to obtain the on-design condition of the engine is solving the aerothermodynamic Brayton cycle associated to the type of engine under study, like the one shown in the T-S diagram in Figure 49. Certain design data from the engine under study are required for this calculation (e.g., pressure ratios, efficiencies, maximum TIT, etc.). This analysis could be approached in different ways, with different degrees of accuracy, detail, and realism. A simplified way to proceed would be by using the applicable algebraic expressions of the one-dimensional flow for a perfect gas model, and by neglecting the influence of the internal air bleeds for cooling purposes (this is a well-known problem, see classic works in [215] and [248]). This approach will obviously lack of certain accuracy and fidelity to the real engine if the simplifying assumptions are not met, particularly in the hot section of the engine. Fortunately, PROOSIS® gets advantage of its powerful set of tools (tables, correlations, etc.) in the TURBO® library to provide a design point that will be close to the real one for the engine taken as reference. The main outcome of the on-design analysis is typically the specific impulse of the engine (I\textsubscript{SP}, thrust per unit of ingested air mass flow), and the required fuel-air ratio. With the I\textsubscript{SP}, the size of the engine can be decided to meet a particular thrust level and consuming a certain amount of fuel. In this study, the total air mass flow ingested by the reference engine was directly provided to the SW, as an input for the on-design point calculations (data available). Because of the divergent nature of the isobars, the higher the TIT, the higher the overall cycle thermal efficiency (given by the ratio of areas indicated in Figure 49).

![Brayton cycle diagram](image)

**Figure 49:** Brayton cycle for a turbofan engine like the one under study. The stations in the diagram are represented over the cutaway of a CF6-80 engine (background images obtained from GE Aviation [107]).
To obtain the *I_{SP}* in a two-spool turbofan engine, it is necessary to obtain previously the conditions of the fluid at the outlet section of the 2 exhaust nozzles (i.e., primary flow nozzle, expanding the flow from the Core Engine, and secondary flow nozzle, expanding the massive amount of air moved by the Fan). In particular, the flow speeds and static pressures. PROOSIS® can calculate the design point in an inexpensive way, but it is convenient at this point, to indicate at least the main principles behind its calculations, by means of the mathematical expressions used when solving such kind of problems (under certain simplifying assumptions that will be commented now). In the next expressions, \( C_p = \gamma R / (\gamma - 1) \) will be the air specific heat at constant pressure, assuming \( R = 287.10 \, \text{J/kg} \cdot \text{K} \) for the air gas constant (its value could vary depending on the molecular weight considered for air), and \( \gamma_{\text{comp}} \approx 1.40 \) the ratio of specific heats for compressors and Fan, given usual temperatures in compressors and Fan, meanwhile \( \gamma_{\text{exp}} \approx 1.33 \) will be the ratio of specific heats for turbines and nozzles, components where the temperatures are considerably higher, being then for reference \( C_{p,\text{comp}} = 1,004.85 \, \text{J/kg} \cdot \text{K} \) and \( C_{p,\text{exp}} = 1,157.10 \, \text{J/kg} \cdot \text{K} \), respectively. So, it will be considered, for this simplified approach, a 1-D compressible flow model and calorically perfect gases.

As it was shown before, when detailing the way the SW modeled the exhaust nozzles, they can be working basically either in critic or in adapted conditions, depending on the value of NPR. The adaptation will occur in any of the nozzles if the following condition is met at outlet section, different version of equation Eq. (3.43):

\[
P_t / P_{\text{amb}} \leq \left( \frac{\gamma_{\text{exp}} + 1}{2} \right) \gamma_{\text{exp}} \left( \frac{P_{\text{amb}}}{P_t} \right)^{\gamma_{\text{exp}} - 1} \tag{3.46}
\]

When that happens, the static pressure equals the ambient pressure at the outlet section, leading to the following expressions for static temperature and flow speed at that section, see equation Eq. (3.42), based on the applicable stagnation (or total) properties on it:

\[
T_s = T_t \left( \frac{P_{\text{amb}}}{P_t} \right)^{\gamma_{\text{exp}} - 1} \gamma_{\text{exp}} \tag{3.47}
\]

\[
V_{\text{out,ad}} = M_{\text{out}} \sqrt{Y_{\text{exp}} RT_s} = \sqrt{\frac{2 Y_{\text{exp}} RT_t}{Y_{\text{exp}} - 1} \left( 1 - \left( \frac{P_{\text{amb}}}{P_t} \right)^{\gamma_{\text{exp}} - 1} \right)} \tag{3.48}
\]

When conditions are critic instead, so the condition in Eq. (3.46) is not met, the Mach number reaches a value of 1.0 (critic conditions in that outlet section, meaning the conditions allowing for the maximum achievable flow in it), and the applicable expressions for static pressure and temperature, and flow speed, are as follows:

\[
T_s = T_t \left( \frac{2}{\gamma_{\text{exp}} + 1} \right) \tag{3.49}
\]

\[
P_s = P_t \left( \frac{2}{\gamma_{\text{exp}} + 1} \right)^{\gamma_{\text{exp}} - 1} \tag{3.50}
\]
\[ V_{\text{out,crit}} = \sqrt{Y_{\text{exp}} R T_s} = \frac{2Y_{\text{exp}}}{Y_{\text{exp}}-1} R T_t \]  \hfill (3.51)

No specific indication has been given in previous equations regarding the exact station where conditions are calculated (i.e., either in station “9” for primary flow nozzle, core, or “19” for secondary flow nozzle, bypass, respectively) so they can be used for both. Nevertheless, most of times the primary nozzle will work choked, so under critic conditions (for sure during a cruise stage). The secondary flow could remain adapted more often, given the low compression developed by the Fan, but that assumption would need to be verified with the value of NPR.

To obtain the stagnation (or total) properties applicable to those sections, it will be assumed that there are not remarkable energy losses between the outlet of the LPT (station “5”) and the primary flow nozzle. No afterburner is considered in the model, as it is not used typically for commercial aviation applications. Knowing the typical adiabatic efficiencies in both HPT (\(\eta_{HPT}\)) and LPT (\(\eta_{LPT}\)), as well as the respective expansion ratios (i.e., \(\Pi_{HPT}\) and \(\Pi_{LPT}\)), it can be linked the conditions in station “5” with the conditions in station “4”, right after the CC and before the first stage HPT nozzle vanes. The value of TIT (or \(T_{4t}\)) is typically given:

\[ P_{5t} = P_{45t} \Pi_{LPT} \]  \hfill (3.52)

\[ P_{45t} = P_{4t} \Pi_{HPT} \]  \hfill (3.53)

\[ T_{5t} = T_{45t} \left[ 1 - \eta_{LPT} \left( 1 - \Pi_{LPT}^{Y_{\text{exp}}-1} \right) \right] \]  \hfill (3.54)

\[ T_{45t} = T_{4t} \left[ 1 - \eta_{HPT} \left( 1 - \Pi_{HPT}^{Y_{\text{exp}}-1} \right) \right] \]  \hfill (3.55)

Then, with the knowledge of the pressure losses at the CC, given by \(\Pi_{CC}\), the respective efficiencies in HPC, booster, and Fan (i.e., \(\eta_{HPC}\), \(\eta_{LPC}\), and \(\eta_{FAN}\)), as well as with the respective pressure ratios (i.e., \(\Pi_{HPC}\), \(\Pi_{LPC}\), and \(\Pi_{FAN}\)), it is possible to link the conditions at station “4” with the conditions at engine’s inlet “2”. Similar consideration could be done for the Fan outlet (station “13”), right before the secondary flow nozzle:

\[ P_{4t} = P_{3t} \Pi_{CC} \]  \hfill (3.56)

\[ P_{3t} = P_{25t} \Pi_{HPC} \]  \hfill (3.57)

\[ P_{25t} = P_{13t} \Pi_{LPC} \]  \hfill (3.58)

\[ P_{13t} = P_{2t} \Pi_{FAN} \]  \hfill (3.59)

\[ T_{3t} = T_{25t} \left[ 1 + \frac{\eta_{\text{comp}}^{Y_{\text{comp}}-1}}{\eta_{\text{HPC}}} \right] \]  \hfill (3.60)
Sometimes, no information about efficiencies is available, and then the mechanical power coupling equations, between compressors and turbines, in the different spools (low-speed spool and high-speed spool, respectively) must be used instead to obtain a total temperature still pending at any engine station, contributing this to the complete resolution of the problem. In these problems with two-spool turbofan engines, counting with several booster stages, it may be necessary to know how the mechanical power demand is shared in between Fan and LPC. Otherwise, it may remain indetermined the conditions at station “13”, right after the Fan and before the LPC.

Assuming small mechanical losses in both spools, and a small portion of mechanical power required for auxiliaries (e.g., electrical generator, lube and scavenge pump, etc., which are mechanically connected to the AGB, and driven by the Core Engine with the transfer gearbox) comparing with the mechanical power transferred from turbines to compressors, which means the mechanical efficiency in both spools is considered to be high enough ($\eta_m \cong 1$) to accept such assumptions, then the following expressions are obtained for the mechanical coupling conditions:

$$ T_{2t} = T_{13t} \left[ 1 + \frac{\gamma_{\text{comp}}^{-1}}{\eta_{\text{LPC}}} \right] $$  \hspace{1cm} (3.61)

$$ T_{13t} = T_{2t} \left[ 1 + \frac{\gamma_{\text{comp}}^{-1}}{\eta_{\text{FAN}}} \right] $$  \hspace{1cm} (3.62)

Finally, with the flight conditions (namely, flight Mach number, altitude $T_{\text{amb}}$ and $P_{\text{amb}}$), with the design characteristics of the inlet (i.e., pressure drop in it, and the associated efficiency during dynamic compression before getting to the Fan), it is possible to obtain the main thermodynamic variables right at station “2”, which corresponds with the end of inlet, where the Fan begins. So, assuming the dynamic compression of the air stream, from the infinite upstream to the Fan’s inlet is isenthalpic (i.e., $h_{2t} = h_{\text{ot}}$):

$$ T_{2t} = T_{\text{amb}} \left[ 1 + \left( \frac{\gamma_{\text{comp}}^{-1}}{2} \right) M_{\text{flight}}^2 \right] $$  \hspace{1cm} (3.68)

$$ P_{2t} = \Pi_{\text{dyn}} \Pi_{\text{inlet}} P_{\text{amb}} \left[ 1 + \left( \frac{\gamma_{\text{comp}}^{-1}}{2} \right) M_{\text{flight}}^2 \right] $$  \hspace{1cm} (3.69)
Where $\Pi_{\text{inlet}}$ takes into consideration the pressure drop associated to the flow evolution inside the inlet, function of its design and manufacturing (i.e., friction with walls, shape, etc.). $\Pi_{\text{dyn}}$ accounts for the potential pressure drop in the air stream before getting into the inlet, function of flight Mach and total air mass flow (typically considered to be 1 for subsonic regimes).

And last, but not least, it could be possible to estimate the FAR in the CC with the equation of energy (simplified in this case, for the sake of simplicity) applied to the chemical reaction that takes place in the CC:

$$\text{FAR} \eta_{\text{comb}} \text{LHV} \cong (1 + \text{FAR}) \left[ C_p, \exp \left( T_4\text{t} - T_{t,\text{ref,prod}} \right) - C_p, \text{comp} \left( T_{3\text{t}} - T_{t,\text{ref,react}} \right) \right]$$ \hfill (3.70)

Where $\eta_{\text{comb}}$ represents the efficiency of the combustion process in the CC, meaning the heat liberated from the total chemical energy contained by the fuel injected, which is given by the Low Heating Value of the fuel (LHV, applicable when the products of the chemical reaction in the CC contain vaporized water, and typically equal to 43 MJ/kg for JET-A). The different reference temperatures for products and reactants are required to evaluate their total enthalpy variation in the chemical reaction hosted in the CC. The expression given by Eq. (3.69) is considerably simplified considering the full enthalpic balance from where it came from, and the calculation in PROOISIS® will certainly provide more accurate results, but it still can be retained to understand how the fuel consumption (and therefore the TSFC) can be estimated, obtaining then higher or lower accuracies depending on the accepted simplifications.

So finally, the expression for the ISP in a turbofan engine, and the expression for the gross thrust (the net thrust is obtained by subtracting the convective terms associated to the flight speed), are given below:

$$\text{ISP} = \frac{\text{Thrust}}{W_{\text{core}} + W_{\text{BP}}}$$ \hfill (3.71)

$$\text{Thrust} = \left\{ C_x \cdot W_{\text{out}} \cdot V_{\text{out}} + A \cdot (P_{\text{Sout}} - P_{\text{amb}}) \right\}_{\text{core}} + \left\{ C_x \cdot W_{\text{out}} \cdot V_{\text{out}} + A \cdot (P_{\text{Sout}} - P_{\text{amb}}) \right\}_{\text{BP}}$$ \hfill (3.72)

In the previous formula, the static pressure difference term at the bypass duct outlet section is often null because, as it was commented before, the exhaust nozzle in the secondary flow is usually working at an adapted condition, see Eq. (3.44). Certainly, the Fan will move a great amount of air, but compressing it moderately to optimize the propulsive efficiency, so the exhaust pressure will not be that high in the bypass duct comparing with the primary flow, where the critical conditions in the exhaust will be normally met, even for relatively low power regimes. That circumstance simplifies the calculations, but it is still required to obtain the flow velocities at both exhaust nozzles and the static pressure for the primary flow’s outlet.

The flow velocity was already developed from Eq (3.42), depending on the conditions for both nozzles. The values for the different thermodynamic magnitudes at each station (like enthalpies, pressures or temperatures) were calculated in PROOISIS®, for the on-design condition, with certain data provided from the reference engine which design was known in some detail, and with the fluid model tables uploaded in the General Model Data module, together with the rest of equations and assumptions in the model that interrelate those different cycle variables, such as mass flow conservation equations between modules (including...
bleeds) or the mechanical power coupling equations between compressors and turbines, among others. The cycle was solved, from the inlet (where the ambient and flight conditions are known and the pressure losses will be considered constant), going component by component, advancing station by station, until reaching to the exhaust nozzles, where the velocities and static pressures will allow to obtain $I_{SP}$, and eventually the thrust (the TSFC will be obtained in the process):

$$h_t = h(\text{fluid}, T_t, \text{FAR}, \text{WAR}) \quad (3.73)$$

$$T_t = t(\text{fluid}, h_t, \text{FAR}, \text{WAR}); \text{ (or given)} \quad (3.74)$$

$$P_t = p(\text{fluid}, T_t, \text{FAR}, \text{WAR}); \text{ (or given)} \quad (3.75)$$

$$\text{Mach} = \text{mn}(\text{fluid}, T_t, V) \quad (3.76)$$

$$\text{PR} = \text{PR}_{design} \quad (3.77)$$

$$\text{PQ} = \text{PQ}_{design} \quad (3.78)$$

$$\eta_{\text{comp}} = \eta_{\text{comp,design}} \quad (3.79)$$

$$\eta_{\text{turb}} = \eta_{\text{turb,design}} \quad (3.80)$$

$$W_{\text{core}} = W_{\text{LPC}} + W_{\text{FAN, bld}} \quad (3.81)$$

$$W_{\text{BP}} = W_{\text{core}} \cdot \text{BPR} \quad (3.82)$$

$$W_{\text{LPC}} = W_{\text{CC}} + W_{\text{HPC, bld}} + W_{\text{CDP, bld}} \quad (3.83)$$

$$W_{\text{HPT}} = W_{\text{CC}} + W_{\text{CDP, bld}} \quad (3.84)$$

$$W_{\text{LPT}} = W_{\text{HPT}} + W_{\text{HPC, bld}} \quad (3.85)$$

$$W_{\text{out, core}} = W_{\text{HPT}} \quad (3.86)$$

$$W_{\text{out, BP}} = W_{\text{BP}} + W_{\text{FAN, bld}} \quad (3.87)$$

So, certain design values required by the SW, like pressure ratios in compressors and turbines, TIT, BPR, or components’ efficiencies, drove to the determination of the searched design point. The size of the engine (meaning areas required to manage the air flow through the different components to get the desired thrust) is finally determined with the propulsive needs (i.e., thrust required). The size of the engine will be determined, station by station, with the on-design problem.

Typical cruise conditions at 35,000 ft (10,668 m) and a flight Mach number of 0.8, on a standard day, were taken as the reference point for a representative commercial flight. Some other values, regarding efficiencies or pressure ratios, were obtained from Igie et al., 2014, [144].
The rest of the inputs for the calculation of engine’s design point, necessary to establish the scaling factors for all the turbomachinery maps (i.e., maps that will be used to determine the performance of the engine’s components during off-design calculations), are shown in Table 7.

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Flight Conditions</td>
<td></td>
</tr>
<tr>
<td>Altitude (m)</td>
<td>10,668</td>
</tr>
<tr>
<td>Mach Number</td>
<td>0.8</td>
</tr>
<tr>
<td>Engine General Data</td>
<td></td>
</tr>
<tr>
<td>TIT (K)</td>
<td>1,400</td>
</tr>
<tr>
<td>BPR</td>
<td>5.0</td>
</tr>
<tr>
<td>Inlet Mass Flow (kg/s)</td>
<td>355</td>
</tr>
<tr>
<td>N1 (rpm)</td>
<td>4,000</td>
</tr>
<tr>
<td>N2 (rpm)</td>
<td>10,000</td>
</tr>
<tr>
<td>Fan Data</td>
<td></td>
</tr>
<tr>
<td>$\beta_{\text{CORE}}$ – coordinate</td>
<td>0.50</td>
</tr>
<tr>
<td>$\beta_{\text{SEC}}$ – coordinate</td>
<td>0.50</td>
</tr>
<tr>
<td>$n_{\text{CFAN,CORE}}$</td>
<td>0.95</td>
</tr>
<tr>
<td>$n_{\text{CFAN,SEC}}$</td>
<td>0.95</td>
</tr>
<tr>
<td>$\text{PR}_{\text{FAN,CORE}}$</td>
<td>1.80</td>
</tr>
<tr>
<td>$\text{PR}_{\text{FAN,SEC}}$</td>
<td>1.60</td>
</tr>
<tr>
<td>$\eta_{\text{FAN,CORE}}$</td>
<td>0.88</td>
</tr>
<tr>
<td>$\eta_{\text{FAN,SEC}}$</td>
<td>0.89</td>
</tr>
<tr>
<td>Booster Data</td>
<td></td>
</tr>
<tr>
<td>$\beta$ – coordinate</td>
<td>0.50</td>
</tr>
<tr>
<td>$N_{\text{CLPC}}$</td>
<td>0.95</td>
</tr>
<tr>
<td>$\text{PR}_{\text{LPC}}$</td>
<td>3.00</td>
</tr>
<tr>
<td>$\eta_{\text{LPC}}$</td>
<td>0.88</td>
</tr>
<tr>
<td>HPC Data</td>
<td></td>
</tr>
<tr>
<td>$\beta$ – coordinate</td>
<td>0.50</td>
</tr>
<tr>
<td>$N_{\text{CHPC}}$</td>
<td>0.95</td>
</tr>
<tr>
<td>$\text{PR}_{\text{HPC}}$</td>
<td>6.60</td>
</tr>
<tr>
<td>$\eta_{\text{HPC}}$</td>
<td>0.87</td>
</tr>
<tr>
<td>HPT Data</td>
<td></td>
</tr>
<tr>
<td>$\zeta$ – coordinate</td>
<td>0.50</td>
</tr>
<tr>
<td>$N_{\text{CHPT}}$</td>
<td>1.00</td>
</tr>
<tr>
<td>$\eta_{\text{HPT}}$</td>
<td>0.92</td>
</tr>
<tr>
<td>LPT Data</td>
<td></td>
</tr>
<tr>
<td>$\zeta$ – coordinate</td>
<td>0.50</td>
</tr>
<tr>
<td>$N_{\text{CLPT}}$</td>
<td>1.00</td>
</tr>
<tr>
<td>$\eta_{\text{LPT}}$</td>
<td>0.91</td>
</tr>
</tbody>
</table>

Table 7: Parameters for the calculation of the design point of a turbofan engine like the CF6-80E1A2.

At the end of the on-design analysis, the required values of $R_1$, $P_0$, $T_1$, Mach number, flow velocities, and specific thrusts were determined throughout the engine, in each module and at each station, but for one operational point only. With the full determination of the size of the engine, the effective areas at the different stations were also fully determined. With a known geometry and the rest of the machine’s technological information available (i.e., data contained in the already scaled maps, the maximum TIT achievable in the CC, etc.), it is possible to start the calculations to test the machine at any other operational condition, inside certain limits given by the technical capabilities of the engine. That is the off-design analysis.

There are several considerations to make regarding the functioning of the turbofan outside the design point:

- As indicated before, the turbines and the nozzles will typically work chocked, excepting the bypass duct nozzle which could be adapted, like in Eq. (3.44). This situation introduces a strong constraint in the way the engine works because the corrected mass flow parameters in the choked stations will be fixed and will have a value determined by their crossed-section areas (calculated during the on-design phase), and the gas flow properties in those
stations. This means the only degree of freedom in the engine will be given by the angular position of the thrust lever (TLA, which controls the TIT by the fuel injected in the CC) managed by the pilot in the cockpit, given certain flight conditions. The HPT will work choked in most of the regimes in which the engine operates, even for relatively low loads. The LPT and the primary exhaust nozzle will need to be at a certain power load to be choked, but such power load is always met during cruise flight. In this sense, for the cruise phase under analysis (the flight phase of highest interest for this study), HPT, LPT, and primary nozzle will be considered to remain always choked. And then, depending on flight conditions and engine’s regime, the secondary nozzle would be also choked. Some engines count with VG in the exhaust nozzles, but that will not be the case for this study as most of the airliners count with fixed convergent exhaust nozzles, both primary and secondary.

- The previous constraint determines the behavior of the compressors and Fan because of the mechanical coupling relationships in both the high-speed shaft and the low-speed shaft, which balance the mechanical power produced in the turbines during the expansion of the hot gases from the CC with the power required to compress the air. The rotational speed in the LPT will be the same than in the booster or Fan, as they are all mechanically connected (see Figure 43). Both HPT and HPC will rotate, coupled as well, at the same revolutions per minute (rpm). Some engines count with geared Fans, introducing thus a variation to this principle, but this technology is not present in the engine taken as reference. When the pilot modifies the TLA, with turbines and nozzles choked, the constraints introduced in those stations determine the compression ratio, the efficiencies, and the mass flows (therefore also the BPR) through compressors and Fan. This is perfectly visible in the turbomachinery maps where the previous constraints and relationships among engine components are reflected in the so-called “running lines” of the engine. The design point (or on-design condition) is just a point of such operating lines.

- When it is stated that the only degree of freedom in the engine will be the TLA, that does not mean exactly that the flight conditions (altitude, \( M_{\text{flight}} \), \( T_{\text{amb}} \), \( P_{\text{amb}} \)) are not relevant anymore. For each set of flight conditions, there is initially a different running line. However, all those lines, for different flight conditions, tend to coincide when they are represented in a turbomachinery map together at certain elevated regimes, basically when the choking conditions are met in the different turbines and nozzles. So, in the practice, even when there are different lines (and this is particularly clear at low regimes where the choking constraints disappear) all of them tend to share points at medium-to-high engine regimes.

- From a computational point of view, as it was indicated before, it is possible to calculate the off-design problem with algebraic expressions. PROOSIS® gives that option to the user. However, it is preferred the use of maps when they are available. And this leads to a thermodynamic matching model (a simplified matching diagram for a two-spool turbofan engine is available in [296]). Once a component’s geometry is fixed, its map is unique (unless it counts with variable geometry, feature that will not apply during cruise flights when the engine’s performance is kept stable), representing performance information at every feasible off-design conditions. For a given
engine, working at a certain operating condition (meaning altitude, ambient conditions, speed, and TLA), its operating point is unique, and this is verifiable in any component map. The different components’ maps will be connected or matched by the principles and constraints already mentioned. Determining a single point in the running line needs of successive guesses of points on the components’ maps. There will be typically a final closing condition (for instance a mechanical coupling equation), that provides an error that must be minimized to find a solution to the matching problem. And that must be repeated at different TLA positions (i.e., TIT values) to obtain the running line of the engine, concluding thus the off-design analysis. That matching model implies a highly iterative process that PROOSIS® numerically solves with refined methods (i.e., improved Powell’s hybrid-based methods [73]). Ideally, at the end of the matching process, the guesses and the solution will coincide. In the practice, it is established a convergence level to guarantee a sufficient accuracy.

- So, once the off-design problem is solved, it is possible to evaluate what would happen to the engine if its H&Q parameters may vary because of some sort of degradation during a cruise flight phase. That will be the final aim of the study. Every single time an off-design result would be needed, it would be just a matter of calling to PROOSIS®, which will contain already a validated model of the engine under study, to obtain the required values that will be fed with the measurements retrieved from the instrumentation mounted in the engine. The thermodynamic variables at different engine stations should ideally coincide with the readings of the sensors installed in those same stations of the real engine. The combination of the verified model with the real measurements will determine the degradation of the engine and the control regime of the engine. Analyzing the difference that might exist between theoretical and real TIT, some strategy to improve engine’s overall efficiency could be set up.

3.3.3. Off-design calculations

In this sense, the model implemented in PROOSIS® follows a well-known theoretical approach that has been studied thoroughly during the last decades. For the off-design problem of a two-spool turbofan engine those main principles can be explained more concisely by using the mathematical expressions that interrelate the different variables involved. For the sake of completeness, those expressions will be provided now, the same way it was done with the on-design problem detailed before, accepting several simplifying assumptions like 1D model, constant efficiencies, choked turbines, and choked primary flow exhaust nozzle.

One of the most relevant expressions to use in the problem is the one to obtain the air mass flow in the different gas turbine stations:

\[
\frac{W_{\sqrt{T_z} P_t}}{\frac{\sqrt{R}}{\sqrt{R}}} = \sqrt[\gamma]{M} \left( 1 + \frac{\gamma - 1}{2} M^2 \right)^{\frac{\gamma}{\gamma - 1}} \frac{A}{\sqrt{R}} = \Gamma(\gamma, M) \frac{A}{\sqrt{R}} \tag{3.88}
\]

When the turbines and the primary flow exhaust nozzle work choked, the mathematical expressions that represent their respective conditions are as follows:
Where “bld” indicates the airflow’s fraction that was bled during the compression and was not recuperated afterwards. For the sake of simplicity in the previous expressions, the reinjection of air bled for cooling purposes to the HPT will be taken as negligible compared with the main airflow going through the CC. However, it is necessary to clarify that PROOSIS® do consider such bled airflow reincorporation to the primary flow, contributing to the propulsive jet exhausted through the primary flow’s exhaust nozzle. The SW models it the way it will be explained later in this section.

FAR represents the overall fuel-to-air ratio (mass flow of fuel added in the CC through the fuel nozzles). \( \Gamma_{\text{exp}} \) stands for the value of the function \( \Gamma(\gamma, M) \) that is typically considered during expansion in these engines, taking into consideration a choked section \((M=1)\), and hot gases \((\gamma_{\text{exp}} \approx 1.33)\). The representative cross-sectional areas for each module are given by \( A_{\text{HPT}}, A_{\text{LPT}}, \) and \( A_{\text{out,core}} \), respectively. In this sense, the mass flow parameters considered in the degradation vector \( \vec{X} \) of the inverse problem, relative to the HPT and LPT, are applied to their respective areas (multiplying factors, similarly used in the rest of engine components, so the effective areas would be: \( A_{\text{eff,i}} = \Gamma_i A_i \)) to account for the erosion suffered typically in those parts of the engine (i.e., \( \Gamma_{\text{HPT}} \) and \( \Gamma_{\text{LPT}} \), model tuning factors, not to be confused with \( \Gamma_{\text{exp}} \), factor that counts with its own specific physical meaning). The stations follow the nomenclature shown in Figure 49. The attention of reader must be focused on the constant value of these 3 expressions, given by \( K_{\text{HPT}}, K_{\text{LPT}}, \) and \( K_{\text{out,core}} \) respectively. Operating the engine under such conditions, typically met at demanding regimes (i.e., cruise, take-off, top of climb), implies a very strong condition among the different variables relative to HPT, LPT, and principal flow’s exhaust nozzle. The mechanical coupling conditions with LPC, HPC, and Fan will complete the determination of engine’s performance for a particular value of \( T_{4t} \) (i.e., TIT, variable controlling engine’s regime), and a certain flight condition. This statement becomes clearer when dividing the previous expressions:

\[
\frac{T_{4t}}{T_{4st}} \frac{P_{4t}}{P_{4st}} = \frac{K_{\text{HPT}}}{K_{\text{LPT}}} \tag{3.92}
\]

\[
\frac{T_{4st}}{T_{5t}} \frac{P_{4st}}{P_{5t}} = \frac{K_{\text{LPT}}}{K_{\text{out,core}}} \tag{3.93}
\]

When those expressions are combined with the ones relative to the efficiencies in HPT and LPT (i.e., the same efficiencies considered for the inverse problem, this is \( \eta_{\text{HPT}}, \) and \( \eta_{\text{LPT}}, \) which are taken as constant here for this simplified mathematical exposition of the main principles ruling engine’s performance), the following remarkable set of equations is obtained:
\[ \eta_{HPT} = \frac{1-T_{45t}/T_{4t}}{1-(P_{45t}/P_{4t})(\gamma\exp-1)/\gamma\exp} \]  
(3.94)

\[ \eta_{LPT} = \frac{1-T_{5t}/T_{45t}}{1-(P_{5t}/P_{45t})(\gamma\exp-1)/\gamma\exp} \]  
(3.95)

\[ \frac{T_{45t}}{T_{4t}} = \alpha \]  
(3.96)

\[ \frac{T_{5t}}{T_{45t}} = \beta \]  
(3.97)

\[ \frac{P_{45t}}{P_{4t}} = \alpha_p \]  
(3.98)

\[ \frac{P_{5t}}{P_{45t}} = \beta_p \]  
(3.99)

Being \( \alpha, \beta, \alpha_p, \) and \( \beta_p \) constant values, independently of regime and flight conditions. The relationships in the last 4 equations do affect to the compressors and Fan by means of the transfer of mechanical power through the 2 different spools in the engine, given by Eq. (3.65) and Eq. (3.67) respectively, and developed now assuming negligible both mechanical losses and auxiliaries power demand:

\[ W_{\text{core}C_{p,\text{comp}}}(1 + \text{BPR})(T_{3t} - T_{25t}) + (T_{25t} - T_{13t})) = (1 - \text{bld})(1 + \text{FAR})W_{\text{core}C_{p,\exp}C_{p}}(1 - \beta) \]  
(3.100)

\[ W_{\text{core}C_{p,\text{comp}}}(T_{3t} - T_{25t}) = (1 - \text{bld})(1 + \text{FAR})W_{\text{core}C_{p,\exp}C_{p}}(1 - \alpha) \]  
(3.101)

Resulting evident the relationship between the mechanical power demanded by the compressors and Fan, and the TIT. Then, combining Eq. (3.101) with the equation that establishes the mass flow conservation throughout the Core Engine, and being \( \Pi_{CC} = P_{4t}/P_{3t} \) to the pressure ratio in CC (i.e., pressure drop to be accounted for):

\[ \frac{W_{\text{core}C_{p,\text{comp}}}}{P_{25t}} \sqrt{T_{25t}/P_{25t}} = \frac{W_{\text{core}C_{p,\text{comp}}}}{P_{4t}} \sqrt{\frac{T_{25t}}{P_{4t}} \frac{P_{4t}}{P_{25t}}} = \frac{K_{HPT} \Pi_{CC}}{(1 - \text{bld})(1 + \text{FAR})} \sqrt{\frac{T_{25t}}{T_{4t}} \Pi_{HPC}} \]  
(3.102)

It is possible to obtain one expression for the running line of the engine (caution must be taken with the accepted assumptions to obtain the previous equations and their implications, as this result constitutes a simplified approach to the full off-design problem PROOSIS® will solve) that could be depicted in Figure 45, once applied adequate corrections to the following expression (to work with \( W_C \)):

\[ \frac{W_{\text{core}C_{p,\text{comp}}}}{P_{25t}} \sqrt{T_{25t}/P_{25t}} = \frac{K_{HPT} \Pi_{CC}}{(1 - \text{bld})(1 + \text{FAR}) \sqrt{C_{p,\exp}C_{p,\text{comp}}}} (1 - \alpha) \eta_{HPC} \frac{\Pi_{HPC}}{\Pi_{HPC}} \left(\gamma_{\text{comp}} - 1\right) \]  
(3.103)

Being \( \Pi_{HPC} = \text{PR} \) in that case. So, the problem relative to the Core Engine would be fully determined this way. However, the portion of the engine relative to the low-speed spool, including the Fan, would need of some additional development to be solved. The Fan introduces a specific variable: BPR. To find its value, it will be
required to analyze the evolution of the airflow in the bypass (BP) duct afterwards the Fan. The BP duct’s exhaust nozzle could work either in critic or adapted conditions as per equations Eq. (3.45) and Eq. (3.44), respectively, depending on the value of NPR. Assuming, for the sake of simplicity, that the BP duct nozzle works in critic conditions, the following expression should be used (for adapted conditions it would be required to impose the condition \( P_{13t} = P_{amb} \) instead):

\[
\text{BPR} \frac{\Pi_{\text{core}T_{13t}}}{P_{13t}} = \Gamma_{\text{comp}} A_{\text{out,BP}} \frac{\sqrt{\Pi}}{\sqrt{R}} = K_{\text{out,BP}}
\]  

(3.104)

Where the applicable cross-sectional area for the BP duct exhaust nozzle, and the value of \( \Gamma(\gamma, M) \) typically considered during compression in these engines, with choked section (\( M=1 \)) and moderate temperatures (\( \gamma_{\text{comp}} \)), are used. Considering, as per Figure 49, that conditions after Fan are applicable for booster’s intake, which PR is then \( \Pi_{\text{LPC}} = P_{25t}/P_{13t} \), and then dividing Eq. (3.104) by Eq. (3.89):

\[
\eta_{\text{LPC}} = \frac{(P_{25t}/P_{13t})^{(\gamma_{\text{comp}}-1)/\gamma_{\text{comp}}}}{(T_{25t}/T_{13t})^{-1}}
\]  

(3.105)

\[
\text{BPR} \frac{\Pi_{\text{core}T_{13t}}}{(1-\text{bld})(1+\text{FAR})} \Pi_{\text{CC}} \Pi_{\text{HPC}} \sqrt{T_{4t}/T_{25t}} \Pi_{\text{LPC}} \sqrt{1 + \frac{\Pi_{\text{LPC}}^{(\gamma_{\text{comp}}-1)}}{\eta_{\text{LPC}} \gamma_{\text{comp}} - 1}} = \frac{K_{\text{out,BP}}}{K_{\text{HPT}}}
\]  

(3.106)

Equation that allows obtaining BPR as a function of \( \Pi_{\text{HPC}} \) and \( \Pi_{\text{LPC}} \). Once this specific variable of the Fan is determined, the rest of the off-design problem in the two-spool turbofan can be solved leaving one acting variable, typically \( T_{4t}/T_{2t} \), which establishes a useful relationship between regime and flight conditions. So, summarizing the unknowns and equations to solve the two-spool turbofan problem:

- 9 Unknowns: \( \Pi_{\text{FAN}} \), \( \Pi_{\text{LPC}} \), \( \Pi_{\text{HPC}} \), BPR, \( T_{13t}/T_{2t} \), \( T_{25t}/T_{13t} \), \( T_{3t}/T_{25t} \), \( T_{25t}/T_{2t} \), and finally \( T_{4t}/T_{2t} \), which will remain as acting variable.
- 2 mechanical coupling equations: given by Eq. (3.100) and Eq. (3.101).
- 1 bypass nozzle equation: shown in Eq. (3.104).
- 3 equations for the thermal efficiencies in Fan, LPC, and HPC.
- 2 additional equations for relationships between temperature ratios.

This system is completed by equations for choking at HPT, LPT and primary exhaust nozzle. It is considered given the efficiencies of the different modules, properties of gas in compression and expansion, FAR, the proportion of bled air not reinjected into the main flow, PR share between Fan and LPC in primary flow (i.e., \( \Pi_{\text{FAN}} = \mu \Pi_{\text{LPC}} \), with \( 0 < \mu < 1 \)), and cross-sectional areas in turbines and nozzles:

1. \[
\frac{T_{4t}}{T_{25t}} = \frac{C_{p,\text{comp}}}{C_{p,\text{exp}}} \left( \frac{\Pi_{\text{HPC}}^{(\gamma_{\text{comp}}-1)/\gamma_{\text{comp}} - 1}}{(1-\text{bld})(1+\text{FAR})(1-\alpha)\eta_{\text{HPC}}} \right)
\]

(3.107)

2. \[
(1 + \text{BPR}) \left( \frac{T_{13t}}{T_{2t}} - 1 \right) + \frac{T_{13t}}{T_{2t}} \left( T_{25t}/T_{13t} - 1 \right) = \frac{C_{p,\text{exp}}}{C_{p,\text{comp}}} (1 - \text{bld})(1 + \text{FAR})\alpha(1 - \beta) \frac{T_{4t}}{T_{2t}}
\]

(3.108)

3. \[
\text{BPR} = \frac{(1-\text{bld})(1+\text{FAR})(K_{\text{out,BP}}/K_{\text{HPT}})}{\Pi_{\text{LPC}} \Pi_{\text{HPC}} \Pi_{\text{CC}}} \frac{T_{25t}}{T_{13t}} \sqrt{T_{4t}/T_{25t}}
\]

(3.109)
The way of solving the previous system, knowing regime (TIT) and flight conditions ($T_{2t}$), is by solving first equations Eq. (3.107), Eq. (3.108), Eq. (3.109), and Eq. (3.114), incorporating $\Pi_{FAN} = \mu \Pi_{LPC}$. Variables in that partial system (i.e., $\Pi_{FAN}$, $\Pi_{LPC}$, $\Pi_{HPC}$, and BPR) are considerably coupled, but variable substitution is possible to just leave $\Pi_{FAN}$ in equation Eq. (3.108). $\Pi_{FAN}$ is obtained by using a solver (e.g., Newton). $\Pi_{LPC}$ is calculated with $\mu$. $\Pi_{HPC}$ is obtained from Eq. (3.114), and finally BPR from Eq. (3.109). Rest of variables are calculated directly with efficiencies and temperatures' ratios. With mass flow continuity equations between different engine components, it is possible to represent the running line for engine’s operation over the different compressor or turbine maps once the appropriate correction factors are applied. For instance, in the case of Fan:

$$
\frac{T_{25t}}{T_{2t}} = \frac{T_{25t}}{T_{13t}} \frac{T_{13t}}{T_{2t}}
$$

(3.113)

With BPR, $T_{25t}/T_{2t}$, and $W_{core} \sqrt{T_{25t}/P_{25t}}$ from Eq. (3.102), it is possible to get the relationship between $\Pi_{FAN}$ and mass flow parameter at Fan intake. And similarly with the rest of engine components and their associated maps. Working with those maps, instead of equations, implies making some initial assumption in one of them, for instance the map of the Fan, choosing one point as a candidate to be part of the running line. That point, together with regime, flight conditions, and the rest of maps and relationships, initiates an algorithm that ends up in one of the mechanical coupling equations. That last equation will be closing the process ($error \rightarrow 0$). That iterative process must be repeated for every candidate point, typically at least one per each constant corrected speed line. PROOSIS® accurately determines the lines with a fast numerical method.

Returning to the specific problem for the present study, the different turbomachinery component maps provided by the TURBO® library [10] were properly scaled with data in Table 7 for the specific engine model used as reference. Maps used in the Fan are shown in Figure 50 and Figure 51, respectively. Maps of booster and HPC are provided in Figure 52 and Figure 53, respectively. And, finally, the maps of both turbines are given in Figure 54 and Figure 55, respectively. Position of design points and associated running lines have been indicated.

$$
\frac{(1+BPR)W_{core} \sqrt{T_{25t}/P_{25t}}}{T_{2t}} = (1 + BPR) \frac{W_{core} \sqrt{T_{25t}/P_{25t}}}{P_{25t}} \sqrt{T_{25t}/T_{2t}} \left(1 + \frac{1}{\mu}\right) \Pi_{FAN}
$$

(3.115)
Figure 50: FAN map (core), as defined in PROOSIS®.

Figure 51: FAN map (bypass), as defined in PROOSIS®.
Figure 52: LPC map (booster), as defined in PROOSIS®.

Figure 53: HPC map, as defined in PROOSIS®.
Figure 54: HPT map, as defined in PROOSIS®.

Figure 55: LPT map, as defined in PROOSIS®.
3.3.4.- Model validation

Table 8 summarizes the comparisons established between the available data and the results obtained with PROOsis®, at steady-state, assessing the validity of the model developed at different off-design conditions, both for the non-degraded engine condition and for one degraded engine scenario documented in [144], where a 5% reduction in the fan flow capacity and a 2% in the fan efficiency were imposed. Differences between data and model results were always below a 5%:

<table>
<thead>
<tr>
<th>Case Number</th>
<th>Flight Condition (Control Setting)</th>
<th>Performance values for the comparison</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>New and Clean engine condition</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Net Thrust (kN) (Model)</td>
</tr>
<tr>
<td>1</td>
<td>Cruise (TIT = 1,400 K)</td>
<td>55.7</td>
</tr>
<tr>
<td>2</td>
<td>Take-off (TIT = 1,620 K)</td>
<td>77.4</td>
</tr>
<tr>
<td>3</td>
<td>Top of Climb (TIT = 1,620 K)</td>
<td>296.9</td>
</tr>
</tbody>
</table>

| Degraded engine: 5% reduction in the fan flow and 2% of reduction in fan efficiency |
|---------------------------------|---------------------------------|---------------------------------|---------------------------------|
| TIT (K) (Model) | TIT (K) (Data) | Δ (%) | TSFC (g/kN·s) (Model) | TSFC (g/kN·s) (Data) | Δ (%) |
| 4  | Cruise (same thrust than in Case 1) | 1,411 | 1,417 | 0.4 | 17.44 | 17.43 | 0.1 |
| 5  | Take-off (same thrust than in Case 2) | 1,652 | 1,637 | 0.9 | 18.99 | 18.36 | 3.4 |
| 6  | Top of climb (same thrust than in Case 3) | 1,629 | 1,637 | 0.5 | 10.34 | 10.30 | 1.4 |

Table 8: Model validation: comparative between engine data and model results. Data obtained from [144].

It is interesting to verify how, because of the degradation, the demand for a constant thrust leads to an increase in TIT. Theory and reality were found coordinated, so the more degraded the engine is, the lower its efficiency will be, the more fuel (i.e., energy) will be needed and, consequently, also the higher the temperature will be in CC and turbines, contributing this to increase the degradation rate in the whole engine. A nice agreement was found in general (errors < 5%), the expected trends were captured correctly, and therefore it can be affirmed that the model created represents, with enough fidelity, the real engine used as reference.

The scope of this research is not constrained to a specific engine. The underlying physics remain identical, no matter the gas turbine model. If a different turbofan was selected, then the data for the design point, as summarized in the Table 7, would change, but the behavior of the model would be practically identical. The turbomachinery maps, the greatest sources of non-linearities for compressors and turbines (and probably the most preserved intellectual property for the OEMs), are not so different from one to another, qualitatively speaking, and the system configuration for every turbofan engine is similar. Therefore, results obtained from PROOsis® could be generalized to any turbofan engine with a standard configuration like the one used for the CF6-80E1A2, and results could be considered extensible to any other similar engine in the market once its model was calibrated.
As the figure of merit to validate the model, the values of thrust-specific fuel consumption (TSFC), exhaust gases temperature, T₅ₜ (EGT) and net thrust can be evaluated in the Table 9. Based on the data available in [144], a value of 17.0 g/kN·s is expected for the TSFC and a value of 600 K for the EGT, while the engine is reportedly producing 53.3 kN of net thrust. The model matches very closely the TSFC and EGT data, with less than 0.25% relative error on the former and 0.5% on the latter, although slightly greater discrepancies are found for the net thrust value. This is a logical result, as the net thrust (an extensive magnitude) is related to the size of the engine, while the TSFC and EGT depend only on the characteristics of the thermodynamic cycle and are, therefore, more useful for the purposes of assessing the validity of this model in terms of capturing the behavior of the engine when degradation takes place. For this reason, more emphasis has been put on accurately modeling the intensive characteristics of the engine than on its actual size.

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Intensive Quantities</strong></td>
<td></td>
</tr>
<tr>
<td>EGT (K)</td>
<td>597</td>
</tr>
<tr>
<td>TSFC (g / kN·s)</td>
<td>17.04</td>
</tr>
<tr>
<td><strong>Extensive Quantity</strong></td>
<td></td>
</tr>
<tr>
<td>Net Thrust (kN)</td>
<td>55.7</td>
</tr>
</tbody>
</table>

Table 9: Model outputs for the design point at cruise conditions.

Some final comments and explanations were still pending, regarding different topics on the model built in PROOSIS®, to complete the overall summary in this chapter about the way the SW provides the required outcome:

- Given the steady condition of the reference engine for this study, it was assumed its warm-up was completed and the usual cruise conditions were reached for the analysis. Therefore, the machine was supposed to be inside thermal equilibrium conditions and no tip clearance effects would be initially accounted for. Additionally, both LPC and HPC were supposed to have fixed geometry under such steady regime, and the effects of variable geometry systems typically installed in the engines under study, such as VSVs or VBVs, were not considered in the compressor maps.

- Reynolds number (Re = ρVl/ũ) and gamma (γ = Cₚ/Cᵥ) corrections were included in the model, and the pressure ratios, efficiencies, corrected mass flows and corrected rotational speed values read from the turbomachinery maps were modified accordingly by the SW. The effect of the Reynolds number on the flow circulating around the turbomachinery internal surfaces is a matter that has deserved attention in the last decades (detailed analysis on the effects of these parameters in air flows can be found in [255] or [278]). The values of efficiencies and corrected mass flows taken from the compressor maps were modified in PROOSIS® with the Reynolds Number
Index (RNI = Re/Re_ref), as per equations given in Eq. (3.116) and Eq. (3.117), respectively. The equation Eq. (3.116) represents a linear interpolation, for a given RNI, between values f₁ and f₂, corresponding to RNI₁ and RNI₂, respectively, when the RNI is plotted on a logarithmic scale.

\[ \eta = \frac{f_1 \log(RNI_2, RNI) + f_2 \log(RNI, RNI_1)}{\log(RNI_2, RNI_1)} \]  \hfill (3.116)

\[ \frac{W_c}{W_{c, \text{map}}} = \sqrt{\eta \eta_{\text{map}}} \]  \hfill (3.117)

The parameters for the interpolation (i.e., \( f_1, f_2, \text{RNI}_1, \) and \( \text{RNI}_2 \)) were determined empirically (they can be found in the map file). RNI for any given, non-standard condition, was calculated with equation Eq. (3.118), so given the dynamic viscosity, \( \mu \), and \( T_\text{t} \), the Reynolds effect could be accounted for:

\[ \text{RNI} = \frac{\delta_t}{\sqrt{\theta _t}} \frac{\mu_{\text{ref}}}{\mu} \]  \hfill (3.118)

The corrected magnitudes, \( \theta_t = T_t/T_{\text{ref}}, \delta_t = P_t/P_{\text{ref}}, \) and the dynamic viscosity, \( \mu \), are evaluated at the inlet of the module where the correction will be done. The reference terms to obtain the corrected values coincide with the standard temperature (\( T_{\text{std}} = 288.15 \) K) and standard pressure (\( P_{\text{std}} = 101,324 \) Pa) at sea level, respectively. The corrected mass flows are obtained immediately once those magnitudes are obtained:

\[ W_{c, \text{corrected}} = \frac{W_c}{\delta_t} \]  \hfill (3.119)

The effects associated to changes of the specific heat coefficient \( \gamma \) on the turbomachinery flow constitute a matter of more detailed and recent study, that has arisen given the need to adapt existing turbomachinery maps to very different working fluids, some of them clearly outside the ideal gas behavior (supercritical \( \text{CO}_2 \), for instance, which typically works very close to its critical point during compression) and thus will need significant corrections to be made to any turbomachinery map (see, for instance [243] and [284]). Because of the inclusion of these effects into the model, pressure ratios, efficiencies, corrected mass flows and corrected rotational speed values need to be modified according to expressions Eq. (3.120), Eq. (3.121), Eq. (3.122) and Eq. (3.123), respectively, where \( \gamma \) and \( R \) are evaluated at the component’s inlet:

\[ \frac{P_R^{(\gamma - 1)}}{P_{R, \text{map}}^{(\gamma - 1)}} = \frac{\gamma - 1}{\gamma_{\text{ref}} - 1} \frac{\eta}{\eta_{\text{map}}} \]  \hfill (3.120)

\[ \frac{1 - \eta}{1 - \eta_{\text{map}}} = \frac{\gamma}{\gamma_{\text{ref}}} \]  \hfill (3.121)
\[
\frac{W_c}{W_{c,\text{map}}} = \frac{\gamma R_{\text{ref}}}{\sqrt{R_\gamma R_{\text{ref}}}}
\]  
(3.122)

\[
\frac{N_c}{N_{c,\text{map}}} = \frac{\gamma R}{R_{\text{ref}} \gamma_{\text{ref}}}
\]  
(3.123)

- The engine’s Fan is modeled in the SW, as it was already mentioned, essentially, the same way the LPC or the HPC are. The Fan module counts with the option of bleeding air which can be configured similarly to the optional bleeds in the other compressor modules. The Fan bleed takes air from the outlet of the inner Fan and diverts it to the bypass duct stream, as indicated in Figure 43. However, at 85% relative corrected rotational speeds (\(N_{\text{CFAN}}\)) and above, no air is bled from the primary stream. When \(N_{\text{CFAN}}\) is at 50% or below, 10% of the Fan core air mass flow is bled into the secondary stream. For \(N_{\text{CFAN}}\) values between 50% and 85%, a linear variation of air mass flow bled is established (see [10] for reference).
- As it was done for the compressors, both turbines will be considered of fixed geometry, and variable area turbine nozzles (VATN) will not be included within the model. This is not very common in the industry, given the maintenance problems associated. The turbines in modern gas turbine engines need cooling flows to resist the high working temperatures in the main gas path. There are several ways of dealing with the effective mixing and work distribution of the cooling flows incorporated into the main gas path by the film cooling technology. The modeling approach followed here is the so-called Equivalent Single-Stage Turbine (ESST) as discussed in [164], by which a fraction of the cooling air is assumed to be injected, and mixed (without pressure losses) upstream of the stator vane, hence expanding in the downstream rotor and producing work, while the rest of the cooling flow is assumed to be injected and mixed (without pressure losses) downstream of the rotor blade, thus cooling the main gas path stream after the expansion has taken place and producing no work, as indicated in Figure 56. If the HPT counts with a second stage, all the mixed flow would be producing some work in it. The film cooling is always more necessary in the first stage of HPT, so the cooling flows to be mixed with the main flow are higher as well there.

\[\text{Figure 56: T-s diagram showing an ESST approach in a cooled turbine (images obtained from [164] and [107]).}\]
This approach, just for a single-stage turbine, has already some implications regarding the assessment on what fraction of the cooling flow would effectively produce work and what fraction would not. For multi-stage turbines, evaluating the work-producing potential of each cooling flow, expressed as a percentage, from 0% to 100% of the whole multi-stage turbine, to be able to build an equivalent single stage following this approach, could result a real challenge. For the sake of simplicity, both the HPT and the LPT will be treated in the same way, and it will be assumed that there are only two main cooling flows: the one used for cooling the nozzle guide vanes (NGV), which will be treated as if mixed upstream of the stator vanes, determining the equivalent stator outlet temperature, $T_{41t}$ (SOT), and producing work; and the flow used for cooling the rotor blades (in practice, this would normally include cooling flows for casing and rotor disks too, which usually lack the capability to produce work because they are typically injected with very little momentum), which will be simply mixed downstream of the completed expansion. In this case, 66% of the cooling flow is assumed for NGV cooling and the rest is for the rotor cooling. This method has some drawbacks, mainly the fact that cooling flow pressures are ignored (fluids are simply mixed energetically, without pressure losses, and cooling flows are supposed to be just at the right pressure to be injected into the main gas path by means of bleeding compressor air at the right stage), but it usually leads to more conservative results, as it tends to predict lower values for the equivalent SOT. Based on the graph in Figure 56, nevertheless, it is apparent that the efficiency for any cooled turbine, treated this way, is directly equivalent to the efficiency definition for any non-cooled turbine, corresponding to a simple ratio of real-to-ideal produced work, $(\Delta H_t / \Delta H_{t(is)})$, which helps in establishing reasonable values for the design point (like the ones in the Table 7). In this model, as it is shown in Figure 43, both turbines receive cooling flows from the HPC: a 10% cooling air mass flow is bled at the CDP, which serves for the cooling of the HPT, while an additional 5% cooling air mass flow is bled at an intermediate stage of the HPC (at 40% of the total enthalpy leap in the compressor) is for the LPT.

- The CC, as every other engine component, was treated as adiabatic (heat soakage effects were neglected for steady calculations). Jet-A was used as fuel, with a LHV of 43.1 MJ/kg. Fuel was assumed to be injected at standard conditions, and an efficiency of 99.9% was considered for the combustion process. There was a constant 3% pressure drop within the combustion chamber itself, plus an additional 2% pressure drop at the diffuser situated immediately upstream, just to avoid complicating the model in excess. Some other options are available for a finer setup.
- Similar constant pressure losses were imposed at the compressors inter-ducts (2%), at the turbines inter-ducts (1%), and at the ducts connecting with the exhaust nozzles (an additional 1% for the primary nozzle, and an overall 3% for the by-pass nozzle).
- Both shafts were considered to incur on 0.5% of mechanical losses each, to account for the small fraction of work dissipated in roller and ball bearings.
- Finally, both nozzles were characterized by means of their discharge ($C_d$) and thrust ($C_x$) coefficients, and both were assumed to be equal in this regard. The thrust coefficient is taken as 0.99, constant throughout the whole
operating envelope of the engine, while the discharge coefficient is interpolated from tabulated values, assuming a nozzle angle $\alpha$ of 10°, depending on the NPR, as detailed in Table 10.

<table>
<thead>
<tr>
<th>NPR</th>
<th>1.1</th>
<th>1.2</th>
<th>1.3</th>
<th>1.4</th>
<th>1.5</th>
<th>1.6</th>
<th>1.7</th>
<th>1.8</th>
<th>1.9</th>
<th>2.0</th>
</tr>
</thead>
<tbody>
<tr>
<td>$C_d$</td>
<td>0.9404</td>
<td>0.9487</td>
<td>0.9623</td>
<td>0.9677</td>
<td>0.9720</td>
<td>0.9778</td>
<td>0.9793</td>
<td>0.9797</td>
<td>0.9797</td>
<td>0.9797</td>
</tr>
</tbody>
</table>

Table 10: Nozzle discharge coefficient as a function of the NPR, with a nozzle angle of 10 degrees.

3.3.4.- The interest on the cruise phase

To finalize this chapter, it will be explained the reasons why the emphasis is put over the cruise phase of a commercial flight. Attending to typical flight profiles covered by commercial two-spool turbofan engines, the most representative phases, in terms of fuel consumption, maintenance impact, and RUL reduction, are:

1. Take-off: (TO) This will be the stage, while the aircraft is still in the ground, where the maximum $T_{4t}$ will be reached.
2. Top of climb: (TOC) This will be the stage, once the aircraft is in the air, where the maximum $T_{4t}$ will be reached, obviously for a limited amount of time.
3. Cruise: This phase constitutes a stable operational condition, between take-off and landing, kept while travelling the longest portion of an established route, optimizing fuel consumption and duration. TIT will be maintained at high values (~1,400 K in commercial turbofans like the one under study) to reach such optimization goal, however lower than those during take-off and climbing, aiming to minimize the impact on maintenance. Initially, most of the operational life of two-spool turbofan engines used in commercial aviation will be consumed during cruise stages. Most of the fuel consumed during a flight, will be burnt in this phase as well.

There are more phases during the operation of the engine in a conventional flight (in fact, there are more every day, thanks to the increased operational flexibility given by modern FADECs, existing for instance different pre-established idle regimes, cruises, etc., in the logic of the CS), but the three stages listed above are the ones that usually imply higher TIT values, and/or longer periods of time, which means demanding operational conditions. Nevertheless, if the limits of the different materials and components of the engine are not overstretched during the cruise stage in each flight, then small reductions in engine’s RUL are expected.

In this sense, establishing a baseline for each flight stage will make possible the development of analytic tools and criteria to evaluate how the operation of the engines, covering a particular route, would be along the time. Such information, aggregated for the aircraft managed by an airline, would be used to estimate fuel consumption, emissions, maintenance impact (i.e., RUL) in a complete fleet, etc. The baselines could be compared with real flights to estimate how far an airline is from a defined theoretical operational scenario. Gathering information from different flights covering a specific route, would be the first step to develop a statistical analysis aiming to find out how an average flight is, and how it can be improved,
when possible. Such analysis could be used to optimize methodologies like the one presented in this thesis (see ROMs in Chapter 4).

Different typical missions could be defined, according with commercial requirements, or attending to the specific type of aircraft (and engines) but, in most of them, the cruise will be systematically the longest stage. To better understand the relative relevance of the cruise stage in a full flight, it is useful to attend to the total range covered in a commercial route. It is normally considered, as a valid criterion to classify flights [299] based on the total range, the next one:

1. **Short-haul flights**: Less than 3 hours flight or routes shorter than 1,500 km.
2. **Medium-haul flights**: Until 6 hours flight or routes shorter than 4,000 km.
3. **Long-haul flights**: More than 6 hours flight or routes longer than 4,000 km.

For short-haul flights, typical profiles are shown in Figure 57. In the first example, a good portion of the flight is dedicated to a progressive descent to the destination, most likely for fuel consumption reduction purposes, and to avoid an unnecessary reduction of useful life in the engines. That is why the cruise stage in this first example of flight is not consuming more time of the total flight duration. Regional flights and short-haul flights like this one, (MAD-CDG) taken as example, could be also covered by turboprop-powered aircraft but, to reduce total flight durations and to optimize the flight experience for the passengers, airlines decide that relevant routes between important cities would be covered by turbofan-powered aircraft instead, even when this could be a sub-optimal solution in terms of performance. A different example of short-haul flight (MAD-LHR) shows a profile with a dominant cruise phase consuming almost a 70% of the total flight duration.

![Figure 57: Examples of short-haul flights inside Europe (see [96]).](image)

There is publicly available information showing how often short, medium, or long-haul commercial flights depart in the world, and how is the real flight profile, in terms of altitude and speed, they keep. It is possible to easily identify the cruise stage, and its relative duration over the full route covered during the flight.
Such information availability makes affordable the preparation of statistical analysis to optimize the operational regime and the monitoring tools used to track the evolution of the engines. It is expected that the longer the duration of the flight is the longer the cruise phase will be as well. This assumption is verified with the information provided in Figure 58, where flight distance, flight duration, and cruise duration are clearly correlated. So, most of the time spent in a commercial flight is inside steady regimes and this circumstance would justify the focus put on such important phase of flight for the methodology presented in this study.

Figure 58: Data obtained from flight distributions: global flights in 2006 and US flights in 2012, respectively (data and back images from [299] and [224], respectively).

Examples of medium-haul, long-haul, and ultra-long-haul flights are provided in Figure 59, Figure 60, and Figure 61, respectively. It is possible to verify in them the previous statement on the preeminence of stable cruise phases in commercial flights. In medium-haul flights, the rate in between cruise and total flight duration reaches the 80%, meanwhile for longer durations, the ratio could achieve the 90%.

Figure 59: Examples of medium-haul flights: MAD-ARN and MAD-SVO, respectively (see [96]).
In the case of gas turbine engines used for marine and industrial applications, the steady operational phase is known as “base load”. In such regime, the unit is typically kept at an elevated power output setting (sometimes these engines work regulating up and down the load, depending on marine propulsion needs, desired production levels, or requirements from Grid Authorities). As it happened with the cruise stage in aviation, the base load means an elevated regime that the engine can afford, given the local ambient conditions where the unit is working, while optimizing fuel consumption, but without implying excessively demanding operational conditions. The higher the TIT and the OPR, the better the efficiency, similarly to what was indicated in Figure 49.

Pure “base loaders” are started up and they usually maintain a similar profile, 24/7, until the next scheduled outage, ideally after 4,000 hours. These engines are typically affected only by the changes in the local ambient conditions and by its own degradation (if there is not a previous unscheduled outage motivated, either by a technical problem, or by an economical decision based on the price of the energy exported and the associated cost of fuel). The ratio of the steady state phase in these units will be even closer to a 100%. Some other gas turbine units (known as “peakers”) are operated only upon demand, they are not linked to any industrial activity beyond the energy production, and could remain stopped during weeks or months, waiting to be started up only when the price established for the exported energy by the Grid Authority could justify the operation of the engine. The role of these unit is crucial to compensate imbalances in the electrical grids, more common these days because of the recent shift to renewable energies (i.e., wind, solar, hydro) with lower or null carbon footprint, but subject to appropriate ambient conditions.

Figure 60: Example of long-haul flight: MAD-DXB (see [96]).
It is usual to find different cruise altitudes in flight profiles when the total duration is long enough [8]. The reason is that the specific range (SR, or the distance covered per unit of fuel consumed), given a fixed Mach number in a steady flight phase, is maximum at a specific altitude known as optimum altitude. That altitude means obviously an optimum lift-to-drag ratio, at the selected Mach number. If that ratio remains constant, then the ratio between weight and ambient static pressure will remain also constant, because of the dynamic balance during a cruise stage. If the weight decreases, the ambient pressure should then decrease to remain in an optimum altitude for maximum SR. And that leads to the convenience of increasing the altitude during long cruise stages, as the fuel on board is burnt.

On the other hand, it is also common to see that the flight speed decreases during the cruise phase in multiple flights (without considering the tail wind). For a given aircraft weight, there is an optimum Mach number providing the maximum range possible. Again, as the weight of the aircraft decreases, that optimum Mach number will vary. It decreases, so the difference between a constant Mach number chosen to complete the route in a pre-established time and the optimum Mach number for maximum range would eventually increase, for a given altitude. Flight scheduled durations take into consideration this circumstance and airlines try to adjust the speed accordingly. For short-haul and medium-haul flights there is some margin to select a flight speed, keeping it almost constant during cruise, that will not mean a big increase in fuel consumption comparing to the flight speed for maximum range. However, Figure 60 and Figure 61 show how, in typical long-haul and ultra-long-haul flights, it is applied a remarkable speed reduction during cruise phase.

![Ultra-Long-haul flight](image)

**Figure 61:** Example of ultra-long-haul flight: MAD-HND (see [96]).
The representative mission considered for this work corresponds, given their higher frequency, to an intermediate case between a short-haul and a medium-haul flight, typically including only one cruise phase (see Figure 62 and Table 11). Similar baselines can be established to compare with real flights or to prepare operational strategies inside a fleet of engines. In case longer flights would be interesting for analysis in the future, it would be just a matter of calculating results at different altitude and Mach numbers, task that is easily fulfilled thanks to PROOSIS®. The methodology will be easily adaptable to other flight profiles to meet commercial requirements, like the one shown in Figure 63 and Table 12.

Figure 62: Main stages of a typical short-haul flight.

<table>
<thead>
<tr>
<th>Stage</th>
<th>Time</th>
<th>Mach Number</th>
<th>Altitude [m]</th>
<th>T₄ [K]</th>
</tr>
</thead>
<tbody>
<tr>
<td>1 - Begin Taxi (Idle)</td>
<td>0:00:00</td>
<td>0.00</td>
<td>0</td>
<td>1,150</td>
</tr>
<tr>
<td>2 - End of Taxi</td>
<td>0:10:00</td>
<td>0.00</td>
<td>0</td>
<td>1,150</td>
</tr>
<tr>
<td>3 - Begin of Take-Off (maximum TIT)</td>
<td>0:10:03</td>
<td>0.00</td>
<td>0</td>
<td>1,620</td>
</tr>
<tr>
<td>4 - End of Take-Off</td>
<td>0:11:00</td>
<td>0.25</td>
<td>0</td>
<td>1,620</td>
</tr>
<tr>
<td>5 - Begin Climb (lower TIT)</td>
<td>0:11:01</td>
<td>0.25</td>
<td>0</td>
<td>1,600</td>
</tr>
<tr>
<td>Climbing</td>
<td>0:24:00</td>
<td>0.75</td>
<td>9,000</td>
<td>1,600</td>
</tr>
<tr>
<td>6 - End of Climb</td>
<td>0:28:00</td>
<td>0.80</td>
<td>10,668</td>
<td>1,600</td>
</tr>
<tr>
<td>7 - Begin Cruise (design TIT)</td>
<td>0:28:02</td>
<td>0.80</td>
<td>10,668</td>
<td>1,400</td>
</tr>
<tr>
<td>8 - End of Cruise</td>
<td>2:10:00</td>
<td>0.80</td>
<td>10,668</td>
<td>1,400</td>
</tr>
<tr>
<td>9 - Begin Descent (Idle)</td>
<td>2:10:02</td>
<td>0.80</td>
<td>10,668</td>
<td>1,400</td>
</tr>
<tr>
<td>Descending</td>
<td>2:17:00</td>
<td>0.65</td>
<td>7,000</td>
<td>1,150</td>
</tr>
<tr>
<td>10 - End of Descent</td>
<td>2:36:00</td>
<td>0.20</td>
<td>0</td>
<td>1,150</td>
</tr>
<tr>
<td>11 - Landing and Braking</td>
<td>2:37:00</td>
<td>0.00</td>
<td>0</td>
<td>1,150</td>
</tr>
<tr>
<td>12 - Taxi and Parking</td>
<td>2:46:00</td>
<td>0.00</td>
<td>0</td>
<td>1,150</td>
</tr>
</tbody>
</table>

Table 11: Typical flight profile considered for a short-haul flight.
Figure 63: Main stages of a typical long-haul flight.

<table>
<thead>
<tr>
<th>Stage</th>
<th>Time</th>
<th>Mach Number</th>
<th>Altitude [m]</th>
<th>T4t [K]</th>
</tr>
</thead>
<tbody>
<tr>
<td>1 - Begin Taxi (idle)</td>
<td>0:00:00</td>
<td>0.00</td>
<td>0</td>
<td>1,150</td>
</tr>
<tr>
<td>2 - End of Taxi</td>
<td>0:10:00</td>
<td>0.00</td>
<td>0</td>
<td>1,150</td>
</tr>
<tr>
<td>3 - Begin of Take-Off (maximum TIT)</td>
<td>0:10:03</td>
<td>0.00</td>
<td>0</td>
<td>1,620</td>
</tr>
<tr>
<td>4 - End of Take-Off</td>
<td>0:11:00</td>
<td>0.25</td>
<td>0</td>
<td>1,620</td>
</tr>
<tr>
<td>5 - Begin 1st Climb (lower TIT)</td>
<td>0:11:01</td>
<td>0.25</td>
<td>0</td>
<td>1,600</td>
</tr>
<tr>
<td>Climbing</td>
<td>0:24:00</td>
<td>0.80</td>
<td>9,000</td>
<td>1,600</td>
</tr>
<tr>
<td>6 - End of 1st Climb</td>
<td>0:28:00</td>
<td>0.80</td>
<td>10,668</td>
<td>1,600</td>
</tr>
<tr>
<td>7 - Begin 1st Cruise (design TIT)</td>
<td>0:28:02</td>
<td>0.80</td>
<td>10,668</td>
<td>1,400</td>
</tr>
<tr>
<td>8 - End of 1st Cruise</td>
<td>4:00:00</td>
<td>0.80</td>
<td>10,668</td>
<td>1,400</td>
</tr>
<tr>
<td>9 - Begin 2nd Climb</td>
<td>4:00:02</td>
<td>0.80</td>
<td>10,668</td>
<td>1,600</td>
</tr>
<tr>
<td>10 - End of 2nd Climb</td>
<td>4:04:00</td>
<td>0.80</td>
<td>11,668</td>
<td>1,600</td>
</tr>
<tr>
<td>11 - Begin 2nd Cruise (design TIT)</td>
<td>4:04:02</td>
<td>0.80</td>
<td>11,668</td>
<td>1,400</td>
</tr>
<tr>
<td>12 - End of 2nd Cruise</td>
<td>6:00:00</td>
<td>0.80</td>
<td>11,668</td>
<td>1,400</td>
</tr>
<tr>
<td>13 - Begin Descent (idle)</td>
<td>6:00:02</td>
<td>0.80</td>
<td>11,668</td>
<td>1,150</td>
</tr>
<tr>
<td>Descending</td>
<td>6:20:00</td>
<td>0.65</td>
<td>7,000</td>
<td>1,150</td>
</tr>
<tr>
<td>14 - End of Descent</td>
<td>7:06:00</td>
<td>0.20</td>
<td>0</td>
<td>1,150</td>
</tr>
<tr>
<td>15 - Landing and Braking</td>
<td>7:07:00</td>
<td>0.00</td>
<td>0</td>
<td>1,150</td>
</tr>
<tr>
<td>16 - Taxi and Parking</td>
<td>7:16:00</td>
<td>0.00</td>
<td>0</td>
<td>1,150</td>
</tr>
</tbody>
</table>

Table 12: Typical flight profile considered for a long-haul flight.
4.- MATHEMATICAL AND NUMERICAL METHODOLOGY

The next two chapters of the thesis contain both the explanation of the mathematical and numerical methodology followed (in this same chapter) and results obtained after its implementation for the resolution of the engine’s inverse problem (Chapter 5) to calculate, as it was schematically indicated in Chapter 1, the H&Q parameters of the aeroengine used as reference, given the required readings from sensors.

In the following pages, regarding the methodology inside a model-based framework, it will be detailed the different techniques that were explored and applied during this research to the resolution of the inverse problem to find out which one could better fit in terms of computational effort and accuracy. Meanwhile PROOSIS® was the SW providing the required performance calculations by running multiple off-design performance cases with the model that was explained in Chapter 3, the different techniques used for the inverse problem resolution were implemented in MATLAB®. Given the continuous use of matrices in the problem-solving process (as well as tensors when dealing with ROMs), and the fact that MATLAB® counts with very powerful and well-known algorithms to manage such mathematical objects, the decision of implementing the codes in that program resulted finally appropriate.

The available literature on the topic, reviewed in Chapter 2, was the starting point to select and incorporate the techniques that were finally used in this study. From the popular Genetic Algorithms (GAs) to the more recent Higher Order Single Value Decomposition (HOSVD), including the Gradient methods within the Sequential Quadratic Programming (SQP), or the Newton-like methods. All of them count with advantageous features that could benefit to the problem-solving process, but they also have some particularities and drawbacks when applied to the specific problem under consideration, so some analysis was required to decide among them. These circumstances will be evaluated in the next pages.

In this sense, novel model-based techniques being developed these days, or some other methods already available but remaining off the radar during the initial research phase (and, consequently, not considered in this document), could be also of great applicability for the calculation of the H&Q parameters of the engine. Certainly, they would be part of the future work to guarantee that the best existing methodology is being systematically applied to the problem under study.

The way the next two chapters are organized is correlative, so the results in Chapter 5 will follow the same order in which the methods are introduced in this chapter, aiming to facilitate to the reader the follow up of the work done through the document. Some partial or qualitative results were directly included in this first theoretical chapter because they were considered useful to follow the different reasonings and theoretical concepts behind the methodology. When additional content was needed in the next chapter with the results, or when some comments were found necessary, they were left at the end of the chapter for the sake of clarity.

Together with the description of the optimization techniques, some notes will be given on the quality of the data provided by the instrumentation. Not all the sensor sets that can be installed in an aeroengine provide the same information from the system. Some undesired effects detected in the H&Q parameters were the result of certain conditioning in the problem when not all the required information from the engine was provided. This leads to the existence of optimum sets of sensors, that could be identified by using the Single Value Decomposition (SVD) technique.
4.1.- Analysis of the inverse problem

Coming back to the formal statement of the problem, the initial target was to calculate the two H&Q parameters associated to each turbomachinery component of the engine (namely, efficiency and mass flows capacity) by using the information available from ten sensors installed at different stations of the machine, given the regime and flight conditions, as it is shown in Figure 64.

![Figure 64: Inverse problem schematic, to obtain the H&Q parameters from the existing sensors’ set in the engine, given the engine’s regime and the applicable flight conditions.](image)

To provide all the H&Q parameters, representing the information from the gas path about the aeroengine’s degradation in a specific time, it was defined a degradation vector, $\bar{X}$, with a size $(N_{\text{deg}} \times 1)$. The $N_{\text{deg}}$ different components of the degradation vector (initially ten in total) gave the status of the relevant engine characteristics chosen to describe how much the engine had degraded from the new and clean state, which is the state of the engine when it is delivered to a customer from the factory, after being fully assembled and tested. There are different ways to show the degradation of the engine and its evolution. It could be taken the values of the efficiencies and flow capacity parameters as they are normally accounted for (values around 1, which represents the state without degradation). In this case, as the engine will continuously degrade, with typically small deteriorations every cycle of use (i.e., flight), it was selected a different way to account for these variables: the new and clean state will be represented by a 0% of degradation in the different components of $\bar{X}$, and the subsequent degradations will be accounted for as a percentage value from that new and clean state (i.e., an isentropic efficiency that evolves from 1 to 0.99, will be accounted for as an equivalent parameter that goes from 0% degradation to 1% degradation). Working with percentages contributed to obtain a higher accuracy in the results. This change to percentages could be considered as a kind of numerical scaling applied to dimensionless variables.

Regarding the data retrieved from the instrumentation mounted in the engine, the values from the different sensors were collected in a different column vector called instrumentation vector, $\bar{Y}$, which size was $(N_{\text{sen}} \times 1)$. The sensors in the instrumentation measure magnitudes such as pressures (typically measured in Pa), temperatures (provided in K), rotational speeds (in rpm) and the fuel mass flow rate (i.e., fuel consumption, which is expressed by the SI of units in kg/s).
This means the value of the components in the instrumentation vector would differ among themselves by several orders of magnitude, and this circumstance represented a remarkable problem for the performance of the methods used in the study, that could lead to longer computational times and convergence issues.

The way found to solve this issue was scaling the data obtained from the sensors, making comparable the orders of magnitude of both the scaled values and the ranges of variation of those scaled values. The scaling procedure applied to the components of the instrumentation vector will be described later. Eventually, both degradations and instrumentation vectors were properly scaled when performing calculations with them, leading to a more convenient numerical approach.

Usual cruise values for flight altitude (35,000 ft or 10,668 m) and Mach number (0.8) were taken as given data for the problem (see Chapter 3 for further justification on the selection of flight conditions). The aircraft instrumentation measures typically ambient magnitudes with enough accuracy, and it is initially inexpensive to select different values for these variables (i.e., $T_{\text{amb}}$, $P_{\text{amb}}$, $M_{\text{flight}}$) in PROOISIS®, depending on the flight condition in which the engine is being operated. Contrarily, the TIT cannot be precisely estimated by the instrumentation currently mounted in the kind of engine under study, with the existing technology. There are not commercial temperature probes (i.e., TCs or RTDs) capable to support the extreme temperatures reached in the CC. An alternative is given by the costly use of pyrometers, as it is done in some military applications. For this study, if this magnitude needs to be calculated together with the other degradations in case no other mean to find out its current value was available, then this circumstance will involve increasing the number of unknowns of the problem, $N_{\text{deg}}$, by one (or two).

Taking all these previous considerations into account, the direct problem can be expressed in the following way, for a given flight condition. Eq. (4.1) indicates the influence of the degradation (current values of the different $\eta_i$ and $\Gamma_i$) and regime (TLA, being translated into fuel supply and TIT, or $T_{4t}$) over the values measured by the instrumentation, which will correspond here with the values of the equivalent magnitudes calculated by PROOISIS® at the right engine stations (data simulation):

$$\bar{Y} = f(T_{4t}, \bar{X})$$

(4.1)

And that leads, naturally, to the approach for the inverse problem for this study, in which given a set of sensor measurements, named as $\bar{Y}_{\text{meas}}$, it is computed the associated degradation vector, $\bar{X}$. Or, in other words, the inverse problem implies solving the following system of equations:

$$f(T_{4t}, \bar{X}) = \bar{Y}_{\text{meas}}$$

(4.2)

The most common solving procedure for this kind of problems [129] consists in substituting the problem given by Eq. (4.2) with an optimization problem in which the degradation vector $\bar{X}$ (alone or together with the TIT, depending if this temperature is calculated as an unknown or not) is obtained by minimizing the value of an objective function (OF), prepared to measure properly the difference in between the two sides of Eq. (4.2). Initially, this can be done as follows, considering that the operator $\| \cdot \|_2$ stands for the usual Euclidean norm:

$$\min \| f(T_{4t}, \bar{X}) - \bar{Y}_{\text{meas}} \|_2$$

(4.3)
Optimizing with an OF is, initially, an attractive approach to solve the problem. However, inverse problems tend to be ill-conditioned, particularly if the information supplied to the engine model by the sensors do not represent properly the behavior of the real system, even if the number of unknown parameters is equal to the number of available sensors. That circumstance happened during the resolution of the inverse problem evaluated in this study, as it will be later detailed.

An inverse problem (given the applicable set of data from sensors) is called ill-conditioned, or badly conditioned, if a small relative error in data causes a big relative error in the computed solution, regardless of the technique of resolution.

One way to evaluate how is the conditioning of the inverse problem is by calculating the condition number (CN) of the Jacobian matrix associated to the left side in Eq. (4.2). That CN (i.e., the ratio between the biggest and the smallest components in a matrix) was found to be initially too high (it was around 6 orders of magnitude greater than 1). This situation was pointing to some instrumentation issue. Nevertheless, the CN in one problem can always be improved by incorporating more data from the sensors in the engine [97], meaning managing several samples of data instead of just one, to solve the inverse problem. This solution was initially adopted, given the high CN obtained in the Jacobian matrices of the problem for the sensors set in the reference engine (CFM56-5A). So, measurements were taken around two reference values of $T_{4t}$, low and high, respectively, during cruise phases:

$$T_{4t}^1 = 1,350\text{K} \quad \text{and} \quad T_{4t}^2 = 1,500\text{K}$$  \hfill (4.4)

This approach is acceptable once it is assumed that the degradation rates in the engine will evolve slowly enough to remain almost constant between the two measurement samples at different TIT values. Fortunately, this is the case in 2-spool turbofan engines (see [221]), when working under normal conditions. As a result, an over-determined problem was obtained which was treated with appropriate data processing tools. It was verified that adding three or four values of $T_{4t}$ did not improve the CN of the problem (like it was indicated in a previous work [247]).

Additionally, certain cautions were taken during the execution of the algorithms in MATLAB®. For instance, some constraints were imposed to guarantee that all the calculated components in the degradation vector would remain non-negative during the optimization process. Obtaining negative degradation vector components was usual during the first iterations with adapted Newton method in this study, and that circumstance caused numerical issues that even prevented the obtention of the required solution in certain cases. Also, when the second sample of measurements, at different TIT, was incorporated to the problem to reduce its CN, then the values of the two different TIT were included to the set of unknowns (it was possible because the resultant problem became over-determined), and it was imposed that the selected $T_{4t}$ values should remain inside appropriate ranges, enough distanced from each other (more than 50 K), to avoid overlapping effects.

There are robust gradient-like constraint optimization methods available to solve the problem presented by Eq. (4.3), typically finding local minima, without the need of a nearby initial guess. That approach would be helpful for the turbofan problem because an accurate initial guess could be not available, or could be not easily findable (e.g., health condition of the engine after several undocumented flight cycles or after a major maintenance in the workshop). One of those methods is the sequential quadratic programming (SQP), which is an active-set descend method.
implemented in MATLAB® (go to [189] for more details on the SQP algorithm version available in MATLAB®), optimized by the application of the Broyden-Fletcher-Goldfarb-Shanno formula (BFGS, see [211] for reference).

The formulation in Eq. (4.2) could be explored using a standard Newton method as well but this technique counts also with some inherent particularities, such as the need of a good initial guess, the difficult imposition of numerical constraints on it and its typical bad suitability for over-determined problems. These drawbacks were avoided with an ad hoc Newton method that will be explained later.

After testing the different methods for the turbofan inverse problem, it was verified how, even with very distant and random initial conditions, the method was capable to find a route from the initial degradation condition to the final one with high accuracy, indicating these good results that the problem could count with a global minimum (or, at least, with a good enough local minimum without too hard competence in a relatively wide area of the solution space). When considering extremely high degradations (in the order of 20% or 50%), multiple problems were found preventing the convergence of the algorithms in a relevant quantity of cases. Obviously, a 20% of degradation is a very unusual condition for any component of a turbofan engine, meaning that some sort of severe failure has happened, and the machine needs for an urgent maintenance intervention. The components’ maps managed by PROOSIS® are mainly prepared for much more standard conditions. Even though, the techniques were capable to find valid solutions, under such extreme situations in certain cases. In the cases where the model of the engine was moving inside well-defined conditions, the numerical methods typically succeeded.

With points more closely located to each other (i.e., standard smaller degradations, 1-2%, from an initial health status) the methods were always capable to provide the right solution with enough accuracy. Obviously, the problem counted with ten or more variables, depending on if $T_{4t}$ values were finally calculated, so it was not possible to visualize the surface of the solution space to have a clear picture of the location of a potential global minimum. Anyway, the evidence after tens of thousands of calculations pointed to the direction of the existence a global minimum (inside the usual region where the performance problem is well-defined) and the solutions resulted unique in all those calculations, circumstance that seemed to suggest the OF was convex (again, inside the appropriate regions of the solution space for the engine under study). These considerations will be discussed later (review [211] and [40] for further details on convexity and optimization).

So, given the high CN of the Jacobian matrices associated to the problem given by Eq. (4.2), the numerical techniques were applied to two sets of sensors:

- The initial set of ten sensors in the CFM56-5A. As it will be exposed, the solution of the inverse problem with this instrumentation was found not optimal, independently of the number of $T_{4t}$ values implied. This situation is consistent with past problems documented by other gas turbine engine diagnosis researchers (see, in this sense, [250]). However, a further study of the correlations detected between degradations led to a valid strategy to overcome this problem by adding an appropriate new sensor ($P_{45t}$).
- The improved engine’s instrumentation, with one extra sensor, was used then with the proposed numerical techniques, and it was confirmed that the new set of eleven sensors improved remarkably the CN of the problem and, consequently, the accuracy of the diagnosis results. In fact, the presence of
that extra sensor is widely extended in commercial aeroengines (e.g., in GE aircraft engines and its aeroderivative gas turbines). CFM56 engines count with one stage less in the HPT than other similar engines (see data in Chapter 3), and such design philosophy (optimization of hardware in the hot section) could be pointing out to the reason behind the CFM56-5A does not have $P_{45t}$.

Table 13 summarizes the definitive list of degradations considered, ordered in the same way than the components of the degradation vector, $X$ (first column), are. It also shows the instrumentation considered, which will be providing data to the problem (second column), ordered the same way the components of the instrumentation vector, $Y$, are. This order will not change, so new variables will be added to the beginning or to the end (like the $T_{4t}$ when calculated as unknown in $X$, or like the new $P_{45t}$ probe in $Y$) of the mentioned lists of components, but always respecting the sequence stabilised in the table.

The third column contains representative values for the measurements in each component of the instrumentation vector. The values in that third column were obtained running direct problems with PROOSIS® at several different and equispaced typical $T_{4t}$ values (seventeen values) during usual flight phases, between 1,300 K and 1,650 K. These temperatures (i.e., regimes) are indicative of less or more demanding regimes, respectively, given the rest of ambient conditions. In this sense, for all the cases, altitude (35,000 ft or 10,668 m) and Mach number (0.8) were maintained, and a new and clean state of the engine was considered (baseline with null degradations). Choosing other $T_{4t}$ values inside that same range would have not meant a remarkable difference in the results. Just representative values were looked for (orders of magnitude for each variable). With the outcome from those calculations in PROOSIS®, the averages of the sensors’ readings in the table were obtained with the arithmetic mean of the obtained values.

<table>
<thead>
<tr>
<th>Degradations</th>
<th>Sensors</th>
<th>Sensors average values</th>
</tr>
</thead>
<tbody>
<tr>
<td>X(1): Fan Efficiency - $\eta_{FAN}$</td>
<td>Y(1): $P_{13t}$</td>
<td>$y_{ave}$r(1): $5.792 \cdot 10^4$ Pa</td>
</tr>
<tr>
<td>X(2): Fan Flow Capacity - $\Gamma_{FAN}$</td>
<td>Y(2): $P_{23t}$</td>
<td>$y_{ave}$r(2): $1.913 \cdot 10^5$ Pa</td>
</tr>
<tr>
<td>X(3): HPC Efficiency - $\eta_{HPC}$</td>
<td>Y(3): $P_{3t}$</td>
<td>$y_{ave}$r(3): $1.322 \cdot 10^6$ Pa</td>
</tr>
<tr>
<td>X(4): HPC Flow Capacity - $\Gamma_{HPC}$</td>
<td>Y(4): $T_{23t}$</td>
<td>$y_{ave}$r(4): $4.302 \cdot 10^2$ K</td>
</tr>
<tr>
<td>X(5): LPC Efficiency - $\eta_{LPC}$</td>
<td>Y(5): $T_{3t}$</td>
<td>$y_{ave}$r(5): $7.751 \cdot 10^2$ K</td>
</tr>
<tr>
<td>X(6): LPC Flow Capacity - $\Gamma_{LPC}$</td>
<td>Y(6): $T_{45t}$</td>
<td>$y_{ave}$r(6): $1.111 \cdot 10^3$ K</td>
</tr>
<tr>
<td>X(7): HPT Efficiency - $\eta_{HPT}$</td>
<td>Y(7): $T_{5t}$</td>
<td>$y_{ave}$r(7): $7.464 \cdot 10^2$ K</td>
</tr>
<tr>
<td>X(8): HPT Flow Capacity - $\Gamma_{HPT}$</td>
<td>Y(8): $N_{It}$</td>
<td>$y_{ave}$r(8): $1.031 \cdot 10^4$ rpm</td>
</tr>
<tr>
<td>X(9): LPT Efficiency - $\eta_{LPT}$</td>
<td>Y(9): $N_{l}$</td>
<td>$y_{ave}$r(9): $4.141 \cdot 10^3$ rpm</td>
</tr>
<tr>
<td>X(10): LPT Flow Capacity - $\Gamma_{LPT}$</td>
<td>Y(10): $W_{F}$</td>
<td>$y_{ave}$r(10): $1.097$ kg/s</td>
</tr>
</tbody>
</table>

Table 13: Degradation vector components and instrumentation vector components, with their description, and average values of the instrumentation vector’s components used. Same number of unknowns than equations.

Figure 65 illustrates the definitive position of the sensors, as considered in the model. The data supplied by PROOSIS® will come from those stations. In Chapter 3, it was shown the position of sensors in the reference engine, CFM56-5A, as decided by the OEM. There were some differences (e.g., measurements in turbine station 49 instead of 45), but both are match reasonably well for this study.
A first overview to the average values in the third column of Table 13 shows there are very relevant differences in the order of magnitude between readings from the available sensors (10^2 vs. 10^6). This circumstance indicates clearly that a proper scaling was, indeed, highly recommendable to avoid numerical problems and lack of convergence in the solving methods that would be used later.

4.2. Scaling of the data provided by the instrumentation

The correct scaling of the information obtained from the instrumentation is of paramount importance regarding the performance of the algorithms employed in this study. There are several orders of magnitude of difference in between the typical values of the variables managed in the problem under consideration (e.g., between pressure values and fuel consumption values). It is commonly understood that a vectorial optimization problem is poorly scaled if changes in the inputs in a particular direction produce much larger variations of the value of the objective function than changes in some other direction, leading to elongated contours in the resultant OF. It is usual to find optimization methods, such as the steepest descent, that tend to be very sensitive to the scaling applied to the data [211]. Some others, like the ones based on the Newton’s method, are not so remarkably impacted.

Sensor scaling factors can be perfectly determined once and offline, just before the diagnosis process is initiated during a flight, meaning no extra time needed for this task. Each sensor reading was scaled following the next procedure:
• First, the average obtained for each sensor (called here \(M(i)\), for \(i = 1, \ldots, N_{sen}\)) was subtracted from each sensor’s reading value and then the result was divided by the same average \(M(i)\). This was done with all the components in the instrumentation vector. With this calculation, the resultant variables would count now with an order of magnitude around one and with zero mean. Resultant values after scaling could be either positive or negative, but their absolute value would be around one.

• A second step was required to have variables’ values with equal variations around their respective means. The standard deviations (called here \(SD(i)\), for \(i = 1, \ldots, N_{sen}\)) calculated with the values from the outcome obtained from the previous PROOSIS® runs were used in this case when dividing each corresponding resultant variable in the first step by the applicable \(SD(i)\).

Table 14: Scaling values obtained from the direct problem results.

<table>
<thead>
<tr>
<th>i</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6</th>
<th>7</th>
<th>8</th>
<th>9</th>
<th>10</th>
</tr>
</thead>
<tbody>
<tr>
<td>(M^*)</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
</tr>
<tr>
<td>(SD^*)</td>
<td>0.05</td>
<td>0.06</td>
<td>0.12</td>
<td>0.04</td>
<td>0.05</td>
<td>0.08</td>
<td>0.08</td>
<td>0.05</td>
<td>0.05</td>
<td>0.20</td>
</tr>
<tr>
<td>(\bar{M}(i))</td>
<td>-0.31</td>
<td>-0.21</td>
<td>-0.29</td>
<td>-0.11</td>
<td>-0.13</td>
<td>0.04</td>
<td>0.12</td>
<td>-0.18</td>
<td>-0.22</td>
<td>-0.11</td>
</tr>
<tr>
<td>(\bar{SD}(i))</td>
<td>0.77</td>
<td>0.99</td>
<td>0.79</td>
<td>0.81</td>
<td>0.77</td>
<td>0.77</td>
<td>0.81</td>
<td>0.80</td>
<td>0.77</td>
<td></td>
</tr>
</tbody>
</table>

Table 14 shows the values obtained in the scaling. In the second row of the table, it is indicated the average (\(M^*\)) of the new sensors’ values obtained directly after subtracting and dividing by \(Y_{\text{aver}}(i)\) in the first scaling step. Asterisks indicates that operation has been already performed. As expected, that new mean value was always equal to zero. Then, the third row gives the \(SD^*\) values obtained from the variables scaled in the first scaling step (\(M^*\)). These \(SD^*(i)\) values were used in the second step. It is necessary to highlight here that the lowest \(SD^*(i)\) value is five times smaller than the highest one in the same row. This difference is what justifies the second step in the scaling process to have \(N_{\text{sen}}\) variables with comparable standard deviations. So, summarizing the two scaling steps indicated before in just one formula, each new reading from a sensor was scaled according to Eq. (4.5):

\[
Y_{\text{scal}}(i) = \frac{Y(i) - Y_{\text{aver}}(i)}{SD(i) \cdot Y_{\text{aver}}(i)}
\]  

(4.5)

Here, \(Y_{\text{scal}}(i)\) is the resultant variable after the scaling process for \(i = 1, \ldots, N_{\text{sen}}\) (\(N_{\text{sen}} = 10\), initially, but that quantity will be increased later with one additional sensor, as it was previously commented). It was finally verified that the scaling process was not too dependent of the sampled data (seventeen equispaced cases were used) by computing many direct problem cases (~100), calculated at the different \(T_{\text{st}}\) values considered before as representative for a typical cruise condition, but with degradations outside the new and clean state of the engine. Random degradations, between 0% and 2% from the baseline, were taken in each PROOSIS® run to verify this. The results are shown in the fourth and fifth rows of the previous table. In the fourth row, the arithmetic mean values of the obtained sensor readings, so called \(\bar{M}^*(i)\), are nonzero as expected, but of order ~1. Also, and more relevant, the new standard deviations obtained, \(\bar{SD}^*(i)\), are all comparable.
This means all the sensors' outcomes scaled with Eq. (4.5) showed comparable values and comparable variations around their respective arithmetic mean values and that finally led to accept the scaling procedure as satisfactory for the purpose of the study. In fact, it could be applied to any other engine configuration with a different instrumentation (and different $N_{\text{sen}}$).

4.3.- Problem exploration with Genetic Algorithms

The first method employed to solve the inverse problem was the Genetic Algorithm (GA), given the fact it is a global optimization (and exploratory) method, used nowadays in a wide variety of problems, and it is typically implemented in scientific software tools inside programs like MATLAB®, circumstance that makes its use considerably easier. The exploratory approach inherent to the GA is especially appropriate for problems in which the global convexity is not guaranteed, or non-existent. When the convexity exists, at least in a vicinity of the solution, there are more efficient methods that can be employed instead, as it will be shown later. Dealing with ten or more variables invited to run a preliminary approximation to the problem, watching from above, to descend once the shape of the field was clearer. The MATLAB-GA® toolbox counts with a very extensive library of standard functions and options, so its graphical user interface was typically used for the sake of simplicity in the analysis with GAs, because it shows, in a systematic way, the different settings that could be considered by the code [188]. Initially, the target, before trying to obtain efficiently a solution to the inverse problem, was verifying such solution to the proposed problem existed, and how affordable was it to find.

An important feature of the GA method is that it avoids calculating derivatives, so no Jacobian or Hessian matrices are used. On the other hand, that lack of information implies longer times until convergence (GAs count with null rate of convergence, versus other methods, like SQP, with super-linear rates of convergence) and potential problems to identify a possible solution. However, the randomness of the process contributes to navigate the solution space providing valuable information on how problematic the search of a global solution could be. Dedicating enough time to this exploratory method, it was expected to find out also what degree of accuracy could be initially achievable. The GA method was used in different ways, always needing of high computational times (~hours).

From a historical point of view, the term Genetic Algorithm appeared officially in 1975, with the work of J. H. Holland [135]. GAs explore the space of feasible solutions for the optimization problem under consideration, and obtain the best among the evaluated ones, employing preestablished criteria to justify such selection. These criteria follow a similar principle to the one popularized by Charles Darwin's work, published in 1859, “On the Origin of Species”, in which the mechanism of the “survival of the fittest” was introduced as the main driver for the natural selection that leads to the evolution of the species on Earth. The individuals with a better condition to survive in the given environment transmit their genetical information, as a heritage, to the next generation of individuals (offspring). In that genetic transmission process, along several generations, different crossovers and mutations could happen to certain members of the population, introducing some degree of randomness to the transmission of the genetic information that contributes to keep some degree of variety among individuals.
All those effects are taken into consideration in the GAs and the iterative process in these algorithms continues until the final value obtained through the optimization is close enough to the targeted solution (the fittest value is finally selected as a valid solution to the problem). The fitness function is the function to optimize, and it will typically coincide with the OF used for the inverse problem, which is expressed by Eq. (4.6):

$$\min F = \sqrt{||f(T_{4t}^1 X) - \overline{Y}_{\text{meas}}^1||_2^2 + ||f(T_{4t}^2 X) - \overline{Y}_{\text{meas}}^2||_2^2} \quad (4.6)$$

Subject to $\text{lb} \leq X \leq \text{ub}$$ \quad (4.7)$

Where two samples of measurements obtained from the instrumentation, namely $\overline{Y}_{\text{meas}}^1$ and $\overline{Y}_{\text{meas}}^2$, were associated at two different values of $T_{4t}$ ($T_{4t}^1$ and $T_{4t}^2$, respectively). The operator $|| \cdot ||_2$ represents the usual Euclidean norm. For the sake of efficiency and conciseness, certain upper bounds (ub, to be defined upon the expected degradation level in the engine) and lower bounds (lb, which must be obviously equal to zero, for a new and clean engine, as a negative degradation would not have any physical meaning) were imposed regarding the degradation vector components. To avoid high CN, two samples were managed in the OF, and that decision allowed to calculate the two involved $T_{4t}$ values together with the degradation vector components. The expression in Eq. (4.6) contains outcomes from PROOSIS®, that had been already scaled, $f(T_{4t}^1 X)$ and $f(T_{4t}^2 X)$, which are compared against the corresponding readings from the instrumentation, also previously scaled. The resolution of the minimization problem provided the actual value of the degradation vector, $\overline{X}$, which was considered as coincident for both samples at different TIT values.

It is important to emphasize that, even when the best individuals have more chances to be selected, that is not guaranteed in the GAs, simply because the selection of the best representatives is done following stochastic procedures. And it is precisely this random choice what typically contributes to the exploration of the whole solutions’ space and avoids the traps of potential local optimizers, in which is relatively easy to get stuck by using some other standard methods. This constitutes a great benefit of this technique but, on the other hand, this exploration is not systematic or methodologic, and there is no certainty on when the computation will end once initiated. This is something that was experienced during this study, in which it was necessary to calculate the value of the OF several hundreds of thousands of times before achieving a satisfactory accuracy in the results, resulting in several computing hours per case. This will be illustrated in Chapter 5. Multiple strategies and variations to improve the performance in GAs have been developed over the last decades, as it was explained in Chapter 2, and a profuse literature on the topic details the different ways a GA can be executed to optimize its performance in certain inverse problems (see [122], [197], and [151] for further reference).

The main data structures in the MATLAB-GA® toolbox are the vectors called “individuals”, considered as artificial chromosomes (which components are then called genes, continuing with the evolutionary analogy), and the fitness function values. The value of the fitness function for an individual determines its “score”. It is possible to use bit strings instead of regular vectors to represent the individuals but the bit strings count with some limitations that could impact negatively to the performance of the technique. The standard options in MATLAB-GA® were used.
In this toolbox, the information of the chromosomes is stored in a single matrix, which organizes the different individuals of the population by rows and their genotypic information by columns. And this situation leads initially to chromosomes of the same length by default if no modification is introduced in the code in this sense. The phenotypes, or decision variables, are obtained after mapping from the chromosome representation into the decision variable space, and they are stored as well in a matrix where the rows contain the specific individual's phenotype. The performance of the phenotypes is evaluated by means of the OF. A different kind of object is represented by the fitness values which are obtained through a ranking process. In terms of results, MATLAB® provides the point at which the final value is obtained, together with its associated final fitness function value.

The initial population is created randomly by the SW, just taking into consideration an initial range of selection, unless a specific population is provided. In this work no initial populations were given to the algorithm, and initial random populations were always considered. In the first GA runs, no strict constraints or boundaries were imposed either, covering an unnecessarily big solution space and resulting eventually in an unaffordable computational time required per generation. So, in the next GAs executed, both upper (ranging from a 2%, up to a 20%) and lower (0%) boundary limits were imposed to the degradation in the H&Q parameters, leading to a considerable reduction in optimization time, but still in the range of hours. These boundaries were the only constraints introduced to the algorithm to narrow down the solution space explored by MATLAB®, avoiding thus the inclusion of individuals that, certainly, would not be close to the desired solution.

There are several main kinds of rules in a GA, applied at each iteration, to create the following generation (offspring) from the current population:

- **Selection rules**: Which determine the mechanism of selection among individuals, in the current generation (the parents), that will generate the next generation of individuals. The algorithm usually selects the individuals that count with better fitness values as parents. An individual could be initially selected more than once as a parent, depending on its fitness value.

- **Crossover rules**: Establish how the combination of parents will be to form the next generation of individuals. Typically, an offspring of two new individuals will be obtained. Each coordinate of one individual vector, provided by the crossover function acting by default in the algorithm, comes from an entry (gene) at the same coordinate from one of the parents. With constrained or bounded problems (like this one), the algorithm uses a randomly weighted average of the values per coordinate from the parents, so the constrains are respected. In this sense, the so-called “Elite” constitutes the group of individuals in the population with best fitness values who pass automatically to the next generation. The percentage of individuals belonging to such group can be defined in the toolbox. The crossover fraction option specifies the fraction of each population, other than elite children, that are made up of crossover children. A crossover fraction of 1.0 means all children other than elite individuals are generated by crossover, while a crossover fraction of 0.0 means that all children are generated by a different mechanism: Mutation.

- **Mutation rules**: These rules introduce some random modifications (usually following a Gaussian distribution by default) to the genetic information provided by the parents to their offspring. Mutation rules specify how the GA
makes small random changes in the individuals of the population to create mutated children. Mutation provides genetic diversity and enables the genetic algorithm to search a broader space. Anyway, for bounded problems (like this one), the child remains feasible inside the allowed ranges for the population, so the mutation rules do not contradict the boundaries or constraints imposed. Typically, the amount of mutation, decreases at each new generation. In this study, mutations had no applicability given the constraints imposed to the possible solution values.

- **Migration rules**: The quality of the solutions delivered depends on the population size, causing larger demand on processing power. Parallel and distributed processing techniques resolve this issue by allocating subpopulations that interact by exchanging parts of their populations through a migration process. Some algorithms allow for parallel processing, establishing subpopulations that evolve independent from each other. From time to time, individuals are exchanged between the different subpopulations. Migration rules specify how individuals move between subpopulations. When migration occurs, the best individuals from one subpopulation replace the worst individuals in the next subpopulation (“forward migration” option) or both in previous and next subpopulations (“both” option). Individuals that migrate, from one subpopulation to another one, are copied. They are not removed from the source subpopulation. The migration rate defines how many individuals move between subpopulations.

After these rules are applied (the ones truly applicable to the turbofan problem, the way it was defined), the previous population is replaced by a new one and its fitness value is evaluated. If the best fitness value is good enough, or the maximum allowed number of iterations is reached, then the process ends, and the best individual found will be taken as the solution. If that is not the case, the evolutionary process continues. The main settings considered in the MATLAB-GA® toolbox for the turbofan problem are summarized below:

- **Population**: 10, 20 and 100 individuals per variable were typically considered to evaluate the impact of this parameter in accuracy and time.
- **Constraints**: Both upper (2% and 20%) and lower (0%) boundary limits were established for the degradation vector values.
- **Fitness scaling**: Rank, which scales the raw scores based on the rank for each individual. A rank value of $R$ implies a scaled score proportional to $\sim (1/\sqrt{R})$. So, this way the spread effect of the raw score is eliminated.
- **Selection**: Following a stochastic Uniform distribution.
- **Elite count**: A 5% of the population was maintained.
- **Crossover rate**: 0.8 in general.
- **Mutation function**: Constraint dependent, otherwise individuals falling out of the acceptable solution region could enter in the population.
- **Crossover function**: Constraint dependent, exactly because of the same reasons considered for the mutation.
- **Migration direction**: Forward, however small effect was expected.
- **Migration rate**: 0.2 but, in general, however small effect was expected.
- **Migration interval**: 20. This parameter defines when a migration between subpopulations will take place. In this case, after twenty generations.
And the termination criteria followed is indicated below (the toolbox allows for their variation on real time while the algorithm is still running):

- **Maximum number of generations**: 100.
- **Function tolerance**: This tolerance value was kept low to force the algorithm to reach a sufficient accuracy (typically between $1 \cdot 10^{-4}$ and $1 \cdot 10^{-5}$).

Regarding the optimization of the GA code itself to try to improve the efficiency of the method, the crossover rates were tuned to measure the sensitivity of the algorithm to their respective variations, without a significant improvement in the computational time or in the accuracy of obtained results. Crossover rates were taken in between 0.2 and 0.8. Values of 0.5 are typically found in the literature. No significant variations were detected, given the imposed constraints, after long times required to reach the desired convergence.

On the contrary, the factor that truly had a very significant impact in the computational time cost was the number of individuals per generation. MATLAB-GA® toolbox recommended using 50 individuals if the problem counted with 5 variables or less, and 200 individuals for a higher number of variables. This advice led to extremely long times because the number of calls to the OF per generation grew considerably. If the size of the population is N, and the OF calls twice to PROOSIS® because of using two samples at different TIT, that means the OF is calculated $(2N)$ times, per generation. It was finally decided, after running several cases with populations of 10, 100 and even 1,000 individuals per variable and generation, to try to maintain a rule of 10 or 20 individuals per variable, which means 100 or 200 individuals when the $T_{4t}$ values where not calculated as variables, and 120 or 240 when considering $T_{4t}$ values from two different data samples with equal H&Q parameters, as unknown variables. Several examples are shown in the Chapter 5 to illustrate the impact of population’s size in the computational time.

It is well known that the definition of the fitness function (OF) impacts the performance of the algorithm. In this case, the selection of the Euclidean norm applied to the difference between sensor readings and calculated values was intuitive, but it finally resulted effective. In case of need, it could be modified with weighted factors to penalize some components over the rest, as the norm counts with some averaging behavior between variables that could impact negatively to the total time needed until convergence. Such kind of ad hoc modifications to the OF were not finally necessary, given the fact that the GAs were soon discarded as a definitive technique, once it was clear the problem had solution, which was the main target to achieve by the GAs. The strategy followed was just setting a function tolerance value low enough to reach the desired level of accuracy, and the slow convergence, comparing with other methods, was the main reason to avoid its use. It could be possible to reduce somehow the computational time with an initial population, for the first generation. This possibility was also discarded given the total computational times, far away from a real-time or near-real-time application. Some comment is necessary regarding the scaling. It is particularly recommended to complete a proper scaling of the data, as mentioned before, when running the GA. Given the fact that in the turbofan problem there are differences of several orders of magnitude in between the values representing the different variables of the problem, some bias could appear in any of the variables when obtaining fitness functions values, increasing even more the time required by the GA [247].
4.4.- Use of the SQP Method

Given the slow convergence obtained with the GAs, the problem provided by Eq. (4.3) and Eq. (4.4) was addressed by means of some other different techniques. In this case, with the sequential Quadratic Programming (SQP). The objective function, \( F \), will maintain the properties of the usual norms, and that leads to the exploration of the typical optimality conditions for continuously twice differentiable functions: gradient equal to zero and Hessian positive semidefinite in the minimum under consideration (see [211]). These optimality conditions apply to local minima in generic problems. However, it will be explained later that the problem studied in this work seemed to always have solution (at least, under most usual conditions), which points in the direction of having some sort of global minimum, for every degradation condition of the engine respecting the physical limits and boundaries established by the model. Most of the cases run, once the respective numerical scheme was properly configured, allowing for the required accuracy, and keeping iterations inside reasonable boundary limits, found a solution.

The problem was solved this time using the function “sqp”, implemented in MATLAB® (go to [189] for further details), that will be briefly explained. Anyway, it is recommended to consult the specialized publications in the topic (like [211]) given the theoretical background that this technique implies. The idea behind this method is not new however, as it is based in the use of Lagrange Multipliers but, obviously, its implementation into scientific software packages like MATLAB® had to wait until the second half of the past century to be successfully completed.

Assuming, by the application of Taylor’s Theorem, that the considered OF in the optimization problem can be approximated by a quadratic function, \( \hat{F} \), in the surroundings of the solution, given the constraints mentioned before, then the multidimensional problem could be numerically expressed in the following way:

\[
\min \hat{F} = F_k + \Delta \bar{X}_k^T \cdot \nabla F_k + \frac{1}{2} \Delta \bar{X}_k^T \cdot (\nabla^2_{XX} F_k) \cdot \Delta \bar{X}_k
\]  

(4.8)

Subject to \( lb \leq X \leq ub \)  

(4.9)

The solution to this quadratic problem will have to comply with the following condition (by deriving and disregarding higher order terms), that leads to the value of the minimizer \( \Delta \bar{X}_k \), needed to continue with the next iteration in the algorithm:

\[
(\nabla^2_{XX} F_k) \cdot \Delta \bar{X}_k = -\nabla F_k
\]  

(4.10)

This way, it is necessary to calculate the inverse of the Hessian matrix of \( F \), \( H_k = \nabla^2_{XX} F_k \), in each iteration “k”, to obtain the next value of \( \bar{X} \) in “k+1”:

\[
\bar{X}_{k+1} = \bar{X}_k + \Delta \bar{X}_k
\]  

(4.11)

The calculation of the Hessian matrix, at every iteration, is expensive from both analytical and numerical perspective (when it is possible). Several alternatives to substitute that matrix, keeping its most relevant properties, have been proposed in the last decades to ease the solving process, emerging thus different versions of the SQP method, like the one used in this work.
The SQP can be applied following a trust region strategy (adapting the length of the iterative steps to the size of the consecutive selected trust region where the minimum is supposed to be located) or with a search line strategy (choosing a direction and searching for lower OF values along it). Regarding the second strategy, probably the most obvious choice for a searching direction would be the steepest descent direction, moving along \(-\nabla F_k\) at every step, but there are more options that can be explored. There is a remarkable variety of line search SQP methods that differ in factors like the way the Hessian approximation is computed, in the step acceptance mechanism, etc. The most accurate one, but also the most expensive computationally speaking, will be the Newton’s direction, based on the direct use of the Hessian matrix from the OF (reaching thus a quadratic rate of convergence).

To facilitate the resolution of the problem in Eq. (4.8), the so-called Quasi-Newton directions were developed, like the one given by the BFGS formula (obtained in 1970 by C. G. Broyden, R. Fletcher, D. Goldfarb, and D. Shanno, separately, but coinciding in time), which substitutes the Hessian by a symmetric and, typically, positive definite approximation, denoted as \(\tilde{H}\) in the Eq. (4.12). This is the key concept behind the updating in the algorithm with BFGS formula: The Hessians, or their inverses, are not recalculated every step, it is just done an update which merges the most recently retrieved information about the OF with the existing knowledge embedded in the current Hessian approximation. And that approach implies a great reduction of computational effort in the whole iterative process. Probably, one of the main issues of this method is the selection of the first approximate matrix for the first iteration, but that problem is not so relevant in globally convex problems. In clearly convex problems, choosing that matrix to be the identity matrix (or a variation of it) would eventually work.

The rate of convergence is super-linear in the BFGS method, but still not as good as using a Newton’s direction, which converges quadratically. Nevertheless, the BFGS is still considered as one of the most effective updating formulae available today. The formula for the update is given below (it is recommended to consult specialized works on the topic, like the one from J. Nocedal and S. J. Wright [211]):

\[
\tilde{H}_{k+1} = \tilde{H}_k - \frac{\tilde{H}_k (\Delta \tilde{x}_k \Delta \tilde{x}_k^T) \tilde{H}_k}{\Delta \tilde{x}_k^T \tilde{H}_k \Delta \tilde{x}_k} + \frac{\Delta \tilde{g}_k \Delta \tilde{g}_k^T}{\Delta \tilde{g}_k^T \Delta \tilde{x}_k}
\]  \(\text{(4.12)}\)

Where: \(\Delta \tilde{x}_k = \bar{x}_{k+1} - \bar{x}_k\), and \(\Delta \tilde{g}_k = \nabla F_{k+1} - \nabla F_k\)  \(\text{(4.13)}\)

The values of \(\Delta \tilde{x}_k\) and \(\Delta \tilde{g}_k\) must satisfy some curvature condition for the BFGS updating \((\Delta \tilde{x}_k^T \cdot \Delta \tilde{g}_k > 0)\), which is the usual case of the problem under consideration in this work but, in case that condition could not be met, the BFGS formula allows for some modification to avoid further problems because of punctual issues at certain iterations.

The function used in MATLAB® follows the search line strategy, based on the Schittkowski’s NLPQL Fortran algorithm [264] and, in turn, has implemented the BFGS formula with some modification, suggested by M. J. D. Powell in 1978, to guarantee the use of positive definite matrices, as substitutes for the Hessians, in each step. At each iteration, the search direction is the solution of a quadratic programming subproblem, obtained from optimal Lagrange multipliers and a positive definite approximated matrix of the Hessian associated. So, the method generates a sequence of quadratic subproblems that must be solved successively.
This method assumes the OF will be sufficiently smooth (i.e., continuously differentiable), regarding variations with degradations and sensors, and that the number of variables will be limited to a maximum of one hundred. Both assumptions are verified in the turbofan problem once the right scales working with PROOSIS® are determined (meaning the size of the selected increment δ to be used with the finite differences when approximating derivatives). Such value for δ > 0 must be chosen carefully. It must be sufficiently small for a finite difference, but not too small to prevent issues because of the inherent round-off done by the numerical solver in PROOSIS®. Figure 66 shows schematically how the selection of δ was made by plotting, just for example, the tenth scaled output sensor component, Y(10), relative to the fuel mass flow injected in the CC (WF), versus the first scaled input degradation component, X(1), which corresponds with Fan’s efficiency (ηFAN).

![Figure 66](image)

Figure 66: Charts representing a plot of Y(10) versus X(1), showing the stair-like structure found (a) and, schematically, how the finite differences were calculated as approximations of the partial derivatives (b). Lower efficiency means higher temperature after Fan, and less fuel needed to achieve the required regime (T4t).
The figure shows several constant “steps” in a stair-like structure. Similar situations were found (with different, but still similar structures) when inspecting the rest of instrumentation vector components after being represented versus the degradation vector components. This situation introduced a clear numerical issue when finding the potential OF minima if no caution was taken dealing properly with it. Typically, a twice continuously differentiable OF would be preferred when attempting to identify those minima. Unfortunately, that would not be the case inside constant steps found when plotting the components of the instrumentation vector, which were found everywhere. Even worse, the derivatives inside those steps, calculated numerically by finite differences, like in Eq. (4.14), will be null, and this is a situation that can make impossible solving a particular case.

\[
\left( \frac{\partial Y_i}{\partial X_j} \right) \approx \left( \frac{\Delta Y_i}{\Delta} \right)
\]  

(4.14)

So, the caveat in this problem when using PROOSIS® was precisely avoiding too small \( \delta \) values in the finite differences that may be calculated, in such a way two values in the same step were not used together for any difference during the calculations of the Jacobian matrices, the Hessians, or the approximated version of the Hessians. If the value of the selected \( \delta \) is small enough to have both components of a finite difference over the same step, that circumstance introduces relevant problems, for instance, when the inverse of one of the previously mentioned matrices is calculated. If one complete column goes to zero, that means the rank of the Jacobian will be lower in one unit to the expected value, which means one potentially regular matrix could become singular, reaching excessive CN (in fact, theoretically, the CN of a Matrix in which one complete column goes to zero would tend to infinite). So, some tests were done, and it was found that \( \delta \approx 10^{-4} \) was a good choice for the finite differences, given the accuracy level provided by PROOSIS®.

Some higher degree of accuracy can be obtained from PROOSIS®, getting thus smaller steps. However, this software, as most of the informatic applications available today to model the behavior of a gas turbine engine, works with components’ maps, and that situation leads to a certain inherent discretization in the numerical problems of interest associated to the performance of the engine.

It is true that some interpolation techniques could help to reduce the discretization found, contributing thus to minimize the problems with the Jacobians mentioned before. Nevertheless, that solution would introduce numerical complexity to the model, and it will end up becoming an artificial approximation to the results from the model. So, in this case, the final decision made was to respect the original values from PROOSIS® (the same decision that would have been made with original maps coming from any gas turbine engine) and choosing the appropriate values of \( \delta \) to only consider the points that would provide the relevant information required for the calculations (i.e., in derivatives). This decision would not introduce artificial expressions to link the points provided by PROOSIS®, just the required points would be selected, disregarding the rest provided by the SW.

Figure 67 shows with some more detail the process followed to determine the appropriate value of \( \delta \), when calculating the variation of \( Y(10) \), scaled fuel consumption, with \( X(1) \), the degradation associated to \( \eta_{FAN} \). The same analysis was done with the rest of involved variables until it was verified that a value of \( \delta \approx 10^{-4} \) was the most appropriate for the calculations to do.
a) Left chart with $1 \cdot 10^4 Y(10)$ points. Right chart plots the detail.

b) Same zoomed-in chart, plotting less points: 1 every 10, and 1 every 20, respectively. Trend line added.

c) Same chart, less points until linearity: 1 every 40, and 1 every 50, respectively. Trend line added.

Figure 67: Process to select the right value for $\delta$, going from a stair-like chart to a linear one, for the partial variation of $Y(10)$ with $X(1)$, directly related with $\eta_{FAN}$, but with opposite influence (higher degradation means less $\eta$). The same method was followed with the rest of functional relationships included in the Jacobians.
The rate of convergence of the SQP is proved to be super-linear, at least in the surroundings near the optimum, which will lead to a faster convergence in the final iterations. The SQP-based methods are typically more efficient if the number of constraints is nearly as large as the number of variables, which happens in this problem. The only constraints that are kept in the problem are the ones which restrict the range of search, in between lower and upper limits, for the degradation values as they were defined. Initially, those constraints were kept in between 0% and 2%, to end up expanding that range to values in between 0% and 10%, 20% or even 50%. Values of degradation below 0% have no physical meaning and values above 10% were tried just to verify the robustness of the numerical method. Again, degradations of more than a 2% in a regular operational cycle of a gas turbine engine (a flight, a certain amount of running hours in an industrial application, etc.) will end up with some investigation to clarify what caused to such high degradation rate, given the technical and economic impact that circumstance would imply.

In the optimizing tool that executes the "sqp" function in MATLAB®, it is possible to choose in between the following two options: Either introducing the gradient's expression of the OF in a separate function file or using the approximation of derivatives calculated directly by the solver. Both options were explored, providing same results with almost identical computing times. Finally, for the sake of simplicity, the second one was selected, by default, as no special benefit was found by keeping the additional file with the lengthy expression of the gradient (especially when T₄t was included in the set of unknowns to the problem).

The SQP tool in MATLAB® also allows to choose in between forward differences or centered differences to approximate the derivatives appearing in the problem (i.e., ∆Y(i)/δ). Both options were explored and remarkable differences between both options were found. With the forward differences, the computing time until reaching the desired level of convergence was shorter (by a 33% average) than using centered differences, but the accuracy was also somehow lower (by one order of magnitude average), keeping same termination conditions in both cases. Finally, the forward differences were used most of the times as they provided enough accuracy and contributed to reduce the computational times until convergence.

When selecting one of both options for the derivatives' approximation, either forward or centered differences, the tool allowed to choose the value for the minimum perturbation (or minimum δ), and this parameter was the lowest perturbation used by the tool when calculating the required finite differences. The value for the minimum δ must be selected carefully to prevent the aforementioned round-off numerical issues introduced by PROOSIS®, because the tool selects by default a minimum perturbation of exactly 1 · 10⁻⁶. After analyzing results from PROOSIS® and calibrating accordingly, the following values were used with satisfactory results:

- For forward differences, a minimum δ_F = 5.50 · 10⁻⁴ was used.
- For centered differences, a minimum δ_C = δ_F / 2 = 2.75 · 10⁻⁴ was used instead.

Finally, two tolerance values had to be given (otherwise the tool would have taken the value 1 · 10⁻⁶, by default, extending the computations typically longer than needed) to stop the iterative process once reached by the errors in the degradations and in the objective function, respectively:
• For the objective function, TolerFunc = $1 \cdot 10^{-4}$ was used.
• For the degradations, the following minimum step was considered as termination criterion: TolerX = $1 \cdot 10^{-4}$.

The SQP method was tested several hundreds of times using random initial and final values (selected by using the command “rand” in MATLAB®) for the degradation vector, approach that is not exactly realistic. However, the method was capable to find a valid route between both engine degraded condition states, initial and final, in the cases analyzed. This fact highlights the robustness of the method. Even more, taking into consideration that, in the real world, the final engine condition after a flight will not count typically with better degradation levels than the ones in the initial state, as it happened with the random tests in which it was a constant to find cases where several H&Q parameters improved. In a real flight, degradations will typically tend to grow slowly but steadily with time.

The random tests proved that the SQP method was capable to find the route between two separate points in the solution space, representing two different degradation conditions, independently of the required process to go from one to the other, and vice versa. This is equivalent to affirming that it is always possible, with the aerothermodynamic model provided by PROOSIS®, to get a particular degradation level in an engine from a random health condition (either by cleaning and repairing, or by degrading, as much as needed, totally or partially, the engine), which is something consistent with the real experience in the field. For instance, a group of damaged blades in the HPC could be partially repaired or replaced, recovering previous efficiency levels. If the Fan is found dirty and that circumstance affects to the performance of that module, it can be cleaned to retrieve previous efficiency and mass flow capacity values. And something similar, but on the contrary, would happen when deteriorating the engine condition. These are just few examples to explain the validity of results obtained with random tests.

The next consideration made in the problem implied including the values of $T_{4t}$ for the two separate samples of data, $T_{4t}^1$ and $T_{4t}^2$ respectively, as unknown variables, the same way it was done before only with degradations. In this case, the calculation of the gradient function required by the SQP toolbox was somehow more complicated and involved a bulkier function file. It was done, again just as a double check, meaning again no remarkable difference, either in results or computational time, when leaving to the SQP tool the approximation of such gradient function, so finally this last option was kept. For this specific problem, it was considered, as a constraint for the temperatures, that they would be inside 100 K intervals around the values given by Eq. (4.4). The size of these two ranges is aligned with the typical rough order of magnitude for the accuracy (~50 K would typically mean around a 2% FS error, so the ranges respect instrumentation’s usual accuracy limits of around a 0.4%) available nowadays in the instrumentation (namely thermocouples) used in the industry when measuring temperatures after the HPT of the engine. Both ranges were kept enough separated to avoid coupling effects between them:

• The range for the first temperature was given by the following upper and lower limits: $T_{4t, \text{min}}^1 = 1,300 \, \text{K} \leq T_{4t}^1 \leq 1,400 \, \text{K} = T_{4t, \text{max}}^1$.
• For the second one, it was considered the following upper and lower limits: $T_{4t, \text{min}}^2 = 1,450 \, \text{K} \leq T_{4t}^2 \leq 1,550 \, \text{K} = T_{4t, \text{max}}^2$. 

- 164 -
To obtain all the unknowns in between 0% and 2%, the following scaling was imposed for both $T_{4t}^1$ and $T_{4t}^2$, coinciding the ranges of degradations’ variations with the ones for these variables. Similar approach was applied between 0% and 20%:

\[
X_T(i) = 2 \cdot \frac{T_{4t}^i - T_{4t,\min}^i}{T_{4t,\max}^i - T_{4t,\min}^i}; \quad \text{for } i = 1,2.
\] (4.15)

4.5.- Calculation of Jacobian matrices

Right after scaling the direct problem in Eq. (4.1), it is possible to calculate the associated Jacobian matrix of the resultant vectorial function with respect to the $N_{\text{deg}}$ degradations considered. With the value of $T_{4t}$ and the degradation vector $\vec{X}$, the Jacobian can be formulated as a matrix, with size $(N_{\text{sen}} \times N_{\text{deg}})$, like in Eq. (4.16):

\[
J(T_{4t}, \vec{X}) = [\vec{j}_1, \vec{j}_2, \ldots, \vec{j}_{N_{\text{deg}}}] \tag{4.16}
\]

Where the columns of the Jacobian matrix, called $\vec{j}_n$, containing the derivatives of the components of the instrumentation vector with respect to the different degradation vector components, were approximated by first order differences. Here, it could be chosen two kinds of differences: centered or forward. In this work the forward differences were finally used as they imply less calls to PROOSIS®, so a faster calculation could be obtained, and the accuracy of results was found to be satisfactory. The calculation to obtain each column is formulated below:

\[
\vec{j}_n = \left[ f(T_{4t}, \vec{X} + \delta \cdot \vec{I}_n) - f(T_{4t}, \vec{X}) \right] / \delta \tag{4.17}
\]

And this must be done, with the right value of $\delta$, for $n = 1, \ldots, N_{\text{deg}}$ until all the columns are obtained. In the previous expression, $\vec{I}_n$ represents the n-th column of the $(N_{\text{deg}} \times N_{\text{deg}})$-unit matrix. This computation implies $N_{\text{deg}} + 1$ calls to PROOSIS®, to run the direct problem and to calculate the associated instrumentation vector. One call is needed for the reference vector, $f(T_{4t}, \vec{X})$, that will be subtracted in all the columns to obtain the forward differences. Using centered differences, then the number of calls to the engine model would have been $2N_{\text{deg}}$ instead.

The initial values of the CN calculated with different Jacobian matrices were in the range between $1 \cdot 10^5$ and $1 \cdot 10^8$, depending on the values of $T_{4t}$ considered and depending on the points taken to calculate the forward differences that constitute the approximations for the derivatives required. The Hessian matrix of the objective function associated with the optimization problem in Eq. (4.3) counted with even higher values, as its condition number scales with the square of the CN of the corresponding Jacobian matrix. Without any doubt, this circumstance would lead to relevant issues when solving the optimization problem with methods based on the use of gradients or Newton-based (references on this topic in [211] and [40]). As it was mentioned, the strategy followed, to improve the situation with the high CN of the Jacobian and Hessian matrices used in the different techniques for the inverse problem optimization, was using more sensor readings, from two different values of $T_{4t}$ instead of only one, that will be called $T_{4t}^1$ and $T_{4t}^2$, respectively.
Both temperature values provided the corresponding Jacobian matrices denoted here as \( J_1 \) and \( J_2 \), respectively, which can be used together in one enlarged Jacobian matrix like the one in Eq. (4.18):

\[
J_{\text{large}} = \begin{bmatrix} J_1 \\ J_2 \end{bmatrix} \tag{4.18}
\]

This new matrix, \( J_{\text{large}} \), will have initially a size \( (2N_{\text{sen}} \times N_{\text{deg}}) \), and will count with a much lower CN than considering the Jacobian matrices separately. Just as an example, taking as representative \( T_{4t} \) values the ones in Eq. (4.4), below and above the previously mentioned typical value of 1,400 K at cruise conditions, it was verified that the CN of \( J_{\text{large}} \) was around \( \sim 350 \), which means a difference of several orders of magnitude comparing with the CN of the Jacobian matrices, \( J_1 \) and \( J_2 \), used individually. Repeating the previous calculation for different pairs of \( T_{4t} \) values the results obtained were very similar. In all those tests, it was left enough distance in between both \( T_{4t} \) values (a minimum of 50K difference in between \( T_{4t}^1 \) and \( T_{4t}^2 \)) to avoid overlapping effects. And it was verified that by including more than two \( T_{4t} \) values, generating an even larger Jacobian matrix, the resulting CN did not improve.

Therefore, it was decided to use two \( T_{4t} \) values initially to solve the inverse problem. As two different samples of data from instrumentation were used to obtain the degradation vector in each calculation case, that led to the situation of having more data \( (2N_{\text{sen}} = 20) \) than unknowns \( (N_{\text{deg}} = 10) \), so the \( T_{4t} \) for those two samples could be now calculated together with the \( N_{\text{deg}} \) degradation parameters, leading to \( (N_{\text{deg}} + 2 = 12) \) unknowns, still a lower number than the number of data available in this over-determined problem.

To do so, a new column was included in the Jacobian matrix of the problem, and the \( T_{4t} \) values were initially scaled making sure the final values would be of order one. The scaled variable for the \( T_{4t} \) was given by the expression in Eq. (4.19):

\[
X_0 = (T_{4t} - 1,400)/1,400 \tag{4.19}
\]

The reference temperature of 1,400 K is representative of usual cruise conditions. The new “zero” column for the temperature was formulated as follows, again, considering forward differences:

\[
j_0 = [f(1,400 \cdot (1 + X_0 + \delta), \bar{X}) - f(1,400 \cdot (1 + X_0), \bar{X})]/(1,400 \cdot \delta) \tag{4.20}
\]

Meanwhile, the other columns were calculated as indicated before, for \( n = 1, ..., N_{\text{deg}} \), but considering the new scaled parameter for \( T_{4t} \), in the following way:

\[
j_n = [f(1,400 \cdot (1 + X_0), \bar{X} + \delta \bar{X}_n) - f(1,400 \cdot (1 + X_0), \bar{X})]/\delta \tag{4.21}
\]

Then, to obtain all the unknowns in between 0% and 2% (or 20% and even higher when applicable), the Eq. (4.15) was used to get homogeneity in the results. With all these considerations, the new Jacobian matrix per \( T_{4t} \) value, which size was finally \( (N_{\text{sen}} \times (N_{\text{deg}} + 1)) \), was provided by the following expression:

\[
J(T_{4t}, \bar{X}) = [j_0, j_1, ..., j_{N_{\text{deg}}}] \tag{4.22}
\]
Calculating these Jacobians per each $T_{4t}$ value, the enlarged matrix, $J_{\text{large}}$, had a size of $((2N_{\text{sen}}) \times (N_{\text{deg}} + 2))$.

### 4.6. Applied Newton-like method

Once the algorithms to calculate the Jacobian matrices were determined, and given the satisfactory results obtained previously by applying techniques such as GAs and SQP to the inverse problem formulated in Eq. (4.3), an iterative method was prepared to solve the problem given by Eq. (4.2), where the vectors composing the enlarged instrumentation vector, $\bar{Y}_{\text{meas-large}}$, as well as its calculated counterpart, already introduced before, led to the following approach:

\[
\begin{bmatrix}
  f(T^{1}_{4t}, \bar{X}) \\
  f(T^{2}_{4t}, \bar{X})
\end{bmatrix} = \begin{bmatrix}
  \bar{Y}^{1}_{\text{meas}} \\
  \bar{Y}^{2}_{\text{meas}}
\end{bmatrix} \equiv \bar{Y}_{\text{meas-large}} \tag{4.23}
\]

Here, the iterations with the degradation vector were performed as follows:

\[
\bar{X}_{k+1} = \bar{X}_{k} + \Delta \bar{X}_{k} \tag{4.24}
\]

In every iteration, the incremental vector, $\Delta \bar{X}_{k}$, was obtained from the linear system given by the next formulation:

\[
J_{\text{large},k} \cdot \Delta \bar{X}_{k} = \bar{Y}_{\text{meas-large}} - \bar{Y}_{\text{large},k} \tag{4.25}
\]

The vector $\bar{Y}_{\text{meas-large}}$ comes from the instrumentation. $\bar{Y}_{\text{meas-large}}$ remains with the same values during all the iterative process until convergence. The final vector $\bar{X}$ must have associated another vector, $\bar{Y}_{\text{large}}$, as closer to $\bar{Y}_{\text{meas-large}}$ as possible. Once again, it was PROOSIS® the source that provided the measured values, as a substitute for the instrumentation of the engine. The enlarged Jacobian matrices were calculated in each iteration, the same way it was formulated in Eq. (4.18) and Eq. (4.22). The $\bar{Y}_{\text{large}}$ vector in each iteration was obtained as follows:

\[
\bar{Y}_{\text{large},k} = \begin{bmatrix}
  f(T^{1}_{4t}, \bar{X}_{k}) \\
  f(T^{2}_{4t}, \bar{X}_{k})
\end{bmatrix} \tag{4.26}
\]

To obtain the incremental vector $\Delta \bar{X}_{k}$ at each iteration, it was necessary to calculate the inverse of the $J_{\text{large},k}$ matrix via SVD. After improving the CN with two samples and after considering right $\delta$ values, no pseudo-inverse matrices were needed, and no truncation was performed for the modes that could be associated to small singular values, when calculating an inverse of a Jacobian matrix. Doing so:

\[
J_{\text{large},k} = U_{k} S_{k} V_{k}^{T} \tag{4.27}
\]

Both $U_{k}$ and $V_{k}^{T}$ are orthogonal, so the calculation of their inverses is done by transposing them, and given that $S_{k}$ is diagonal (which inverse is therefore obtained by inverting the elements in the main diagonal), $\Delta \bar{X}_{k}$ is then given by Eq. (4.28):
\[
\Delta \bar{X}_k = V_k S_k^{-1} U_k^T [\bar{Y}_{\text{meas-large}} - \bar{Y}_{\text{large},k}]
\]

(4.28)

During the first steps of the iteration process, some components of the degradation vector could count with negative values. To avoid this inconvenient circumstance when applying Newton-like methods, the negative degradation vector components, without valid physical meaning, were set to zero and used in the next iteration with such null value. After few steps, the degradation values went closer to the exact ones, which corresponded with the readings in \( \bar{Y}_{\text{meas-large}} \).

Newton-like methods are globally convergent if the Jacobian matrices used have a bounded condition number and are positive definite (condition needed to ensure a descent direction). The convergence was typically reached after no more than 10 iterations, which means this adapted method was considerably faster than the optimization-based ones. And this better performance was present even when the initial conditions were also selected randomly (initially, between 0% and 2%, and later between considerably wider ranges, as it will be explained).

The criterium of convergence (i.e., algorithm termination) was met if any of the next expressions was verified, where the thresholds, \( \varepsilon_1 \) and \( \varepsilon_2 \) respectively in this case, were chosen sufficiently small (typically between \( 1 \cdot 10^{-2} \) and \( 1 \cdot 10^{-3} \)):

\[
\| \bar{X}_{k+1} - \bar{X}_k \|_2 < \varepsilon_1 \quad \text{and} \quad \| \bar{Y}_{\text{meas-large}} - \bar{Y}_{\text{large},k} \|_2 < \varepsilon_2
\]

(4.29)

Where \( \| \cdot \|_2 \) represents the Euclidean norm. When the convergence was not obtained after the maximum number of iterations allowed (typically not more than 30), which was a very rare situation, that case was treated specifically, taking some additional action like, for example, repeating the iterative process allowing a higher number of iterations, or giving a different value to the mentioned thresholds.

### 4.7.- Singular Value Decomposition (SVD)

To solve the inverse problem, several numerical techniques were used, and their associated algorithms implemented in MATLAB® called iteratively to PROOSIS® to get the necessary performance estimations until enough degree of convergence was achieved. The techniques worked with vectors. The input vector (instrumentation vector, \( \bar{Y} \)) contains the data coming from the instrumentation, and the output vector (degradation vector, \( \bar{X} \)) contains the information regarding the H&Q parameters of the engine. In some of those techniques, it was necessary to calculate multiple times the Jacobian matrix and the Hessian matrix (or any more affordable approximation of it) of the equation system that linked sensor data with H&Q parameters.

In particular, the Jacobian matrix contained in its columns the gradients of the instrumentation vector with respect to the components in the degradation vector (first derivative). More elaborated Jacobian matrices would contain more variables (\( T_{4b}, P_{45b} \), etc.) later but, by now, just (10x10) matrices will be considered:

\[
\mathbf{J}(T_{4b}, \bar{X}) = \begin{bmatrix}
\frac{\partial P_{13t}}{\partial \eta_{FAN}} & \ldots & \frac{\partial P_{13t}}{\partial \Gamma_{LPT}} \\
\vdots & \ddots & \vdots \\
\frac{\partial W}{\partial \eta_{FAN}} & \ldots & \frac{\partial W}{\partial \Gamma_{LPT}}
\end{bmatrix}_{(10 \times 10)}
\]

(4.30)
The first derivatives can be numerically estimated by first order finite differences (forward or centered):

\[
J(T_{4l}, X) \approx \left[ \frac{\Delta P_{13l}}{\delta} \ldots \frac{\Delta P_{13l}}{\delta} \right] \ldots \left[ \frac{\Delta W_{F}}{\delta} \ldots \frac{\Delta W_{F}}{\delta} \right]_{(10 \times 10)} \tag{4.31}
\]

The calculation of the Hessian is considerably more time-consuming than the Jacobian because it implies a second derivative and, therefore, more operations to be performed. Fortunately, just with the information contained in the Jacobian matrices it is possible to obtain valuable information from the problem by means of the SVD. The SVD is an algebraic technique that has been extensively employed along the research documented in this work given its great applicability in the solving process of the inverse problem under consideration. Furthermore, the information it provided from the data contributed to develop a better analysis and to reach a deeper understanding of the problem, becoming thus a powerful tool to evaluate the results obtained through the solving process. This section will briefly introduce the theory behind the SVD and will try to clarify how it was used and why.

For a better understanding of the SVD, it is usually helpful to start its introduction with the well-known concept of diagonalization of square matrices, to end up with the presentation of the concept of singular values associated to generic rectangular matrices and their applications. In Linear Algebra (see [277] for a more elaborated explanation), it is proved that, given a generic square matrix \( M \), with a size \((I \times I)\), where the index \( I \) counts for both rows and columns of \( M \), with linearly independent eigenvectors, it is always possible to obtain the following factorization (any symbol, for products between matrices, or between matrices and vectors, will be obviated for the sake of clarity):

\[
M = PD P^{-1} \tag{4.32}
\]

Where \( P \) is a matrix which columns are given by the eigenvectors of \( M \) and \( P^{-1} \) stands for the inverse matrix of \( P \) (superindex “\(-1\)” stands for inverse in this section, by default). \( D \) is a diagonal matrix containing the eigenvalues of \( M \) along its diagonal, ordered from the highest to the lowest in absolute value. This technique, so called diagonalization of a matrix, is a very useful algebraic tool that eases greatly, for instance, the exponentiating of any square matrix, especially when the matrix is big or when the exponent is a number with a high value:

\[
M^n = [\prod_n PDP^{-1}] = PD(P^{-1}P)DP^{-1} \ldots PD(P^{-1}P)DP^{-1} = PD^n P^{-1} \tag{4.33}
\]

When the matrices under consideration are not only square but also symmetric, the associated eigenvectors are mutually orthogonal. Here it is good to remember that some of the eigenvalues could have algebraic multiplicity higher than 1, and the Gram-Schmidt algorithm could be always used to obtain an orthonormal basis. The symmetric square matrices and their orthonormalized eigenvectors are always compliant with the following diagonalization:

\[
B = PDP^{-1} = PDP^T \tag{4.34}
\]
Where \( \mathbf{B} \) is a generic symmetric matrix, with a size \((1 \times 1)\), and \( \mathbf{P}^T \) stands for the transpose matrix of \( \mathbf{P} \) (superindex "T" stands for transpose). It is typically much easier the calculation of the transpose than the calculation of the inverse. So, the calculation of \( \mathbf{B}^{-1} \) is considerably faster, comparing with the nonsymmetric case, especially when dealing with big matrices.

From terminology standpoint, it is also useful to recall, from the elemental Linear Algebra, that a real symmetric matrix \( \mathbf{B} \) is called positive definite when \( \mathbf{X}^T \mathbf{B} \mathbf{X} > 0, \forall \mathbf{X} \neq \mathbf{0} \), and if, and only if, each eigenvalue of \( \mathbf{B} \) is positive, being \( \mathbf{X} \) a column vector, with a size \((1 \times 1)\). \( \mathbf{B} \) is called positive semi-definite if \( \mathbf{X}^T \mathbf{B} \mathbf{X} \geq 0, \forall \mathbf{X} \neq \mathbf{0} \). In this last case, some of the eigenvalues could be null, but always non-negative.

So far, the mentioned concepts result familiar as they are used extensively in 2D and 3D algebraic and geometric problems. Nevertheless, the situation changes when the matrices are not square, \((1 \times 1)\), but rectangular, \((1 \times J)\), so with different numbers of rows and columns, because the previous factorizations cannot be directly used, and some additional steps are required to be capable to apply the mentioned theory. This situation could appear in the inverse problem for the health condition determination of the engine if the number of sensors and the number of H&Q parameters do not coincide, maybe because more sensors are available or maybe because the TIT is calculated as an additional H&Q parameter. This circumstance will be treated now with some more detail.

In this sense, it is known that, given a generic rectangular matrix \( \mathbf{A} \), with a size \((1 \times J)\), where \( I \) counts for the rows and \( J \) for the columns of \( \mathbf{A} \), respectively, the matrices \( \mathbf{A}^T \mathbf{A} \) \((1 \times 1)\) and \( \mathbf{AA}^T \) \((J \times J)\) are both square and symmetric (this result is easy to prove by Mathematical Induction), which means they can be orthogonally diagonalized, as it was mentioned previously, in the following way:

\[
\mathbf{A}^T \mathbf{A} = \mathbf{VD}_1 \mathbf{V}^T, \quad \text{and} \quad \mathbf{AA}^T = \mathbf{UD}_J \mathbf{U}^T
\]  \(\text{(4.35)}\)

Where \( \mathbf{V} \) and \( \mathbf{U} \) are matrices which contain the associated orthonormal eigenvectors of the eigenvalues associated to \( \mathbf{A}^T \mathbf{A} \) and \( \mathbf{AA}^T \), respectively. \( \mathbf{D}_1 \) and \( \mathbf{D}_J \) are the respective diagonal matrices for each factorization. The matrix \( \mathbf{A} \) could be a Jacobian matrix, like the one in Eq. (4.18), when different number of sensors (or samples of sensors) and H&Q parameters are considered. Paying attention to \( \mathbf{A}^T \mathbf{A} \) (a similar approach could be followed for \( \mathbf{AA}^T \)), this matrix will be, typically in the inverse problem under consideration for the present research work, positive definite, and its eigenvalues (all of them positive) and associated eigenvectors will verify the following expression:

\[
\mathbf{A}^T \mathbf{A} \mathbf{v}_i = \lambda_i \mathbf{v}_i \; \text{for} \; i = 1, \ldots, Q \quad \text{(being} \; Q = \text{rank}[\mathbf{A}]\quad )
\]  \(\text{(4.36)}\)

Where \( \lambda_i \) stands for the “ith” eigenvalue of \( \mathbf{A}^T \mathbf{A} \) and \( \mathbf{v}_i \) will correspond with an eigenvector associated to \( \lambda_i \). Operating few more steps with Eq. (4.36), it is possible to reach the following result that will naturally lead to the concept of SVD:

\[
\mathbf{v}_i^T \mathbf{A}^T \mathbf{A} \mathbf{v}_i = \lambda_i \mathbf{v}_i^T \mathbf{v}_i = \lambda_i > 0, \quad \text{so} \quad \|\mathbf{A} \mathbf{v}_i\|_2 = s_i, \quad \text{with} \; s_i = \sqrt{\lambda_i} \; \text{for} \; i = 1, \ldots, Q.
\]  \(\text{(4.37)}\)
The operator \( \| \cdot \|_2 \) corresponds with the usual Euclidean norm and the \( s_i \) terms, obtained as the square root of the different eigenvalues of \( A^T A \), are the **Singular Values** from the generic rectangular matrix \( A \).

In this sense, the singular values of a matrix \( A \) will be the non-negative square roots of the eigenvalues of either \( A^T A \) or \( AA^T \). The previous result allows to generate an orthonormal basis of the image subspace through \( A \), given by the following set of vectors:

\[
A \overline{v}_i / s_i = \overline{u}_i \text{ ; for } i = 1, ..., Q \text{ (being } Q = \text{ rank}[A]).
\]  
(4.38)

Where \( \overline{u}_i \) stands for the orthonormal vectors that will constitute an orthonormal basis in the image subspace through \( A \). It is easy to verify that the vectors generated this way are orthonormal, just by recalling the previous steps:

\[
\overline{u}_i^T \overline{u}_j = \frac{(A \overline{v}_i)^T A \overline{v}_j}{s_i s_j} = \frac{\overline{v}_i^T A \overline{v}_j}{s_i s_j} = \delta_{ij}
\]  
(4.39)

Here, \( \delta_{ij} \) represents the Kronecker delta. Since \( A \overline{v}_i = s_i \overline{u}_i \) for \( i = 1, ..., Q \), then, it has been proved that the generic rectangular matrix \( A \) can be expressed in the following way:

\[
A V = U S, \text{ or, in the more usual form: } A = U S V^T
\]  
(4.40)

In Eq. (4.40), \( U \) and \( V^T \) could be represented by square matrices with sizes \((I \times I)\) and \((J \times J)\), respectively, or alternatively by \((I \times Q)\) and \((Q \times J)\) matrices, where \( Q \) will be the minimum value between \( I \) and \( J \). The difference in between both representations is in the null rows and columns that will be retained (full SVD), or not (reduced SVD), after the decomposition, in the new matrices generated. The null rows and columns can be discarded when performing SVD analysis in MATLAB\textsuperscript{®}. Typically, those null rows and columns are not retained to save memory storage and processing time, and that approach was followed in the calculations done for this study as well. Either way, the columns of the matrices \( U \) and \( V \) will be, as previously indicated, mutually orthonormal with the Euclidean inner product:

\[
U^T U = V^T V = I_{Q \times Q}
\]  
(4.41)

Those same columns in the matrices \( U \) and \( V \) are so called left and right modes of the decomposition, respectively, in the terminology typically employed when using this technique. In Eq. (4.41), \( I_{Q \times Q} \) is the \( Q \)-order unit matrix.

The matrix \( S \) corresponds with a \((Q \times Q)\) size diagonal matrix which elements, the previously introduced as singular values of the decomposition, and denoted as \( s_i \), are real and non-negative values. They are the square roots of the nonzero eigenvalues of both \( AA^T \) and \( A^T A \). This can be seen more clearly with the following expression, obtained from Eq. (4.40):

\[
A^T = V S^T U^T, \text{ then } A^T A = V S^T S V^T, \text{ and } A A^T = U S S^T U^T
\]  
(4.42)

It can be easily proved that both matrices, \( S^T S \) and \( S S^T \), will count with the same values in their respective diagonals, so both \( AA^T \) and \( A^T A \) will count with the
same eigenvalues, and the associated singular values from \( \mathbf{A} \) will be then uniquely determined. There will not be negative singular values in \( \mathbf{S} \) but, depending on the rank of the matrix, there could be null single values in its diagonal.

As it was mentioned before, in the inverse problem explored in this research work, the Jacobian matrices obtained typically count only with positive singular values in the diagonal of \( \mathbf{S} \). Nevertheless, in some cases, they could be very close to zero. That circumstance will be commented in more detail later as it will have important implications on the optimum instrumentation that the kind of gas turbine engines under study should have installed. Returning to the singular values in \( \mathbf{S} \), they are organized in the diagonal, as usual, decreasingly. So, the singular value with the highest absolute value will occupy the \((1,1)\) position in the matrix \( \mathbf{S} \) (or \( s_{11} \)):

\[
s_1 \geq s_2 \geq \cdots \geq s_Q \geq 0
\]  

(4.43)

The expression in Eq. (4.40) constitutes the single value decomposition (SVD) of the generic rectangular matrix \( \mathbf{A} \), as it is schematically shown in the Figure 68. Using the MATLAB command \([\mathbf{U}, \mathbf{S}, \mathbf{V}] = \text{svd}(\mathbf{A}, \text{‘econ’})\) with a generic rectangular real matrix \( \mathbf{A} \), like the ones treated so far, the decomposition of this matrix is obtained in an inexpensive way, from a computational time perspective. The option "econ", in the "svd" function implemented in MATLAB®, simplifies the output from the algorithm, removing rows and columns populated by zeros (leading to the reduced version of the SVD).

![Figure 68: Schematic representation of a full SVD of a generic rectangular matrix A.](image)

If the expression in Eq. (4.40) is developed, after obtaining the singular values of \( \mathbf{A} \) (see next Figure 69), breaking down the matrix in different terms based on the singular values obtained, it can be verified how the elements in the original matrix \( \mathbf{A} \) can be expressed in the form of products of terms containing singular values and the components of the left and right modes, pairing properly the different components, like in Eq. (4.44). This expression evinces that a good approximation of \( \mathbf{A} \) could be obtained if terms with small singular values are neglected.
\[ A_{ij} = \sum_{q=1}^{Q} s_q U_{iq} V_{jq} = s_1 U_{i1} V_{j1} + \cdots + s_Q U_{iQ} V_{jQ} \quad (4.44) \]

Figure 69: Schematic representation of the product of the singular values and left and right modes, when \( I > J \).

Here is where the concept of the decomposition of the original generic rectangular matrix \( A \), in a sum of terms weighted by the different singular values, is more evident. A truncation, like the one represented in the Figure 70, retaining just the first most remarkable \( \bar{Q} < Q \) singular values, leads to a rank-\( \bar{Q} \) matrix, \( A^{\text{trunc}} \), approximation of \( A \), given by the expression in Eq. (4.45).

\[ A_{ij}^{\text{trunc.}} = \sum_{q=1}^{\bar{Q}} s_q U_{iq} V_{jq} = s_1 U_{i1} V_{j1} + \cdots + s_{\bar{Q}} U_{i\bar{Q}} V_{j\bar{Q}} \quad (4.45) \]

Figure 70: SVD, Product of the singular values and left and right modes.

That truncation will reduce the complexity in the problem by keeping only the most remarkable information from the original matrix \( A \). However, it will be necessary to establish a criterion to decide how much information must be retained within the truncation. Attending to the properties of the traces in matrices, the relative root-mean-square (RRMS) error could be obtained from Eq. (4.46), where the operator \( \|\cdots\|_F \) corresponds with the Frobenius norm, as in Eq. (4.47).
RRMS error \( \equiv \frac{\|A_{\text{trunc}} - A\|_F}{\|A\|_F} = \sqrt{\frac{s_{Q+1}^2 + \cdots + s_Q^2}{s_1^2 + \cdots + s_Q^2}} \) \hspace{1cm} (4.46)

\[ \|A\|_F = \sqrt{\sum_{i,j} |a_{ij}|^2} \] \hspace{1cm} (4.47)

And here is where the mentioned criterion must be imposed to keep that error inside acceptable limits. For instance, requiring conditions like the following one for some (typically small) threshold \( \varepsilon_{\text{SVD}} > 0 \):

\[ \text{RRMS error} \leq \varepsilon_{\text{SVD}} \] \hspace{1cm} (4.48)

If the matrix \( A \) was square, symmetric, and definite positive, then its factorization with eigenvalues and its decomposition with singular values would be totally coincident. As the matrix deviates from the symmetry and positive definition, the difference between eigenvalues and singular values will be more evident.

The factorization with eigenvalues is the preferred tool when analyzing applications operating within the same vectorial subspace. SVD is the adequate one to analyze applications in between different vectorial spaces, eventually with different dimensions. Regarding this last comment, it could be possible to provide a geometrical interpretation of the SVD. The SVD breaks down a linear transformation, given by the matrix \( A \), into a canonical decomposition of three consecutive transformations given by \( U \), \( S \), and \( V^T \). Matrices \( U \) and \( V^T \) are orthonormal and represent either rotations or reflections of the subspace formed by the vectors in their respective columns and rows. Matrix \( S \) represents a scaling of each coordinate by a factor given by each singular value. The singular values of a rectangular matrix \( A \), with a \((I \times J)\) size, can be considered as the relative length of the semiaxis of an \( J \)-dimensional ellipsoid in an \( I \)-dimensional subspace, meanwhile the associated vectors of the decomposition determine the direction of those semiaxis. This is easier to visualize with a 3D example, like in the Figure 71, with geometric bodies such as spheres, ellipsoids, circumferences, or ellipses.

![Figure 71: Geometrical interpretation of the SVD, in 3D, given a rectangular matrix.](image-url)
Another relevant relationship that will be used later in this work comes from multiplying in Eq. (4.45) by $V_{jq}$ and adding in $j$. Doing so, the following $Q - \tilde{Q}$ correlations among the columns of $A^{\text{trunc.}}$ are obtained:

$$\sum_{j=1}^{I} A_{ij}^{\text{trunc}} V_{jq} = \left( \sum_{q=1}^{\tilde{Q}} s_q U_{iq} \right) \left( \sum_{j=1}^{I} V_{jq} V_{jq}' \right) = 0 \text{ ; for } i = 1, ..., I, \tilde{Q} < q' \leq Q. \quad (4.49)$$

This happens because the right modes are mutually orthonormal and $q \leq \tilde{Q}$ is different to $q' > \tilde{Q}$. If the truncating threshold $\varepsilon_{\text{SVD}}$ is small enough, then the matrices $A$ and $A^{\text{trunc.}}$ will be still quite similar and the correlations in Eq. (4.49) will be converted into the following approximate correlations among the columns of $A$:

$$\sum_{j=1}^{I} A_{ij} V_{jq} \approx 0 \text{ ; for } i = 1, ..., I, \tilde{Q} < q' \leq Q. \quad (4.50)$$

It can be proved that approximate correlations among the rows of $A$ are similarly calculated using left modes instead of the right modes. The importance of these approximate correlations will become clearer when evaluating the optimal distribution of sensors in the engine under consideration.

Another useful application of the SVD is found when performing the calculation of the CN of a generic matrix $A$. This number is also called the 2-norm condition number and is obtained as the ratio of the highest singular value of $A$ to the lowest (i.e., between the first and last values in the diagonal of matrix $S$). The CN of a matrix, as it was indicated before, has a critical influence when computing the inverse of that matrix and, consequently, when solving linear systems involving that same matrix (like it will happen with the Jacobian matrices of the inverse problem).

The higher the CN of the matrix, the worse the accuracy obtained because the equation system will not be representing properly the problem. In this sense, monitoring the value of the CN of relevant matrices in optimization problems, such as Jacobians or Hessians, gives valuable information on the quality of the solution obtained. Inverse problems in which the associated matrices count with high CN will likely go through serious convergence issues and the accuracy of the solution, when obtained, will be far from the chased value. MATLAB® directly calculates the CN of a matrix with the command “cond”.

Another important feature of the SVD of a generic rectangular matrix $A$ is the easy way of computing its inverse, $A^{-1}$, that it provides if the inverse matrix exists (in a regular matrix with rank = $Q$), or its most approximated version so called pseudo-inverse, $A^{-1}_{\text{pseud}}$, otherwise (with lower ranked matrices). Here, it will be necessary to pay attention to the size of the different matrices involved in the decomposition. In Eq. (4.40), applying the inverse operator to the three generated matrices, it is possible to reach to the following expression:

$$A^{-1} = VS^{-1}U^T \quad (4.51)$$

As the matrices $U$ and $V$ are orthonormal, the calculation of their inverse is quick and easy, just by transposing. The singular values ($s_1, ..., s_Q$) are in the diagonal of the matrix $S$, which size will be generically (1 x $J$), so the reciprocals of those singular values ($1/s_1, ..., 1/s_Q$) will be in the diagonal of the matrix $S^{-1}$, which size will be therefore ($J$ x 1).
If the rank of $A$ is less than $Q$, the inverse does not exist. However, an approximate matrix can be used by means of the SVD using only the first $\tilde{Q}$ singular values. $S$ will be then then an $(\tilde{Q} \times \tilde{Q})$ matrix and $U$ and $V$ will be shrunk accordingly. The pseudo-inverse of the pseudo-inverse of a matrix will return the very same matrix. When the inverse of a matrix exists, it coincides with its pseudo-inverse. This calculation was considered when solving the inverse problem by means of the adapted Newton method. However, no pseudo-inverse matrices were used, as no totally null singular values were obtained. Such circumstance led during certain calculations to very high CNs. That situation will be explored in the next section.

Beyond the applications of the SVD mentioned so far, there is a varied and extremely helpful list of applications in numerical analysis, including the calculation of the effective rank of a matrix, information filtering, least squares approximations, latent semantic analysis, etc., that will not be described here for brevity and conciseness to the problem relative to the research within this work.

For the sake of the historical reference, the standard SVD was first developed by E. Beltrami and C. Jordan, separately, in 1873 and 1874, respectively, for square matrices [276]. C. Eckart and G. J. Young [81] generalized the technique for rectangular matrices. Another milestone is owed to G. Golub and W. Kahan who introduced the SVD in numerical analysis in 1965 [123]. Since those studies were published, many other versions of SVD have appeared. The one used in this work is the one available in MATLAB®, based on the LAPACK (Linear Algebra PACKage for FORTRAN90) library subroutines, developed by T. M. Toolan.

### 4.8. Correlations between degradations

Coming back to the Jacobian matrices as they were formulated in Eq. (4.30), $J(T_{4t}, \bar{X})$, which size was $(10 \times 10)$, they were found with a large CN, higher than $\sim 1 \cdot 10^5$, when considering no degradations (this means $\bar{X} = \bar{0}$), and different typical $T_{4t}$ values. When analyzing the singular values of these Jacobian matrices, $J(T_{4t}, \bar{0})$, it was evident that the tenth singular value was systematically much lower than the rest, as it is shown in Table 15:

<table>
<thead>
<tr>
<th>$T_{4t}$</th>
<th>$S_1$</th>
<th>$S_2$</th>
<th>$S_3$</th>
<th>$S_4$</th>
<th>$S_5$</th>
<th>$S_6$</th>
<th>$S_7$</th>
<th>$S_8$</th>
<th>$S_9$</th>
<th>$S_{10}$</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.325</td>
<td>0.717</td>
<td>0.455</td>
<td>0.113</td>
<td>0.102</td>
<td>0.074</td>
<td>0.051</td>
<td>0.042</td>
<td>0.023</td>
<td>0.013</td>
<td>$1.8 \cdot 10^{-6}$</td>
</tr>
<tr>
<td>1.375</td>
<td>0.820</td>
<td>0.413</td>
<td>0.114</td>
<td>0.105</td>
<td>0.082</td>
<td>0.057</td>
<td>0.043</td>
<td>0.024</td>
<td>0.014</td>
<td>$2.1 \cdot 10^{-4}$</td>
</tr>
<tr>
<td>1.425</td>
<td>0.878</td>
<td>0.392</td>
<td>0.117</td>
<td>0.109</td>
<td>0.090</td>
<td>0.063</td>
<td>0.045</td>
<td>0.025</td>
<td>0.015</td>
<td>$1.8 \cdot 10^{-4}$</td>
</tr>
<tr>
<td>1.475</td>
<td>0.895</td>
<td>0.377</td>
<td>0.121</td>
<td>0.115</td>
<td>0.097</td>
<td>0.068</td>
<td>0.047</td>
<td>0.026</td>
<td>0.016</td>
<td>$5.2 \cdot 10^{-4}$</td>
</tr>
<tr>
<td>1.525</td>
<td>0.920</td>
<td>0.390</td>
<td>0.130</td>
<td>0.120</td>
<td>0.101</td>
<td>0.072</td>
<td>0.049</td>
<td>0.027</td>
<td>0.017</td>
<td>$3.2 \cdot 10^{-4}$</td>
</tr>
</tbody>
</table>

**Table 15:** Associated singular values to the Jacobians, $J(T_{4t}, \bar{0})$, for different typical $T_{4t}$ cruise values.

The expression provided by Eq. (4.50), leads to the same quantity of approximate correlations as the number of singular values obtained from the Jacobian matrices that are much smaller than the rest (almost equal to zero), just by making $J(T_{4t}, \bar{0}) = A$, given the fact the degradations correspond with the columns of the Jacobians. This approach leads naturally to the next expression (note the products are between scalars):
\[ \sum_{j=1}^{N_{\text{deg}}} b_j J_{ij} \approx 0 \text{ ; with } b_j = V_{jN_{\text{deg}}} \quad (4.52) \]

For \( j = 1, \ldots, N_{\text{deg}} \). In this case, the number of degradations is equal to ten. The term \( V_{10} \) corresponds with the elements in the last column of the right SVD modes. That column is the one associated to the smallest singular value obtained.

Given the low value of the last singular value, if any incremental degradation vector is projected in the orthogonal base determined by the right SVD modes, the component in the direction of the tenth singular value will be always near to zero. Which is like establishing an orthogonality condition with the tenth right SVD mode. There would be as many orthogonality conditions as singular values close to zero. Then, as per the expression given by Eq. (4.16), the approximate correlations formulated in Eq. (4.52) can be considered approximate correlations between the ten incremental degradation vector components (and eventually, given the way the Jacobians are calculated, between the degradation vector components):

\[ \sum_{j=1}^{10} b_j \Delta X(j) \approx 0 \quad (4.53) \]

Table 16 contains the values of the \( b_j \) coefficients for the same \( T_{4t} \) values used in Table 15. Consistently, the highest coefficients found, by far, were \( b_7 \), \( b_9 \) and \( b_{10} \) (note the other coefficients are multiplied by 100 in the table). The rest were almost negligible. In this sense, the value of the typical Euclidean norm for the \( \bar{V}_{10} \) mode associated to \( s_{10} \) is equal to 1, given the fact the column vectors in the matrix \( \bar{V} \), coming from the SVD of the Jacobian, form an orthonormal base. As only those 3 components of \( \bar{V}_{10} \) almost already gave a value of 1 in the norm by themselves, that led to an almost negligible contribution from the rest of components:

<table>
<thead>
<tr>
<th>( T_{4t} )</th>
<th>100b₁</th>
<th>100b₂</th>
<th>100b₃</th>
<th>100b₄</th>
<th>100b₅</th>
<th>100b₆</th>
<th>b₇</th>
<th>100b₈</th>
<th>b₉</th>
<th>b₁₀</th>
</tr>
</thead>
<tbody>
<tr>
<td>1,325</td>
<td>-0.01</td>
<td>0.06</td>
<td>0.28</td>
<td>0.36</td>
<td>-0.30</td>
<td>0.24</td>
<td>-0.53</td>
<td>0.05</td>
<td>0.31</td>
<td>-0.79</td>
</tr>
<tr>
<td>1,375</td>
<td>0.57</td>
<td>0.27</td>
<td>0.08</td>
<td>-0.04</td>
<td>-0.06</td>
<td>0.11</td>
<td>0.53</td>
<td>0.49</td>
<td>-0.32</td>
<td>0.79</td>
</tr>
<tr>
<td>1,425</td>
<td>1.16</td>
<td>0.60</td>
<td>0.37</td>
<td>0.13</td>
<td>-1.17</td>
<td>-0.86</td>
<td>0.52</td>
<td>1.15</td>
<td>-0.32</td>
<td>0.79</td>
</tr>
<tr>
<td>1,475</td>
<td>0.61</td>
<td>0.16</td>
<td>0.41</td>
<td>-0.51</td>
<td>-0.92</td>
<td>-0.94</td>
<td>-0.54</td>
<td>1.00</td>
<td>0.32</td>
<td>-0.78</td>
</tr>
<tr>
<td>1,525</td>
<td>-0.40</td>
<td>0.16</td>
<td>-0.08</td>
<td>-0.75</td>
<td>0.66</td>
<td>0.68</td>
<td>-0.53</td>
<td>-0.68</td>
<td>0.32</td>
<td>-0.79</td>
</tr>
</tbody>
</table>

Table 16: List of \( b_j \) coefficients for the \( T_{4t} \) values used in the previous table.

As can be seen in the previous table, the values of the three mentioned coefficients (highlighted in a different color) were almost coincident for all the \( T_{4t} \) considered. The only difference found between temperatures was in the sign of these coefficients. However, given the combination of signs between coefficients, that difference did not impact the following expression:

\[ b_7 \Delta X(7) + b_9 \Delta X(9) + b_{10} \Delta X(10) \approx 0 \quad (4.54) \]

With this expression, it would be evident that one of the involved incremental degradations could be perfectly ignored during the solving process of the inverse problem, because it could be obtained as a linear combination of other two terms. So, once the rest of the degradation vector components (a total of 9 in this problem) was calculated, then Eq. (4.54) would allow to get the value of the remaining one.
At least, this situation was detected for the clean state (null degradation values) keeping the initial set of sensors. It is doubtful that a so clear correlation like the one in Eq. (4.54) would appear for whatever other degradation condition. Some numerical effects could hide this correlation in case of remain. The reality is that the physical nature of the degradation vector components (H&Q parameters) indicates that these are typically independent among each other, so eliminating one of the three mentioned degradations from the inverse problem would require, as a minimum, a further analysis before applying such result. And so, it was done.

After performing different tests with various sets of sensors, the correlation disappeared when selecting different instrumentation to provide information to the inverse problem. The three correlated degradations are related to both the HPT and LPT of the engine. More specifically, the tenth degradation is the one relative to the flow capacity in the LPT ($\Gamma_{\text{LPT}}$), which depends on the total pressure before this module, $P_{45t}$, and this sensor was not included in the original set of sensors.

A test was done, including a new sensor to the instrumentation mounted in the engine capable to provide readings of $P_{45t}$, leading to Jacobian matrices of size (11 x 10). It was calculated the condition number for the resultant matrices, considering a new and clean state (i.e., $\mathbf{J}(T_{4t}, \mathbf{0}))$, and the exact $T_{4t}$ values used before. And it was found that those numbers were in the order of $\sim 50$, which was a remarkable lower value when comparing with the ones obtained with the initial set of sensors without $P_{45t}$ (above $\sim 1 \cdot 10^5$). With this result, just by adding a new sensor, it would be possible to avoid calculations at two different $T_{4t}$ values because the resultant condition numbers of the associated Jacobian matrices would be good enough. And this is also applicable to the case in which the $T_{4t}$ was left as an unknown. Now the Jacobian matrices would count with a size of (11 x 11), and only a slightly higher condition number comparing with the case without $T_{4t}$ as unknown.

Unfortunately, substituting one of the existing sensors in the original sensors’ set by the new one measuring $P_{45t}$ is not always equally efficient. For instance, replacing the sensor monitoring $P_{25t}$ by a $P_{45t}$ probe will lead to a condition number of $\sim 1 \cdot 10^4$, which is better than the initial condition number obtained with the original set of sensors, but it is still very high. And the same situation was found replacing other existing sensors with the new $P_{45t}$ probe. This will be illustrated with some examples in the next chapter. So, the strategy to follow, when designing the instrumentation system in an aircraft engine, should be as follows:

- First, analyzing which is the minimum instrumentation that provides a better condition number, making sure no correlations appear in between degradations as exposed before.
- Even when obtaining more information from the engine, by adding extra sensors, seems appealing from a monitoring perspective, the extra sensors would introduce also a higher technical complexity and some more maintenance efforts to cover with the added hardware. So, it would be always desirable to maintain the instrumentation installed in the engine to the minimum required when a new monitoring system is being designed.
- In this problem, given the initial set of sensors already available in the CFM56-5A, adding a new $P_{45t}$ probe to the engine’s instrumentation would be the preferred solution, as installing a new sensor will not modify the existing instrumentation’s philosophy (which would affect to existing certifications and airworthiness), it will just be upgraded.
Several references were found showing how the sensors could produce spurious data due to discrepancies between the engine’s model and the measurements taken in the practice [3]. Any disagreement between the model assumptions and reality, given by incompleteness in the information retrieved by the instrumentation, for instance (see [2]), potentially introduces bias and other numerical artifacts into the input parameters and their relations. Here, inconsistencies were identified and excluded based on a comparison of the results from sensor data against alternative instrumentation configurations, decorrelating thus the data produced, as it has been proposed by other authors in the past.

4.9. Calculations including one additional sensor

In this section of the work, it will be presented the strategy of adding the new $P_{45t}$ probe to the initial instrumentation mounted in the engine. The resultant new sensors’ set was called “improved set of sensors”. Given the analysis developed in the previous sections, only one $T_{4t}$ value was considered. Together with the adapted Newton method, the improvements achieved with the new sensor were verified in the other techniques used in the study, like the SQP method or the GA.

Initially, the 10 unknown degradation vector components were calculated, permitting their variation in between 0% and 2%. As it will be proved later, the method is capable to solve numerically cases allowing the degradations to move within far wider ranges, not necessarily corresponding with likely situations in the practice. This implies a considerable degree of robustness in the method, as it will be illustrated in the next chapter.

Now, the only $T_{4t}$ value considered was in the range between 1,300 K and 1,550 K. This range is considerably wider than the ones used for the two $T_{4t}$ values used previously for the enlarged Jacobian matrices. Values that also needed a separation of, at least, 50 K to avoid coupling problems in the results that could lead to higher condition numbers in the inverse problem.

The unknown parameter used to obtain $T_{4t}$ was scaled as follows (similar expressions were used with identical results), repeating the same formulation used in Eq. (4.15) to keep the values of the parameter in between 0% and 2%, as the rest of the degradations (similar approach followed for wider ranges up to 20% or 50%):

$$X_T = 2 \cdot \frac{T_{4t} - T_{4t,\min}}{T_{4t,\max} - T_{4t,\min}} \quad (4.55)$$

Where $T_{4t,\min}$ and $T_{4t,\max}$ correspond with the minimum and maximum $T_{4t}$ values allowed (i.e., 1,300 K and 1,550 K, respectively). The rest of components were scaled the same way it was done in the previous sections of the work. With this approach, the formulation shown in Eq. (4.2) is substituted by:

$$f (\bar{X}) = \bar{Y}_{\text{meas}} \quad (4.56)$$

Here, the vector $\bar{X}$ contained the 10 initial components of the degradation vector plus the scaled parameter for the TIT, which constituted the solution of the inverse problem. So, it was a column vector of size $(11 \times 1)$. Again, $\bar{Y}_{\text{meas}}$ was the vector with the exact readings from the instrumentation (simulated by PROOSIS®).
Once an initial value, $\bar{X}_0$, was considered, the iterative process could be initiated, similarly to how it was indicated in Eq. (4.24) and Eq. (4.25):

$$\bar{X}_{k+1} = \bar{X}_k + \Delta \bar{X}_k$$  \hspace{1cm} (4.57)

Where the incremental vector $\Delta \bar{X}_k$ was obtained from:

$$J_k \Delta \bar{X}_k = \bar{Y}_{\text{meas}} - \bar{Y}_k$$  \hspace{1cm} (4.58)

The Jacobian matrix of the left side of Eq. (4.58) at $\bar{X}_k$ was represented by $J_k$. The vector $\bar{Y}_k$ was given by $f(\bar{X}_k)$. To avoid again negative values for the degradations, especially during the first steps in the iteration process, a constraint was imposed to set the degradations to zero when outcomes below were obtained. The iterative process continued until reaching convergence given by a similar criterion to the one in Eq. (4.29):

$$\| \bar{X}_{k+1} - \bar{X}_k \|_{\text{RMS}} < \varepsilon_1 \text{ and } \| \bar{Y}_{\text{meas}} - \bar{Y}_k \|_{\text{RMS}} < \varepsilon_2$$  \hspace{1cm} (4.59)

The thresholds were chosen small enough, in this case $\varepsilon_1 = \varepsilon_2 = 1 \cdot 10^{-4}$. Initially, only degradations up to a 2% were allowed and the $T_{4t}$ was computed as an unknown together with the rest of degradations. The associated scaled parameter for the TIT was also kept in between 0% and 2%. So, improved results were obtained with the improved set of sensors, after adding the $P_{45t}$ probe. Comparing with the cases in which only 10 sensors were employed, there were then several remarkable differences that must be highlighted (results in Chapter 5):

- Only one $T_{4t}$ value was required to keep the condition number of the problem inside acceptable limits. With the previous set of sensors, it was necessary to use two $T_{4t}$ values, leading eventually to an overdetermined problem.
- The correlation found before between incremental degradation vector components 7th, 9th, and 10th disappeared. This change led to the conclusion that a wise selection of sensors will avoid similar situations in the future.
- The accuracy was also improved for both degradation vector components and $T_{4t}$, getting convergence after few iterations, with computational times relatively low (but not yet in the order of the fast real-time applications with the computer used during the study).

After solving cases where the degradations allowed were up to 2%, the next step was trying cases with higher degradation levels. Continuing with this analysis and trying to provide more evidence pointing in the direction of the global convexity of the inverse problem, cases with random initial and exact conditions were run with degradation vector components varying up to a 50%. These cases fall inevitably outside the usual regions in the maps of the components, so the results must be managed carefully and must be understood purely as proofs of numerical robustness and punctual convexity. The model developed with PROOsis® covered numerically somehow these regions of the maps, and it managed to provide numerical solutions for the direct problem, however such kind of results must be understood as clear technical issues, either in the engine or in the instrumentation.
4.10. Analysis of the model application, convexity, uniqueness, and convergence

One of the most relevant concepts in optimization is the convexity, geometrical property that facilitates greatly the solving process. The convexity is very intuitive, especially in 2D and 3D problems, nevertheless the lack of visualization in problems involving higher dimensions could lead to misleading conclusions, so it is convenient to provide a formal definition, showing then some of its most remarkable features when the right premises are met.

Both sets and functions can be categorized as convex or non-convex [211]. In Figure 72, both the set of possible degradations in the engine (set \(X\)) and the set of possible readings from instrumentation (set \(Y\)) are represented. Additionally, the application between both sets, the model of the engine (provided by PROOSIS®), is indicated with dotted arrows between sets (also the norm). Those sets could be considered, for the sake of simplicity and without lack of generality, to be inside \(\mathbb{R}^{11}\).

That would be the case in which the \(P_{45t}\) probe was added to the engine’s instrumentation and \(T_{4t}\) was calculated as another unknown component of the degradations’ vectors. However, the following explanation would be also applicable to the cases in which vectors in \(\mathbb{R}^{10}\) (initial problem definition) or \(\mathbb{R}^{12}\) (after adding two \(T_{4t}\) values to the unknown degradations) where employed. The only caution to keep in this last case has to do with the way the surjection would be done in the application providing the values in the set \(Y\) from the values in the set \(X\).

The convexity of a set is defined as follows: a set \(A \subset \mathbb{R}^n\) is convex when the straight segment connecting whatever two points of \(A\) (e.g., \(C\) and \(D\)) lies completely inside \(A\). This condition can be generically expressed in the following way:

\[
given \ C \in A \text{ and } D \in A \rightarrow (\lambda C + (1 - \lambda) D) \in A, \ \forall \lambda \in [0,1] \subset \mathbb{R} \quad (4.60)
\]

As per the previous definition, it results evident the degradations form a convex set in \(\mathbb{R}^{11}\) as it is always included inside the set \(X\) the intermediate segment of points connecting two different degradation states (\(X_1\) and \(X_2\) in Figure 72).
This is particularly clear when one of the points represents the new and clean condition of the engine (baseline). All the intermediate degradation points that lead to the second degradation point are feasible degradation points, independently of the route followed by the engine to get to the second point. Maybe the degradation in the engine did not follow a linear evolution, but there are chances to observe such kind of routes, particularly when the degradations are small.

Similar consideration could be made with two different readings ($\bar{Y}_1$ and $\bar{Y}_2$) and the segment that connects them. From a physical standpoint, this means that any intermediate sensor reading in between two valid different readings is always possible, which is true irrelevant of how likely those intermediate readings would appear in the instrumentation during the degradation of the engine. Sensors count with ranges of operation specifically designed to cover any potential engine condition, and it is assumed that the engine will move always inside those ranges.

As per Eq. (4.60), regarding the concept of convexity of a set, it is not relevant if the engine actually traveled through the intermediate points between $\bar{C}$ and $\bar{D}$. It is just relevant to evaluate if those points are part of the set.

Obviously, this statement will not be applicable to the whole $\mathbb{R}^{11}$ space for both degradations and sensors’ readings. Some cases are simply incompatible with the physical model of the engine. Some other cases, even when feasible from a numerical methodology standpoint, will be so unusual in the practice that could be perfectly discarded. Gas turbine engines are exposed to certain conditions, sometimes truly extreme, but not all imaginable situations are equally likely to happen. In fact, it is very important to determine very carefully the limits of the sets, $X$ and $Y$, for the cases contemplated in the inverse problems solved in this work. Otherwise, solutions without sense could be obtained and taken as valid, or a long calculation time could be invested to reach to a solution that typically will not be ever present during real operation.

The physical model of the engine that has been used in this work, as it happens with the ones used in the industry, and as it was detailed in the previous chapter, does not rely only on algebraic equations or only on any other kind of mathematical relationship in between input parameters and output variables. The performance models of every existing gas turbine engine manufactured nowadays utilize specific performance maps, which contain the possible values that the relevant magnitudes defining the behavior of the main components of the engine, such as the compressors and turbines. The existence of those different maps is, initially, evidence of the independent nature of the different H&Q parameters considered in this work for the main engine components, which count with a specific map for each one of them. They are also evidence of the clear nonlinear nature of the problem under consideration. The maps provide a thorough information, obtained after demanding and rigorous testing processes carried out in instrumented test cells specially prepared to monitor the same variables that have been mentioned in this work, and they arejealously preserved by the OEMs.

Figure 73 shows a pair of performance maps used by PROOSIS®, for both HPC and HPT, highlighting the part of the maps in which degradation values associated with efficiencies are up to a 20% ($\eta$ curve of 0.8 indicated in the map). As can be verified, most of the information shown in the map is already contained withing a 20% of degradation. The running line of the engine will fall inside that region. The usual degradation values during operation will be considerably lower, so the operational points will be typically located clearly inside the colored zones.
So, the reason why the range of the possible degradations was maintained initially defined inside more restricted values (up to a 2%) was because those degradations would cover already most of the accumulated deterioration in the engine since it enters to revenue service, and even after thousands of operating hours. Similar situation occurs with the maps of the rest of main components, used by PROOSIS® for the FAN, LPC or LPT, regarding the gas turbine engine under consideration. Similar situation will be found dealing with other gas turbine engine models and configurations. Their respective maps will not be so different:

- Up to a 2% will cover most of potential cases.
- Up to a 20% in case of unexpected very severe events may occur.
- Up to a 50%, catastrophic event, just to test the robustness of the method.

Statistically speaking, the engine degradations will remain most of the time inside acceptable limits for this methodology, so no numerical convergence problems will happen in the practice. However, the available experience and the knowledge on the technology under study must guide the numerical calculations. Maps’ acceptable regions, according with the different OEMs, must be clearly determined and some interpretation must be provided if the engine would eventually go outside those acceptable regions. PROOSIS® can show warning messages if such situation might occur. In this sense, the methods mentioned in previous sections took random values for both initial and exact final degradations. This was done to ensure the robustness of the method, but this situation will never happen in the reality. Degradations will not improve with time. These values will increase, slowly but steadily. So, given a certain degradation status in one engine, its future condition after some operational time will be typically a bit worse than what it originally was, with slightly lower efficiencies, and with more dirt accumulated in compressors meanwhile turbines’ internal surfaces result more eroded.

When the methods moved inside expectable scenarios (i.e., degradations up to 2% and initial conditions better than the final ones) the convergence was always verified. Only when these conditions were extended to far less likely situations the methods showed some difficulty to reach the required convergence (e.g., high number of iterations needed to reach a solution with the adapted Newton method).
Additionally, the sensors’ readings will normally be inside certain operational limits and that circumstance made possible the scaling process explained before, assuming typical average values in the usual cruise conditions for the engine. When one sensor or a group of sensors provide unusual values then that circumstance indicates that either a problem with the instrumentation, or with some components of the engine, is going on and needs of technical intervention and troubleshooting to restore the previous operational condition.

Regarding the OF used with the GAs or with the SQP, \( f(\bar{Y}) \rightarrow \mathbb{R} \), in which an optimization takes place, that function is a convex function if its domain is a convex set (already commented and verified), and if for any two different points in the set \( Y \), for instance \( \bar{Y}_1 \) and \( \bar{Y}_2 \) in Figure 72, the following property is satisfied:

\[
f(\lambda \bar{Y}_1 + (1 - \lambda) \bar{Y}_2) \leq \lambda f(\bar{Y}_1) + (1 - \lambda) f(\bar{Y}_2) \quad \text{for all } \lambda \in [0, 1] \subset \mathbb{R} \quad (4.61)
\]

The OF, given by Eq. (4.6), used with the GA and SQP techniques in this work, was built with norms, respecting their most relevant properties. And all the norms are convex, by definition, given the fact that they are positive definite applications, scalable in terms of absolute values, and compliant with the usual triangular inequality of Minkowski, leading to the next result:

\[
\|\lambda \bar{Y}_1 + (1 - \lambda) \bar{Y}_2\|_2 \leq \|\lambda \bar{Y}_1\|_2 + \|(1 - \lambda) \bar{Y}_2\|_2 = |\lambda| \|\bar{Y}_1\|_2 + |1 - \lambda| \|\bar{Y}_2\|_2 \quad (4.62)
\]

When the objective function used in the optimization process and the feasible set taken as domain of definition of such function are both convex, then any local solution of the problem is in fact a global solution, theorem easily demonstrable by reaching to a contradiction when assuming there is another minimum, which could be a global minimum (this is one of the first results provided in [211], given its simplicity and also its relevance).

The only point pending to verify if the inverse problem in this study counts, theoretically, with a global minimum, is the application between degradations and sensors’ readings. That application has been provided by PROOSIS® during the thousands of calculations performed in this research. The previous theorem would fail if there would not be a direct correspondence, one by one, between the points in the set of the degradations with the points in the sets of sensors’ readings. It is needed what is called in topology a relationship of injection in between these two sets of points. An absolute surjection, mathematically speaking, between these two sets is not really required (even when it could be desirable), as it is just interesting to find out the solution for the degradation conditions of the problem and not for all the possible conditions (some of them would be very unlikely to happen). The combination of injection and surjection would lead to a bijective mapping between sets, becoming the application given by PROOSIS® an isomorphism. Such bijective application would be guaranteed in case of existing a fully monotonic relationship between the points in the degradation sets and the points in the sensors’ readings set. In other words: If the variation of the different H&Q parameters implied a monotonic variation in the values from the sensors (simulated by the SW), entirely nonincreasing or nondecreasing, then the relationship between sets will be bijective, and therefore also injective, being then verified the premises of the convexity theorem. Unfortunately, as it can be seen in the following charts with plots of variations between 0% to 10% in the components of X, that is not guaranteed.
So, the next sensitivity analysis, evaluating instrumentation readings vs. degradation values, was performed upon a typical generic cruise flight condition, in a new and clean engine (baseline), described by the data in Table 17:

<table>
<thead>
<tr>
<th>Sensors</th>
<th>H&amp;Q Parameters</th>
</tr>
</thead>
<tbody>
<tr>
<td>$P_{13t} \text{ [Pa]}$</td>
<td>$\eta_{\text{FAN}}$ 1.00</td>
</tr>
<tr>
<td>$P_{23t} \text{ [Pa]}$</td>
<td>$\Gamma_{\text{FAN}}$ 1.00</td>
</tr>
<tr>
<td>$P_{3t} \text{ [Pa]}$</td>
<td>$\eta_{\text{HPC}}$ 1.00</td>
</tr>
<tr>
<td>$T_{23t} \text{ [K]}$</td>
<td>$\Gamma_{\text{HPC}}$ 1.00</td>
</tr>
<tr>
<td>$T_{3t} \text{ [K]}$</td>
<td>$\eta_{\text{LPC}}$ 1.00</td>
</tr>
<tr>
<td>$T_{43} \text{ [K]}$</td>
<td>$\Gamma_{\text{LPC}}$ 1.00</td>
</tr>
<tr>
<td>$T_{5t} \text{ [K]}$</td>
<td>$\eta_{\text{HPT}}$ 1.00</td>
</tr>
<tr>
<td>$N_e \text{ [rpm]}$</td>
<td>$\Gamma_{\text{HPT}}$ 1.00</td>
</tr>
<tr>
<td>$N_H \text{ [rpm]}$</td>
<td>$\eta_{\text{LPT}}$ 1.00</td>
</tr>
<tr>
<td>$W_F \text{ [kg/s]}$</td>
<td>$\Gamma_{\text{LPT}}$ 1.00</td>
</tr>
</tbody>
</table>

**Flight Data and Main Engine Performance Values**

| Altitude [m] | $F_N \text{ [N]}$ | 49,576.42 |
| Mach Number | $I_{\text{SP}} \text{ [m/s]}$ | 140.19 |
| $T_{4t} \text{ [K]}$ | $\text{TSFC} \text{ [g/(kN\cdot s)]}$ | 18.71 |

Table 17: Representative data of the engine (new and clean condition), simulated in a generic cruise flight, that were used for the sensitivity analysis. The performance values obtained from the engine (i.e., Thrust or $F_N$, specific impulse or $I_{\text{SP}}$, and TSFC) are also indicated.

The following charts show the value of the instrumentation, without scaling and the value of the degradations scaled (from 0% up to a 10% was enough) for reader’s convenience, to facilitate the understanding of the results.

![Figure 74: Variations in $P_{13t}$ when any of the components of $\bar{X}$ increases from the new and clean state.](image-url)
In Figure 75, it is indicated with thicker lines the lack of monotony in $P_{25t}$ when it varitates with the 3rd and 7th degradation components in $X$. The change in tendency happens in between 0% and 5% of degradation. So, for usual flights with lower degradation rates, that change in tendency would not be appreciated and the inverse problem would be solved without major issues, even when there would be flat areas in some tendencies. For higher degradations, the model is not injective.
Figure 77: Variations in $T_{25t}$ when any of the components of $\bar{X}$ increases from the new and clean state.

Figure 78: Variations in $T_{31t}$ when any of the components of $\bar{X}$ increases from the new and clean state.

Similar considerations must be done in Figure 77 than in Figure 75. Thicker lines show the lack of monotony in $T_{25t}$ when it variates with the 3rd and 7th components in $\bar{X}$. The rest of components in $\bar{X}$ do not provoke changes in tendency. Monotony in variables is common, and that circumstance explains the high rate of success with the techniques used before, even for great degradations. On the other hand, the lack of injection would explain the problems detected in certain cases.
Figure 79: Variations in $T_{45t}$ when any of the components of $\bar{X}$ increases from the new and clean state.

Figure 80: Variations in $T_{56t}$ when any of the components of $\bar{X}$ increases from the new and clean state.

Figure 79 and Figure 80 show monotonic behaviors in the sensors up to a 10% of degradation in the components of $\bar{X}$. However, some flat behaviors are also present in the charts, pointing to changes of tendency. For degradations higher than a 5% the numerical methods could find some issues when trying to find the global minimum, as different degradation values in $\bar{X}$ could correspond with the same sensor reading in $\bar{Y}$. This circumstance could redound in longer computational times to find the right solution. The more intense the degradation is during a cruise, the more chances of suffering these potential problems caused by the loss of injectivity.
Monotonic behavior found also in Figure 81, Figure 82, and Figure 83, with some flat behavior in certain variables. It must be clarified that for very severe damages or sudden catastrophic events, implying degradations higher than a 20% during operation, maps should be verified. It could happen that the numerical methods operate with data that fall outside acceptable regions for the engine. In such case, the complete method would fail. That would explain some problems when obtaining the solution in several cases of the thousands that were run. These problems of convergence were rare with degradations until a 20% (in the order of less than one case per one hundred cases run), needing of extra iterations and more relaxed termination criteria to end up finding the solution, when it was possible.

Figure 81: Variations in $N_L$ when any of the components of $\bar{X}$ increases from the new and clean state.

Figure 82: Variations in $N_H$ when any of the components of $\bar{X}$ increases from the new and clean state.
Figure 83: Variations in $W_f$ when any of the components of $\bar{X}$ increases from the new and clean state.

So, the problem described in the previous sections will count with a global unique solution if the variations in the H&Q parameters are not excessive. The following criteria could be established to categorize the issues that may happen:

- When considering degradations up to a 2%, situation that represents most of cases in the reality, the techniques used in this study were perfectly capable to find the solution with acceptable levels of accuracy. Degradations reaching a 5% are considered rare during normal operation in a flight. Even so, as per the previous sensitivity analysis, there should not be issues in such range. In fact, that was the experience when running cases in this study. It is not expected that any engine operator would allow to the machine continue working after such degradation without inspection or repair. This is the main case of interest for this study, as it is mainly pursued the monitoring and identification of slight but continuous degradations.

- For degradations between 5% and 20%, implying severe degradations caused by an unexpected event, techniques worked well most of the times, but certain cases needed of extra iterations to find the solution with the required accuracy. Some cases reached convergence in degradations but not in TIT, evincing the model lack of injection. The TIT was restricted in those cases to a range of TIT $\pm$ 50 K to separate the desired solution from other spurious results. Some modifications could be considered for the termination techniques in these rare situations to reach to the solution. It would be recommendable to verify the maps of the different components of the engine to make sure the methods are working inside safe operational regions.

- For catastrophic events, hundreds of cases were still solved as it was previously indicated given the robustness of the method, but also serious issues were found in certain cases preventing the convergence. Even if the model delivers results, they must be analyzed (both maps and uniqueness).
The way to address uniqueness problems, by restricting the solution candidates’ list to ranges of TIT ± 50 K, is illustrated in Figure 84. This decision was made because, in certain cases, the technique was capable to calculate with high accuracy the degradations when dealing with points inside acceptable regions, but with TIT values ~100 K different to the solution. The lack of uniqueness was mitigated, for most of the cases, with this action. In the rare event of finding the TIT still few degrees different (~5 K), the case was considered acceptable, bearing in mind the full scale of the TC used in the engine to measure high temperatures. The TIT will not vary too much during a real steady cruise stage from previous values.

![Figure 84: Restriction imposed to the TIT values. The target was to solve uniqueness' lacks in the model.](image)

Regarding the points that may be out of the acceptable regions in the different components’ maps, meaning by acceptable regions the loci inside the solution space where the OEM of the engine expected the machine to be operated, some comment should be made. Applying the methodology to points, in the 10D or 11D solution space (depending on if \(P_{4st}\) and TIT are included or not), which components are up to a 5% of degradation, there was low risk of finding a problematic case. As soon as the considered degradations reached values ~10%, the chances to find a problematic case were higher. Multiple test cases were run with certain problematic points, trying to find convergence from many different initial points. Some cases could be solved, when tried from appropriate initial points, but some others remained unsolved, even when the methodology was always close to the exact solution. Unfortunately, the convergence required by termination criteria, was not finally met. These results suggested the existence of non-acceptable regions where maps were not correctly set up by the OEM. These loci represented regions where it is not guaranteed the unit will be working inside efficient or safe conditions. OEMs do not produce machines to work severely deteriorated, so some combination of degradations in the different components of the engine will fall into regions where OEMs did not expect the unit to be operated. In this sense, PROOSIS® counts with a code (so-called ESI error code, see Figure 85) to identify potentially problematic cases, because of points located out of acceptable regions, including warnings for user’s information. With that code, and the associated message in the Command Console of the user’s interface, it is possible to identify which component is affected. This feature could be also used to improve the diagnostic capabilities of the methodology, by defining a comprehensive list of potential issues that may occur when interacting with non-acceptable regions of the solution space.

![Figure 85: Couple of ESI error code numbers and associated messages in PROOSIS® (see [234] for further details).](image)
4.11.- The use of tensors and HOSVD for the resolution of the inverse problem

The previous set of techniques applied to the inverse problem counted with one common characteristic: They all made use of the complete engine performance model provided by PROOSIS®. Selecting wisely among them, it was possible to provide accurate results in few seconds (this is summarized in the next chapter), when the level of degradation of the different H&Q parameters was below 2%, or in longer times (ranging from seconds to hours) when the degradations considered were considerably higher, but always guaranteeing the effectiveness of the whole method (guarantee eventually given by the GAs’ pure exploration). Whatever the technique that might be finally applied, the use of the whole model was a fact.

The question that arose then was if the full model would be always necessary, given the usual degradation levels that a turbofan engine experiments, statistically speaking, during its revenue service life. Severe sudden degradations do not happen frequently, meanwhile slow but steady losses of efficiency in the engine components, caused by progressive accumulation of dirt in compressors and other internal surfaces, and by the increasing erosion in the hot section, are phenomena continuously suffered by these machines. One alternative to the previous methodology in this chapter is counting with a partial model, focused on the usual conditions and degradation levels that the engine would experiment before the next planned visit to the workshop. The turbofan engines used in commercial aviation work most of time inside well-known operational ranges. Similar consideration could be done for other kinds of gas turbine engines used in different applications, subject to local ambient conditions and applicable power regimes.

Such strategy, thought for usual operational conditions, could be implemented by means of a tensor (multidimensional array), large enough to contain all the potential representative operating points in which the engine will typically work. Obviously, one tensor is built by separated points in a multidimensional grid, so it does not contain all the information provided by the algebraic relationships between variables and components’ maps that the model in PROOSIS® manages. It will just contain a limited part of it. Nevertheless, that tensor could contain enough data to obtain a relatively good estimation, for engineering and maintenance purposes, of the H&Q parameters evaluated before in this study, becoming a functional version of the whole model, and replacing it in the practice. A “good” estimation would certainly mean counting with a sufficient degree of accuracy regarding the evaluation of the status of the engine, enough to capture small changes in the H&Q parameters during their continuous deterioration (typically, one order of magnitude below their usual values, as a minimum), and without incurring in an excessive computational cost from CPU time perspective.

It would be just a matter of populating enough densely the array with representative data (which would be supplied by PROOSIS®). On the other hand, managing big dense tensors could imply long computational times so, as it happened in the previous techniques, the commitment between accuracy and computational time would have to be previously established. The so-called “curse of dimensionality” must be seriously considered when obtaining and managing a tensor like the one required for the turbofan problem, because the addition of one additional point to the grid created for one of the parameters, aiming to increase the degree of detail obtained from it, would imply the incorporation of millions of new values that must be processed and stored. And tensor’s size should be kept as reduced as possible.
Initially, the use of a partial version of the full performance SW constitutes already a Reduced Order Model (ROM) of the real system, however some additional compression techniques could be considered to ease, as much as possible, the inverse problem solving to the CPU. The smaller the number of operations to obtain a certain result from the tensor, the more efficient the method would be. The target in this study was simplifying the tensor without drastic accuracy reductions.

The main reasons to use tensors instead of a full model are varied and they deserve some comment to understand the applicability of this methodology:

- First, the use of a powerful SW to run direct performance problems implies certain hardware requirements, normally demanding from computational standpoint, that must be met. The commercial version of PROOSIS® runs on computers that count, at least, with Microsoft® Windows™ 64 bit, with 4GB RAM memory, 3GB of free disk space, and the appropriate C++ compilers. This circumstance could imply problems when installing the SW in the onboard computers in an aircraft, depending on its age. Regarding aeroderivative gas turbines, used for marine and industrial applications, there are not typically so many restrictions or requisites to meet (lower criticality level, but also depending on the application). Additionally, each computer running the SW must count with a valid license, which means extra costs to be added, probably per engine, in an aircraft. The accumulated licensing costs in a complete aircraft fleet could become a relevant financial burden. In this sense, a medium size airline like Iberia (including Air Nostrum and Iberia Express) counts with 278 engines powering the 139 aircraft in active service [140]. Usual licensing costs for industrial applications could mean for this airline final prices above one million of euros. On the contrary, the management of a tensor, as a mathematical object, could be performed by free access SW, needing of far less requisites to work in most of computers.

- Second, there are several available analytic techniques that, depending on the structure and content of the tensor, compress the information contained on it by keeping the most relevant part of it, and discarding the rest if it is not really providing a remarkable value to the results, making its management more affordable from a computational perspective, and leading potentially to shorter CPU times. These techniques play with the inherent scale factor associated to the tensors (particularly, when they are large and count with many dimensions), and try to get advantage of potential symmetries, anti-symmetries, or some other useful features that could be present in the tensor. Obviously, such kind of compressions, when possible, would imply a certain loss of accuracy that must be analyzed to avoid excessive numerical errors or losses of effectiveness. It is worthy, at least, to explore such options, even if they could not be finally used to obtain some remarkable profit.

- Finally, the tensors, as well as the required optimization that may be applicable to facilitate its use, could be calculated and pre-processed before its implementation for the diagnostics, prognostics, and performance estimation of the engine, so there is no need to dedicate long computational times during the operation of the machine to produce a potentially dense multidimensional array. It could be calculated, for instance, before the first flight, and uploaded meanwhile the engine is being installed in the aircraft.
Alternatively, in an aeroderivative gas turbine, the tensor could be obtained during the installation and commissioning phase in a power plant or a ship. Then, that same tensor could be theoretically used, continuously, during thousands of cycles and operational hours, until the next visit to the workshop. It is expected that some parameters in the engine would change after the required maintenance (e.g., minor cross section areas change in the gas path), and some correction to the model (and, therefore, to the tensor) would be needed. Ideally, just one tensor could be valid for all the new units leaving a factory, if the mechanical tolerances and manufacturing processes allow for it (i.e., if the engines can be considered identical). And that adds a great value regarding the potential implementation of this technique in a full fleet of engines, as the associated SW costs could get greatly reduced.

The distribution of information in the tensor used in this study for the turbofan problem was organized as it was indicated in the previous chapters, so the array counted with twelve dimensions. Flight conditions were chosen (altitude of 10,668 m, Mach number 0.8, and ISA conditions) to remain constant for the sake of simplicity in this study but, obviously, those conditions may slightly vary during the cruise phases of a flight, and that circumstance deserves some comment. 3 additional dimensions, with 2 or 3 points in the grid each, could be added to the tensor to reflect more accurately the variations in those flight conditions. That would imply certainly a bigger tensor. However, depending on the duration of the flight, working with weather forecasts, and organizing the flights in advance, the size of the resultant tensor would not be so greatly increased. Also, if the data storage is not a problem, several tensors could be calculated to cover the different flight conditions, not incurring in long computational times by managing just one big tensor, but smaller ones instead for each operational scenario (selected with a previous “if” command, for instance). Nevertheless, counting with enough historical data of the completed flights in a particular route, both flight conditions and regime (i.e., TIT) could be selected to optimize the layout of the grid (see Figure 86). Airlines try to operate always optimizing the fuel consumption by respecting routes, timetables, and schedules as much as possible, so there are not so many sources of variation during a commercial regular flight. That means there will be multiple options to optimize the layout of the grid, aiming to avoid time penalties. Similar considerations could be done for the aeroderivative versions used in industrial or marine applications (i.e., considering constant altitude and null speed at the inlet).

![Figure 86: Schematic of the application of a ROM to determine the progressive deterioration of the engines mounted in an aircraft covering a standard commercial route (short-medium haul).](image-url)
Once the applicable flight conditions were properly reflected in the tensor to be used in the study, the distribution of the grid was setup. The first dimension in the grid was assigned to the sensor selection (instrumentation), the second one was for the range of applicable TIT, and then the ten degradation vector components were considered after, making a total of 12 dimensions, following the next layout:

- **Sensor readings**: The 10 initial components of the instrumentation vector were included. Each new component (like a new $P_{45}$ probe could be) must be evaluated carefully, given the scale factor associated. Adding a new sensor to the tensor would have meant a 9.1% increase in the array's size.

- **TIT**: A usual range between 1,300 K and 1,650 K for cruise phases was covered, with a grid of 18 points. This grid could be refined if most of the time the engine regimes are fixed around dominant TIT values. Again, the cruise phases are designed to keep the engine inside well-defined ranges.

- **Regular degradation components (not affected by correlations)**: Degradation components 1-6 and 8 were considered with a grid of three points (0%, 1%, and 2%, respectively). Such election of degradation range and grid layout should suffice to provide enough accuracy in the results and to maintain the same tensor during extended time periods, considering a new and clean engine as a baseline (corresponding with a 0% degradation level). It is not expected that engine components would reach higher deteriorations levels, beyond a 2%, before thousands of flights and operational hours.

- **Rest of degradation components (affected by correlation)**: The troublesome components, 7th, 9th, and 10th, found for the instrumentation system installed in the CFM56-5A, were initially considered with a finer grid of 5 points (0%, 0.5%, 1.0%, 1.5%, and 2.0%, respectively) given the lower accuracy typically found on them to try to improve the results on those specific components. This decision, obviously, implied a severe penalization in terms of data storage and CPU efficiency. The benefit of such decision was found to be limited, or almost imperceptible (correlation was associated to the sensor distribution of the engine, so the drop in numerical accuracy would always happen, with three, five or more points in the grid), for the improvement of the level of accuracy in the solution for those components. So, the next action was to use only 3-point grids instead, for all the degradation components, with the distribution already mentioned. Just this change meant a decrease in the size of the tensor of more than a 78% (a relevant liberation of data storage, from more than forty-nine million values to less than eleven million). This is an example of the impact that the “curse of dimensionality” implies (exponential dependence of the number of entries), and how different decisions could improve the resolution of the inverse problem.

According with the final grid distribution, the number of times the direct problem might be executed to populate all the points in the 12D grid defined by the previously defined tensor was $18 \cdot 3^{10}$ (i.e., 1,062,882 times), consuming 20.67 CPU hours in a i7 desktop PC. This calculation could be considered as an expensive computational process but, once it is done, it could be used for thousands of cycles and operational hours. Also, if the maps and the rest of model settings for the engines in the fleet (or a portion of that fleet) are supposed to be equivalent, the same tensor could be used for all those engines installed in different aircraft.
Something that deserves some additional comment is the data storage needed in the computer given the size of the tensor itself. As per the previous data, more than 10 million values (10,628,820) were stored initially in the memory of the computer running the inverse problems for the reference engine. That made around 80 MB of internal memory storage that had to be available, just for the tensor. Again, depending on the potential optimization applied to the tensor layout, such storage may vary.

Beyond the hardware requirements to allocate such number of entries in the grid, the SW must find the right values for the instrumentation vector and, obviously, the higher the number of points in the 12D grid, the longer the time needed to get the appropriate values during the calculations to be done (e.g., during the obtention of the Jacobian matrices for the Newton method). Some compression could be useful to speed up the calculations if the loss of accuracy was not excessive. However, the tensor generated did not count, a priori, with any useful feature such as symmetries (also called super-symmetries in Tensor Algebra, meaning being invariant after a permutation of tensor indexes) or accumulations of null values in certain regions of the array. On the contrary, the tensor was completely full of values from which a solution to the inverse problem had to be obtained, with enough accuracy to perceive small changes in the H&Q parameters. This circumstance certainly limited the effectiveness of any compression that might be applied. Initially, all the points in the grid were equally relevant for the problem, and the accuracy required would imply counting with enough data to approach to the right solution.

In this sense, it is necessary to clarify that the target is not identifying patterns or structures in the tensor. The structure of the information included in the grid, the nature of the information contained in the tensor, is already known as it was given by the full model. The intention behind the use of tensors is trying to achieve similar results to the ones already obtained with the full model, in terms of numerical accuracy, and in short computational times, if possible, but instead of the model. The qualitative results chased in some research works found in the literature would not be directly applicable for this specific problem. Here, the quantitative results (measurable in decimals), together with the CPU times, are crucial.

Another relevant question when dealing with tensors was the way to obtain the required intermediate values (navigation inside the tensor). Tensors are objects destined to allocate values in a structured multidimensional grid, but the calculations to be done will require the use of values that would be in an intermediate position inside the grid, given the accuracies that have been managed so far, otherwise the only degradation values that would be delivered as outcome would be 0%, 1%, and 2%, with no higher accuracy possible. Trying to perform a multi-dimensional linear interpolation, the same way it would be done for one or two dimensions, without some rational approach regarding the dimensions involved, would lead to unaffordable numerical errors. Some systematic procedure, allowing to select the right position in each dimension with enough accuracy, must be followed. If each parameter used for the direct problem (i.e., degradations and TIT) is vectorized, then it would be required counting with a set of matrices (i.e., bases) to serve as a bridge between the vectorized parameters and the tensor.

All these questions regarding tensor management and optimization, pointed to the need of a theoretical framework to deal with tensors, as mathematical objects, aiming to integrate them with the methodology stablished before for the resolution of the inverse problem.
And such required background was found in the theory associated with the Tensor Algebra and, more specifically, with the Higher Order Single Value Decomposition, or HOSVD (from the profuse literature on the topic, probably the best starting point to work with tensors could be found in De Lathauwer et al., 2000, [66], Kolda et al., 2009, [159], and Cichocki et al., 2015, [47]).

Given the complicated (and bulky) nature of the operations with multidimensional arrays, there are available several software tools to develop the required computations, including decompositions and optimizations. TENSORLAB® [292] was the chosen one for the calculations that were done in this study. It provided useful features for visualization (see Figure 87) and optimization that were used systematically during the application of HOSVD to the turbofan problem.

![Figure 87](image)

Figure 87: Several examples of charts obtained with TENSORLAB®, showing the variations in several instrumentation components (TST, T2ST, Wf, and P13T, respectively) with TIT, or different degradation components.

The first issue that was addressed during this study was finding a valid way to obtain the required data from the initial tensor generated with PROOSIS®. To do so, it was necessary to point properly to the required value inside the grid of the tensor. Selecting a TIT from the range covered was easy, when dealing only with such variable (just by adequately averaging in between points of the grid established for the TIT). Similar approach was followed with the degradations (see Figure 88). So, it was found easy to calculate vectors with the exact coordinates for the different parameters involved in the tensor (this operation is required whenever a value is needed from the tensor). Then, those vectors (12 in total) had to be used with the tensor in such a way they would point to the desired value inside the grid.
However, a set of vectors like the ones with coordinates cannot be used directly with the tensor, there must be a set of matrices (i.e., bases of the required vectorial subspaces per dimension) to convert the 1D information in the vectors to be used in the 12D mathematical object containing the model.

1 - Vectorization of the parameter (degradation):

![Diagram of vectorization](image)

2 - Use of the adequate matrix (base per mode) prior accessing to the tensor:

![Diagram of matrix-vector product](image)

Those matrices (indicated as \( \mathbf{U}^{(i)} \) in the previous figure) can be obtained with the HOSVD of the tensor, which is a direct generalization of the SVD technique explained before in this chapter. In the literature, the multilinear SVD (MLSVD, generalization of the SVD for matrices) is known as Tucker Decomposition since the 1960s (or TD, see [285] for reference), and as HOSVD since 2000. However, this concept has experienced some evolution in the last years and it is now used in a more generic sense. Like in the SVD, where the matrices obtained were compound by orthonormal bases for the column and row space (left and right modes), respectively, the HOSVD computes \( n \) orthonormal bases, \( \mathbf{U}^{(n)} \), for the \( n \) different subspaces of mode-\( n \) vectors. These orthonormal bases have a similar interpretation as in the SVD case for matrices. Basically, the HOSVD considers a tensor, \( \mathbf{T} \), as \( n \) sets of mode-\( n \) vectors and computes the matrix SVD of these sets, as it will be explained later. With the matrices containing the bases associated to the different subspaces, it is possible to introduce the right coordinates into the tensor, pointing adequately to obtain the right values of the instrumentation vector.

The multilinear rank (n-rank) of the tensor would correspond to the \( n \)–tuple consisting of the dimensions of the different subspaces established by the HOSVD. The information provided by different singular values obtained with the HOSVD, as a function of the multilinear rank, is of great relevance for the compression of the information contained in the tensor, as it will be exposed in this section.
If any singular value becomes small (or null) for some values of its associated n-rank component, then that circumstance can be used as a criterium to bound the associated grid to that mode, avoiding bigger grids than necessary. But before using this criterion for the turbofan problem tensor, it is necessary to explain some more concepts of the HOSVD.

Within the Tensor Algebra and HOSVD framework, several useful operations are defined. It is possible to calculate the usual scalar product with tensors (represented by the \((,\) operator and applicable when dimensions match) and, consequently, it could be determined when two tensors are orthogonal (circumstance occurring when their scalar product is null), as well as the norm of a tensor, which will be the usual Frobenius norm applied to the components of the tensor (direct generalization of the same norm for a matrix seen before).

Very relevant for the HOSVD is the product of a tensor by a matrix, as different decompositions or tensors are provided as sets composed by a Core Tensor, \(\vec{S}\), and mode-n vectors inside the respective n matrices, \(U^{(n)}\), that will have to be multiplied that way (the associated vectors in the matrices of the subspaces, \(\vec{U}_{i}^{(n)}\), will be the ith n-mode singular vectors). In fact, the HOSVD of every tensor can be always provided in the form of n-mode products, like it is indicated in Figure 89.

The n-mode product of a tensor \(\vec{S}\), with a size \((I_1 \times I_2 \times \ldots \times I_N)\), by a matrix \(U\), of size \((J_n \times I_n)\), denoted formally as \(\vec{S} x_n U\), is a new tensor of size \((I_1 \times I_2 \times \ldots \times I_{n-1} \times J_n \times I_{n+1} \times \ldots \times I_N)\), with the following components:

\[
\vec{S} x_n U = \sum_{i_n} a_{i_1 \ldots i_{n-1} i_n i_{n+1} \ldots i_N} \cdot u_{j_n i_n}
\]  

(4.63)

Such product counts with several interesting (and crucial) properties. For instance, it does not matter the order in which two matrices, \(U\) and \(V\), of sizes \((J_n \times I_n)\) and \((J_m \times I_m)\), respectively, dimensionally compatible with one tensor \(\vec{S}\), with a size \((I_1 \times I_2 \times \ldots \times I_N)\), multiply consecutively to the tensor, because the outcome will be the same. This property, as indicated in Eq. (4.64) can be generalized to whatever number of n-products, but always respecting the different dimensions involved:

\[
\vec{S} x_n U x_m V = \vec{S} x_m V x_n U
\]  

(4.64)

Figure 89: Decomposition applied to a 3D tensor, \(\vec{T}\), in the form of 3-mode products of the Core Tensor, \(\vec{S}\), and matrices containing the respective n-modes, \(U^{(n)}\), for \(n = 1, 2, 3\). Indexes are indicative, dimensions must match.
Formally, the multi-linear SVD (MLSVD) of a nD generic tensor $\mathbf{T}$, with a size $(I_1 \times I_2 \times \ldots \times I_n)$, like the one used to replace PROOSIS®, consists in the following operation (go to [66] for further details):

$$
\mathbf{T} = \mathbf{S} \mathbf{x}_1 \mathbf{U}^{(1)} \mathbf{x}_2 \mathbf{U}^{(2)} \ldots \mathbf{x}_n \mathbf{U}^{(n)}
$$

(4.65)

Or, alternatively, considering the individual components of the arrays:

$$
t_{i_1i_2\ldots i_n} = \sum_{j_1}^{I_1} \sum_{j_2}^{I_2} \ldots \sum_{j_n}^{I_n} s_{j_1j_2\ldots j_n} u_{i_1j_1}^{(1)} u_{i_2j_2}^{(2)} \ldots u_{i_nj_n}^{(n)}
$$

(4.66)

Expression where the different indexes $i_1, i_2, \ldots, i_n$ vary from 1 to $I_n$, and $u_{i_1j_1}^{(1)} u_{i_2j_2}^{(2)} \ldots u_{i_nj_n}^{(n)}$ are the entries of orthogonal matrices $(\mathbf{U}^{(i)})$, containing the bases of the different associated subspaces, multiplying with tensor-matrix products (or n-products) to the so-called Core Tensor, $\mathbf{S}$, which is a tensor with a size that will be initially $(I_1 \times I_2 \times \ldots \times I_n)$, like the original tensor $\mathbf{T}$, counting with the important property of “all-orthogonality” for the “n” subspaces of mode-n vectors determined by the grids in each dimension or, in other words:

$$
\sum_{i_1i_2\ldots i_n} s_{i_1i_2\ldots i_n} a_{j_1j_2\ldots j_n} = \sum_{j_1}^{I_1} \sum_{j_2}^{I_2} \ldots \sum_{j_n}^{I_n} s_{j_1j_2\ldots j_n} a_{i_1i_2\ldots i_n} = \ldots = \sum_{i_1i_2\ldots i_n} s_{i_1i_2\ldots i_n} a_{j_1j_2\ldots j_n} \neq 0 \quad (4.67)
$$

The previous expression in Eq. (4.67) is equivalent to affirm that every two sub-tensors (i.e., $\mathbf{S}_{i_n=\alpha}$ and $\mathbf{S}_{i_n=\beta}$) of the given tensor, $\mathbf{S}$, obtained by two different values of any of the tensor indexes (e.g., $i_n$ in this case) are orthogonal, so their scalar product will be null. This condition would be accomplished whenever $\alpha \neq \beta$. The different matrices obtained in the MLSVD (this is $\mathbf{U}^{(1)}, \mathbf{U}^{(2)}, \ldots, \mathbf{U}^{(n)}$) are mutually orthogonal because of the “all-orthogonality” property in the Core Tensor, and unique (once given $\mathbf{S}$), analogously to the matrix case. Keeping in mind that analogy, it is also possible to obtain the Core Tensor as it is indicated by Eq. (4.68):

$$
\mathbf{S} = \mathbf{T} x_1 \mathbf{U}^{(1)T} x_2 \mathbf{U}^{(2)T} x_n \mathbf{U}^{(n)T}
$$

(4.68)

In the MLSVD, the Core Tensor would be the multidimensional equivalent of the diagonal matrix in the SVD for matrices, but without symmetries or diagonalities whatsoever. The Core Tensor of the decomposition will be typically a dense tensor, and this circumstance was verified in fact for the turbofan problem. Additionally, the Core Tensor will contain different (N-1)th-order sub-tensors that will be compliant with the expression in Eq. (4.69):

$$
\| \mathbf{S}_{i_n=1} \|_F \geq \| \mathbf{S}_{i_n=2} \|_F \geq \ldots \geq \| \mathbf{S}_{i_n=I_n} \|_F \geq 0
$$

(4.69)

This order can always be stablished for all the possible values of $n$. The operator $\| \cdot \|_F$ represents here the usual Frobenius norm of the tensor. Those norms in Eq. (4.69) represent the previously mentioned n-mode singular values of the original tensor, $\mathbf{T}$. Depending on their values, the influence of the associated parameter will be higher or lower (no influence if the value is null). It can be proved that the MLSVD of every tensor is always possible, but it is not unique.
And here is where the more generic approach of the HOSVD appears, introducing the concept of optimization. Different sizes of Core Tensors (and their associated matrices) would lead to different decompositions, with higher or lower n-rank (a Core Tensor will count with a n-rank equal or lower than the one of the original tensor). The basic MLSVD would lead to a Core Tensor of the same size than the original one, but smaller Core Tensors are also possible by truncation and optimization (HOSVD). The MLSVD of a 3D tensor, in which the Core Tensor counts with a smaller size, is represented in Figure 90.

![Figure 90: MLSVD applied to a 3-D tensor, \( \overline{T} \), with a smaller Core Tensor. Indexes are indicative.](image)

This kind of decomposition opens the door to different useful applications involving the management of big sets of values organized in tensors, among them, the dimensionality reduction of multidimensional problems (for a given compressed size of the Core Tensor), which is the one of interest for this study. However, it is still to be clarified if such application of the HOSVD will be useful for the turbofan problem because a sufficient accuracy should be maintained.

Initially, it is not clear what level of accuracy could be obtained from a ROM in which an important data compression had been applied. In several applications, like image processing, face recognition, certain kinds of medical diagnostics, etc., in which the desired outcome of the analysis would be of a qualitative nature, or in which an aggregate or averaged result could be valid, there might be a motivation to compress the information in the original tensor to work with the most reduced Core Tensor possible, speeding up this way the resolution of the problem. In Cichoki et al. [47], it is indicated how different applications including classification, feature extraction, and subspace-based harmonic retrieval, are well covered by the HOSVD. The treatment of certain types of electronic signals can be addressed by this technique as well. There are novel techniques based networks of tensors (hierarchically interconnected by graphs), thought to minimize the effect of the “curse of dimensionality” [223], that certainly could be evaluated for the turbofan problem in the future, to try to speed up the calculations while keeping a similar accuracy level (in case the data in the problem could be decoupled in some way before using the adapted Newton to solve the inverse problem, for instance).
Nevertheless, the number of dimensions in the turbofan problem is high, the data in the tensor do not follow any a priori identifiable structure, the variables that are involved in the turbofan problem are highly interrelated, the direct performance problem in a turbofan manages parameters that are coupled, and enough degree of accuracy to detect small degradations is a must for the results. These circumstances will impact necessarily the solving speed, as it will be shown later in the results.

It is particularly interesting the concept of "matrix unfolding" (or matrization) of a tensor, which is of great help for the visualization of its internal structure, at least in simple cases, as well as for the explanation of the multilinear SVD (MLSVD), and HOSVD with smaller Core Tensors. In fact, the way used by De Lathauwer et al. in [66] (2000) to compute the MLSVD, was by applying systematically the regular SVD to the matrices obtained from the matrization (or unfolding) of a given tensor.

The i-th matrix unfolding of a generic real tensor, \( \mathbf{T} \), of size \((l_1 \times l_2 \times \ldots \times l_n)\) is given by a matrix, \( \mathbf{T}_{(ij)} \), of size \( l_i \times \left(l_{i+1}l_{i+2} \ldots l_{i+j-1}l_{i+j}l_{i+j+1} \ldots l_{i-1}l_{i-2} \ldots l_1 \right) \), in which the element \( t_{i_1i_2\ldots i_n} \) is located at the row \( i_j \in \{1, \ldots, l_j\} \), and at the column indicated by Eq. (4.70):

\[
(i_j + 1 - 1)l_{j+2}l_{j+3} \ldots l_{i-1}l_1 l_{i+1} \ldots l_{i+j-1} + (i_j + 2 - 1)l_{j+3} \ldots l_n l_{i+1} l_{i+j-1} + \cdots + (i_n - 1)l_1 \ldots l_{i-1} + (i_1 - 1)l_2 l_3 \ldots l_{i-1} + (i_2 - 1)l_3 \ldots l_{i-1} + \cdots + i_{i-1} \tag{4.70}
\]

The formal definition of this operation of unfolding could result complicated initially. However, it is relatively intuitive. In that operation the elements of the tensor are organized inside arrays of less dimensions than the original tensor, one per each element of the index selected for the unfolding. For instance, in a 3D tensor, there will be three possible indexes for the unfolding. Once one of the indexes is selected, the others will define as many column vectors as elements are covered by the selected indexes, until all the elements of the original tensor are allocated. The full explanation gets bulkier when the number of dimensions is higher than three, but the concept behind is a generalization. The order of the subindexes in the expression given by Eq. (4.70) is essential to complete the operations that will be made after a tensor matrization (in particular, the SVD of \( \mathbf{T}_{(ij)} \)).

Even when the MLSVD technique is considered as a generalization of the SVD for matrices, there are some differences. For instance, regarding the definition of the rank of a tensor, or when rank-related properties are involved. Working with tensors, it is managed the concept of column rank and row rank for column and row vectors of nth-order tensor. The column and row vectors of such tensor are defined as the \( l_n \)-dimensional vectors obtained from the tensor by varying the index \( i_n \), keeping the others fixed. The n-mode vectors of a tensor are the column vectors of the matrix unfolding of the tensor. Then, the n-rank of a nth-order tensor \( \mathbf{T} \) (or \( R_n = \text{rank}_n(\mathbf{T}) \)), would be the dimension of the vector space spanned by the n-mode vectors \( (R_n = \text{rank}(\mathbf{T}_{(in)})) \), and there will be one n-rank per dimension of the tensor. The different n-ranks of a tensor are not necessarily the same, and this circumstance is aligned with the strategy of looking for compression of data. If the n-rank in one dimension is lower than the number of vectors in its associated \( \mathbf{U}(i) \) matrix, then there is a clear chance to reduce the number of elements in the tensor.

A nth-order tensor, \( \mathbf{T} \), has rank 1 if it equals the outer product of n vectors like in the next expression (remember that the outer product of two vectors, \( \mathbf{u} \) and \( \mathbf{v} \), delivers a matrix, \( (u \otimes v)_{ij} = u_i v_j \)), for all the values of the indexes:
The rank of an arbitrary nth-order tensor (or \(R = \text{rank}(\mathbf{T})\)) would be defined as the minimum number of rank-1 tensors that yield \(\mathbf{T}\) in a linear combination and it can be verified that the rank of a tensor does not necessarily have to be equal to an n-rank, even if all the n-ranks are the same. Nevertheless, the most useful rank definition for the analysis done in this work was the n-rank.

Continuing with the HOSVD and the compressibility of a given tensor, it was introduced before the concept of MLSVD of a tensor as a direct algebraic generalization of the matrix SVD, operation that can be computed in a similar way once the tensor has been unfolded in matrices.

Initially, by applying the MLSVD to a generic tensor, the associated Core Tensor would count with the same dimension than the original one, as it happened theoretically in the SVD applied to matrices (matrices \(\mathbf{A}\) and \(\mathbf{S}\) had initially the same size, independently of the number of null terms in \(\mathbf{S}\)). The reasoning behind the HOSVD, the way it is known today, is finding the components for the Core Tensor that best capture the variation in the n-modes, independently of the other modes.

As it happened with the developed expression of the SVD for matrices, the MLSVD could be truncated by discarding the terms associated to the lower singular values, leading thus to a not-optimal tensor in terms of giving the best fit (measured by the norm of the difference between the truncated decomposition and the original tensor). In this sense, such truncation could be considered as a strategy to reduce CPU times, and it would be also a good starting point for some iterative algorithms. In this case, the n-mode rank of the decomposition will be smaller or equal than the n-mode rank of the original tensor.

The truncated MLSVD will lead to a lower rank solution which can be optimized to try to obtain the best truncated solution possible given a particular size of the Core Tensor. This optimization is not a pure algebraic operation as it involves the obtention of one minimizer. This search for an optimum Core Tensor is what would set the difference between a pure MLSVD with truncation (pure algebraic operations), and the more generic HOSVD, as it is known today, in which an optimization takes place to obtain the best truncated decomposition possible. The potential lower rank solutions that would be obtained this way, with feasible smaller Core Tensors, are the ones that will be explored, to evaluate if, for any lower rank configuration, the commitment between CPU times and accuracy could be met.

Given a generic tensor, \(\mathbf{T}\), which is a tensor with a size \((I_1 \times I_2 \times \ldots \times I_n)\), the HOSVD (and not the MLSVD with truncation explained before) would be the solution to the following optimization problem \([159]\):

\[
\mathbf{\mathbf{T}} = \mathbf{\mathbf{S}} \times_1 \mathbf{V}^{(1)} \times_2 \mathbf{V}^{(2)} \ldots \times_n \mathbf{V}^{(n)}
\]

Subject to the Core Tensor of given size \((J_1 \times J_2 \times \ldots \times J_n)\), equal or smaller than the original’s tensor size, and subject to a new set of matrices \(\mathbf{V}^{(n)}\) of respective size \((I_n \times J_n)\). Again, the norm is the Frobenius norm applied to tensors.

Here, the Core Tensor is compliant with the "all-orthogonality", which means that the associated set of matrices are mutually orthogonal column wise.
Then, by means of the usual operations regarding scalar product and norms with tensors, and keeping in mind that the set of matrices $V^{(n)}$ are orthogonal:

$$
\|\mathbf{T} - [\tilde{S} x_1 V^{(1)} x_2 V^{(2)} \ldots x_n V^{(n)}]\|_F^2 = \|\mathbf{T}\|_F^2 - 2 \cdot (\tilde{S} x_1 V^{(1)}\mathbf{T} x_2 V^{(2)} \ldots x_n V^{(n)}\mathbf{T} \tilde{S}) + \|\tilde{S}\|_F^2 \tag{4.73}
$$

Expression that can be elaborated and simplified to lead to the following one, which contains a constant term given by the norm of the original generic tensor $\mathbf{T}$:

$$
\|\mathbf{T} - [\tilde{S} x_1 V^{(1)} x_2 V^{(2)} \ldots x_n V^{(n)}]\|_F^2 = \|\mathbf{T}\|_F^2 - 2 \cdot (\tilde{S} \tilde{S}) + \|\tilde{S}\|_F^2 = \|\mathbf{T}\|_F^2 - \|\tilde{S}\|_F^2 \tag{4.74}
$$

And that means that the original minimization problem of the error between the original tensor and the decomposition, given by Eq. (4.72), can be converted into a set of maximization problems for each $V^{(i)}$ matrix, which can be expressed in the following way:

$$
\max_{V^{(i)}V^{(2)}\ldots V^{(n)}} \|\mathbf{T} x_1 V^{(1)^T} x_2 V^{(2)^T} \ldots x_n V^{(n)^T}\|_F \tag{4.75}
$$

These problems can be solved matrix wise, once the “matrix unfolding” of the original tensor $\mathbf{T}$ is done (leading to $n$ $T^{(i)}$ terms after deploying its components by one index), by applying regular SVD to the resultant matrices, truncating, and finding the optimum solution. The resultant matrix version of the previous maximization problem would be given by the next expression:

$$
\max_{V^{(1)}V^{(2)}\ldots V^{(n)}} \|V^{(i)^T}\mathbf{W}\|_F \tag{4.76}
$$

Where the term $\mathbf{W}$ is computed as follows:

$$
\mathbf{W} = T^{(i)} (V^{(n)} \otimes_K \ldots \otimes_K V^{(i+1)} \otimes_K V^{(i-1)} \ldots \otimes_K V^{(1)}) \tag{4.77}
$$

The operator $\otimes_K$ represents the usual Kronecker product between matrices and the term $T^{(i)}$ represents the mode-i matricization of the tensor $\mathbf{T}$ which arranges the mode-i fibers to be the columns of the resulting matrix. It must be indicated that a fiber is a set of components of the tensor defined by fixing every index but one. A matrix column is a mode-1 fiber, and a matrix row is a mode-2 fiber. Third-order tensors would have column, row, and tube fibers accordingly.

With the previous approach, the solution to the problem given by Eq. (4.76) could be determined using then the SVD. It is required that the vectors compounding the different $V^{(n)}$ matrices must be used as singular vectors of the resultant SVD matrix problems after the unfolding. That condition must be respected to verify the “all-orthogonality” of the Core Tensor. This method will converge to a solution where the OF given by Eq. (4.72) ceases to decrease, but it is not guaranteed its convergence to a global optimum or even to a stationary point (on the ill-posedness of the problem see De Silva et al., 2008, [68]). The details of the matrix approach to the optimization problem are explained in [66] and [159], so they are not fully included here for the sake of conciseness (just enough background will be provided to justify the next steps and results).
It is now evident, depending on the solution obtained to the previous optimization problem based on the assumed size of the Core Tensor, that the HOSVD of a generic tensor $\mathbf{T}$ is not unique (several different Core Tensor configurations have been evaluated for the turbofan problem in this study). This circumstance calls for choosing transformations that could simplify the problem reducing the size of $\mathbf{S}$ by optimizing a lower rank version of it. In this sense, the selection of smaller (truncated and optimized) Core Tensors should redound, initially, in less operations to be done by the computer and shorter processing times. It is expected that there will be a cost regarding numerical accuracy that will have to be determined. In this sense, it must be indicated, that the impact in terms of accuracy could end up leading to a higher number of iterations if the solution is not so easy to find for the techniques used so far (e.g., Newton) comparing with a non-truncated version.

It might happen that the acceleration obtained by the size reduction in the Core Tensor and its associated matrices could be compensated by a higher number of iterations until reaching a satisfactory solution because of the reduction in accuracy of the available data. Even more, some cases could fail to reach convergence if the reduction of accuracy is too drastic. The effectiveness of the new reduced versions of the problem must be analyzed, together with accuracies and CPU times, to make sure that no convergence issues occur in the practice (and finding out a potential solution if that situation finally was unavoidable).

The direct multilinear SVD performed in TENSORLAB® ("mlsvd" command) of a given generic tensor relies on the SVD of its matrix unfolding, which is sometimes expensive to obtain, particularly with large tensors. TENSORLAB® gives the option of performing lower rank approximations ("lmlra" command) to the decompositions given by the direct or truncated MLSVD, moreover given the fact that some truncated decompositions could be not-optimal. Those refined approximations can be obtained by the SW using different optional methods such as non-linear least squares ("lmlra_nls" command) or non-linear unconstrained optimizations ("lmlra_minf" command). The least squares option is the one used by default (and used in this study), but this is configurable in the SW.

Several studies have provided additional methods to solve the problem in Eq. (4.72), leading to different algorithms with varied convergence rates (more expensive, computationally speaking, when the Hessian of the problem needs to be obtained). There are also several methods to choose the most applicable size for the Core Tensor. The tools implemented in TENSORLAB® will be the ones used in this work to select the size of the Core Tensor and to obtain the associated decomposition.

In this sense, the chart in Figure 91 shows the outcome obtained with the command "mlrkest", which provides different Core Tensor sizes that can be chosen, their associated compression ratios (given by the ratio between the number of elements of the HOSVD decomposition chosen and the number of elements of the original tensor), and their relative error (calculated as the ratio between the Frobenius norm of the HOSVD decomposition chosen and the Frobenius norm of the original tensor). That chart is so-called "L-curve" referring to the shape of the red line plotted in with the error associated to truncated MLSVD solutions of the original tensor below the upper bounded error considered for a given compression ratio. The corner of this L-curve is often a good estimate of the optimal trade-off between accuracy and compression in certain problems [292], but not initially for the turbofan problem, as it will be detailed in the next chapter.
Another way to choose the right size for the Core Tensor would be, as mentioned before, by analyzing the value of the different multilinear singular values associated to the dimensions of the tensor under evaluation ($\mathbf{T}$, in the case of the turbofan problem, counts with 12, see Figure 92). If one of the singular values had gone to zero for any grid point of the charts (abscissa axis), it would have made sense to choose a mode-$n$ rank equal or below that point (i.e., keeping only the points of the grid where the singular value is not null), as no remarkable influence of the associated parameter would have been expected beyond that point in the results. Although there are great differences, of several orders of magnitude, in between the values in the charts (apparently pointing to an ideal Core Tensor size of $[3 4 2 2 2 2 2 2 2 2 2 2]$ as indicated in Figure 91), the losses of accuracy associated to lower rank sizes were found unaffordable for the target of the study. This made more complicated the selection of the Core Tensor, and less convenient the use of the tensor and HOSVD comparing with some other techniques previously analyzed.

In this situation, there was not a clear decision regarding which size could be better for the Core Tensor (apart of respecting the original size of the tensor), as whatever compression option that had been chosen would have eventually led to a potentially undesirable loss of information (and accuracy). It was required to evaluate the different possibilities indicated in the chart showed in Figure 91 to get some more insight on the potential validity of the technique, as it will be shown in the next chapter with a Core Tensor size study.
Figure 92: Multilinear singular values of the original tensor, provided by TENSORLAB®. For no grid size (abscissa axis), any of the charts goes to zero, and this makes complicated the selection of the Core Tensor size.

However, the size and layout of the tensor, together with the highly coupled and interrelated nature of the information contained on it, and the demanding targets associated to the EHM systems in which the ROM methodology would be implemented, make difficult an efficient compression for any of the 12 dimensions considered in this problem. The next chapter will show how the potential benefits associated to the data optimization are counteracted by its inherent loss of accuracy. The use of ROM and HOSVD would be limited to non-real time applications, but that circumstance could be still valid for an affordable strategy focused on mid-term and long-term engine monitoring, focused on slow degradations (prognostics) and not on sudden changes (diagnostics).
4.12. Conclusions

In this chapter of the thesis, a methodology has been proposed for the efficient health condition evaluation in turbofan engines (in fact, for any kind of gas turbine engine). For all the analyzed cases, results will show consistency with the applicable theory in the next chapter. The methodology provided additional tools allowing to:

- Calculate not only the health condition of a gas turbine engine, by obtaining the values of the degradations in the different main modules of the engine, but also the $T_{4t}$, at which the sensors' reading were taken.
- Validate the sensors in the engine's instrumentation (which configuration used as reference came from a real engine mounted in a commercial aircraft), providing a criterion to decide if a particular set of sensors was acceptable regarding the diagnosis of issues in the system. Just by analyzing the CN of the resultant Jacobian matrices, finding out if the inverse problem is ill-conditioned as it was explained in this chapter. It will be possible to evaluate the set of sensors under consideration, and how it could be improved, if needed, as it will be explained in the next chapter.
- Count with different options to solve the initial problem, with inherent benefits and drawbacks, that will be commented in the next chapter.

In particular, the accurate calculation of the $T_{4t}$ is something that deserves attention as it is not economical to measure this parameter in commercial gas turbine engines (for civil applications) with the current technology. In the military world the TIT is measured (i.e., by pyrometers), but requirements on operational hours and cycles are different. No TC probe tolerates extremely high temperatures, $\sim 1,400$ K, right before the first stage nozzle vanes at the HPT inlet. Typically, $T_{4t}$ is estimated in civil engines via correlations using readings from some other sensor, such as $T_{45t}$, installed downstream in the gas path. That estimation involves relying in the accuracy of the correlation (which is doubtful as the engine degrades), and depending on the condition of another specific sensor, to determine the value of $T_{4t}$.

Fortunately, the methodology explained in this chapter provided results based on the readings from a complete set of sensors, which will be more reliable and aligned with the real condition of the whole engine. But not only that, because the $T_{4t}$ obtained in the previous sections do also take into consideration the degradations in the different main modules of the engine, factor that determines dramatically the final value of the TIT needed to reach a certain performance level.

The values of the engine's degradations, and the associated $T_{4t}$, were calculated using two types of tools:

a) Minimizing the norm of the difference between actual sensors' readings and the equivalent values obtained from the engine model (PROOSIS®). With the standard set of sensors, initially considered, readings at two different values of $T_{4t}$ were needed, given the lack of valid diagnosis information supplied, as it was discussed in previous sections. Only after including a $P_{45t}$ probe to the engine's instrumentation enough information was retrieved with only one value of $T_{4t}$. Results obtained suggested the existence of one unique minimum under low degradations, and that uniqueness invited to use an adapted Newton method, able to work with constraints, as required.
b) The adapted Newton method resulted to be very robust, and remarkably faster and more accurate than the predecessors, based on the SQP and GA.

The cumulative nature of the degradations, and the slow deterioration of the turbomachinery under normal operation, led to the use of two samples, with different TITs, to try to improve the CN. In the next chapter it will be shown how both methods systematically provided worse results for some of the degradations (the ones relative to HPT and LPT). The deficient amount of information provided to the inverse problem by the initial standard set of sensors led to this situation. The detected correlations were spurious and were corrected by adding (not replacing one by one) a convenient new sensor. The improved set of sensors with $P_{45t}$ made the adapted Newton method work faster, more efficiently, and more accurately.

Finally, once the previous methodology based on the use of the full model (given by PROOSIS®) was fully established, the use of ROMs, based on the application of the HOSVD to a given tensor with the portion of information from the full model relative to the flight conditions (and regime) expected during the cruise phase of a flight, made possible the consideration of an alternative to such methodology. The commercial routes are well-defined, and airlines try to optimize the operation of the aircraft to minimize costs. Similar consideration could be made for engines used in industrial or marine applications. This means that statistically representative parameters’ ranges could be inferred to produce a tensor containing enough information to reproduce the outcome from the full model, but just for the cruise phase (by now), which is the one the aircraft will fly more often.

Such approach counts with clear benefits, as the computations with tensors can be made in many existent free-access software languages, and just one tensor could be used during long periods of time in every engine that was compatible with the data contained in it (e.g., it could be calculated once for the new engines delivered from the factory, if mechanical and manufacturing tolerances allowed for it). On the other hand, the “curse of dimensionality” associated with the tensors implies severe CPU time penalties every time one additional parameter must be considered, or when higher accuracy levels are required. There are techniques to try to compress the information contained in the tensor, based on the truncation of the original tensor and the optimization of the resultant decomposition to minimize the relative error, aiming to reduce CPU times by minimizing the number of operations that the CPU must process. However, the commitment between accuracy and speed must be respected, otherwise the results could not reflect the reality, or could be obtained too late for the purpose of the engine diagnostics and prognostics. It will be always required, within this methodology, to verify that small degradations (meaning, at least, one order of magnitude lower than the full range) are effectively and consistently detected. On the other hand, excessively long computational times could be a serious issue if the information arrives too late. Evaluating the slow deterioration of one engine along thousands of cycles and operational hours is not as critical as the diagnostic of certain sudden events. So, depending on the CPU times obtained, a valid strategy could be established, deciding which methodology is more suitable for different operational scenarios and EHM applications.

The next chapter will certainly clarify some of these points with a set of representative results obtained from the repetitive use of each technique. That information will be essential to decide which is the best technique, or how those techniques must be combined, for the problem under consideration.
5. - RESULTS

The main results obtained from the methodology described in the previous chapter, will be presented now. The order followed in the next exposition will be correlative with the sections in Chapter 4, to make easier the evaluation for the reader:

- First the most remarkable results obtained with GAs will be commented.
- Next, some examples on the use of SQP will be provided.
- Adapted Newton method will be next with more examples and comments.
- Results from the analysis of sensors’ sets by SVD will be also included.
- The improvements after adding a new sensor will be illustrated.
- The results obtained with the ROM, replacing the full model, will be next.
- Finally, the conclusions will bring extra comments on the calculations.

Several relevant results were already provided in the previous chapter, given its interest to understand essential concepts, that should be recalled, in particular:

- Charts regarding the way PROOSIS® rounded off performance results in the direct problem, for a given level of accuracy. A strategy to prevent numerical issues when estimating derivatives with finite differences, determining the values of δ that should be used, was established accordingly.
- Some relevant results on the application of SVD to the problem’s Jacobian matrices, required for the adapted Newton method, evincing the existence of spurious correlations which were affecting systematically to the same degradation vector components (7th, 9th, and 10th). This circumstance was caused by a suboptimal selection of sensors. That set of sensors was installed in a real engine used as reference in this sense (CFM56-5A, with 1 HPT stage).
- Results from a brief sensitivity analysis were provided to better understand when, and why, numerical issues (i.e., mainly lack of uniqueness, but also working out of maps’ acceptable regions) could appear during the application of the different selected techniques, and leading to a classification of cases depending on the severity of the degradations: Usual cases (up to a 2%), very severe degradation cases (up to a 20%), and catastrophic events cases (higher degradation levels). Something to be highlighted is the methodology exposed do not linearize for low degradation levels, The use of maps and the integrity of the rest of the full model in the SW was kept in the calculations. The range of degradations for which losses of uniqueness may happen was estimated with the sensitivity analysis, given the loss of monotony between degradations and sensor readings. The monotonic behavior in the variables guaranteed the existence of a global minimum in the problem. If that condition was not met, there could be more than one solution, complicating the convergence (constraining acceptable solution ranges to leave out spurious results, TIT ± 50 K, solved the issue).
- Several interesting charts regarding the HOSVD of the tensor for the cruise.

Given the great quantity of cases that were run, just representative results will be provided now. For all the techniques, variables were scaled as indicated in the previous chapter. Calculations were done in a desktop PC, i7 Core, at 3.4 GHz.
5.1. - Results obtained with GAs

First, GAs were used to solve the inverse problem. Results obtained with GAs were good enough in terms of accuracy, when the final numerical value of the OF was close enough to zero after several generations. The following tables and figures from different practical examples gather those results, obtained in different cases, involving the calculation of random degradations only (limited to a 2% or 20%, respectively), or obtained together with the T₄T. As it was indicated in the previous chapter, this method does not use derivatives, and that circumstance led to considerably longer CPU times comparing with other techniques in which some more information about the shape of the OF was used to speed up the convergence.

5.1.1. - Example 1

Table 18 shows the results obtained when applying the GA to a random case in which the degradations were kept below a 2% and the TIT was randomly given, not calculated, for two samples of data (1,395.3 K and 1,473.2 K, respectively). \( \bar{X}_{GA} \) and \( \bar{X}_E \) stand both for the result obtained by GA, and the exact solution, respectively. It has been highlighted in the table the rows of components with higher errors, corresponding with components 7\textsuperscript{th}, 9\textsuperscript{th} and 10\textsuperscript{th}, being the 10\textsuperscript{th} clearly the worst in this case. The drop of accuracy in the affected components is of several orders of magnitude comparing with other components not affected by spurious correlations.

| Component | \( \bar{X}_{GA} \) | \( \bar{X}_E \) | Error = |\( |\bar{X}_{GA} - \bar{X}_{EXACT}| \) |
|-----------|-----------------|-----------------|---------|
| X(1)      | 0.2958          | 0.2911          | 4.7E-03 |
| X(2)      | 0.2730          | 0.2721          | 9.4E-04 |
| X(3)      | 1.8389          | 1.8386          | 3.5E-04 |
| X(4)      | 1.1574          | 1.1594          | 2.0E-03 |
| X(5)      | 1.0970          | 1.0997          | 2.7E-03 |
| X(6)      | 0.2865          | 0.2899          | 3.4E-03 |
| X(7)      | 1.2454          | 1.1761          | 6.9E-02 |
| X(8)      | 1.2465          | 1.2441          | 2.4E-03 |
| X(9)      | 0.6586          | 0.7019          | 4.3E-02 |
| X(10)     | 1.1350          | 1.0265          | 1.1E-01 |

Table 18: Results obtained with GAs. Degradations up to a 2% and \( T_{4T} = 1,395.3 \) K, \( 1,473.2 \) K. 100 Individuals.

The Figure 93 contains some interesting results provided by the MATLAB-GA® toolbox. The bar chart corresponds with the previous table (each bar is one calculated degradation component). The population managed was compound by 100 individuals (10 per component), not following the recommendation of the SW, which suggested 200 individuals. A total of 400,220 calls to the OF were required finally to get a value of the OF equal to 8,78 · 10⁻⁵. Each call to the OF implied 2 calls to PROOSIS®, consuming 0.07 CPU seconds each, because 2 samples of data were used to improve the CN of the problem with the original set of sensors. That number of calls implied a total duration, until reaching the desired level of convergence, of more than 15 CPU hours. An acceptable solution for most of degradations was found,
but very slowly. The mechanisms and rules used by the GAs are fixed, not adaptive (and no derivatives are used). So, there are no changes in the way the individuals are selected, just the evolution mechanisms operate iteratively over a set of increasingly evolved individuals. The random factor that could be introduced by the mutation mechanism was clearly limited by the strict constraints imposed to the upper and lower values for the individuals. So, its effectiveness for this problem would be negligible, when applied. And something similar would happen to the migration mechanism, given the size and characteristics of the population.

![Charts representing the resolution, by GAs, of a random case with degradations up to a 2% and given $T_{44} = 1,395.3 \, K$ and $1,473.2 \, K$. Only one TIT was used this time.](image.png)

The value of the overall Fitness Function managed by the GA (coinciding with the OF in the inverse problem) decreased very quickly after few generations, and then the rest of the computational time, for a total of 74 generations, was dedicated to the achievement of a sufficient degree of accuracy to reach convergence (determined by the termination conditions). The average distance between individuals shows some variety in the members of the population until the generation 42, when the individuals seemed to be already very close to each other, and to the final optimum solution. However, the individuals seemed to remain struggling to reach the required degree of convergence. The fourth chart, providing the Fitness Function's value for every individual in the population (which is potentially a solution for the problem) at the last generation, shows some uniformity with already very low values between $1.8 \cdot 10^{-4}$ and $8.7 \cdot 10^{-5}$, which suggests that few variations were to happen in the population if the iterations would have continued, unless the applied evolution mechanisms had introduced some variation to the already very uniform distribution of the population.
5.1.2.- Example 2

Table 19 shows the results obtained when applying the GAs to the resolution of a random case in which the degradations were allowed to reach considerably higher values (up to a 20%), meaning a remarkably worse health condition for the engine, and the TITs were again randomly given, not calculated, for the two samples of data used to abate the CN's value in the problem (1,317.9 K and 1,495.6 K, respectively). It has been highlighted again the components with higher errors, corresponding with the same 3 components than in the previous case. This was a constant circumstance that occurred when the $P_{45t}$ probe was not included in the instrumentation vector. The population in this case was compound by 1,000 individuals (100 per component), aiming to check the impact of this parameter (population’s size) over the duration of the algorithm until its termination.

| Component | $\bar{X}_{GA}$ | $\bar{X}_{E}$ | Error = $|\bar{X}_{GA} - \bar{X}_{EXACT}|$ |
|-----------|----------------|--------------|-------------------------------------|
| X(1)      | 5.3449         | 5.3654       | 2.1E-02                             |
| X(2)      | 1.8981         | 1.8976       | 4.6E-04                             |
| X(3)      | 10.1232        | 10.1265      | 3.3E-03                             |
| X(4)      | 7.0124         | 7.0003       | 1.2E-02                             |
| X(5)      | 9.5623         | 9.5439       | 1.8E-02                             |
| X(6)      | 12.3713        | 12.3421      | 2.9E-02                             |
| X(7)      | 11.7046        | 11.2378      | 4.7E-01                             |
| X(8)      | 17.8621        | 17.8976      | 3.5E-02                             |
| X(9)      | 7.6994         | 7.9088       | 2.1E-01                             |
| X(10)     | 5.1414         | 4.4532       | 6.9E-01                             |

Table 19: Degradation vector obtained using GAs. Degradations were up to a 20%. $T_{4t} = 1,317.9$ K and $1,495.6$ K, respectively, for the two samples of data managed. 10 unknowns, in total, were calculated in this case.

A total of 1,006,750 calls to the OF were required finally to get a value of the Fitness Function (for the fittest individual in the population) equal to $2.99 \cdot 10^{-4}$. That number of calls implied a total duration, until reaching the desired level of convergence, of more than 39 CPU hours. Here is where the impact of the number of individuals of the population is more evident. This was also a “motivating” result to try to keep the population under certain limits, managing minimum quantities of individuals necessary to avoid excessive computational times. In any case, the use of GAs always involved solving times of several hours, with hundreds of thousands of calls to the OF, when the target was to achieve solutions after a maximum of few seconds. This was the main reason to discard this technique for future cases.

On the other hand, the GAs provided a valuable analytical service during the beginning of this study, as they were used to verify that the method was capable to find a solution to the problem, even with an engine extremely deteriorated. So, the effectiveness of the whole methodology would be always guaranteed with the GAs. With some other (quicker) techniques, the degradation’s envelope was expanded, solving cases even with degradations up to 50%. Such level of degradation is, fortunately, not common for most of the cases in the real life but it was reached in the simulations to evince the robustness of the numerical methodology developed.
5.1.3.- Example 3

Table 20 shows the results obtained when applying the GAs to the resolution of a random case in which the degradations and the TIT were calculated together. This example shows a case in which the correlation between the three components previously mentioned was not so clear, most likely because of other numerical effects intervening during the solving process, obtaining a homogeneous error level for all the components, between $1 \cdot 10^{-2}$ and $1 \cdot 10^{-3}$. To solve this case, it was necessary to call to the OF a total of 591,426 times, implying 2 calls to PROOSIS® each, as 2 samples of data were used, leading to 23 hours CPU time required to reach a minimum value in the function of $8.61 \cdot 10^{-4}$. The population was compounded this time by 240 individuals (20 per degradation component), and 2 more variables were added as unknowns, circumstances that would contribute to explain the longer time required when comparing with the first example, even when the total number of generations was lower this time (29 versus the 74 required in the first example).

| Component | $\overline{X}_{GA}$ | $\overline{X}_{E}$ | $\text{Error} = |\overline{X}_{GA} - \overline{X}_{EXACT}|$ |
|-----------|----------------------|---------------------|-------------------------------------------------|
| X(1)      | 1.5084               | 1.5432              | 3.5E-02                                         |
| X(2)      | 1.2617               | 1.2654              | 3.7E-03                                         |
| X(3)      | 0.0018               | 0.0021              | 1.9E-03                                         |
| X(4)      | 0.4667               | 0.4563              | 1.0E-02                                         |
| X(5)      | 0.2551               | 0.2378              | 1.7E-02                                         |
| X(6)      | 0.8110               | 0.7899              | 2.1E-02                                         |
| X(7)      | 0.1515               | 0.1209              | 3.1E-02                                         |
| X(8)      | 1.0993               | 1.1111              | 1.2E-02                                         |
| X(9)      | 0.3368               | 0.3423              | 5.5E-03                                         |
| X(10)     | 1.9999               | 1.9654              | 3.5E-02                                         |

Table 20: Degradation vector obtained using GAs. Degradations were up to a 2%. $T_{AT} = 1,376.2$ K and $1,523.4$ K, respectively, calculated together with $\bar{X}$, for the two samples of data managed.

The accuracy level was also typically higher in the first example (excepting for the usual 7th, 9th, and 10th degradations). If the GA executed for this third example would have been allowed to continue during more generations, the accuracy obtained could have been improved.

The problem with this exploratory technique is the impossibility to estimate how much computational time will be required to reach a certain level of accuracy in the solution. Even in a constrained problem like this one, in the way it is indicated. Again, the technique was capable to find the solution, even after adding 2 more unknowns to the problem (and typically the accuracies obtained in the determination of TIT are the highest, being below $1 \cdot 10^{-5}$ K), but the time required was very long (almost one complete day using a standard desktop computer).

In Figure 94, it is possible to verify how the GA reduced again very quickly the value of the OF from the first generation (initiated by a random population of individuals), to continue working in the refinement of the solution until reaching the termination criteria. This is something found in the solutions to the different cases that were run. They were fast initially to slow down after few iterations.
The last chart in the previous figure indicates some level of dispersion in the population, comparing with the first example. The fitness values of the different individuals were considerably more homogeneous and reduced in the first example. In this third case, there are several orders of magnitude between individuals, pointing to the direction that some more generations could have been given to the algorithm to improve the accuracy of the results.

That potential overall lack of accuracy in the results, because of a premature termination of the GA, could be the explanation to the “masked” correlation between the three usual components that has been commented before. With more generations, the differences between components would have grown to the usual accuracy values (with differences of one or two orders of magnitude). This circumstance links with the discussion about the commitment between CPU time and accuracy, difficult to establish when using GAs, given their random nature.

In regards with the most influential parameter in this sense, the population's size, given the results obtained, it was finally decided to respect the following rule:

- 100 individuals per generation (10 per variable) for the cases with degradations up to a 2%, without calculating the TIT.
- 240 individuals per generation (20 per variable) for the cases with degradations up to a 2%, calculating the TIT.
- 1,000 individuals per generation for degradation values reaching a 20%.
5.2. Results obtained with SQP

The second technique used in this study was the SQP, as applied by MATLAB®, making use of the BFGS formula to update the approximate version of the Hessian matrix. The SQP allowed to reduce considerably the computational time required to solve random cases, comparing with the GAs, passing from hours to minutes. The nature of this technique is totally different to the one in GA. This method does make use of the derivatives of the OF (containing the necessary information on the shape of such locus) in its theoretical development. When the calculation of the Hessian is done, the rate of convergence of the technique is quadratic, or in other words:

$$\lim_{k \to \infty} \frac{|X_{k+1} - X_E|}{|X_k - X_E|^p} = \epsilon; \quad \text{with } p = 2$$  \hspace{1cm} (5.1)

Being $X_k$ and $X_E$ the value obtained in the “k-th” iteration and the exact solution, respectively, and $\epsilon$ a real positive number lower than 1. But that calculation is expensive and must be done every iteration. As one approximation of the Hessian will be used instead, given by the BFGS method, the rate of convergence will be only super-linear ($p > 1$). However, that improvement shortened considerably the computational times required by the GAs (with $p = 0$), as it will be shown now.

5.2.1. Example 4

The results obtained in 2 random cases from the whole set of runs, solved with the SQP method, are shown in Table 21. More than 100 cases were calculated but, for the sake of brevity and conciseness, just a pair of them are shown here as a sample of all the rest. As the errors obtained were always relatively small, only the initial conditions ($\bar{X}_0$), and the exact degradation status of the engine ($\bar{X}_E$), are provided, together with the errors ($Error_X$) in between the exact and the calculated conditions (i.e., $|\bar{X}_E - \bar{X}_{SQP}|$). Same consideration will be taken in the rest of tables of results in this section. The $T_{4t}$ values were always randomly chosen (and given in this case):

<table>
<thead>
<tr>
<th>X</th>
<th>Case 1</th>
<th>Case 2</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>$\bar{X}_0$</td>
<td>$\bar{X}_E$</td>
</tr>
<tr>
<td>X(1)</td>
<td>0.5129</td>
<td>1.2146</td>
</tr>
<tr>
<td>X(2)</td>
<td>1.2269</td>
<td>0.9003</td>
</tr>
<tr>
<td>X(3)</td>
<td>1.1645</td>
<td>0.9175</td>
</tr>
<tr>
<td>X(4)</td>
<td>1.0815</td>
<td>1.3239</td>
</tr>
<tr>
<td>X(5)</td>
<td>1.7399</td>
<td>1.5406</td>
</tr>
<tr>
<td>X(6)</td>
<td>0.5296</td>
<td>0.7004</td>
</tr>
<tr>
<td>X(7)</td>
<td>0.6361</td>
<td>1.3240</td>
</tr>
<tr>
<td>X(8)</td>
<td>0.2384</td>
<td>0.8323</td>
</tr>
<tr>
<td>X(9)</td>
<td>1.8797</td>
<td>1.6839</td>
</tr>
<tr>
<td>X(10)</td>
<td>1.2911</td>
<td>1.6658</td>
</tr>
</tbody>
</table>

Table 21: Results obtained in two cases by using SQP, with degradations up to a 2%, for given values of $T_{4t}$: 1,387.6 K and 1,489.1 K for Case 1, and 1,377.5 K and 1,467.3 K for Case 2, respectively.
The worst results obtained with the GAs before, in the components $X(7), X(9)$ and $X(10)$, corresponding with $\eta_{\text{HPT}}, \eta_{\text{LPT}},$ and $\Gamma_{\text{LPT}}$, respectively, were again obtained by the SQP. The errors in those components were still considerably larger than in the rest of degradations.

On the other hand, the method converges reasonably well, even when both the initial degradation and the exact final condition of the engine were randomly chosen, considering the usual degradation values that can be expected in a real flight. This circumstance suggested that the OF would count with only one local minimum inside the constrained region of the solution space in which the example is calculated, as it was commented in the previous chapter.

The average computational time required to solve each case was around 1.6 CPU minutes, needing around 35 iterations until the established convergence criteria were met (same threshold value $\sim 1 \cdot 10^{-4}$, applied either to the OF, or to the step size between iterations to terminate the algorithm), choosing finite forward differences instead of centered to try to reduce, preferably, the time dedicated per case. When the centered differences were selected instead, aiming to gain accuracy in the results, just 20 iterations were required, but meaning longer CPU times, as the centered differences imply the double of calls to the full model in PROOSIS®.

The high number of iterations required is consistent with the inherent ill-conditioning of the inverse problem, aggravated by the suboptimal set of sensors considered for the engine. Even when that time improved considerably the one obtained with the GAs, this technique was still far from the barrier of a second.

### 5.2.2. Example 5

When the $T_{4t}$ values of the 2 different data samples were taken as unknowns, $T_{4t}^1$ and $T_{4t}^2$ were calculated together with the degradations. Keeping the rest of the options in the SQP identical to the case when $T_{4t}$ were given (forward differences, termination criteria, and values of $\delta$) and choosing randomly both initial and exact degradation values. More than 100 cases were finally solved. Table 22 shows the results obtained in two of those cases only, for the sake of brevity:

<table>
<thead>
<tr>
<th>X</th>
<th>Case 1</th>
<th>Case 2</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>$\overline{X}_0$</td>
<td>$\overline{X}_E$</td>
</tr>
<tr>
<td>$X(1)$</td>
<td>1.7268</td>
<td>1.9432</td>
</tr>
<tr>
<td>$X(2)$</td>
<td>1.0124</td>
<td>1.6431</td>
</tr>
<tr>
<td>$X(3)$</td>
<td>1.4309</td>
<td>0.2354</td>
</tr>
<tr>
<td>$X(4)$</td>
<td>1.2315</td>
<td>0.7786</td>
</tr>
<tr>
<td>$X(5)$</td>
<td>1.8766</td>
<td>0.0874</td>
</tr>
<tr>
<td>$X(6)$</td>
<td>0.5637</td>
<td>1.6875</td>
</tr>
<tr>
<td>$X(7)$</td>
<td>0.4501</td>
<td>1.8655</td>
</tr>
<tr>
<td>$X(8)$</td>
<td>0.3567</td>
<td>0.4356</td>
</tr>
<tr>
<td>$X(9)$</td>
<td>0.4580</td>
<td>0.2121</td>
</tr>
<tr>
<td>$X(10)$</td>
<td>1.9872</td>
<td>0.7864</td>
</tr>
</tbody>
</table>

Table 22: Results obtained by using SQP, calculating the values of $T_{4t}$ as unknowns.
The temperature values were very accurate, again with an error of less than $1 \cdot 10^{-5}$ K. The randomly chosen $T_{4t}$ values, in the first case, were $(T_{4t1}, T_{4t2}) = (1,348.3 \text{ K}; 1,473.2 \text{ K})$. Meanwhile, in the second case, the values were $(T_{4t1}, T_{4t2}) = (1,378.9 \text{ K}; 1,490.3 \text{ K})$.

5.2.3. Example 6

The SQP resulted, as expected, somehow slower with the two extra unknowns, needing about 2.0 CPU minutes (in average) to solve each case. The same components $X(7)$, $X(9)$ and $X(10)$ obtained worse results but the method still converged satisfactorily well, considering that both initial and exact values were randomly determined. Figure 95 provides several examples from different cases where the TIT values were provided as given data to the problem. The total number of function evaluations were in an average order of 400 to 600. Nevertheless, some cases were solved more quickly (in about 1.0 CPU minute).

Figure 95: Charts and data from four random cases (2% degradation) where the values of $T_{4t}$ were given. Components of vectors $\bar{X}_0$ (initial) and $\bar{X}_E$ (exact) are shown in the associated tables.
Meanwhile, for the cases where the TIT values were considered as unknown variables, the total number of function evaluations were in an average order of 600 to 800 (again, some cases needed less calls, as it is shown in Figure 96), meaning times above 3.0 CPU minutes. The number of function calls increased with the number of iterations, so a more demanding effort for the SQP to approach to the final solution was required (and a slower performance). If the level of accuracy obtained in the results were higher than the one finally necessary (given the accuracy afforded by the real instrumentation mounted in the engine), the total number of iterations could be reduced to try to minimize computational times. A commitment between accuracy and CPU time would always have to be established.

![Figure 96: Charts and data from four random cases (2% degradation) where the values of $T_{\text{IT}}$ were calculated. Components of vectors $\hat{X}_0$ (initial) and $\hat{X}_E$ (exact) are shown in the associated tables.](image)

Some of the cases could look standard and easy to calculate, but the reality is both initial and exact points were randomly selected, leading to situations in which the engine components improved while others continued growing. It is expected to find more standard degradation patterns in real cases. To challenge this technique, certain very rare cases were evaluated. Results are shown in the next example.
5.2.4. Example 7

To test the robustness of the SQP, rare cases were suggested, combining modules that showed an expected incremental degradation with others that improved. One improvement could be expected after a reparation or a water-wash, not during a flight. Figure 97 shows three cases (a, b, and c) of this kind, with the situation in the initial point and the final one. It was achieved a good accuracy in all of them.

![Figure 97](image_url)

**Figure 97:** Charts with three rare cases (up to a 2% degradation and TITs being unknown) to test the robustness of the method. That evolution of degradations (improving and worsening this way) never typically happens.

Components of vectors $\vec{X}_0$ (initial) and $\vec{X}_E$ (exact) are shown in the associated tables.
Some comments should be made on the previous three rare cases:

- First, higher number of iterations were purposedly allowed, comparing with previous examples, to obtain very accurate solutions. This was part of the challenge, finding out what the computational cost of solving such rare cases, with a very good level of accuracy, would be. It led to CPU times between 4 and 5 minutes. Typically, the termination conditions would have ended the algorithm before, leading to more usual times below 2 CPU minutes.

- Additionally, it is important to highlight that, for the optimization problem, there are no such “initial” or “final” conditions, there are just possible health conditions compatible with the OF and the constraints that were imposed. No temporal considerations whatsoever are made. That is the reason why it is perfectly possible to find routes in the solution space between two engine conditions, apparently unrelated and impossible to link during its normal operation. This result contributes to clinch the model’s robustness.

- However, as it was indicated in the previous chapter, depending on the behavior (monotonic or not) of the variables of the problem and the OF’s shape, always inside acceptable maps’ regions, solutions could be hard to obtain. This is probably the main caveat for the SQP method. Certain routes tested with the SQP were found troublesome, requiring the use of centered differences instead of forward differences (meaning more time) to get the required level of accuracy, so this possibility must be contemplated.

- This circumstance suggests that some strategy should be established to solve the different cases, based on the difficulties found, to get a solution.

5.2.5.- Example 8

Finally, to complete this section, a pair of cases (Figure 98) will be shown for degradations up to a 20%, calculating the TIT values of the 2 samples of measurements. The SQP solved the cases (centered differences), after many more iterations (in the order of 2,000 to 4,000) and longer times (7 to 8 CPU minutes).

\[
\begin{array}{|c|c|c|c|}
\hline
\text{T}_{\text{at}}=1,338.8 \text{ K and } \text{T}_{\text{at}}=1,506.8 \text{ K} \hline
X & X_0 & X_1 & \text{Error} \\
\hline
X(1) & 8.2463 & 1.5171 & 3.88 \text{ E-02} \\
X(2) & 15.8827 & 1.0790 & 9.96 \text{ E-04} \\
X(3) & 9.2243 & 10.6160 & 1.85 \text{ E-03} \\
X(4) & 10.5707 & 15.5533 & 6.56 \text{ E-03} \\
X(5) & 3.3130 & 18.6802 & 4.33 \text{ E-02} \\
X(6) & 12.0386 & 2.3981 & 5.98 \text{ E-02} \\
X(7) & 5.2994 & 11.3785 & 1.38 \text{ E-01} \\
X(8) & 15.0816 & 9.1878 & 1.28 \text{ E-02} \\
X(9) & 12.7843 & 0.2380 & 6.72 \text{ E-02} \\
X(10) & 14.9630 & 6.7425 & 2.22 \text{ E-01} \\
\hline
\end{array}
\]

\[
\begin{array}{|c|c|c|c|}
\hline
\text{T}_{\text{at}}=1,338.8 \text{ K and } \text{T}_{\text{at}}=1,512.1 \text{ K} \hline
X & X_0 & X_1 & \text{Error} \\
\hline
X(1) & 6.0229 & 16.8389 & 6.76 \text{ E-02} \\
X(2) & 9.9114 & 2.6171 & 6.52 \text{ E-03} \\
X(3) & 5.1632 & 3.7830 & 5.16 \text{ E-04} \\
X(4) & 14.0576 & 3.0737 & 1.47 \text{ E-03} \\
X(5) & 2.3532 & 0.5780 & 2.22 \text{ E-03} \\
X(6) & 14.9208 & 0.1817 & 2.78 \text{ E-03} \\
X(7) & 16.1057 & 11.9200 & 1.45 \text{ E-02} \\
X(8) & 14.9046 & 12.1809 & 2.48 \text{ E-04} \\
X(9) & 6.7428 & 18.3784 & 7.68 \text{ E-02} \\
X(10) & 11.6864 & 14.6714 & 2.95 \text{ E-01} \\
\hline
\end{array}
\]

Figure 98: Charts and data from four random cases (20% degradation) where the values of $T_{4\text{t}}$ were calculated. Components of vectors $\mathbf{X}_0$ (initial) and $\mathbf{X}_E$ (exact) are shown in the associated tables.
5.3. - Results obtained with the adapted Newton method

As the results from the optimization problem pointed to the existence of a global minimum, at least for the typical problem conditions (e.g., up to a 2% of components’ degradations), an adapted Newton method was developed accordingly to try to speed up the solving process. This adapted method did not need of a good initial guess (in fact, initial values of degradation of ~50%, and higher, were successfully managed) and allowed for imposing that all degradations would be non-negative, just by correcting, as necessary, the results in the algorithm if a negative component was obtained after any iteration. Negative degradations could likely appear during the first iterations (most of the times only in the very first iteration).

5.3.1. - Example 9

The convergence thresholds, as indicated in the previous chapter, were typically kept like $\varepsilon_1 = 1 \cdot 10^{-2}$ and $\varepsilon_2 = 1 \cdot 10^{-3}$, allowing no more than fifteen iterations. Even so, accuracies were higher than in the case of SQP, as shown in Table 23:

<table>
<thead>
<tr>
<th>X</th>
<th>Case 1</th>
<th>Case 2</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>$\bar{X}_0$</td>
<td>$\bar{X}_E$</td>
</tr>
<tr>
<td>X(1)</td>
<td>1.2895</td>
<td>0.0652</td>
</tr>
<tr>
<td>X(2)</td>
<td>0.7525</td>
<td>1.1224</td>
</tr>
<tr>
<td>X(3)</td>
<td>0.3818</td>
<td>1.7637</td>
</tr>
<tr>
<td>X(4)</td>
<td>0.8565</td>
<td>1.3384</td>
</tr>
<tr>
<td>X(5)</td>
<td>0.9640</td>
<td>0.3809</td>
</tr>
<tr>
<td>X(6)</td>
<td>0.2412</td>
<td>0.7378</td>
</tr>
<tr>
<td>X(7)</td>
<td>1.1790</td>
<td>0.9215</td>
</tr>
<tr>
<td>X(8)</td>
<td>0.4524</td>
<td>1.9633</td>
</tr>
<tr>
<td>X(9)</td>
<td>0.7692</td>
<td>0.3128</td>
</tr>
<tr>
<td>X(10)</td>
<td>1.1660</td>
<td>1.7110</td>
</tr>
</tbody>
</table>

Table 23: Results obtained with the adapted Newton-like method. $T_{uf}$ values used are given by Eq. (4.4).

The robustness of this new method was tested by successfully running several thousands of cases, always with both random initial and exact engine conditions. The small errors obtained highlight the good convergence reached. The CN of the Jacobian matrices was in an average of ~300, meanwhile the CN of the Hessian matrix typically used in the SQP method (or their approximations) would be around $1 \cdot 10^5$. As expected, X(7), X(9), and X(10) obtained worse results.

The computational time needed was about ~10 CPU seconds, so this new method was much faster than the SQP method and the level of accuracy reached was also higher. These values were aligned with the computational cost per each PROOSIS® run, which took about 0.07 CPU seconds in average. MATLAB® required 11 OF’s evaluations to calculate its gradient by forward differences (20 evaluations if centered differences would be used instead, but this was not necessary in the cases that were run after several tests).
5.3.2.- Example 10

Similar approach was followed leaving the $T_{4t}$ values as unknown variables. Several hundreds of cases were solved successfully, being this result evidence of the robustness of the method. Table 24 provides just two of those solved cases:

<table>
<thead>
<tr>
<th>X</th>
<th>Case 1</th>
<th>Case 2</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>$\bar{X}_0$</td>
<td>$\bar{X}_E$</td>
</tr>
<tr>
<td>X(1)</td>
<td>1.9004</td>
<td>1.4863</td>
</tr>
<tr>
<td>X(2)</td>
<td>0.0689</td>
<td>0.7845</td>
</tr>
<tr>
<td>X(3)</td>
<td>0.8775</td>
<td>1.3110</td>
</tr>
<tr>
<td>X(4)</td>
<td>0.7631</td>
<td>0.3424</td>
</tr>
<tr>
<td>X(5)</td>
<td>1.5310</td>
<td>1.4121</td>
</tr>
<tr>
<td>X(6)</td>
<td>1.5904</td>
<td>0.0637</td>
</tr>
<tr>
<td>X(7)</td>
<td>0.3737</td>
<td>0.5538</td>
</tr>
<tr>
<td>X(8)</td>
<td>0.9795</td>
<td>0.0923</td>
</tr>
<tr>
<td>X(9)</td>
<td>0.8912</td>
<td>0.1943</td>
</tr>
<tr>
<td>X(10)</td>
<td>1.2926</td>
<td>1.6469</td>
</tr>
</tbody>
</table>

Table 24: Results obtained with the adapted Newton method. The values $T_{4t}$ were calculated as unknowns.

After including 2 TIT values to the list of variables to be computed, the size of the enlarged Jacobian matrix was as follows:

$$\left(2 \cdot N_{sen}\right) \times \left(N_{deg} + 2\right)$$

(5.2)

Random values were taken for both initial and exact engine condition status, as it was done before. The temperature values were calculated again with an error even lower than $1 \cdot 10^{-3}$ K (sometimes reaching $1 \cdot 10^{-7}$ K). The randomly chosen $T_{4t}$ values, in the first selected case, were $(T_{4t}^1, T_{4t}^2) = (1,367.9$ K; $1,525.8$ K). Meanwhile, in the second case, the values were $(T_{4t}^1, T_{4t}^2) = (1,380.5$ K; $1,507.7$ K).

Robustness, efficiency, and accuracy are desirable goals that seemed to be reached satisfactorily with the adapted Newton method, without remarkable conflicts in between them. However, for real-time applications, the target would be to obtain results in times of the order of 1 CPU second. Here, some commitment between accuracy and computational time could be invoked to improve the later by worsening not too much the former.

If by dedicating 5 iterations, the less accurate components count with errors in the order of $\sim 10^{-3}$ then, with some less iterations allowed, the accuracy reached will be certainly lower, but the improvement in time may compensate that reduction (errors in the order of $\sim 10^{-2}$, could result acceptable depending on the accuracy managed by the instrumentation).

Ideally, changes in the degradations of two order of magnitude below the values managed for the instrumentation will capture with guarantees the progressive deterioration in the engine.
5.3.3. Example 11

Multiple cases with a higher severity of random degradation in the engine were also tried with this methodology, obtaining successful results. A pair of examples, considering the $T_{4t}$ values as known data, are shown in Table 25. The average CPU times needed were about $\sim 15$ CPU seconds when the TITs were given.

<table>
<thead>
<tr>
<th>X</th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>$X_0$</td>
<td>$X_E$</td>
<td>Error$_X$</td>
<td>$X_0$</td>
</tr>
<tr>
<td>X(1)</td>
<td>10.0373</td>
<td>2.6145</td>
<td>1.5·10$^{-3}$</td>
<td>11.4038</td>
</tr>
<tr>
<td>X(2)</td>
<td>5.5326</td>
<td>1.0274</td>
<td>3.6·10$^{-4}$</td>
<td>6.9294</td>
</tr>
<tr>
<td>X(3)</td>
<td>2.3932</td>
<td>12.5497</td>
<td>4.0·10$^{-4}$</td>
<td>11.1501</td>
</tr>
<tr>
<td>X(4)</td>
<td>17.731</td>
<td>0.5809</td>
<td>8.2·10$^{-4}$</td>
<td>5.9956</td>
</tr>
<tr>
<td>X(5)</td>
<td>19.4057</td>
<td>2.7235</td>
<td>2.9·10$^{-4}$</td>
<td>3.1814</td>
</tr>
<tr>
<td>X(6)</td>
<td>18.8505</td>
<td>13.8903</td>
<td>8.0·10$^{-4}$</td>
<td>13.3051</td>
</tr>
<tr>
<td>X(7)</td>
<td>12.7625</td>
<td>10.3355</td>
<td>2.2·10$^{-2}$</td>
<td>13.6840</td>
</tr>
<tr>
<td>X(8)</td>
<td>1.8118</td>
<td>10.8520</td>
<td>4.1·10$^{-4}$</td>
<td>15.8481</td>
</tr>
<tr>
<td>X(9)</td>
<td>1.4941</td>
<td>16.1552</td>
<td>1.4·10$^{-2}$</td>
<td>6.9724</td>
</tr>
<tr>
<td>X(10)</td>
<td>3.6490</td>
<td>15.9245</td>
<td>5.1·10$^{-2}$</td>
<td>5.0014</td>
</tr>
</tbody>
</table>

Table 25: Results obtained with the adapted Newton-like method for degradations up to a 20%. $T_{4t}$ values were given: 1, 349. 2 $K$ and 1, 482. 1 $K$ in Case 1 and 1, 302. 9 $K$, and 1, 459. 7 $K$ in Case 2.

5.3.4. Example 12

Table 26 provides 2 similar cases but obtaining the $T_{4t}$ values as unknowns. Times $\sim 20$ CPU seconds were obtained when the TITs were calculated together with X(i).

<table>
<thead>
<tr>
<th>X</th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>$X_0$</td>
<td>$X_E$</td>
<td>Error$_X$</td>
<td>$X_0$</td>
</tr>
<tr>
<td>X(1)</td>
<td>13.1909</td>
<td>8.9577</td>
<td>1.9·10$^{-4}$</td>
<td>15.6718</td>
</tr>
<tr>
<td>X(2)</td>
<td>5.8954</td>
<td>13.0249</td>
<td>9.9·10$^{-5}$</td>
<td>13.5412</td>
</tr>
<tr>
<td>X(3)</td>
<td>19.0073</td>
<td>3.3900</td>
<td>4.1·10$^{-4}$</td>
<td>2.9962</td>
</tr>
<tr>
<td>X(4)</td>
<td>13.8857</td>
<td>10.6289</td>
<td>3.5·10$^{-4}$</td>
<td>13.9323</td>
</tr>
<tr>
<td>X(5)</td>
<td>4.1361</td>
<td>12.6760</td>
<td>4.5·10$^{-4}$</td>
<td>2.5802</td>
</tr>
<tr>
<td>X(6)</td>
<td>11.0952</td>
<td>0.2819</td>
<td>6.2·10$^{-4}$</td>
<td>18.9189</td>
</tr>
<tr>
<td>X(7)</td>
<td>17.5855</td>
<td>9.4074</td>
<td>2.0·10$^{-2}$</td>
<td>17.7282</td>
</tr>
<tr>
<td>X(8)</td>
<td>11.1571</td>
<td>17.7265</td>
<td>6.6·10$^{-4}$</td>
<td>10.3000</td>
</tr>
<tr>
<td>X(9)</td>
<td>15.0466</td>
<td>2.2805</td>
<td>1.1·10$^{-2}$</td>
<td>13.5881</td>
</tr>
<tr>
<td>X(10)</td>
<td>17.8980</td>
<td>8.8508</td>
<td>2.9·10$^{-2}$</td>
<td>19.5358</td>
</tr>
</tbody>
</table>

Table 26: Results obtained with the adapted Newton-like method for degradations up to a 20%. $T_{4t}$ values were calculated: 1, 385. 6 $K$ and 1, 460. 0 $K$ in Case 1 and 1, 351. 4 $K$, and 1, 529. 9 $K$ in Case 2.
5.3.5.- Example 13

Figure 99 shows absolute errors, obtained for all the degradation vector's components and for both calculated TITs, after running 100 random cases allowing degradation levels up to a 2%. Average values are indicated with red squares. Most of the results were clearly below $1 \cdot 10^{-2}$, however few outliers were detected with higher errors, corresponding typically with the already mentioned degradation vector's components 7th, 9th, and 10th, affected by spurious correlations. The non-correlated components counted with errors below $1 \cdot 10^{-3}$, excepting few outliers. TIT values had always an error below 0.1 K. To get these results, an average of 5 iterations was needed in the 100 cases. If the correlations affecting to the 3 mentioned components disappeared, then the accuracy could be challenged to get better CPU times by reducing the number of iterations (to go from an accuracy of $1 \cdot 10^{-3}$ to another one still valid of $1 \cdot 10^{-2}$, for instance).

Figure 99: Results from 100 random cases, initial and final degradations up to a 2%.

The following 3 figures explore this potential strategy, aiming to reduce the CPU time. Figure 100 means a promising result, as it shows how by limiting the number of iterations to 3 it is still possible to achieve accuracies of $1 \cdot 10^{-1}$ (and below), in average, for the degradations not affected by correlations and for the TITs. The conclusion seems to be evident: It would be very interesting for the purpose of this study to solve the existing problem with the spurious correlations to get closer to a real-time solution. The next two charts, in Figure 101 and Figure 102, showing similar results but limiting even more the number of iterations, to 2 and 1 respectively, could mean a too aggressive approach to capture small changes in X, given the characteristics of the instrumentation without $P_{45t}$. The errors in these two last cases may be too high in average to be useful. 2 iterations seem to be the limit. The next 3 charts show results for 100 cases each, only for the sake of clarity in the figures, because 1,000 cases were calculated. Average values for 100 cases and for 1,000 were similar. Average TIT errors remained equal or below to 1 K.

Nevertheless, this situation changes when the increments in degradation are not too high. Figure 103, Figure 104, and Figure 105 are the counterpart to the previous 3 charts but considering variations in degradation of only 0.1% over 1% levels. In such case, even only 1 iteration could be enough, meaning 3.89 CPU seconds.
Figure 100: Results from 100 random cases, initial and final degradations up to a 2%, allowing 3 iterations.

Figure 101: Results from 100 random cases, initial and final degradations up to a 2%, allowing 2 iterations.

Figure 102: Results from 100 random cases, initial and final degradations up to a 2%, allowing 1 iteration.
Figure 103: Results from 100 random cases, degradations up to a 1%, increments up to 0.1%, allowing 3 iterations.

Figure 104: Results from 100 random cases, degradations up to a 1%, increments up to 0.1%, allowing 2 iterations.

Figure 105: Results from 100 random cases, degradations up to a 1%, increments up to 0.1%, allowing 1 iteration.
5.4.- Analysis of the results obtained after applying SVD

In the previous sections of this chapter, it was highlighted that the results obtained for the components X(7), X(9), and X(10) were systematically less accurate than the results for the rest of degradations. And this situation was detected in all the techniques used along the study. The solution to this issue was already mentioned in the previous chapter: The inclusion of an additional sensor, a $P_{45t}$ probe, would be enough to improve considerably the CN of the inverse problem, allowing also to obtain the value of $T_{4t}$ with just 1 sample of data, and not with 2.

Next results will spread some more light on the reasons why the inclusion of that sensor was clearly convenient, and why the substitution of any other existing sensor by the new probe would have not improved the situation comparing with the initial instrumentation considered. Figure 106 shows different sensors’ replacements that have been evaluated (cases a, b, and c), and the modules that resulted affected by correlations right after those decisions were made.

![Figure 106: Figure showing the sensors replaced (red), sequentially a) $P_{25t}$, b) $N_{Ht}$, and c) $T_{5t}$, by the new $P_{45t}$ probe (green). Back images obtained from [145] and [14], respectively.]

5.4.1.- Example 14

Following data contained in Table 27, Table 28, and Table 29 show highlighted the coefficients associated to the components that would be affected by new correlations, originated when replacing any of the existing sensors by a new $P_{45t}$ probe. It is evident from the tables below that the situation does not improve replacing sensors. The problem is just moved to some other modules of the engine:
• In Case a), after removing the $P_{25t}$, the 2 main modules, before and after this sensor (i.e., booster and HPC), which are affected by the coefficients given by $b_i$ (for “i” values from 3 to 6, colored in the table), are showing a similar behavior to the one seen previously in Chapter 4, where the $P_{45t}$ probe had not been installed yet, and the main modules, upstream and downstream of its location in the engine (i.e., HPT and LPT), were also affected by a spurious numerical correlation. And again, the same correlation independently of $T_{4t}$.

• In Case b) removing the information provided by the rotational speed sensor for the high-pressure spool, $N_{H}$, will lead now to a new correlation affecting precisely the 2 modules rotating simultaneously at that same speed (i.e., HPC and HPT). So, the missing data will avoid again getting a more accurate result.

• Finally, in Case c), by removing the $T_{5t}$, the affected modules were finally the Fan, LPC and LPT. Similar situation.

<table>
<thead>
<tr>
<th>$T_{4t}$</th>
<th>100$b_1$</th>
<th>100$b_2$</th>
<th>$b_3$</th>
<th>$b_4$</th>
<th>$b_5$</th>
<th>$b_6$</th>
<th>100$b_7$</th>
<th>100$b_8$</th>
<th>100$b_9$</th>
<th>100$b_{10}$</th>
</tr>
</thead>
<tbody>
<tr>
<td>1,300</td>
<td>0.19</td>
<td>−0.22</td>
<td>−0.37</td>
<td>−0.61</td>
<td>0.67</td>
<td>0.20</td>
<td>0.17</td>
<td>−0.23</td>
<td>−0.10</td>
<td>0.06</td>
</tr>
<tr>
<td>1,350</td>
<td>−0.51</td>
<td>0.11</td>
<td>−0.40</td>
<td>−0.68</td>
<td>0.60</td>
<td>0.11</td>
<td>−0.73</td>
<td>0.28</td>
<td>0.21</td>
<td>0.15</td>
</tr>
<tr>
<td>1,400</td>
<td>−0.17</td>
<td>−0.17</td>
<td>0.44</td>
<td>0.73</td>
<td>−0.51</td>
<td>−0.07</td>
<td>0.15</td>
<td>−0.10</td>
<td>−0.06</td>
<td>−0.03</td>
</tr>
<tr>
<td>1,450</td>
<td>0.08</td>
<td>0.25</td>
<td>−0.47</td>
<td>−0.76</td>
<td>0.44</td>
<td>0.04</td>
<td>0.25</td>
<td>−0.37</td>
<td>0.09</td>
<td>−0.26</td>
</tr>
<tr>
<td>1,500</td>
<td>0.19</td>
<td>0.35</td>
<td>−0.49</td>
<td>−0.78</td>
<td>0.38</td>
<td>0.02</td>
<td>0.03</td>
<td>0.24</td>
<td>0.04</td>
<td>−0.12</td>
</tr>
</tbody>
</table>

Table 27: Case a). List of $b_i$ coefficients for different typical cruise $T_{4t}$, after replacing $P_{25t}$ by $P_{45t}$. Condition number in the order of $\sim 1 \cdot 10^4$. Correlations affect both HPC and LPC.

<table>
<thead>
<tr>
<th>$T_{4t}$</th>
<th>100$b_1$</th>
<th>100$b_2$</th>
<th>$b_3$</th>
<th>$b_4$</th>
<th>$b_5$</th>
<th>$b_6$</th>
<th>100$b_7$</th>
<th>100$b_8$</th>
<th>100$b_9$</th>
<th>100$b_{10}$</th>
</tr>
</thead>
<tbody>
<tr>
<td>1,300</td>
<td>0.04</td>
<td>−0.52</td>
<td>0.04</td>
<td>−0.99</td>
<td>−0.94</td>
<td>−0.39</td>
<td>−0.15</td>
<td>−0.03</td>
<td>−0.03</td>
<td>−0.25</td>
</tr>
<tr>
<td>1,350</td>
<td>0.25</td>
<td>−0.10</td>
<td>−0.06</td>
<td>0.99</td>
<td>0.24</td>
<td>0.05</td>
<td>0.16</td>
<td>0.02</td>
<td>−0.24</td>
<td>−0.27</td>
</tr>
<tr>
<td>1,400</td>
<td>−0.33</td>
<td>−0.53</td>
<td>−0.07</td>
<td>0.99</td>
<td>1.13</td>
<td>−0.19</td>
<td>0.15</td>
<td>0.02</td>
<td>−0.16</td>
<td>−0.01</td>
</tr>
<tr>
<td>1,450</td>
<td>0.07</td>
<td>−0.41</td>
<td>−0.07</td>
<td>0.99</td>
<td>0.48</td>
<td>−0.76</td>
<td>0.14</td>
<td>0.03</td>
<td>−0.13</td>
<td>0.18</td>
</tr>
<tr>
<td>1,500</td>
<td>−0.18</td>
<td>−0.32</td>
<td>−0.06</td>
<td>0.99</td>
<td>−0.22</td>
<td>−0.38</td>
<td>0.14</td>
<td>0.03</td>
<td>0.05</td>
<td>0.19</td>
</tr>
</tbody>
</table>

Table 28: Case b). Analogous table, replacing $N_{H}$ by $P_{45t}$. Condition number of $\sim 1 \cdot 10^4$. Correlations affect both HPC and HPT.

<table>
<thead>
<tr>
<th>$T_{4t}$</th>
<th>$b_1$</th>
<th>$b_2$</th>
<th>100$b_3$</th>
<th>100$b_4$</th>
<th>$b_5$</th>
<th>100$b_6$</th>
<th>$b_7$</th>
<th>100$b_8$</th>
<th>$b_9$</th>
<th>100$b_{10}$</th>
</tr>
</thead>
<tbody>
<tr>
<td>1,300</td>
<td>0.79</td>
<td>0.05</td>
<td>−0.11</td>
<td>0.34</td>
<td>−0.44</td>
<td>−0.18</td>
<td>0.26</td>
<td>−0.42</td>
<td>−0.36</td>
<td>−0.55</td>
</tr>
<tr>
<td>1,350</td>
<td>0.81</td>
<td>0.05</td>
<td>0.18</td>
<td>−0.06</td>
<td>−0.43</td>
<td>−0.18</td>
<td>−0.17</td>
<td>0.07</td>
<td>−0.36</td>
<td>−0.46</td>
</tr>
<tr>
<td>1,400</td>
<td>−0.81</td>
<td>−0.05</td>
<td>0.06</td>
<td>0.24</td>
<td>0.41</td>
<td>−0.18</td>
<td>0.03</td>
<td>0.11</td>
<td>0.36</td>
<td>0.47</td>
</tr>
<tr>
<td>1,450</td>
<td>−0.82</td>
<td>−0.05</td>
<td>0.18</td>
<td>−0.26</td>
<td>0.41</td>
<td>0.17</td>
<td>−0.04</td>
<td>0.49</td>
<td>0.36</td>
<td>1.01</td>
</tr>
<tr>
<td>1,500</td>
<td>−0.82</td>
<td>−0.06</td>
<td>0.42</td>
<td>−0.62</td>
<td>0.39</td>
<td>0.17</td>
<td>−0.32</td>
<td>0.24</td>
<td>0.36</td>
<td>0.61</td>
</tr>
</tbody>
</table>

Table 29: Case c). Analogous table, replacing $T_{5t}$ by $P_{45t}$. Condition number of $\sim 1 \cdot 10^4$. Fan, LPC and LPT affected by this change.

The modules affected by correlations were always monitored (or coupled to modules that were monitored) by the sensor that had been previously removed.
When one of these correlations appeared, it indicated that the variation in degradation could be obtained with the rest of variations of degradations affected by the correlation, which is not possible, given the independent nature of each degradation. The lack of information to solve the inverse problem introduced spurious relationships between degradations, that eventually led to less realistic results. As no improvement was obtained by replacing any of the existing sensors with a new pressure probe, it was admitted that the right solution to optimize the existing instrumentation would be completing it by adding that new pressure probe.

5.5.- Calculations after the upgrade by including one additional sensor

The additional information provided by the new sensor made possible obtaining more accurate and representative solutions, like the ones shown in Table 30, for degradations up to a 2%, obtained by the adapted Newton method working with (11 x 11) Jacobians in which $P_{45t}$ and $T_{4t}$ were included.

5.5.1.- Example 15

The previously introduced techniques were benefitted by this instrumentation change, reducing computational times, and improving accuracy in results.

<table>
<thead>
<tr>
<th>X</th>
<th>Case 1</th>
<th>Case 2</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>$X_0$</td>
<td>$X_E$</td>
</tr>
<tr>
<td>X(1)</td>
<td>0.7609</td>
<td>0.5022</td>
</tr>
<tr>
<td>X(2)</td>
<td>1.1356</td>
<td>1.2321</td>
</tr>
<tr>
<td>X(3)</td>
<td>0.1517</td>
<td>0.9466</td>
</tr>
<tr>
<td>X(4)</td>
<td>0.1079</td>
<td>0.7033</td>
</tr>
<tr>
<td>X(5)</td>
<td>1.0616</td>
<td>1.6617</td>
</tr>
<tr>
<td>X(6)</td>
<td>1.5583</td>
<td>1.1705</td>
</tr>
<tr>
<td>X(7)</td>
<td>1.8680</td>
<td>1.0994</td>
</tr>
<tr>
<td>X(8)</td>
<td>0.2598</td>
<td>1.8344</td>
</tr>
<tr>
<td>X(9)</td>
<td>1.1376</td>
<td>0.5717</td>
</tr>
<tr>
<td>X(10)</td>
<td>0.9388</td>
<td>1.5144</td>
</tr>
</tbody>
</table>

Table 30: Results obtained with the adapted Newton method including $P_{45t}$ in the instrumentation. $T_{4t}$ value calculated: 1,349.2 $K$ in Case 1 and 1,302.9 $K$ in Case 2.

Several thousands of cases were solved to prove the robustness of the method with degradations up to a 2%. In the previous table only two of them are shown for the sake of simplicity. It was always randomly selected both initial and exact engine degradations. Convergence was always reached in about 4 iterations, dedicating 5.0 CPU seconds until convergence. In the unlikely event that no acceptable outcome would have been obtained, then the iterative process would have been repeated allowing for a higher number of iterations until convergence (but that situation never occurred up to an 8% of degradations after the upgrade).
Few cases with lower accuracy than expected were identified and repeated accordingly to obtain a better result, with positive results. The accuracy obtained for the $T_{4t}$ values was in the order of $1 \cdot 10^{-4}$ K, and even lower.

The correlations in the troublesome components disappeared completely in the calculations in which the new probe was taken into consideration. This circumstance allowed to obtain overall better accuracies in the solutions and opened the door to play with the balance between accuracy and CPU time, as some more margin of accuracy was obtained. In the case a relevant reduction of computational time could be obtained by renouncing to “excellent” degrees of accuracy for the results, given the accuracy potentially achievable by the sensors, then that possibility should be contemplated.

The improvement in the results could be better understood with the information provided in Figure 107. The chart represents results obtained from 100 random cases considering degradations up to a 2%. The differences of accuracy that were found before are clearly mitigated with the new sensor. All the components counted, after the upgrade in the instrumentation, with very homogeneous error average values, between $1 \cdot 10^{-4}$ and $1 \cdot 10^{-5}$. Similar situation was found with TIT. Few outliers counted with values above $1 \cdot 10^{-4}$ in any component, or below $1 \cdot 10^{-6}$. This scenario would make possible a reduction in the number of iterations to be closer to real-time applications, given the existing margin in accuracy.

With that intention in mind, the following charts were prepared, by limiting the number of iterations allowed. Figure 108 shows the errors obtained in 100 random cases when allowing a maximum of 3 iterations, needing less than 4 CPU seconds in average per case, and keeping errors below $1 \cdot 10^{-2}$. Figure 109 provides a similar information for another 100 random cases but allowing 2 iterations this time (average CPU time was around 2.6 seconds, while errors were below $1 \cdot 10^{-1}$), and finally Figure 110 is dedicated to 100 random cases in which only 1 iteration was performed in an average time of 1.3 CPU seconds, but with errors exceeding $1 \cdot 10^{-1}$. Next 3 charts in Figure 111, Figure 112, and Figure 113 contain same information but considering smaller deterioration changes (up to 0.1%) between initial and final state. These results evince this methodology will be close to be a real-time engine monitoring application. It would be a matter of deciding when a shorter version of the adapted Newton should be used instead of the full version.
Figure 108: Results from 100 random cases, initial and final degradations up to a 2%, allowing 3 iterations.

Figure 109: Results from 100 random cases, initial and final degradations up to a 2%, allowing 2 iterations.

Figure 110: Results from 100 random cases, initial and final degradations up to a 2%, allowing 1 iteration.
Figure 111: Results from 100 random cases, degradations up to a 1%, increments up to 0.1%, allowing 3 iterations.

Figure 112: Results from 100 random cases, degradations up to a 1%, increments up to 0.1%, allowing 2 iterations.

Figure 113: Results from 100 random cases, degradations up to a 1%, increments up to 0.1%, allowing 1 iteration.
Table 31 shows results of other two representative cases in which random initial and exact conditions were considered for degradations up to a 20%, situation that would correspond with a serious deterioration in the engine. An average number of 8 iterations were needed, dedicating an average of 10.3 CPU seconds to meet the convergence criteria. Some cases took more than 20 iterations until the sufficient degree to accuracy in the solution was obtained, circumstance that evinces some routes between initial and final engine health conditions were more challenging than others for the adapted Newton method. As the degradations are higher, the chances to be close to non-appropriate maps’ regions are higher too.

<table>
<thead>
<tr>
<th>X</th>
<th>Case 1</th>
<th>Case 2</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>$X_0$</td>
<td>$X_F$</td>
</tr>
<tr>
<td>X(1)</td>
<td>0.5998</td>
<td>16.2623</td>
</tr>
<tr>
<td>X(2)</td>
<td>10.7133</td>
<td>7.6661</td>
</tr>
<tr>
<td>X(3)</td>
<td>1.7415</td>
<td>12.3456</td>
</tr>
<tr>
<td>X(4)</td>
<td>16.0418</td>
<td>11.5099</td>
</tr>
<tr>
<td>X(5)</td>
<td>19.7829</td>
<td>10.6010</td>
</tr>
<tr>
<td>X(6)</td>
<td>1.3389</td>
<td>5.5014</td>
</tr>
<tr>
<td>X(7)</td>
<td>18.7880</td>
<td>4.9726</td>
</tr>
<tr>
<td>X(8)</td>
<td>0.3636</td>
<td>9.0328</td>
</tr>
<tr>
<td>X(9)</td>
<td>13.6768</td>
<td>4.5543</td>
</tr>
<tr>
<td>X(10)</td>
<td>15.6747</td>
<td>16.0890</td>
</tr>
</tbody>
</table>

Table 31: Results obtained with the adapted Newton method including $P_{45r}$, with degradations up to 20%. $T_{4t}$ calculated: 1, 486.5 $K$ in Case 1 and 1, 454.9 $K$ in Case 2. The accuracy for $T_{4t}$ was much below ~0.1 $K$.

Figure 114 shows results from 100 cases with a similar degree of accuracy to the ones obtained in cases that counted with similar degradations. Dispersion found was still limited with degradations up to a 20%. When the adapted Newton was used to find a solution inside an acceptable region the results counted with an excellent level of accuracy. The difference comparing with the previous example is that the chances of finding a solution which is outside the acceptable region of the engine’s model are higher. Working with degradations of this magnitude led to cases in which several tries, using different initial conditions, were needed to reach convergence. Those cases implied dealing with points “in the border” of the model’s acceptable region. Some other cases could not reach convergence after trying multiple times from different initial conditions. The issue was not related to the initial conditions, as they were tested by trying multiple cases with initial degradations up to a 70%, and final conditions up to 9%, that were always successful. The problem had to do with the approach to the final point. When a problematic point was found while running random cases, it was not possible for the Newton to reach it with enough accuracy. Even when the technique was capable to get relatively close to it, it was not enough to meet the termination criteria, reaching solutions with errors of higher orders of magnitude than usual (1 · 10$^{-1}$ and higher).
Figure 114: Results from 100 random cases, initial and final degradations up to a 20%, involving an average of 8 iterations until convergence when the solution was inside acceptable regions for the different maps.

Regarding these potential situations, in which there might be issues to find a valid solution with the desired level of accuracy (and it must be recalled that such cases were only detected for degradations above 9% so, statistically speaking, that would affect to a minority of cases), some support from PROOSIS® could be invoked to complete the diagnostic of the engine. The messages and warnings that could be configured in that SW when moving outside acceptable regions, or nearby their borders, should be included in the methodology, contributing to identify faulty components, or other reason causing a punctual lack of convergence.

5.5.3. - Example 17

The robustness of the method was challenged with much higher degradations, representative of catastrophic failures in the engine, getting in certain cases excellent results, as it is exposed in Table 32. However, this was not the norm.

<table>
<thead>
<tr>
<th>X</th>
<th>Case 1</th>
<th>Case 2</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>$X_0$</td>
<td>$X_0$</td>
</tr>
<tr>
<td>X(1)</td>
<td>49.4870</td>
<td>21.8466</td>
</tr>
<tr>
<td>X(2)</td>
<td>9.1838</td>
<td>15.2149</td>
</tr>
<tr>
<td>X(3)</td>
<td>43.0828</td>
<td>14.5430</td>
</tr>
<tr>
<td>X(4)</td>
<td>1.6316</td>
<td>12.1258</td>
</tr>
<tr>
<td>X(5)</td>
<td>16.5979</td>
<td>46.8342</td>
</tr>
<tr>
<td>X(6)</td>
<td>37.4373</td>
<td>43.0095</td>
</tr>
<tr>
<td>X(7)</td>
<td>32.2183</td>
<td>19.8614</td>
</tr>
<tr>
<td>X(8)</td>
<td>8.4619</td>
<td>23.9710</td>
</tr>
<tr>
<td>X(9)</td>
<td>47.6103</td>
<td>28.2498</td>
</tr>
<tr>
<td>X(10)</td>
<td>27.1635</td>
<td>24.4810</td>
</tr>
</tbody>
</table>

Table 32: Results obtained with the adapted Newton-like method including $P_{45\%}$, with degradations up to 50%. $T_{4\%}$ calculated: 1,393.7 K in Case 1 and 1,430.6 K in Case 2. The accuracy for $T_{4\%}$ was well below ~0.1 K.
The adapted Newton technique found more problems to reach convergence, circumstance that was expected given the previous cases. However, results obtained were reasonably good, after several hundreds of cases calculated, especially bearing in mind that the combination of such random initial and final conditions would be certainly impossible in the practice, being this case selection criterion clearly overdemanding for the method. Despite this consideration, a pair of results were shown in the previous tables, as a sample of the hundreds of cases solved anew.

In this sense, the average of required iterations until convergence, when considering degradations up to a 50%, was 13 for the cases that could be solved, consuming about 17.0 CPU seconds. Figure 115 shows how the dispersion grew considerably in the errors of the solutions’ components comparing with other cases run before with lower degradation levels. Most of average errors were typically below 1 \cdot 10^{-2}, but certain outliers moved some errors up to 1 \cdot 10^{-1}, or higher.

The number of iterations was limited to 30, to avoid to the algorithm getting stuck in pathologic cases, excessively time consuming. That restriction left without a definitive solution to a significant number of cases (even when they ended close to the exact values), which means that for such level of degradations, the routes between different health conditions become considerable troublesome for the method (and potentially outside acceptable maps’ regions).

Almost the 50% of the cases that were initiated found problems after 30 iterations. When the maximum number of allowed iterations was exceeded, it was automatically tried a different combination of initial conditions, instead of allowing a higher number of iterations until convergence. Those problematic cases could be then tried again from a different initial point, allowing for more iterations. As it was previously explained for the cases up to a 20%, some pathologic cases never reached convergence because of being outside acceptable regions.

However, even if the method was capable to solve numerically the problem, the obtained solution would need of careful evaluation given the remote areas in the maps that were used to get to such outcome. The reality is the effectiveness of the
method is compromised under such degradation conditions. This circumstance reflects the problems that appeared when exploring unusual zones in the performance maps of the main engine components. Figure 116 shows the distribution of iterations needed, with degradations up to a 20% and 50%, respectively, for cases inside acceptable regions. The bars represent the percentages in between ranges, of a width of 3 iterations each. There is dispersion for high degradations as indicated in Figure 115. Working with more usual degradations, most of cases were solved in a specific range of iterations (the lowest).

With 2% degradation levels, the 100% of the cases were solved within three or four iterations (that is why that distribution chart was not included, as it would be compound by only one bar with the totality of solved cases). Up to 8% no problems whatsoever were detected after the upgrade with P₄₅⁵-

Figure 116: Distribution of iterations after 100 cases solved, for unusual (20%) or extreme (50%) degradations.

5.5.4. - Example 18

Additionally, the SQP method was tested with the new sensor, as it is shown in the following Figure 117. Correlations disappeared, and accuracy improved. Both forward and centered differences were tested to see differences between them.

Figure 117: Charts and data from two random cases (20% degradation) where the value of $T_{4t}$ was calculated. First case was run with forward differences and the second one with centered differences. Components of vectors $X_0$ (initial) and $X_E$ (exact) are shown in the associated tables.
In the previous figure, maintaining the maximum degradations up to a 20%, it is possible to see the difference in function evaluations with forward (first case) and centered differences (second one). With the forward differences the computational times were in the order of 2 to 3 CPU minutes in average. Meanwhile with centered differences the times were in the order of 4 to 5 minutes. The double number of calls to PROOSIS® was clearly detected.

5.5.5. - Example 19

Degradations up to a 50% were tried as well, always with centered differences this time. The SQP was capable to solve several random cases as well once the new sensor was included in the model. Computational times were somehow higher comparing with previous cases (up to a 20%), accuracies obtained were typically lower, and a good portion of the cases run did not provided a satisfactory result once the termination criteria were met, evincing certain lack of capability to succeed like the adapted Newton method did when dealing with high deterioration conditions.

![Image](image.png)

**Figure 118:** Charts and data from two random cases (50% degradation) where the value of $T_{4t}$ was calculated. All the cases with this level of degradations were run with centered differences.

Regarding the improvements in time obtained with GAs after the inclusion of the new $P_{45t}$ sensor, the change contributed positively to reduce the CPU times given it was necessary to call PROOSIS® only once every time the OF had to be obtained. However, the reduction was limited, given the inherent exploratory nature of the technique, being the complete computing times still in the order of several hours. The number of necessary calls to the OF, trying to keep past accuracy levels, was like the number obtained in the previous cases in which 2 TIT values were considered. So, the benefit in terms of CPU time were only the result of the reduction of calls to PROOSIS® in one half. With this computational efficiency in mind, the conclusion is this technique would be left exclusively for those cases in which the rest of techniques are not capable to find the solution to the engine’s inverse problem, and only a direct exploration of the solution space seemed to be the way to succeed.
5.6.- Results obtained with the ROM and the application of HOSVD

The next tables and figures show results obtained with the information from the tensor calculated to replace PROOSIS®, for the typical cruise conditions selected for the engine’s operation. The different cases were solved with the adapted Newton method to try to obtain the fastest times possible, given the experience with the full model. Several sets of Core Tensors and associated matrices, meaning different sizes, relative errors, and compression ratios, as it was explained in the previous chapter, were obtained by means of the HOSVD technique.

The different Core Tensors were tested more than one hundred times each with random cases, the same way it was done with PROOSIS®, to obtain representative results, statistically speaking, regarding accuracy, computational times, and effectiveness. In this sense, Figure 119 indicates the relative position of the different Core Tensors in the “error-compression” chart provided by TENSORLAB®, already introduced in the previous chapter. The configuration in the corner of the L-curve, often considered an optimum commitment between relative error and compression ratio (based on the charts of the different singular values) meant a compression of more than a 99% comparing with the original tensor. The accuracy obtained with it was considered insufficient as it will be now explained.

The ROM was used to obtain not only the degradations, but also the TIT, using two sets of data at different TIT values, by working with one tensor without the additional pressure probe that was suggested before to improve the CN of the problem, aiming to keep dimensions in the tensor to the minimum necessary.

[Diagram: Location of different Core Tensors used to solve the problem in the “error-compression” chart. ROM.

Figure 119: Location of different Core Tensors used to solve the problem in the “error-compression” chart. ROM.]
Table 33 provides a summary of the results obtained after using the different Core Tensors, indicated in Figure 119, which deserves some comment given the evolution of the different parameters contained on it:

<table>
<thead>
<tr>
<th>Option</th>
<th>Core Size</th>
<th>Compression Ratio</th>
<th>Relative Error</th>
<th>Cases</th>
<th>Avg. Time (s)</th>
<th>Avg. Abs. Error</th>
<th>Avg. Iterations</th>
<th>Effectiveness</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>[10 18 3 3 3 3 3 3 3 3]</td>
<td>100.00%</td>
<td>0.0E+00</td>
<td>110</td>
<td>41.01</td>
<td>6.6E-02</td>
<td>3.07</td>
<td>87.27%</td>
</tr>
<tr>
<td>2</td>
<td>[9 18 3 3 3 3 3 3 3 3]</td>
<td>90.00%</td>
<td>8.9E-09</td>
<td>110</td>
<td>39.62</td>
<td>7.3E-02</td>
<td>3.36</td>
<td>97.27%</td>
</tr>
<tr>
<td>3</td>
<td>[8 18 3 3 3 3 3 3 3 3]</td>
<td>80.00%</td>
<td>7.8E-07</td>
<td>110</td>
<td>46.60</td>
<td>9.0E-02</td>
<td>4.27</td>
<td>94.55%</td>
</tr>
<tr>
<td>4</td>
<td>[7 18 3 3 3 3 3 3 3 3]</td>
<td>70.00%</td>
<td>1.6E-06</td>
<td>110</td>
<td>49.36</td>
<td>1.5E-01</td>
<td>5.30</td>
<td>88.18%</td>
</tr>
<tr>
<td>5</td>
<td>[7 17 3 3 3 3 3 3 3 3]</td>
<td>66.12%</td>
<td>3.3E-06</td>
<td>110</td>
<td>47.85</td>
<td>1.5E-01</td>
<td>5.35</td>
<td>87.27%</td>
</tr>
<tr>
<td>6</td>
<td>[6 15 3 3 3 3 3 3 3 3]</td>
<td>50.00%</td>
<td>9.9E-06</td>
<td>110</td>
<td>107.46</td>
<td>2.6E-01</td>
<td>16.20</td>
<td>80.00%</td>
</tr>
<tr>
<td>7</td>
<td>[6 14 3 3 3 3 3 3 3 3]</td>
<td>31.11%</td>
<td>2.2E-05</td>
<td>110</td>
<td>164.90</td>
<td>3.0E-01</td>
<td>17.70</td>
<td>79.39%</td>
</tr>
<tr>
<td>8</td>
<td>[4 14 3 3 3 3 3 3 3 3]</td>
<td>20.74%</td>
<td>3.0E-05</td>
<td>110</td>
<td>85.73</td>
<td>3.9E-01</td>
<td>12.07</td>
<td>97.27%</td>
</tr>
<tr>
<td>9</td>
<td>[4 10 3 3 3 3 2 3 3 3]</td>
<td>9.88%</td>
<td>4.8E-05</td>
<td>110</td>
<td>38.19</td>
<td>4.0E-01</td>
<td>11.68</td>
<td>98.18%</td>
</tr>
<tr>
<td>10</td>
<td>[4 8 3 3 3 2 3 3 2 3]</td>
<td>5.27%</td>
<td>8.2E-05</td>
<td>110</td>
<td>21.16</td>
<td>4.1E-01</td>
<td>11.82</td>
<td>94.55%</td>
</tr>
<tr>
<td>11</td>
<td>[3 4 2 2 2 2 2 2 2 2]</td>
<td>0.12%</td>
<td>3.8E-04</td>
<td>110</td>
<td>2.04</td>
<td>5.3E-01</td>
<td>13.05</td>
<td>98.18%</td>
</tr>
</tbody>
</table>

- In all the cases that were run, with different Core Tensors and associated matrices, the same values for δ = 10−3, ε₁ = 1 · 10−2, and ε₂ = 1 · 10−2 were kept during the execution of the adapted Newton to ease the comparison among sizes (wide thresholds, allowing the candidates to solve their cases).
- The compression ratio and the relative error allowed are the coordinates to identify the different sizes in Figure 119 (indicated by green circles).
- It is possible to verify how, as the compression ratio decreases and the size of the Core Tensor gets smaller, the relative error increases, as well as the average absolute error in the different components of the degradations (including the scaled TIT components). This was an expected result.
- What was not so expected was the evolution of the average time per case and the average number of iterations per case. Both have a maximum for a compression ratio of 31.11%. That option was particularly troublesome, counting with 79.39% of effectiveness (ratio between cases successfully solved and the total number of cases run).
- Only the three first options kept average absolute errors in degradations and TIT components below 1 · 10−1. Among them, the one that showed a better behavior was the second one, with a compression ratio of 90%. Figure 120, Figure 121, and Figure 122 show the poor accuracies achieved in some cases.
- The optimum Core Tensor size provided by TENSORLAB® is the one with the fastest execution times but, unfortunately, it is also the one with the highest average absolute error in the degradation and TIT components.
- The removal of information from the original tensor makes possible a faster management of the data contained in the resultant tensor, but it also makes more difficult the convergence for the Newton.
- Given the sensors’ set considered (without P₄₅), correlations between components 7th, 9th, and 10th were present in the results, contributing to reduce the overall accuracy level achievable with this method.

Table 33: Summary of the cases run with the different Core Tensors. ROM.
Figure 120: Results from 100 random cases, using the Core Tensor of size $[7 \ 18 \ 3 \ 3 \ 3 \ 3 \ 3 \ 3 \ 3 \ 3 \ 3 \ 3]$, initial and final degradations up to a 2%, allowing multiple iterations. ROM.

Figure 121: Results from 100 random cases, using the Core Tensor of size $[6 \ 15 \ 3 \ 3 \ 3 \ 3 \ 3 \ 3 \ 3 \ 3 \ 3 \ 3]$, initial and final degradations up to a 2%, allowing multiple iterations. ROM.

Figure 122: Results from 100 random cases, using the Core Tensor of size $[3 \ 4 \ 2 \ 2 \ 2 \ 2 \ 2 \ 2 \ 2 \ 2 \ 2 \ 2]$, initial and final degradations up to a 2%, allowing multiple iterations. ROM.
5.6.1.- Example 20

In Table 34 there is a pair of representative solutions obtained with the second option of the set of analyzed Core Tensors, the one with a 90% of compression ratio (the best candidate found in terms of the combination of timing, accuracy, and effectiveness). As indicated before, possible degradations included in the tensor were limited to up to a 2%. Accuracies in components not affected by the mentioned correlation are typically good, and sometimes excellent, depending on the location of the solution inside the limited portion of the model contained in the tensor.

<table>
<thead>
<tr>
<th>X</th>
<th>Case 1</th>
<th>Case 2</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>$X_0$</td>
<td>$X_E$</td>
</tr>
<tr>
<td>X(1)</td>
<td>0.1579</td>
<td>0.2738</td>
</tr>
<tr>
<td>X(2)</td>
<td>0.9181</td>
<td>0.1154</td>
</tr>
<tr>
<td>X(3)</td>
<td>0.8958</td>
<td>0.3435</td>
</tr>
<tr>
<td>X(4)</td>
<td>0.6789</td>
<td>0.8444</td>
</tr>
<tr>
<td>X(5)</td>
<td>0.5161</td>
<td>0.9033</td>
</tr>
<tr>
<td>X(6)</td>
<td>1.4040</td>
<td>1.0925</td>
</tr>
<tr>
<td>X(7)</td>
<td>1.9012</td>
<td>1.6343</td>
</tr>
<tr>
<td>X(8)</td>
<td>0.9242</td>
<td>0.6643</td>
</tr>
<tr>
<td>X(9)</td>
<td>1.3210</td>
<td>0.2221</td>
</tr>
<tr>
<td>X(10)</td>
<td>1.4332</td>
<td>0.2209</td>
</tr>
</tbody>
</table>

Table 34: Results obtained with random calculated TIT values. $T_{4t} = 1,367.6$ K and 1,395.5 K, respectively, in the first case run. $T_{4t} = 1,514.3$ K and 1,585.1 K, respectively, in the second one. ROM.

When analyzing the errors in another 100 random cases, like the ones in the previous table, the following results in Figure 123 regarding the errors per component and TIT are obtained, which deserve some comment:

![Figure 123](image.png)

Figure 123: Results from 100 random cases, initial and final degradations up to a 2%, involving an average of 4 iterations until convergence. ROM.
The errors remain typically below \(1 \cdot 10^{-1}\), excepting few outliers and the 3 correlated components. Paying attention to which degradation components count with better accuracy, the ones associated to Fan, HPC, and booster seem to be capable to track slow deteriorations of engine’s components with enough detail (one decimal of degradation percentage, meaning for instance the detection of a change from 1.5% to 1.6%). Another important degradation component that will count with enough accuracy to capture small variations would be the one relative to \(\Gamma_{HPT}\), which is crucial to monitor the erosion suffered by the HPT along the operational hours and cycles. Both TIT values are calculated obtaining errors close to \(1 \cdot 10^{-2}\) K, which seems to be enough to monitor them with guarantees.

The degradations associated to the efficiencies in HPT and LPT (i.e., \(\eta_{HPT}\) and \(\eta_{LPT}\)), and the one relative to the mass flow coefficient in LPT (i.e., \(\Gamma_{LPT}\)) are not well monitored (same situation when using ROM or the full model). It must be recalled where the instrumentation without \(P_{45t}\) came from: The CFM56-5A. As it was indicated in Chapter 3, CFMI established a design advantage, becoming the most successful OEM during decades, by limiting the cost associated to the hot section parts in the engines they produced. Meanwhile similar engines manufactured by other OEMs counted with 2 expansion stages in HPT, and 5 stages in LPT, the CFM56 family was characterized by typically incorporating less expansion stages in both HPT (1 stage) and LPT (4 stages). This observation depends, obviously, on the engine variant under consideration, but the historical trend has followed this path, as it was documented. CFMI’s engines are less efficient, because they count with lower expansion capabilities needed to transfer more mechanical power to the compressing stages. But they are shorter, lighter (which translates directly into fuel savings), less complex regarding maintenance, the reparations in the hot section components are considerably less costly comparing with other models from competitors, and several improvement programs on performance retention have been launched by CFMI to maintain during longer periods of time the initially lower efficiencies, contributing this way to reduce the impact of the inherent limitations coming from design. The lack of a relevant sensor like the \(P_{45t}\) probe is probably not so critical for CFM56 engines, and this could explain why it is not installed in any of the different series of the family (hypothesis). In this sense, the proposed methodology in this study will provide a good service to monitor the evolution of the compressors and Fan, the TIT, and the erosion in HPT, so it could be perfectly incorporated to the CFMI maintenance philosophy, regardless the 3 components that are not well captured, the same way it could be incorporated to the monitoring practices from other OEMs more interested in the efficiencies achieved in both HPT and LPT. The flexibility of the proposed methodology counts with such advantage.

Regarding accuracy and CPU times, it would be interesting to understand how accurate this method could be when the iterations allowed is restricted to 3 or less. In this sense, Figure 124, Figure 125, and Figure 126 collect data when random degradations, up to a 2%, are considered for both initial and exact conditions, permitting 3 iterations (meaning 40 CPU seconds in average), 2 iterations (around 27 CPU seconds average), and 1 iteration (still 14 CPU seconds average), respectively. With 3 iterations, average values for errors in non-correlated components, and TIT, are still below \(1 \cdot 10^{-1}\). The average error in the 3 correlated components is above. With less iterations, the accuracy in all the components go worse, as expected. Leaving an extra iteration, up to 4-5, to try to get results like the ones in Table 34 in terms of accuracy, will mean CPU times in the order of 1 minute.
Figure 124: Results from 100 random cases, initial and final degradations up to a 2%, allowing 3 iterations. ROM.

Figure 125: Results from 100 random cases, initial and final degradations up to a 2%, allowing 2 iterations. ROM.

Figure 126: Results from 100 random cases, initial and final degradations up to a 2%, allowing 1 iteration. ROM.
Figure 127: Results from 100 random cases, degradations up to a 1%, increments up to 0.1%, 3 iterations. ROM.

Figure 128: Results from 100 random cases, degradations up to a 1%, increments up to 0.1%, 2 iterations. ROM.

Figure 129: Results from 100 random cases, degradations up to a 1%, increments up to 0.1%, 1 iteration. ROM.
Figure 127, Figure 128, and Figure 129 show random cases run allowing for 3 iterations, 2 iterations, and only one iteration, in case small increases in degradations (0.1% of increment from initial values up to a 1%). These 3 figures, as well as the previous 3, are the counterpart obtained with ROM of those, Figure 100 to Figure 105, obtained with the full model. As it happened when using PROOSIS®, smaller degradations meant also easier conditions for the adapted Newton when using the ROM instead, leading to larger accuracy margins, and better chances to obtain acceptable results with less than 3 iterations per case (2 iterations seem to be acceptable, and even 1 could be fine, depending on the instrumentation), even considering the corelated variables associated with the typical instrumentation set in CFMI engines. This circumstance constitutes a promising scenario for the future, pointing to the right direction to close the existing gap with real-time applications.

However, the CPU times achieved with the HOSVD directly applied to the 12D tensor generated for the engine are still worse than those considered typically for real-time applications, ranging from values around 14 seconds (1 iteration allowed with random cases around 1% degradation and slight variation of 0.1% between initial and final conditions) to 1 minute (around 5 iterations, when running random cases with degradations up to a 2%). Considering the previous results, dedicating less than 3 iterations per case, depending on the magnitude of the degradations to calculate, would imply excessive errors, not only for the corelated components, but also in the rest of H&Q parameters and TIT (cases with degradations up to a 2%).

Ideally, it would be advisable to perform 4 or more iterations to get accuracies like the ones shown in Figure 123, when considering the maximum degradation levels included in the tensor, but faster. The way to get such goal has to do with the way the different solving techniques get access to the information contained in the 12D tensor. By accelerating that data “pointing” process, quickly identifying the desired values in the multidimensional array, it would be possible to reduce the overall time per case. Obviously, working with a tensor of many dimensions, and counting some of them with fine grids like it happens with the TIT, does not help to speed up the computations, but the contrary. The tensor used demanded the inputs regarding 10 degradations, and the TIT, to provide the applicable output of 10 sensor values (direct problem). And, initially, the effects associated to the different degradations, given their independent nature, should be maintained to keep acceptable accuracy levels in the different solved cases.

Today, there are several interesting open research lines chasing this target. Stablishing a hierarchical approach to retrieve data from tensors, like the one associated to the engine’s inverse problem in this thesis, could be the way to reduce CPU times by working with mathematical objects of smaller sizes. Different factorization techniques (see, Liu et al., 2012, [172]), randomized algorithms (see, Ahmadi-Asl et al., 2021, [7]), or the so-called Latent Semantic Analysis (see, Landauer et al., 1998, [165]) are few examples of techniques, applied to different fields, that look for alternative ways to efficiently obtain the required information from tensors, generally of big size (e.g., image processing, data bases, etc.).

In this sense, the tensor used in this study does not count with easily identifiable internal structures, super-symmetries, sparsity, etc. In addition, the involved variables are coupled by different fluid-dynamic and mechanical relationships, which makes difficult to isolate the different effects associated to each degradation. And, finally, the accuracy in the results is essential to try to detect small variations in the health condition of the engine.
All these circumstances contribute to make the engine's inverse problem a true challenge when trying to solve it on real-time with a ROM counting with more than 10 dimensions.

The original method created in 2020 [252] by F. Sastre and A. Velázquez, goes in the direction of getting advantage of the potential redundancy of information contained in the outcome from engine's direct problem. Not all the values of the outcomes are different from each other, numerically speaking, as the problem counts with a non-linear nature, so different combinations of inputs could deliver initially identical outputs. The Latent Semantic Analysis (LSA), taken as reference for the work described in the paper, is widely used today in several web search engines on the Internet, and it relies on the systematic use of the SVD, establishing a hierarchical set of solutions to an inverse problem. The solutions are designed as answers to a question or “query” from the user of the method. In this case, applied to a gas turbine engine, which engine inputs (i.e., degradations and TIT, for instance) are causing certain outputs in the engine (values of sensors at specific stations).

The creators of the method generated several partial tensors (not only 1, like it was done in this thesis), containing each of them the values of the different output variables from the direct performance problem, as a function of the input variables that determined the size of the dimensions of those partial tensors. The tensors contained basically all the useful information from the full model of the engine for the purpose of the problem. So, 10 partial tensors were generated from the same 10 output parameters considered in the direct problem of this thesis, that were evaluated after running multiple times the direct problem (also with PROOSIS®), like in this thesis, counting those 10 tensors finally with 4 dimensions corresponding with the 4 input parameters relative to $\eta_{\text{FAN}}$, $\Gamma_{\text{FAN}}$, $\eta_{\text{LPC}}$, and $\Gamma_{\text{LPC}}$, respectively (only those inputs were considered, for the sake of clarity, in the paper). The target was obtaining the solution to the associated engine’s inverse problem by projecting the data in reduced sub-spaces.

To do so, it was performed an organized unfolding of the information existing in the different tensors associated to each output by selecting a particular dimension for the unfolding, normalizing conveniently the data available in the tensors before (similarly to what was done in Chapter 4). With the information unfolded into smaller sub-tensors, which dimensions were determined with the amount of data in the original 10 tensors, it was possible to define matrices, one per dimension of the original tensor, containing the number of times a given term appeared in the associated sub-tensors. In the case of the paper 4 matrices were generated per output tensor. SVD was then applied to the generated matrices aiming to obtain a representative lower rank version of the original mathematical object.

That was the case when working with only one of the original 10 tensors. The way to proceed when the 10 tensors were considered simultaneously was by piling the information from all the tensors, column-wise, in the previous matrices. And here is when the redundancy of numerical values can appear more easily, contributing thus to improve the performance of the method (i.e., it would be easier to find a potential solution matching the searching criteria).

Typically, the higher the rank of the resultant decomposition, the better the approximation to the reality, but also the higher the CPU effort required to apply the method. However, that was not the case here as the CPU times shown in the paper were always very similar, regardless of the dimensions retained in the SVD, mainly because of the nature of the operations repeatedly performed through the method.
The SVD generated a subspace (bases for that subspace could be generated with the modes associated to the retained Singular Values, see Chapter 4) of lower dimension than the original which will be used to project the original sub-tensors and a “query” vector representing the request of information for the problem resolution (meaning the desired solution). The global “query” vector would be created also by piling the individual queries associated to the different original tensors. That query vector contains the properties of the solution that must match with the information contained in the tensors. However, instead of searching a perfect match directly in the tensors, task that would have implied a longer computational time, the search was done in reduced sub-spaces, by projecting the available information systematically in them, trying to get advantage of the redundant information contained on the tensors, while avoiding missing the target if possible. This is the genius of the proposal, and the philosophy behind the numerical strategy assumed: divide and conquer. Dealing with smaller objects will be beneficial because of scale factors and by the excellent set of existing functions and subroutines available today to deal with matrices and projections. At the end of this elaborated process, 4 sets of sub-tensors (one set per dimension) per original tensor, were generated. They would be used to answer to the request of information contained in the global “query” vector.

This approach allowed to establish a hierarchy of input parameters that could best answer to the request as indicated in the query of the problem. The method could eventually fail when trying to obtain a solution, because it projects systematically the original information in tensors, and the request contained in the query, into reduced sub-spaces that could lead to the disappearance of the solution, if the information removed with the projection contained all the necessary data to obtain such value. In this sense, a cosine-based criterion was used to select the closest projected sub-tensor to the projected query. The more information was retained in the SVD, and the more output tensors were considered simultaneously, the more chances to find redundancies, and the higher the probability of success in finding the right information from the tensors.

It was mentioned in the paper the possibility of using 3 main parameters to calibrate the resulting algorithm, providing thus numerical tools to optimize its effectiveness and accuracy (i.e., a normalizing factor, the number of dimensions in the sub-spaces generated with SVD, and the number of candidate solutions per axis, respectively). Projecting upon too reduced sub-spaces, the information could become “blurry”, leading to miss finally the solution. Creators were nonetheless capable to obtain promising results by retaining just 6 dimensions in the SVD from the original resultant matrices after the unfolding (matrices in the problem had 20 columns and a number several order of magnitude more elevated of rows), and by calibrating the method with a sensitivity analysis. The CPU times needed were in the order of less than $1 \cdot 10^{-3}$ seconds, and the performance showed an improved performance with the dimensionality of the problem. No specific indications on accuracies were given in the paper, which is something essential for the detection of progressive deteriorations in the engine, so this is a factor of the method that still needs to be explored.

Nevertheless, novel and innovative methods like the one explained in [252] show the way to follow in the future when dealing with big tensors: It is necessary to break down the information contained in the multidimensional array, creating ad hoc hierarchical structures as needed to access faster to the information contained
on it, and getting advantage of the existing excellent algorithms for projecting purposes and matrix operations. The method needed of certain calibration, depending on how likely the mentioned redundancies in the outcomes will appear in the different tensors, and it is still to be tested when more inputs are taken into consideration (and not only 4, associated to Fan and booster, as it was done in the original paper). Once that calibration could be completed and systematically applied to the different engines to be monitored, this is a clear candidate to be incorporated to the ROM methodology explored in this thesis. Dealing with partial tensors, instead with a global one, could contribute to boost the calculations, so this option should be carefully evaluated, and incorporated to the presented methodology in the future, when possible.

![Diagram](image1)

**Figure 130:** Schematical concept of the LSA applied to the tensors generated for the aeroengine’s inverse problem solving. This technique constitutes a way to explore in the future to speed up CPU times. ROM.

### 5.7. Comments on the calculations performed and conclusions

Some comments will be done at this point regarding the results obtained from the methodology exposed in this study:

- The 17 different temperatures used for the calculation of the average values in the instrumentation vector components’ scaling process were obtained by dividing the temperature range between 1,300.0 K and 1,650.0 K in 16 equispaced intervals. In all the cases that were run, altitude (35,000 ft or 10,668 m) and Mach number (0.8) remained constant. Degradations were all set to 0% (baseline) when obtaining the scaling averages. At certain point in the study, it was decided to include the value 1,400 K into the TIT grid, as it was considered a relevant TIT value, leading finally to 18 values in the grid.

- The CPU times mentioned in the document were obtained from a standard desktop PC, with a microprocessor Intel Core i7-3770 at 3.4 GHz, with 16 GB RAM memory. The typical capabilities of the processors used in the CS managing gas turbine engines are considerably less powerful (working in the range of MHz, as it was indicated in Chapter 3). Additionally, those CS typically count with considerably limited integrated data storage capabilities. This means that a different computer could be needed to run the algorithms used for this study, if a fast and reliable connectivity to the cloud is not an option (challenging during long transoceanic routes).
• In this sense, the full functionality of PROOSIS® will need, as a minimum to replicate the operations performed for this thesis, of a WIN7 computer (64-bit version), with 4 GB RAM memory, and with a minimum free memory storage capability of 3 GB [235].

• The accuracy obtained by the different techniques needs of some comment as well. It is necessary to highlight that the H&Q parameters are expressed in values around the unity, meanwhile the components in the degradation vector are indicated in percentages. So, a value of a 2% in any of those components would mean a value of 0.98 or 1.02 in its associated H&Q parameter. The mass flow capacities in the turbines evolve towards values higher than the unity because of the erosion and TBC deterioration suffered in the hot section of the engine, the rest of H&Q parameters evolve to lower values than the unity. Therefore, an error level in the order of $\sim 10^{-1}$ for a degradation will affect to its first decimal, but for the associated H&Q parameter that error will be affecting its third decimal. And this could be acceptable, depending on the application, considering the existing accuracy levels reached by the existing instrumentation (and there are many different models of engines active today, from different ages and, potentially, with also different sensing technologies). The format of the data shown in human-machine interfaces (HMI), such as EICAS in an aircraft, must be analyzed as well. Very rarely a magnitude is presented to the pilots with more than two decimals in the screen, even if the stored data could retain values with considerably higher numerical accuracy. In this sense, the results from the sensitivity analysis exposed in Chapter 4 have been used to estimate if the different changes in the degradation vector components would be captured by the selected instrumentation during a typical cruise. Considering the full scales and accuracies, gathered in Table 35, for the different sensors, obtained from the information and references in Chapters 3 and 4 (see Igie et al., 2016, [144]), it was analyzed if variations of at least a 1% in the different 10 degradations, keeping the rest constant, ceteris paribus, could be detected by the sensors. Results are shown in Table 36, where its 10 columns, representing the variations in the instrumentation for the 10 different degradation vector components when a change of a 1% occurs in any of them, count with 1 green cell at least, meaning that the associated sensor to that row in the column relative to a particular degradation is capable to detect such 1% change in that component. Red cells indicate not enough accuracy in the sensor to capture the 1% changes, meanwhile green cells indicate the opposite. A similar exercise was done for degradation changes of 0.5%, which results are shown in Table 37 (just by applying linear changes to the values in the previous table, for the sake of simplicity). And the selected instrumentation is still valid for such level of accuracy. However, it will not be enough to detect changes of 0.1% in the degradations (i.e., sensors will capture variation in efficiencies and mass flow capacities from 0.985 to 0.980, but not from 0.985 to 0.984). Keeping accuracies of $1 \cdot 10^{-1}$ in the calculations of degradations seems to be enough conservative for the methodology proposed, given the available instrumentation nowadays. In any case, improvements in sensors’ technology will bring higher demands in terms of numerical accuracies and it is convenient to be capable to guarantee even higher levels, as it was done with the adapted Newton technique.
Table 35: Instrumentation’s considered full scales and accuracies, based on the information detailed in Chapter 3 and Chapter 4. The ranges have been selected to fit exactly to engine’s operation to avoid accuracy losses. Such selection could be refined, either by design or by calibration, to try to gain some more accuracy.

<table>
<thead>
<tr>
<th>SENSOR</th>
<th>TECHNOLOGY</th>
<th>FULL SCALE CONSIDERED</th>
<th>ACCURACY CONSIDERED</th>
</tr>
</thead>
<tbody>
<tr>
<td>P_{13t}</td>
<td>MEMS (FADEC mounted). Silicon-based components.</td>
<td>180,000 Pa</td>
<td>0.1% FS, or ±90 Pa</td>
</tr>
<tr>
<td>P_{25t}</td>
<td>MEMS (FADEC mounted). Silicon-based components.</td>
<td>530,000 Pa</td>
<td>0.1% FS, or ±265 Pa</td>
</tr>
<tr>
<td>P_{3t}</td>
<td>MEMS (FADEC mounted). Silicon-based components.</td>
<td>3,300,000 Pa</td>
<td>0.1% FS, or ±1650 Pa</td>
</tr>
<tr>
<td>T_{25t}</td>
<td>RTD (Pt-100).</td>
<td>500 K</td>
<td>0.6% FS, or ±1.5 K</td>
</tr>
<tr>
<td>T_{3t}</td>
<td>Thermocouple (K-Type).</td>
<td>800 K</td>
<td>0.4% FS, or ±1.6 K</td>
</tr>
<tr>
<td>T_{45t}</td>
<td>Thermocouple (K-Type).</td>
<td>1200 K</td>
<td>0.4% FS, or ±2.4 K</td>
</tr>
<tr>
<td>T_{ST}</td>
<td>Thermocouple (K-Type).</td>
<td>900 K</td>
<td>0.4% FS, or ±1.8 K</td>
</tr>
<tr>
<td>N_{H}</td>
<td>Magnetic Pickup.</td>
<td>11,000 rpm</td>
<td>0.5% FS, or ±28 rpm</td>
</tr>
<tr>
<td>N_{L}</td>
<td>Magnetic Pickup / Hall effect.</td>
<td>4,800 rpm</td>
<td>0.5% FS, or ±12 rpm</td>
</tr>
<tr>
<td>W_{f}</td>
<td>Turbine flow-meter.</td>
<td>2.2 kg/s</td>
<td>0.5% FS, or ±0.006 kg/s</td>
</tr>
</tbody>
</table>

Table 36: Verification of instrumentation when capturing changes of 1% in the different H&Q parameters. There is always at least 1 green cell per column, meaning at least 1 sensor will capture the change in that degradation.

<table>
<thead>
<tr>
<th>SENSOR</th>
<th>( \bar{\eta}_{\text{HAN}} ) (( \Delta = 1% ))</th>
<th>( \bar{\eta}_{\text{HAN}} ) (( \Delta = 1% ))</th>
<th>( \bar{\eta}_{\text{HPC}} ) (( \Delta = 1% ))</th>
<th>( \bar{\eta}_{\text{HFC}} ) (( \Delta = 1% ))</th>
<th>( \bar{\eta}_{\text{LFP}} ) (( \Delta = 1% ))</th>
<th>( \bar{\eta}_{\text{LFT}} ) (( \Delta = 1% ))</th>
<th>( \bar{\eta}_{\text{HPT}} ) (( \Delta = 1% ))</th>
<th>( \bar{\eta}_{\text{HPT}} ) (( \Delta = 1% ))</th>
</tr>
</thead>
<tbody>
<tr>
<td>P_{13t} (Pa)</td>
<td>93.6</td>
<td>20.7</td>
<td>299.1</td>
<td>12.5</td>
<td>172.3</td>
<td>20.0</td>
<td>252.6</td>
<td>23.5</td>
</tr>
<tr>
<td>P_{25t} (Pa)</td>
<td>924.1</td>
<td>2715.6</td>
<td>1414.2</td>
<td>294.6</td>
<td>824.7</td>
<td>2407.1</td>
<td>2394.1</td>
<td>875.2</td>
</tr>
<tr>
<td>P_{3t} (Pa)</td>
<td>5620.5</td>
<td>8920.8</td>
<td>16434.5</td>
<td>11498</td>
<td>7037.2</td>
<td>9534.8</td>
<td>15564.7</td>
<td>11634.5</td>
</tr>
<tr>
<td>T_{25t} (K)</td>
<td>0.02</td>
<td>1.99</td>
<td>0.63</td>
<td>0.16</td>
<td>0.54</td>
<td>1.46</td>
<td>1.34</td>
<td>0.57</td>
</tr>
<tr>
<td>T_{3t} (K)</td>
<td>0.084</td>
<td>1.75</td>
<td>0.53</td>
<td>0.47</td>
<td>0.53</td>
<td>1.33</td>
<td>3.56</td>
<td>2.21</td>
</tr>
<tr>
<td>T_{45t} (K)</td>
<td>0.02</td>
<td>0.22</td>
<td>0.99</td>
<td>0.55</td>
<td>0.065</td>
<td>0.12</td>
<td>4.22</td>
<td>2.40</td>
</tr>
<tr>
<td>T_{ST} (K)</td>
<td>0.11</td>
<td>0.11</td>
<td>1.15</td>
<td>0.42</td>
<td>0.25</td>
<td>0.10</td>
<td>3.48</td>
<td>1.74</td>
</tr>
<tr>
<td>N_{H} (rpm)</td>
<td>0.58</td>
<td>5.76</td>
<td>114.73</td>
<td>69.63</td>
<td>0.04</td>
<td>2.97</td>
<td>127.90</td>
<td>20.62</td>
</tr>
<tr>
<td>N_{L} (rpm)</td>
<td>0.49</td>
<td>17.56</td>
<td>20.65</td>
<td>0.86</td>
<td>11.97</td>
<td>1.39</td>
<td>17.58</td>
<td>1.63</td>
</tr>
<tr>
<td>W_{f} (kg/s)</td>
<td>0.0044</td>
<td>0.0046</td>
<td>0.0117</td>
<td>0.000063</td>
<td>0.0062</td>
<td>0.0057</td>
<td>0.0069</td>
<td>0.0032</td>
</tr>
</tbody>
</table>

Table 37: Verification of instrumentation when capturing changes of 0.5% in the different H&Q parameters. There is always at least 1 green cell per column, so at least 1 sensor will capture changes in that degradation.

<table>
<thead>
<tr>
<th>SENSOR</th>
<th>( \bar{\eta}_{\text{HAN}} ) (( \Delta = 0.5% ))</th>
<th>( \bar{\eta}_{\text{HAN}} ) (( \Delta = 0.5% ))</th>
<th>( \bar{\eta}_{\text{HPC}} ) (( \Delta = 0.5% ))</th>
<th>( \bar{\eta}_{\text{HFC}} ) (( \Delta = 0.5% ))</th>
<th>( \bar{\eta}_{\text{LFP}} ) (( \Delta = 0.5% ))</th>
<th>( \bar{\eta}_{\text{LFT}} ) (( \Delta = 0.5% ))</th>
<th>( \bar{\eta}_{\text{HPT}} ) (( \Delta = 0.5% ))</th>
<th>( \bar{\eta}_{\text{HPT}} ) (( \Delta = 0.5% ))</th>
</tr>
</thead>
<tbody>
<tr>
<td>P_{13t} (Pa)</td>
<td>46.8</td>
<td>10.4</td>
<td>148.6</td>
<td>6.3</td>
<td>86.2</td>
<td>10.0</td>
<td>126.8</td>
<td>11.8</td>
</tr>
<tr>
<td>P_{25t} (Pa)</td>
<td>462.1</td>
<td>1357.8</td>
<td>707.1</td>
<td>147.3</td>
<td>412.4</td>
<td>1263.6</td>
<td>1197.1</td>
<td>437.6</td>
</tr>
<tr>
<td>P_{3t} (Pa)</td>
<td>2810.3</td>
<td>4460.4</td>
<td>8217.3</td>
<td>574.4</td>
<td>3518.6</td>
<td>4767.4</td>
<td>7782.4</td>
<td>5817.3</td>
</tr>
<tr>
<td>T_{25t} (K)</td>
<td>0.01</td>
<td>0.10</td>
<td>0.32</td>
<td>0.08</td>
<td>0.27</td>
<td>0.73</td>
<td>0.67</td>
<td>0.29</td>
</tr>
<tr>
<td>T_{3t} (K)</td>
<td>0.002</td>
<td>0.88</td>
<td>0.27</td>
<td>0.24</td>
<td>0.27</td>
<td>0.67</td>
<td>1.78</td>
<td>1.11</td>
</tr>
<tr>
<td>T_{45t} (K)</td>
<td>0.01</td>
<td>0.11</td>
<td>0.50</td>
<td>0.28</td>
<td>0.003</td>
<td>0.06</td>
<td>2.11</td>
<td>1.20</td>
</tr>
<tr>
<td>T_{ST} (K)</td>
<td>0.06</td>
<td>0.06</td>
<td>0.58</td>
<td>0.21</td>
<td>0.13</td>
<td>0.05</td>
<td>1.74</td>
<td>0.87</td>
</tr>
<tr>
<td>N_{H} (rpm)</td>
<td>0.29</td>
<td>2.88</td>
<td>57.37</td>
<td>34.82</td>
<td>0.02</td>
<td>1.49</td>
<td>63.95</td>
<td>10.31</td>
</tr>
<tr>
<td>N_{L} (rpm)</td>
<td>3.25</td>
<td>8.78</td>
<td>10.43</td>
<td>0.43</td>
<td>5.99</td>
<td>0.70</td>
<td>8.79</td>
<td>0.82</td>
</tr>
<tr>
<td>W_{f} (kg/s)</td>
<td>0.0022</td>
<td>0.0023</td>
<td>0.0059</td>
<td>0.0000015</td>
<td>0.0031</td>
<td>0.0029</td>
<td>0.0035</td>
<td>0.0016</td>
</tr>
</tbody>
</table>
• This consideration leads naturally to the previously mentioned commitment between accuracy and computational time. The adapted Newton method reached accuracy levels, for usual degradations up to a 2%, in between $10^{-4}$ and $10^{-7}$, after typically 3 or 4 iterations, needing 4.0 CPU seconds in average. With less iterations, maybe 2 to 3, the reduction in CPU time would lead to values that will range in between 1.0 to 2.0 CPU seconds, depending on the case. It must be highlighted that there is an increase in the computational time required to achieve convergence when the demanded accuracy becomes more exigent. This is something that was observed in all the techniques previously analyzed, but more clearly in the optimization techniques: GAs and SQP. While the initial stages of each run were very efficient, and the most promising solution region was identified quickly, the last stages, in which the search was refined to achieve higher accuracies, became a very slow process. In the case of GAs, the search process must conduct to explore the surrounds of any promising candidate without any information that could be leveraged to speed up the search (i.e., the derivatives of the associated OF). Additionally, the higher the variation in degradations, between initial and final conditions per case run, the more iterations will be needed to get an acceptable solution. However, the usual changes in degradation (i.e., always to worse under normal operating conditions) are typically small and that implies that 1 or 2 iterations would be sufficient most of the times when using the adapted Newton. Using a ROM, that would imply times of about 14 to 26 CPU seconds, which could be perfectly acceptable for engine health monitoring purposes (not for pure diagnostics and prompt failure detection). And the accuracies would be aligned with the capabilities of the typical available instrumentation in commercial aviation (it was already mentioned that engines used for military applications count with different sensor specifications, so not all the available sensors installed in aircraft engines share the same characteristics).

• In this sense, the concept of real-time SW counts with different meanings, definitions, and interpretations depending on the technical context where it is used. A CS working in real-time conditions could be described, for instance, as a controller that receives data, process the information, and returns an outcome, quickly enough to affect its environment at that time (and preventing issues in the overall system). That description lacks a clear time scale to decide what is a real-time application, and what is not, leaving finally the decision on the applicable time scale as a function of the associated process. For simulation purposes, it is typically considered that the simulation's clock runs at the same speed then a real system's clock. In process control it typically means management "without significant delay", but this is always subject to the criticality of the application and opens the door to establishing classifications based on such criterion [152]. If deadlines of real-time tasks (even under worst case processing loads) must be respected by the application, then it is typically considered a hard-real-time application. On the other hand, if application deadlines can be managed on average, and that is acceptable for the user, then such application could be considered a soft-real-time application. In both categories, faster and slower speeds of response are contemplated (see Table 38 with some examples of the different categories).
Table 38: Examples of real-time applications based on the criticality of the associated processes, and on their speed of response (see [59] for further details).

<table>
<thead>
<tr>
<th>Soft Criticality</th>
<th>Fast Response</th>
</tr>
</thead>
<tbody>
<tr>
<td>Condition Monitoring</td>
<td>HMI</td>
</tr>
<tr>
<td>Missile Defense Systems</td>
<td>Airbag Control System</td>
</tr>
</tbody>
</table>

A hard-fast system would correspond to the one associated to a critical system, such as an airbag, for instance. On the contrary, a soft-slow system is one in which the deadlines are not so critical and significant tolerance can be permitted on the speed of response. Applications for engine’s health condition monitoring, or trend analysis, are typically included in this category. Frequently, such kind of SW is large and complex, like it happens with PROOSIS®, and some of the solving techniques implemented in MATLAB®. An arbitrary boundary between slow and fast real-time applications is 1 second, which is chosen because technical problems shift, from individual computing issues to overall system aspects, at around that time (see J. Cooling, 2019, [59]). According to this classification, the methodology developed in this study would be framed into the soft-slow real-time applications, but the fast-response will not be far if the accuracy-time commitment is optimized by limiting the number of iterations when the degradations are reduced, as it normally happens during normal operation.

- In this sense, trying to optimize the application of the different techniques explored for this methodology could lead to some sort of hierarchized algorithm, which could execute first the faster techniques, getting advantage of any possible accuracy-time commitment, changing to another one, more explorative, in case the faster ones would fail with a quick try, requiring then some more detail and analysis of the surroundings of the solution point. Most of the cases, according to the results obtained so far, the faster techniques will suffice to provide the right solution in a near-to-real-time basis. Then, when some unexpected event happened involving a troublesome route in the solution space, a more detailed technique could be applied instead. If a damage implying degradations higher than 5% should happen during a real flight, then the most probable reason would be a severe mechanical damage, or auxiliary systems failure, that may imply the loss of the affected engine in the worst-case scenario, being necessary to shut it down and, in that circumstance, there would be time enough later to analyze the situation after landing in an alternative airport. Similar considerations may apply for marine or industrial applications.

- The direct use of ROMs did not improve finally the CPU times that were obtained with the full model. The high dimensionality of the problem, the coupled nature of the data involved, and the required accuracy to capture small degradations during the operation of the engine made unaffordable the obtention of trustable results in less than 14 seconds with the proposed approach when calculating potential degradation increments up to a 2%. However, considering the inherent benefits associated to the use of ROM-
based methods (e.g., the use of tensors avoids costs for SW licensing and hardware requirements, including potentially the ones relative to data storage), and keeping in mind that the main target is tracking small and progressive degradation levels in the different engine components, together with the values of TIT, it is still to be determined if this methodology could be applicable for typical cruises (or for extended operational periods in aeroderivative gas turbines). It is not typically required to store an elevated number of points per flight to capture a slow trend in the degradation of the engine. Few readings per hour (around 100 with the average times obtained with the second option) could suffice for the purpose of the diagnostic and prognostic of the engine based on the current condition. This circumstance invites to consider seriously the use of ROMs for commercial aviation purposes, not being too problematic obtaining results in more than 10 seconds. In this sense, further research and analysis will be done in the future to incorporate novel tensorial techniques, like the one relative to LSA commented in this Chapter, that may appear and could contribute to boost the calculations presented in this study. The key to do so seems to be in finding faster the required information from the existing tensor. SVD-based projections into reduced sub-spaces, together with the use of the powerful available functions and subroutines for matrix operations, could be the way to follow.

- The next 2 summarizing tables, Table 39 and Table 40, contain the most relevant data from the previous examples and explanations given in this chapter. The average times contained on them were double-checked accounting for the number of iterations, generations, function calls, etc., necessary in each technique, and multiplying the total number of calls to the model by the time required by PROOSIS® to solve a direct problem (0.07 CPU seconds), getting to an equal result, as expected. Similar check was done with the ROMs verifying how long it took to solve a direct problem with the tensor (0.49 CPU seconds). In both tables, the typical accuracies and CPU times are indicated for each technique incorporated to the methodology, either by using the full model contained in PROOSIS® or the associated ROM to the typical cruise. The commitment between accuracy and computational time is indicated in certain cases where it was possible to choose (e.g., in the SQP, by using forward or centered differences, or in the adapted Newton by limiting the number of iterations). It seems the technique that will be used most of times will be the adapted Newton, given its effectiveness and fast convergence. The most likely scenario will be having small increments of degradation each flight, circumstance that will contribute to obtain accurate results in short CPU times, by dedicating only 1 iteration when possible (meaning 1.3 CPU seconds with PROOSIS® and 14 CPU seconds with the ROM). The GAs and the SQP would be left for troublesome cases, when the adapted Newton could fail, potentially because of some local issue with the engine’s maps in the surroundings of the solution. The SQP could be a first try to obtain information of the shape of the OF around the desired point and solving the problem when possible, leaving the GAs as a last resource to fully explore the area around the solution in case of finding serious difficulties to get convergence. An organized scheme for the potential hierarchical approach will be shown in the last chapter of the thesis.
### Table 39: Summarizing table for GA and SQP.

<table>
<thead>
<tr>
<th>TECHNIQUE</th>
<th>EXAMPLE NUMBER</th>
<th>TYPICAL ACCURACY IN DEGRADATIONS</th>
<th>TYPICAL CPU TIME</th>
<th>ADDITIONAL INFORMATION</th>
</tr>
</thead>
<tbody>
<tr>
<td>Genetic Algorithm</td>
<td>Example 1</td>
<td>$10^{-3}$</td>
<td>$2 \cdot 10^4$ hours</td>
<td>Global accuracy was affected by correlations in $X(7)$, $X(9)$, and $X(10)$. Those components were in the order of $10^{-3}$, or even $10^{-3}$, 100 individuals. 2 TTF values were necessary because of lack of $F_{TDO}$. Degradations up to 2%.</td>
</tr>
<tr>
<td></td>
<td>Example 2</td>
<td>$10^{-2}$</td>
<td>$4 \cdot 10^4$ hours</td>
<td>Global accuracy was affected by correlations in $X(7)$, $X(9)$, and $X(10)$. Those components were in the order of $10^{-2}$, 1,000 individuals. 2 TTF values were necessary because of lack of $F_{TDO}$. Degradations up to 2%.</td>
</tr>
<tr>
<td></td>
<td>Example 3</td>
<td>$10^{-2}$</td>
<td>$2 \cdot 10^4$ hours</td>
<td>Global accuracy was affected by correlations in $X(7)$, $X(9)$, and $X(10)$. Those components were not clearly affected this time, 240 individuals. 2 TTF values were calculated because of lack of $F_{TDO}$. Degradations up to 2%.</td>
</tr>
<tr>
<td></td>
<td>Example 4</td>
<td>$10^{-3}$</td>
<td>$2 \cdot 10^4$ minutes</td>
<td>Global accuracy was affected by correlations in $X(7)$, $X(9)$, and $X(10)$. Those components were in the order of $10^{-3}$, forward differences. 2 TTF values were necessary because of lack of $F_{TDO}$. Degradations up to 2%.</td>
</tr>
<tr>
<td></td>
<td>Example 5</td>
<td>$10^{-3}$</td>
<td>$3 \cdot 10^4$ minutes</td>
<td>Global accuracy was affected by correlations in $X(7)$, $X(9)$, and $X(10)$. Those components were in the order of $10^{-3}$, forward differences. 2 TTF values were calculated because of lack of $F_{TDO}$. Degradations up to 2%.</td>
</tr>
<tr>
<td>SQP</td>
<td>Example 6 [summary of the two previous examples]</td>
<td>$10^{-3}$</td>
<td>$2 \cdot 10^4$ minutes</td>
<td>Global accuracy was affected by correlations in $X(7)$, $X(9)$, and $X(10)$. Those components were in the order of $10^{-3}$, forward differences. 2 TTF values were calculated because of lack of $F_{TDO}$. Centred differences.</td>
</tr>
<tr>
<td></td>
<td>Example 7</td>
<td>$10^{-4}$</td>
<td>$5 \cdot 10^4$ minutes</td>
<td>Global accuracy was affected by correlations in $X(7)$, $X(9)$, and $X(10)$. Those components were in the order of $10^{-4}$, forward differences. 2 TTF values were calculated because of lack of $F_{TDO}$. Degradations up to 2%.</td>
</tr>
<tr>
<td></td>
<td>Example 8</td>
<td>$10^{-3}$</td>
<td>$8 \cdot 10^4$ minutes</td>
<td>Global accuracy was affected by correlations in $X(7)$, $X(9)$, and $X(10)$. Those components were in the order of $10^{-3}$, forward differences. 2 TTF values were calculated because of lack of $F_{TDO}$. Centred differences.</td>
</tr>
<tr>
<td></td>
<td>Example 18</td>
<td>$10^{-5}$ (med. diff.)</td>
<td>$3 \cdot 10^4$ minutes</td>
<td>No more correlations in $X(7)$, $X(9)$, and $X(10)$ after the incorporation of $F_{TDO}$. Only 1 TTF value was calculated. Degradations up to 20%. Accuracy-time commitment by choosing forward of centred differences.</td>
</tr>
<tr>
<td></td>
<td>Example 19</td>
<td>$10^{-3}$ (cont. diff.)</td>
<td>$6 \cdot 10^4$ minutes</td>
<td>No more correlations in $X(7)$, $X(9)$, and $X(10)$ after the incorporation of $F_{TDO}$. Only 1 TTF value was calculated. Degradations up to 20%. Accuracy variations depending on each case, and the component.</td>
</tr>
</tbody>
</table>

### Table 40: Summarizing table for adapted Newton used with both the full model and ROM.

<table>
<thead>
<tr>
<th>TECHNIQUE</th>
<th>EXAMPLE NUMBER</th>
<th>TYPICAL ACCURACY IN DEGRADATIONS</th>
<th>TYPICAL CPU TIME</th>
<th>ADDITIONAL INFORMATION</th>
</tr>
</thead>
<tbody>
<tr>
<td>Adapted Newton</td>
<td>Example 9</td>
<td>$10^{-5}$</td>
<td>$1 \cdot 10^4$ seconds</td>
<td>Global accuracy was affected by correlations in $X(7)$, $X(9)$, and $X(10)$. Those components were in the order of $10^{-5}$, forward differences. 2 TTF values were necessary because of lack of $F_{TDO}$. Degradations up to 2%.</td>
</tr>
<tr>
<td></td>
<td>Example 10</td>
<td>$10^{-5}$</td>
<td>$1.2 \cdot 10^4$ seconds</td>
<td>Global accuracy was affected by correlations in $X(7)$, $X(9)$, and $X(10)$. Those components were in the order of $10^{-5}$, forward differences. 2 TTF values were calculated because of lack of $F_{TDO}$. Degradations up to 2%.</td>
</tr>
<tr>
<td></td>
<td>Example 11</td>
<td>$10^{-4}$</td>
<td>$1.5 \cdot 10^4$ seconds</td>
<td>Global accuracy was affected by correlations in $X(7)$, $X(9)$, and $X(10)$. Those components were in the order of $10^{-4}$, forward differences. 2 TTF values were necessary because of lack of $F_{TDO}$. Degradations up to 20%.</td>
</tr>
<tr>
<td></td>
<td>Example 12</td>
<td>$10^{-4}$</td>
<td>$2 \cdot 10^4$ seconds</td>
<td>Global accuracy was affected by correlations in $X(7)$, $X(9)$, and $X(10)$. Those components were in the order of $10^{-4}$, forward differences. 2 TTF values were calculated because of lack of $F_{TDO}$. Degradations up to 20%.</td>
</tr>
<tr>
<td>Example 13 [development of previous examples]</td>
<td>$10^{-4}$</td>
<td>$1.5 \cdot 10^4$ seconds</td>
<td>Global accuracy was affected by correlations in $X(7)$, $X(9)$, and $X(10)$. Those components were in the order of $10^{-4}$, forward differences. 2 TTF values were calculated because of lack of $F_{TDO}$. Degradations up to 20%.</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Example 15</td>
<td>$10^{-5}$</td>
<td>$5 \cdot 10^4$ seconds</td>
<td>Global accuracy was affected by correlations in $X(7)$, $X(9)$, and $X(10)$ after the incorporation of $F_{TDO}$. Only 1 TTF value was calculated. Forward differences. Degradations up to 5% and up to 3% with up to 0.1% $\Delta$(X). Several tests to evaluate the impact in accuracy by limiting iterations. Condition Number $\approx 50$. (1x11) Jacobian matrix.</td>
</tr>
<tr>
<td></td>
<td>Example 16</td>
<td>$10^{-4}$</td>
<td>$1 \cdot 10^4$ seconds</td>
<td>No more correlations in $X(7)$, $X(9)$, and $X(10)$ after the incorporation of $F_{TDO}$. Only 1 TTF value was calculated. Forward differences. Degradations up to 20%. (1x11) Jacobian matrix.</td>
</tr>
<tr>
<td></td>
<td>Example 17</td>
<td>$10^{-3}$</td>
<td>$1 \cdot 10^4$ seconds</td>
<td>No more correlations in $X(7)$, $X(9)$, and $X(10)$ after the incorporation of $F_{TDO}$. Only 1 TTF value was calculated. Forward differences. Degradations up to 50%. (1x11) Jacobian matrix.</td>
</tr>
<tr>
<td>ROM + Adapted Newton</td>
<td>Example 20</td>
<td>$10^{-2}$</td>
<td>$1 \cdot 10^4$ minutes</td>
<td>Global accuracy was affected by correlations in $X(7)$, $X(9)$, and $X(10)$. Those components were in the order of $10^{-2}$, forward differences. 2 TTF values were calculated because of lack of $F_{TDO}$. Degradations up to 2%.</td>
</tr>
</tbody>
</table>
6.- APPLICABILITY AND FUTURE WORK

The final chapter of the thesis will be dedicated to the last considerations regarding probably the most interesting part of the study, which is the applicability of the presented methodology to two-spool turbofan engines working in real situations (and, by extension, to any kind of gas turbine engine which model could be prepared in the future), how a potential preliminary business case could be prepared to justify the commercial interest on the services that may be associated to its use, and regarding, finally, to the future work that should contribute to continue, maintain updated, and improve when possible this research line initiated, several years ago, with the first steps given in gas turbine performance analysis and health condition estimation.

6.1. Applicability of the methodology

About the applicability of the methodology previously explained to real cases, it is necessary to emphasize that the intention will be, once it was studied under which conditions the analyzed techniques would typically obtain a valid solution, to solve as fast as technically feasible the inverse problem for the calculation of the degradation status of the engine, given the data provided by the instrumentation. Statistically speaking, during most of the operational time of the engine, the degradations experienced by its main components during a flight will be considerably lower than a 2%. Otherwise, in few days of continuous operation, the engine would have to be sent to repair to the workshop (see works from NASA [221], [148], and [17], in this sense, for further reference). Figure 131 illustrates this circumstance for a typical turbofan engine model, like the ones mentioned in this thesis. Scheduled maintenance (e.g., hot section repairs) will contribute to keep during a longer time the overall condition of the engine inside acceptable values. Some other interventions would have to be necessarily performed based on engine’s health condition. Some degradation level will be unrecoverable because several key engine’s components will remain during the whole (or most of) engine’s service life.

![Figure 131: Evolution of the degradation in a typical turbofan engine (P&W JT9D-7A) with flight cycles, considering no repair whatsoever, and the improvement after a hot section repair (chart obtained from [221]).](chart.png)
The exact degradation rate will depend on the conditions under which the engine is operated but, certainly, meanwhile the degradation values for the different engine components are below 5%, it will be inside a range in which the explored techniques will typically find a docile behavior in the variables or, at least, a behavior without problematic changes of tendency and within fully trustable regions in the different components’ maps. Most of time, each degradation value will follow a soft variation (to worse) until the end of the Remaining Useful Life (RUL) of the associated component under consideration. The adapted Newton technique could be applied extensively and, preferably, not reaching an excessively high accuracy level to try to deliver the value of $\bar{X}$ in one to three seconds maximum. ROM-based methods could be also valid alternatives to avoid excessive implementation costs once the typical operational conditions are known with enough level of confidence.

Then, if any specific case needs of some more iterations, either because the final degradation suffered by the machine components is higher than expected, or maybe because the case involves a numerically troublesome route between the previous engine health condition and the new one, then more iterations should be applied until convergence. However, this is an unlikely situation under 5% of degradation in the components, as per the experience gathered along this research.

Finally, in an unusual situation in which the solution could not be obtained by the adapted Newton technique, then alternative well-known techniques, like SQP, could be used instead to explore with some more detail the surroundings nearby the solution point, getting additional information on what the exact condition of the engine could be, and on the potential reasons why the solution is not being obtained with the adapted Newton technique. This circumstance is expected only after severe deteriorations in the engine (above 5% in TSFC). Under such circumstance, the most likely scenario (once discarded problems with instrumentation) will be the engine’s removal to receive the proper maintenance at the first chance. And then, the necessary time to get to a conclusion on the reasons behind the degradation level of the engine will not be urgent, so the analysis can be finalized later, as required.

The GAs could be eventually used too in case a detailed scanning of the solution space would be needed when a troublesome degradation condition was detected. In regions of the solution space in which the model is not perfectly defined (OEMs obtain maps in test cells for regular operational conditions, and not typically for heavily damaged components), or in which several different potential candidate points may be interfering against the obtention of the chased solution, the GAs might be the way to solve the problem by using the full model, as this method is fully exploratory. It does not depend on derivatives (i.e., the shape of the OF) when exploring the solution space. This technique could be helpful in case a Root Cause Analysis (RCA) had to be completed after the engine had reached an unusual working condition, not contemplated initially by the OEM, and/or not well defined by the model. Needless to mention, ROMs would not be useful to cover such unusual conditions (i.e., limited data in the tensor), but only to monitor soft and progressive deteriorations. If a severe degradation occurred, the full model would be necessary.

The previous approach, based on different available techniques, would lead to a hierarchical scheme to solve the different cases that may rise during operation. This scheme is the one that should be implemented in the algorithm that would deliver the information about engine’s health condition given the data coming from the instrumentation (with its accuracy), the regime, and the flight conditions. An example of such scheme is conceptually represented, at a high level, in Figure 132.
The most relevant task of the EHM will be finding out what is the degradation of the engine during operation, showing then that information to the operator. The algorithm represented in Figure 132 would count with enough calls to the right functions, containing commands to calculate and decide what to do based on results.

There are multiple ways to present afterwards the health condition of the engines to the crew on board in an aircraft (similarly to the engine operators in a ship or in industrial facilities), like the one suggested for the Electronic Centralized Aircraft Monitor (ECAM) of an Airbus A320 in Figure 133 below, to inform about the current health condition of each main component of the engines:

It is convenient to recall here that the methodology presented deals with data from the instrumentation, but it does not contain elements to treat signals coming from the different sensors. Signal conditioning is a task that can be performed by means of hardware (e.g., using specific circuits, like the one presented in [30], or in [229]), by software (e.g., applying HOSVD to the measured values, similarly to the way shown in [149]), or by a combination of both.
Knowing the previous engine conditions (so, counting with a baseline), and the current one (after running the previously introduced algorithm), it would be possible to initiate the analysis of what could have happened to the engine so far during operation, and what would be recommended to do next. Such analysis could be approached from several different perspectives:

- From a **diagnostic** point of view, based on the exact degradations’ value, it is possible to determine the condition in which the components of the engine are working, in the sense of serviceability (go or no-go). This decision should be based mainly in the use of adequate thresholds obtained from the knowledge of the design of the engine model and the experience in the fleet. If any of the components in the degradation vector goes beyond the applicable thresholds, the associated component should be considered as faulty. Once those limit values were established (and $\bar{X}$ will not depend on the operational conditions and regime, but only on engine’s condition), the diagnostic could be performed on real-time. Determining if one engine has been affected by a DOD, a FOD, by contamination (e.g., volcanic ashes), or by any other circumstance (e.g., icing) is something that is not typically possible until a further inspection (i.e., by borescope camera) is completed. However, finding out that only the Fan module has suffered a relevant degradation could be the evidence of a bird strike, for instance, and that information would be certainly useful for the crew onboard, for the maintenance staff, and for the emergency assistance that could be required after the event. If one aircraft engine shows higher degradation levels in the compressors after flying over an active fire, that information would tell how bad the contamination has been for the machine, and where the further inspections and maintenance actions should be more intensively focused. If the HPT module shows high erosion (mass flow capacity higher than expected given the number of hours the engine has been operated), that could be an indication of the need for a hot section repair or exchange to be done soon. And more examples could be found to illustrate the potential usefulness of the results once the components in the degradation vector $\bar{X}$ are available.

The maps provided by the OEMs would be the first resource to understand if the engine is being operated inside valid regions (i.e., operational conditions to work approved by the manufacturer for the engine). When any engine’s main component drifts into an unacceptable region on its associated map, the convergence of the methodology (e.g., adapted Newton) could not be guaranteed, and that circumstance should be indicated to the engine operator, pilot, or maintenance crew. If the methodology is not capable to provide the degradation condition of the engine given the information from the instrumentation, the reason behind will be the operation of the engine outside acceptable regions of the maps. When the engine is being operated outside the recommended conditions by the OEM, at a minimum, a further evaluation of the event should be carried out. If the results obtained by the methodology do not match the reality, that could be caused by a problem with the instrumentation (i.e., one main module showing null degradation levels constantly, showing unexpectedly high levels, or constantly varying, up and down, instead of showing a clear trend to higher degradations). A diagnostic approach for this methodology is represented in **Figure 134**.
The **prognostics** over one engine, given the values of $\bar{X}$, will be developed also upon some external information. The determination of the applicable thresholds will certainly be used to decide if one engine can continue working and for how long, based on the current engine condition, but the prognostics (like the alarm forecasting shown in Figure 134) imply a temporal dimension that does not necessarily have to be present in a diagnostic analysis. The main outcome from the prognostic analysis will be the RUL of the engine, and/or of any of its main components. The typical evolution of the RUL is softly declining during the first hours of operation but, as the accumulated amount of running hours increases, the decline starts to be more evident, ending up abruptly at the no-go threshold when the end of the actual RUL is near, as shown in Figure 135. That evolution will depend, not only on the current engine health status and the maximum degradation allowed during operation, but also on how fast those two conditions are approaching to each other. And this means several basic scenarios can be considered, based on the way the unit is handled by the pilot (or engine operator). It could be conservative (not-demanding operating profile), most likely evolution, or aggressive (incurring in more hours than expected at high TIT and EGT, and/or completing more frequent operating cycles than scheduled, having landing aborts, etc.). It will be important to track and store, in some CS memory storage device, the previous values of $X$ to analyze the trend the degradations are following (i.e., getting the first derivatives of degradations with time). That estimation based, for instance, in a quadratic approximation of the evolution of the degradation (i.e., $X_i(t) = At^2 + Bt + C$, or in any other suitable analytical function), is inexpensive to obtain, once the value of $\bar{X}$ is calculated as a function of time. With the value of the components of $\bar{X}$, the thresholds established by the OEM, and their temporal derivatives, the estimation of the respective RUL, evaluated inside different
scenarios characterized by the value of the derivative of the function when reaching the applicable threshold, can be accomplished (Figure 135). In this sense, it is mainly considered in the literature the evolution of the degradations associated to the respective adiabatic efficiencies of the main components of the engine to evaluate their respective RUL. The reason behind could be they are more easily calculable with the available instrumentation. The evolution of the degradations associated to the mass flow coefficients in the engine are not typically mentioned in an explicit way, but they are convenient too, and they should be coherent with the ones relative to adiabatic efficiencies.

![Figure 135: Evolution with time of one of the components of the degradation vector, and determination of the RUL of the component when considering potentially different scenarios (conservative, average, and aggressive).](image)

In any case, the exact moment in which the required maintenance must be applied, when no imminent safety risk has been detected yet, will be typically decided upon an economical consideration, because the higher the degradation of one component on the engine is, the more expensive its reparation will be. Stopping one engine to perform inspections, or repairs, implies an operational cost that must be considered as well (potential impact in scheduled flights, industrial power supply outages, etc.). So, it is a matter of the highest interest for the operator of the engine to decide wisely when it is the best moment to perform the applicable maintenance. Relevant factors like the cost of the reparation itself (including the costly hardware used in aeroengines, or the specialized labor required), the Return on Investment (ROI) of the asset, and the direct and indirect costs associated to the time the engine will remain out of service must be incorporated to the decision process. A business case will be needed to decide, and that analysis can be consistently done only when the expected evolution of the degradation can be estimated. This means the methodology proposed in this thesis will provide a technical criterion to establish when one engine must be stopped to perform some maintenance on it to make an informed final decision.
• With the **calculation of TIT**, together with the rest of degradations, the methodology determines the regime at which the engine is supposed to be working, given certain ambient and flight conditions. That temperature is extremely important because it is directly linked to the fuel consumption, and it has a direct impact on the RUL of the components in the hot section of the engine. The TIT evolves during a flight, being different during its different stages (i.e., idle, take-off, top of climb, cruise, etc.) and the airlines have a clear interest in optimizing its value, modifying flight altitude and Mach during medium and long-haul flights, as it was indicated in Chapter 3. The value of TIT could be higher than expected during normal operation for several reasons, given certain engine’s overall degradation level and flight conditions. When the efficiency of the different main components of the engine decreases, the engine is less capable to convert the chemical energy from the fuel into mechanical energy, and then it is necessary to supply more fuel to reach the same power level or thrust, as in the case of the gas turbines used in aviation, and that implies higher TIT comparing with non-degraded conditions (and less EGT margin). One of the simplest reasons why the TIT could be higher than expected would be a miscalibration of the variable geometry devices (e.g., VBVs and VSVs) leading to a misuse of the air going through the compressors. If any of the VBV doors remained unexpectedly open, then a certain amount of air would be bled to the Fan duct when it should be going to the CC instead. Such circumstance would involve more fuel, necessary to reach the same thrust level that would be reached without such bleeding. Same consideration may be done with other bleed valves (e.g., active clearance control, aircraft handling for cabin pressurization, anti-icing, etc.). A calibration of the variable geometry and bleed valves would eventually solve such problems. But also, during a flight, a pilot could intentionally demand a higher thrust level than required for a specific flight condition (given Mach number, temperature, and altitude), penalizing the RUL of the engine, because of time constraints or changes in wind direction and intensity. The value of TIT provided by the methodology in this work is the one theoretically expected for the engine, based on the model of the engine, given its health status and the current flight conditions. With the proposed methodology, it could be selected automatically as a setting, if it is calculated quickly enough by the techniques seen in the previous chapters, always guaranteeing the appropriate TIT value in the engine, therefore the right fuel consumption, regardless of the cruise phase, according with the model, and incorporating to the autopilot during flight the input of the degradations in the main components of the engines (and something similar could be done with industrial or marine applications, as usual). Higher TIT values mean higher deterioration rates for the engine’s materials, particularly for the components in the hot section of the engine. The variable that has been traditionally used to track the deterioration of the engine was the EGT (the higher the EGT, the more fuel used, the lower the efficiency of the engine). Obviously, the information supplied by the proposed methodology improves the degree of knowledge on the engine’s overall health status. As the main target of the methodology would be contributing to the diagnostics and prognostics in gas turbines providing useful information, the determination of TIT is fully aligned with such final aim.
6.2.- Business Case

This section will elaborate with some more detail on the way the outcome from the methodology presented in this work could help to end customers to make informed decisions regarding the operation and maintenance of their engines, and why the targeted segments of market would be interested in the information generated.

As it was introduced in the first chapter of this thesis, it results evident, after consulting the specialized press and the last technical papers on the topic, that the engine performance optimization, the prompt diagnostic, and damage prognostic is a matter of the highest interest nowadays for every aeroengine and gas turbine operator worldwide. The search for efficiency retention, reliability improvements, and higher availabilities has extended along the different industries and markets.

The topic of diagnostics and prognostics in an aeroengine, based on the information retrieved from the sensors installed in the machine, is not new. This was commented in Chapter 2, where the first works on GPA were dated in the late 1960's and early 1970's. Nevertheless, the current computational capabilities have made possible the incorporation of some novel elements, unaffordable years ago computationally speaking, with respect to the classic specialized papers (e.g., the use of specific components’ maps, like it was shown in the presented methodology, trying to avoid the linearization of equations from the engine theoretical performance model, strategy almost omnipresent in the consulted literature).

In this sense, the main OEMs of gas turbine engines have been working continuously during decades to create software tools that allowed to get accurate performance estimations of their engines, with the final aim of improving their design, operation, and maintenance practices. These companies developed (and continue developing) applications that are used by them and commercialized as valuable services for their customers. In those applications, it has been incorporated information from the whole fleet of engines monitored and maintained by the manufacturer. The quality of the information and service that the main OEMs offer to their customers, based on engines’ health condition, and obtained by the massive human and material resources they count with, is certainly the best possible.

The manufacturers own all the existing detailed information about the design characteristics of the machines they produce, so they do not have to do reverse engineering to understand how their engines work. They have the most detailed engine models available to reproduce in a computer their expected behavior with the highest accuracy level possible. They also manufacture their machines according with the targets required by customers, and that means OEMs not only know what their machines do, but also what customers are looking for.

And the available sources of information for the OEMs do not end there, because they monitor a good portion of the engines produced, storing massive sets of data about the operation of hundreds of engines working around the world, for diagnostics, prognostics, and performance optimization purposes. That information is shared with all the fleet in the form of Service Letters, Product Bulletins, etc. When a potential issue is detected with any design, maintenance procedure, or produced part of any engine model, the strategy to fix usage to be immediately deployed in all the fleet. However, it is doubtful that both end customers and manufacturers, or approved maintenance service providers, would share the same strategies and commercial targets. Each party will look for the optimization of their own profit and all will need the right information to do so.
To get advantage of the great amount of information retrieved from the fleet, the main OEMs collaborate with big technological companies, like Microsoft® [195] or Apple® [16], to leverage their advanced informatic technologies, providing highly valuable engine condition analysis and operation summaries for the performance and maintenance optimization of the assets being monitored (potentially, at perpetuity). Attending to the previous scenario, it may look like there is no competence possible with the different OEMs regarding EHM services. However, these services imply several drawbacks that deserve some comment:

- The integral services provided by the manufacturers based on their knowledge of the fleet imply some costs (typically installation and periodic subscription) that not all the customers can afford, simply because the final price of the service provided is typically based, not only on the costs associated to the development, installation, and put into service of the solution itself, but also on the potential benefit that the end customer should obtain with it, and not necessarily on the actual improvement achieved. It is true that certain companies operating gas turbine engines, such as big airlines [254] or energy suppliers [82], are signing off multimillion contracts in this sense. However, not all the customers are so big, and some of them do not count with financial resources to incorporate those advanced features into their reduced organizations.

- It is still to be clarified if mid-sized or small companies would get all the potential benefits from the integral and extensive digital EHM services provided by the manufacturers. Big companies count with a clear scale factor, given by the fleet of engines they operate, that easily justifies the cost, so a small improvement in one unit could mean automatically a much bigger profit when considering the whole fleet. For smaller customers, with few engines, that scale factor does not exist initially. Maybe a simpler software solution that was focused on the most relevant point for the end customers, as the health condition of the different main components of the engines is, could fit better into the maintenance strategy of certain market sector. There could be a niche to explore that main OEMs do not find appealing yet.

- To apply a Condition Based Maintenance (CBM) philosophy, which is inherent to the use of such EHM tools, and requires of an eminently proactive attitude, instead of a reactive and scheduled maintenance approach, means a change that requires of time for education, so its implementation will not be quick in the practice. In this sense, the change from old operational and maintenance paradigms to new ones is something that may take years to happen, as it is related to the culture and philosophy of the end customers.

- Customers will normally prioritize some other urgent costs over EHM services. At least, at the cost levels managed nowadays. The control of the engine and the identification of potential technical issues are critical features that customers truly need, so it is in their interest to allocate the necessary budget for them. Nevertheless, the optimization of the operation and maintenance management of the asset is still perceived by certain customers as a “good-to-have” feature, but not critical. The reparation of a real damage that could prevent the operation of one asset will be always a priority. This perception could change if the offered EHM services were more affordable, clearer when showing results, faster to implement, and easier to interpret.
All these reasons invite, at least, to propose the described methodology to the targeted end customers that may not be interested in the complete EHM services provided by the different OEMs, but only in those that would improve more clearly the operation of the gas turbine engines in the fleet and the associated maintenance practices. The way to improve those practices must begin with the timely knowledge of the condition of the engine. Once that information is at hand, it will be just a matter of deciding what action must be done next (e.g., engine water wash, repair, etc.), and when. In this sense, the price of the JET-A1 fuel in Spain was about 0.596 USD/litter by November 2021, [65]. Or converting to mass, considering the density of the fuel at low temperatures, 0.709 USD/kg (-44°C is the threshold that is commonly considered for cold conditions with JET-A1 in civil aircraft, see [200] and [125]), meaning 0.657 USD/s or 2,368 USD per hour (and per engine, according with the engine model used in this thesis) of cruise flight, assuming no degradation affects to any of the components in the turbofan and keeping a constant thrust level of 49.58 kN. Taking into consideration the progressive expected deterioration of the engine with the continuous operation, the fuel consumption cost will grow as the performance of the engine gets impacted. An easy estimation can be done, based on a sensitivity analysis like the one presented in Chapter 4. In that analysis, which considered a typical cruise condition for the engine model utilized along this thesis, the increment of TSFC experienced by the engine, assuming only one component was degraded while the rest of degradations remained constant and equal to 0%, is given by Figure 136. This simplistic analysis would be valid only for relatively low degradations, where the ceteris paribus assumption does not introduce too high calculation errors. For higher degrees of deterioration, there will be coupling effects that may introduce high variations to the results. Nevertheless, it can be used to visualize the severity of the increasing economic impact that the continuous deterioration of the engine could imply. The full model can be always used to get a more accurate estimation of the associated TSFC increase. Fortunately, more than 2,000 flight cycles would be required (with no maintenance) to get a degradation of 1% in any of the main engine’s components, according with [221] (see Figure 137):

![Figure 136: TSFC increase (cruise) with degradation in the main engine components, according with model.](image-url)
In the sensitivity analysis results, it is evident that the components of the degradation vector $\bar{X}$ that most contribute to the increase of TSFC are the ones associated with the efficiencies of the different components (i.e. $X(1)$, $X(3)$, $X(5)$, $X(7)$, and $X(9)$, respectively), and more specifically, the degradations associated with components in the hot section, HPT and LPT, being the degradation of the HPC efficiency the next one in relevance. The degradation associated with the Fan does not contribute that much to the TSFC increase in this model. Similar trends in the sensitivity analysis can be also found in the data provided in [221], however values are different as both results correspond with different engines from different OEMs.

It is also clear from Figure 136 that degradations associated with mass flow coefficients contribute less to the increase of TSFC (they contribute less, but their addition is not negligible, so the effect of all the degradations over the TSFC will be retained). Nevertheless, beyond their role increasing TSFC, these degradations are important to assess the condition of the engine because it may happen that some damage in the engine could remain hidden if only the efficiencies are taken into consideration. It is possible to find damages and deterioration in a borescope inspection to a gas turbine engine component without experiencing great losses of efficiency after a sudden event. Mass flow coefficients should be therefore accounted for, and its use and analysis should be extended in the future.

Operational thresholds must be established for every degradation based on the knowledge of the engine model and the experience from the fleet. Typically, both degradations regarding efficiency and mass flow should evolve in a coherent way. Some operators may pay more attention to the efficiencies, as the operational cost will be driven by them (transfer of energy from fuel), but the mass flow parameters could detect severe maintenance issues (erosion, dirt accumulation, tip clearance, etc.). A severe local HPT TBC loss, detected by $X(8)$, could compromise the RUL of the HPT, not affecting necessarily to the overall efficiency. Another example is the CC of the engine, that could suffer of great deterioration without affecting to the engine’s performance, reason why it is typically left out of this kind of diagnostic and prognostic analysis. This is another reason that justifies the convenience of getting accurate results for the computed calculations. Detecting small changes in mass flow coefficients would be the way to foresee more serious effects in the long term.

Information like the one shown in Figure 137, indicating how the degradation in the different components will affect to the overall TSFC of the engine, could be used to establish decision criteria on the maintenance actions to make, aiming to minimize the operational costs along time, and optimizing thus the profit obtained. There are typically four main maintenance interventions to consider:
Engine water wash: A complete engine water wash is a common maintenance procedure to recover performance in the engine lost by fouling (not all the losses caused by fouling are recoverable, as it is indicated in [194]), that consists of several steps. The engine is hydraulically (or pneumatically) cranked while several nozzles inject water solutions through the engine's inlet (hot water with industrial soap, fresh water, or water with glycol in cold climates). The water wash system counts with pumps, pipework, and filters to guarantee that the water solution is injected into the engine at the desired conditions (usually established by the OEMs). Portable equipped trolleys are used in airports to wash engines, on-wing, while one aircraft is grounded. Industrial units count with permanent systems, including fixed nozzles and water manifolds in the inlet plenum area. The first step implies the use of a chemical substance (industrial soap, with a controlled pH value, with corrosion inhibitors, and limited alkali metals after dilution, see [306] for further reference) mixed with hot water (above 50 °C, and up to 90 °C). This water is also normally treated to limit its conductivity (it is used demineralized water with a typical conductivity lower than 10 uS/cm, meaning very low impurities dissolved in it). The water supply pressure ranges between 10 barg and 90 barg (higher pressures allow for a better droplet size control, critical factor for the washing effectiveness). After several minutes to leave the chemicals work over the internal surfaces of the engine, several rinses with fresh water are required to remove any rest of soap. A final engine start up is usually done to help to dry the machine. With this practice, known as off-line water wash (the engine is cranked with no fire in the combustion chamber), the recoverable performance loss in the compressing stages of the engine caused by dirt accumulation (also in some internal ducts) is mostly recovered. Traditionally, it has taken between 6 and 8 hours to be fully completed, per engine. Now, some service providers advertise novel methods that could reduce considerably that amount of time (down to 45 minutes, according with [182]). In industrial applications, some customers operating gas turbines in dirty environments combine off-line water washes (performance recovery) with on-line water washes (while the engine is working to try to improve the retention of performance). In this last case, it is necessary to decrease the power load of the engine, slowing down the rotational speed, to avoid an excessive erosion in the blades that will receive more directly the water injection. In both cases, the decision to perform a water wash is made based on economic reasons: the cost of keeping the engine stopped between 6 and 8 hours, the cost of use of the required auxiliaries to complete the washing, the cost of water and chemicals, and finally the associated cost of labor must be more than compensated by the improvement in TSFC. The cost of completing the water wash is reasonably low (see [131] for a cost evaluation in a LM6000 engine during a year, meaning between 25,000 and 30,000 USD per year, washing twice per month, but engine-specific intervals will depend on usage) comparing with the cost of keeping the engine stopped (so the loss of passengers in an aircraft, or the production loss in a power plant). If an improvement in TSFC of a 1.5% average was feasible with water washes,
in the case of the engine used in this thesis, that would mean a direct fuel cost saving in the order of a minimum of 105,000 USD per year (assuming 3,000 hours of cruise operation each year service, see US public data available from 2020 in [196]), and per engine, with fuel prices taken by November 2021. That would be an optimistic scenario, but it serves to visualize the potential gain the engine washes imply. Keeping the engine clean and removing dirt accumulation potentially could imply additional maintenance cost mitigations, but they are more difficult to estimate. Customers usually focus on EGT increases or power losses, magnitudes easy to measure, to decide when a water wash must be done. Planning an effective on-line water wash program needs to consider potential fuel losses, materials, and labor required as well as lost production (see [198] and [11]). The proposed methodology could help in this sense as the degradations provided (clear physical meaning) are directly related with the increments of TSFC, which could be calculated by PROOSIS®. The water wash program could be designed to maintain an average level of performance in the Fan, LPC, and HPC. The methodology also allows to differentiate which component has suffered of a higher deterioration, so the wash program can be more incisive in those modules specifically.

2. Hot Section Repair: The engine parts installed in the CC and HPT (e.g., combustor liners, HPT vanes and blades, shrouds, associated seals, etc.) are the ones exposed to higher thermal stresses along the consecutive flights or operational cycles (starting up, warming up, keeping the engine working at a particular load, and shutting down). As it was commented before, the degradation in the HPT has a clear negative impact on the TSFC of the engine, meanwhile the combustor follows a degradation path without impacting that much to the engine’s performance. Nevertheless, a damaged combustor could affect negatively to other engine components (fuel nozzles, ignitors, internal cooling flows in the engine, casings’ heat rejection, etc.), so it is crucial to assess its serviceability. The replacement of these parts is nowadays driven by the number of running hours and/or operational cycles, but also by its condition. In this sense, some manufacturers establish in their manuals the need for Hot Section Inspections (HSI) in between overhauls, to validate the condition of combustor and turbines. During those inspections, it is possible to detect and repair or replace any damaged component. Some OEMs count with a Hot Section Exchange (HSE) program, in which the combustor liners and HPT module are exchanged by repaired parts (rotatable modules pool), so no delays are caused by the potential findings during inspection. A time limit is defined by contract and the Hot Section is exchanged, sending the removed parts to be repaired before they are sent for another customer. This kind of programs are very popular these days as the cost of a new combustor and a new set of vanes or blades is very high (a new HPT blade set for a CFM56-5B/7B was around 1.2 million USD in 2018, see [185], so a complete Hot Section Replacement, including HPT blades, vanes and combustor liner will reach easily prices above 2 million USD). Customers pay a fee by contract within HSE programs, but their risk is kept inside limits. Some parts’ providers (e.g., Chromalloy [57]) could offer alternatives to the parts and programs offered by the OEMs, these are the

- 269 -
so-called Parts Manufacturer Approval (PMAs, see FAA [89] in this sense). Initially, the PMAs are equivalent, in terms of licensing, to the parts offered by the OEMs, but typically at a lower cost. PMAs could not count with the durability of the parts provided by OEMs, but depending on the scenario, they could constitute an acceptable solution in certain cases. There are examples of customers in industrial applications that extend the HSE intervals thanks to the information supplied by condition monitoring systems, increasing the uptime substantially (and limiting the unnecessary downtime resulting in an estimated 30 million USD/event in savings, as documented in [25]). The proposed methodology could help also in this sense, providing a technical resource to make an informed decision on the possibility of shortening (if the impact of TSFC is higher than expected) or extending the use of the hot section components installed (if the performance behaves reasonably well once the recommended life of the components is completed, as per supplier, being always guaranteed first the operational safety of the engine), based on engine’s condition. Anyway, the first visit of one engine to the workshop should be for its first HSI/HSE, if the operation of the engine goes as expected, with no unscheduled outages caused by other technical issues. A full HSE can be accomplished typically in one week.

3. Major Overhaul (MOH): The major overhaul implies the disassembly of the engine to complete a deep inspection of Life Limited Parts (LLPs, such as shafts, disks, rotating spools, etc., parts that cannot be contained in case they failed, so they must be replaced beforehand) and its replacement, or repair, as required. As it happened with the hot section components, there is a market for used and repaired LLPs, but the availability for the most demanded engines is limited, so some customers have to buy completely new components (as indicated in [185], the core engine LLPs could reach up to 2 million USD, 2018 price level, meanwhile the complete set of LLPs will mean for certain models 4 million USD, 2018 price level). The LLPs timely availability plays a relevant role in the final price. The overhaul finishes in a certified test cell to validate the performance guarantees after the intervention. A turbofan like the one used for the model implemented in PROOsis® will typically go for the first MOH after 7,300 flight cycles (see [155] for reference), however this number will vary depending on the engine model and OEM. The whole engine overhaul cycle should mean four visits (MOH cycles) to the workshop over its entire life, however that number will be normally higher, as introduced before with the HSI/HSE, because of the different engines’ components maintenance intervals. The work from Petcharin and Ren, 2012, [228], shows how the intervals between visits to the shop in CF6-80C2 engines, given a certain reliability level, follow a two-parameter Weibull statistical distribution, not matching necessarily with OEM forecasts. The Weibull distribution, which describes the type of failure mode experienced by the population and determines the mean time between visits to the shop based on the characteristic life curve determined by the mentioned parameters, is very popular for RUL estimations in complex mechanical systems like engines (see [119] for further reference from GE). In this sense, no operator is allowed to fly
with one engine exceeding the Life Limit in any of its parts, so the MOH constitutes one of the principal milestones for each gas turbine engine maintenance scheduled program (see FAA Circular AC 120-113, January 2016, in this sense [90]). The MOH can take up to 60 days in a workshop. The proposed methodology could not help to extend the time between overhauls (TBO) of a given engine, as the life of the LLPs is the critical parameter used to decide the length of that interval, but the information obtained with it could be used to initiate a program to extend that life in the LLPs, for future engines, if the results invite to do so.

4. Minor repairs: To be done on condition, like faulty sensor replacements or oil leaks' fixings. They will usually mean unexpected delays, while the on-wing (in the field) reparation is completed, but not long operational outages or visits to the workshop. The cost dedicated to spare parts is minimum comparing with the cost associated with the delay itself. In this sense, even some components can be repaired at a lower cost for the engine operator comparing with a completely new component (a speed probe repair, with its test included, is in the range of 2,500 USD price, see [183]). The proposed methodology could be helpful as well with this kind of diagnostics, helping to identify and confirm the component in which the problem has been detected, contributing to minimize delays.

The maintenance program over one engine, and the estimated effects on the overall performance (TSFC %), can be represented in charts like the one in Figure 138. The engines will go through several MOH cycles, of different duration given the limitations imposed by the LLPs. The best moment to perform the different maintenance actions indicated in the chart depends on complex economic models that try to optimize the overall operational cost for each engine. The methodology in this thesis, could be integrated inside those models, so the degradations and their associated extra fuel costs are balanced with the costs associated to the required maintenance to try to recover previous health conditions. An additional washing could be scheduled before the next Shop Visit (SV), if the average degradation reduction compensates the costs of stopping to wash one more time.

Figure 138: Maintenance program schematic for the 1st MOH cycle (“honeymoon”), including washes (CF6-80E1).
It must be highlighted here, as it was done in Chapter 1, that the operational costs associated to fuel consumption constitute approximately a third of the total cost structure of a conventional commercial airline, being the costs associated to the engines' maintenance a 4% of the total. The relevance of the topic for the industry explains the plethora of studies and works publicly available in this sense, detailing many different approaches followed today by the airlines and industrial gas turbine operators to optimize both fuel consumption and required maintenance.

On this regard, the presented methodology will not contribute initially to change substantially the practices needed to recover performance losses, or the reasoning behind the decision process to send one engine to repair, even when it potentially counts with the capability to improve the accuracy of the diagnostics over the engine, pointing directly to the components that show a higher degradation level. It will mainly provide real-time information on the engine's condition to make an informed decision, when incorporating to it the knowledge on the engine's design and construction, as well as the available experience from the fleet. The methodology could nonetheless help to select among different practices and maintenance philosophies, if the results are not as good as expected after applying one of them. Once it is possible to establish a correlation between components' degradation and the increase in TSFC, given a certain thrust level or power (which can be estimated with enough margin of confidence, statistically speaking, when the engines are covering the same route, or developing similar operational profiles along time), and a fuel cost, it is possible to elaborate a cost function. That function, used to decide the best moment to perform any of the three first maintenance actions indicated before (the fourth one will be normally addressed by the mere application of thresholds to sensor readings, inside a simple “high-low” diagnostic logic implemented in the CS), which cost is usually also known in advance by contract at certain extent (some variations may appear, depending on engine's real condition and the use made of it), could be estimated on real-time with PROOSIS®.

If one engine in the fleet has experienced a sudden damage in the hot section, relatively soon for its maintenance cycle, that is causing a higher cost in its operation than expected, it could be justified to do a HSE before it was scheduled to try to mitigate that financial drain (avoiding more severe incidents, or even accidents as well). As it was indicated before, the typically highest impact on TSFC comes from the degradation in HPT and LPT (HPC as well, but its deterioration can be more easily restored with an adequate washing schedule, unless it was damaged). On the contrary, if the performance of the components in the hot section was inside limits, the HSI/HSE could be postponed until the performance deterioration level justified such action. That decision, extended to a complete fleet, could mean a remarkable saving in maintenance costs for an airline or for a big industrial gas turbine operator.

Similar considerations could be done with the water washes. An unexpected higher deterioration rate could justify a sooner stop of the engine for washing, or even a top case intervention (opening HPC casings) to use more effective techniques like foam washing or dry ice. And, again, the justification to do so would be to avoid a continuous extra cost in fuel, given the current energy prices worldwide.

In all these cases, a higher knowledge of the degradation status in the engine, provided by the presented methodology, will be a powerful tool in the decision-making process, adapting the maintenance to the current engine condition, and
contributing this way to optimize the management of each engine in the fleet. And this methodology will focus just on the engine, limiting its implementation cost.

So, just to emphasize the way the presented methodology could be helpful to end customers, the following strategy is suggested to be implemented:

1. Water wash program: With a more detailed information on the evolution of the degradation along time of the engine it is possible to optimize the washing intervals (i.e., it may be necessary washing just the fan, or the complete engine with an additional water wash per year, depending on the evolution of the degradation values obtained from the methodology), and it is also possible to make a more informed decision on the convenience of completing a more intense washing during the visit of the engine to the workshop (e.g., using dry ice or foam washing in HPC or LPC). If, because of the continuous use and analysis of the information regarding the degradation evolution, an average improvement of a 0.5% in TSFC was possible comparing with the situation without a detailed knowledge of the degradation of the engine, then a medium-sized airline (125 aircraft with 250 engines in the fleet) using engines like the one used as reference in this thesis, at a constant thrust level to avoid inconveniences to the passengers, considering the cost of the fuel by 2021, and flying nine hours per day and per aircraft, could save more than 9.5 million USD per year. Similar analysis could be done for industrial gas turbines, leading to relevant savings per unit given the continuous (24/7) operation of many of these units (base-loaders), and the increasing evolution of the natural gas prices in 2021 (see [86] in this sense).

2. Hot section schedule: Keeping a cleaner engine, with the previously mentioned potential improvements in the washing plan, will redound in a more efficient machine using less fuel to deliver the same level of thrust or power. The work from Wehrspohn et al., 2021 [298], regarding the Lifecycle Cash Flow Environment program, known as LYFE, developed by researchers from DLR, to estimate the impact of engine washes over the lifetime of the engines in an aircraft fleet, shows that engine washings (analyzing scenarios ranging from 4 to 14 washes per year) could improve the time on wing of one engine up to 2,240 FC, reducing the fuel cost in an average of 1.2% (nevertheless, contrarily to the methodology presented in this thesis, no estimation on the impact of dirt accumulation in the variation of TSFC could be provided in the paper, only one educated guess). That change will have a positive impact in the hot section components’ wear, difficult to measure. However, if the hot section components exchanges or repairs are determined on condition after a HSI, it could be decided to extend the life of those components. That measure could potentially lead to avoid a complete HSE after two or three complete MOH cycles, meaning a potential saving of 2.0 million dollars after 15 or 20 years. Again, the scale factor associated to a full fleet of engines in an airline would justify more than enough the implementation of the proposed methodology.

3. Major overhaul impact: The LLPs are clearly a major limitation for the potential upgrades in the costs associated with the MOH schedule that each engine needs to follow. There are clear regulations from the Aviation
Authorities to prevent the operation of engines with LLPs exceeding their established lifetime. Nevertheless, the information provided by this methodology, together with the results found during the MOH inspections, documented, and shared with the OEMs and the applicable Authorities, could lead to programs aiming to extend the life of those critical parts, contributing to save a portion of the 4.0 million USD cost, previously mentioned, for a complete set of new LLPs. The recovery of performance after a MOH can reach values up to an 80% of TSFC [298]. Some engines are sent to the shop depending on the remaining EGT margin (difference between current EGT value and the threshold established by OEM and Authorities). In that case, mostly defined by the aircraft and the routes, the improvement in efficiency over the years could imply an extension of the on-wing time of the engines.

4. Minor detected issues: Having a more precise information on the condition of the engine could help to an easier identification of the potential faults detected in one engine, leading to a faster resolution of the problems and shorter delays in its return to normal operation. That improvement would redound in lower operational costs for the airline, again, being potentially extensive to all the fleet of engines.

All the potential improvements mentioned so far will depend greatly on many factors, including the human factor. What is evident is the first step to try to improve the operation and maintenance in each engine under consideration is counting with the right information regarding its health condition and degradation of its main components. The cost of an industrial software license is typically in the range of about 5,000 to 50,000 USD, per computer, depending on the functionalities incorporated to it, the validity and periodical update of the associated software, and the potential benefit the end customer will obtain with its use. In the case of the methodology presented so far, even if every engine counted with one PROOSIS® license to run the model of the engine being monitored, that cost may be justified if it contributed to improve the efficiency and maintenance practices of all the engines in the fleet. Similar consideration could be always done for industrial applications. But the business case could be even more clear for the end customers if the use of ROM-based methods was feasible, as only one license would be required for most of the engines working under usual and well-known conditions, statistically speaking.

These reasons, including the considerable fuel and maintenance savings that may be derived of the implementation of the methodology, the diverse market segments in which it could be offered, and the relatively simple approach it means comparing with some other more sophisticated (and costly) solutions offered by OEM and service providers, truly invite to try to commercialize the concept.

Just to summarize the potential profit that could be obtained by a better knowledge of the degradation in the engines of a mid-sized airline, counting with 250 engines in total, Table 41 gathers some of the estimations that were previously mentioned, just for the direct operational impact, excluding potential maintenance beneficial side effects. It was included an estimation for the costs associated to the potential increase in the rate of washings per year, for the different SW associated costs either using the full model or the ROM, and finally considering the loss of commercial margin associated to losing flights because of the washings. Pre-COVID commercial margins from IATA, 2021 [138], were used to complete the exercise.
Results could look appealing, because of the scale factor associated. The benefit will be less interesting for the operator of one gas turbine only for marine or industrial purposes. Estimations can be, obviously, refined to reflect more precisely the reality, but the potential operational gains invite, at least, to think in implement the methodology. Then, it will be a matter of integrating it with success into the maintenance practices of the candidate customer. The adequate use of the new available information will be definitory for the final improvement achieved.

| Fuel costs for JET-A1 (to be updated, as required, with the current fuel cost): |
|---------------------------------|-----------------|
| Cost of JET-A1 (USD per liter):  | 0.60            |
| Density at 15°C (kg/m³):         | 800.00          |
| Density at -44°C (g/ft³):        | 404.16          |
| Cost of JET-A1 (USD per kg):     | 0.71            |

Results from Sensitivity Analysis:

- TSPC from Sensitivity Analysis (g/kN at cruise phase, with no degradation, and 49.58 kN of thrust):
  - 18.71
- Accumulated percentage increase (%) in TSPC when all the degradation grows by 1%:
  - 3.19
- Total cost in USD of fuel per hour per engine, in such cruise regime with no degradation (baseline):
  - 2,269.01
- Increase in fuel cost per hour, USD/hr per engine, when all degradations grow 1% (assuming no coupling effects):
  - 7.52
- Operational scenario (average cruise hours per year):
  - 3,000.00
- Aggregated costs considering a mid-size fleet of 125 aircraft:
  - 1 engine 250 engines
- Increase in fuel cost per hour, USD/hr, because of a global 2% increase in TSPC, comparing with the baseline (with no degradations):
  - 47.88 11,845.05
- Increase in fuel cost per year, USD/yr, because of a global 2% increase in TSPC, comparing with the baseline (with no degradations):
  - 142,140.66 35,535,164.36
- Target to reach with the integration of the new methodology in the maintenance of a mid-size aircraft, per year:
  - 1 engine 250 engines
- Operational costs reduction, in USD per year, by decreasing the TSPC in a 1% with a better knowledge of the engines’ condition:
  - 71,070.33 17,767,582.18
- Costs, in USD, of implementing the full model with PRODISH®:
  - ESTIMATED 10,000.00 2,500,000.00
- Costs, in USD, of increasing the rate of compressor washings per year, from 3 to 6:
  - 3,750.00 937,500.00
- Costs of performing 3 engine washings per year (as per consulted reference in the chapter, in USD):
  - 30,000.00 7,500,000.00
- Costs of performing 24 engine washings per year (as per consulted reference in the chapter, in USD):
  - 10,000.00 2,500,000.00
- Costs of flights lost to wash the engines (assuming CF6-80B1 engines, powering a 277 seat A330-200, and a 9% commercial margin):
  - 33,240.00 8,310,000.00
- Cost of extra 2 flights lost per wash, with a ticket price of 250 USD/Pax (total of 6 flights lost, per year). Assuming IATA’s pre-COVID data:
  - 24,080.33 6,020,082.18
- Total savings per year, in USD, associated to the improvement of 1% TSPC with the full model:
  - 24,080.33 6,020,082.18
- Total savings per year, in USD, associated to the improvement of TSPC with the use of ROM:
  - 24,080.33 6,020,082.18

Table 41: Summarizing table for potential business case: Target of 1% TSFC improvement by increasing engines’ washing rate per year, deciding the right moment based on the information obtained from the proposed methodology. Costs associated to the extra engines’ washes, required SW licensing, and associated operational margin losses (lost flights because of the washings’ stops) have been considered to complete the analysis.

Finally, just to complete the analysis in this last section, the calculation of the fuel consumed during a flight (needed to evaluate the fuel costs reductions) can be done just by numerical integration of the readings from the sensor \( W_f \) (in kg/s). That is one of the signals used by the methodology, so its value would be available during a flight. The variable can be tracked with time the way shown in Figure 139:
Figure 139: Integration of the variable $W_f$. This result will be necessary to evaluate the improvement in fuel consumption after the implementation of the proposed methodology.

The numerical integration (Dahlquist, 1974, [64]) can be done following different integral rules, like the one shown in the figure. The final value of the fuel consumption will be the more accurate the more samples are obtained with time.

6.3. Future Work

After having analyzed how the methodology in this study could be applied to a real case, and the potential economic benefits that could be obtained with it, this section will summarize which next steps could be followed in the future to continue with this research line, growing in targets (and research ambition) when possible.

The following list is not comprehensive, as several other actions could be considered in addition, or instead, but it gathers the steps that seem to be more aligned with the worked developed so far. These steps are thought to be a natural continuation of the previous work done. In particular:

- Initially, new models in PROOSIS® should be created, representing different turbofan configurations or, in general, other models of gas turbine engines. The configuration evaluated in this work is probably the most common nowadays in the commercial aviation industry, but it is not the only one. Diversifying the offer of potential diagnostic, prognostic, and control optimization services will certainly redound in a better positioning in the market and a helpful advertisement strategy, in case this methodology was finally offered for commercial purposes. As it was mentioned in the first chapters, there are turbofan configurations with more than 2 spools (or shafts), the geared turbofan will be a reality in the next years, and the open-rotor configuration will come presumably in the next years because of the continuous search for better efficiency. Turboprops and turboshafts are engines that could be also modelled once it is possible to represent them by
a flexible and powerful SW. Aeroderivative variants used for industrial and marine applications represent very relevant fleets of engines around the world, working for very relevant potential customers, and they could count with specific models as well. However, the most challenging part of the modelling phase will be obtaining validation data to calibrate the results obtained. This is the point where the interest of potential customers is crucial given the scarcity of real data publicly available. This methodology could be also applicable to aircraft thought to use electric engines once the model of the engines was configured and validated.

- The quality of the data sent to the engine model is extremely important as it was indicated during the SVD analysis of this study. In real engines, data provided by the instrumentation will be typically affected by noise in the different sensor channels. It would be advisable counting with some own technique to avoid the pernicious effects of the noise, even when the noise levels were low. The required de-noising could be obtained by HOSVD, by neglecting the smallest modes in the decomposition of the signals form the different sensor channels. Obviously, this data pre-processing could mean extra computational time that must be optimized to remain as a real-time application, when possible. Non-optimized instrumentation installed in the engines can be treated with this methodology and, when possible, such instrumentation could be improved to provide the required information if spurious data correlations appear, as it was shown in Chapters 4 and 5.

- In the future, it would be desirable to avoid heavy SW applications or an excessive use of SW licenses that could mean extra costs for the research. This target is not only aiming to reduce costs. As it was indicated in previous sections of this work, the hardware used typically in aircraft, or managing the operation of gas turbine engines, is highly robust and reliable (optimized for the mission), but not very powerful in terms of processing capabilities comparing with other available commercial computers (they work in the scale of MHz, when nowadays every standard home desktop PC works inside the range of several GHz). Counting with heavy, or unnecessarily detailed, SW applications will redound in potential problems to implement the developed EHM solutions in industrial computers. In this sense, it is important to highlight that PROOSIS® is a very sophisticated application that needs to work installed in OS beyond WIN7 (64 bits). Here, the support from informatic experts would be of great interest. How the increasing connectivity with a cloud framework of IoT will impact the way the information is managed (data from the fleet, delivery of results and analysis during operation, etc.) is also something to be explored in the next years.

- It is very important to maintain the attention over the profuse literature about the topic of diagnostics, prognostics, control, and performance optimization relative to gas turbine engines that is constantly refreshed with new works and ideas, given the dynamism and relevance of this subject. The reason is trying to identify potential new techniques and theories that could contribute to boost the methodology described in this study. The target is providing results in the fastest way possible to optimize the decision-making process on a real-time basis, independently of the condition in the engine. This is a very time-consuming task, not always rewarding, but still necessary to understand where the state of the technology will be in the near future.
• Another relevant topic has to do with the improvements both in instrumentation and in the hardware installed in engines. Instrumentation will evolve to new types of sensors that will count with higher accuracies and reliabilities (i.e., by using optic sensors, piezoelectric components, etc.) being progressively less immersed in the gas path of the engines. Those improvements will come together with higher demands, in terms of accuracy, for the models providing EHM. Potential fusion of data coming from different kind of instrumentation (i.e., not only relative to the gas path, also relative to vibrations, oil temperatures, etc.) should be also considered in the future. In addition, new materials will be progressively installed in the new versions of engines that will be introduced in the aircraft fleets of the different airlines (also in industrial applications), like CMC or carbon fiber, among others. These new materials and hardware will certainly impact to the models that will be created and it is necessary to understand the implications of their implementation. It is very important to understand what can be expected from those new materials to deliver the best analysis possible to future potential customers.

• New alternative fuels are being analyzed these days with the final target of reducing pollutant emissions. From renewable diesel to hydrogen, new substances will be used in gas turbine engines in the next years and this technological change will mean a certain impact, not only regarding emissions, but also over the degradation expected in the main engine components located in the hot section. Just to give an idea, the hydrogen used in gas turbine engines counts with a much faster flame speed than other usual fuels, with a considerable higher diffusion coefficient inside metals, with higher LHV at equal amounts of mass, and with much wider flammability limits when compared against different standard fuels with compositions based on hydrocarbons (see [280], [56], and [9] for further reference). The chemistry associated to these new fuels should be incorporated into the models developed in the future. In this sense, the estimation of emissions is also of great interest for the customers nowadays, given the restrictions applied by the Environmental Authorities worldwide (and the costly fines associated when not met, locally and globally). Even when the chemical composition of the products obtained from the combustion in gas turbine engines depends greatly on the point where it is measured (the Chemistry needs time to be fully developed), the expected global composition of the products in equilibrium is a valuable information for potential customers that are not only interested in the performance capabilities or health condition of a specific engine, but also in the environmental impact of operating such asset. These requirements will be more evident in the future given the increasingly stringent restrictions from the relevant Authorities in the matter.

• The development of the previous steps could be supported by other students or R&D personnel, from different grades and backgrounds given the interdisciplinary nature of the research (i.e., machinery, engine performance, numerical analysis, software development, etc.), that may be interested in this specific subject. This circumstance implies an effort to teach the fundamentals of the research, making sure new candidates will contribute effectively to the targets established. Ideally, individuals from
different backgrounds (aerospace, industrial, marine, software engineering, mathematics, etc.) could be eligible for this purpose.

- As a result of the activity performed by the original team involved in this research line, several papers have been produced during the last years. It is expected therefore the publication of new technical papers because of the advances that may be obtained, contributing this way to create a sound basement for the future research activity. The acquired technological background should be published and shared with the R&D community contributing to the development of this important matter for the industry.

- Finally, the commercial target would be offering formally the EHM services, developed inside the research team, to different end customers operating engines for different applications, aiming to create a business ecosystem inside a corporate structure, self-maintained financially speaking, focused on the topics treated in this research, held mainly by a workforce compound by researchers and students dedicated to the development of the required models, customer-specific SW, continuous improvement of the available technology, etc. To do so, it is important to initiate the project with achievable goals, maybe offering advisory services to small airlines (regional, cargo, etc.), small ferry lines operating gas turbines in their ships for the propulsive waterjets, or small industrial customers (paper mills, water treatment plants, chemical plants, steam suppliers, etc.) trying to gain progressively more workforce capabilities. Considering an aeroderivative gas turbine operator, the steps to proceed with the first unit to be monitored could be the next ones:

  - Signing off the necessary formal legal documents including non-disclosure agreements to share information, and the required contract for the services that will be supplied.
  - Retrieving enough information from the engine to calibrate and validate the model. Most customers count with enough information stored for reference that can be of great help in this phase.
  - Setting up the model, installing at site a separate computer to perform the calculations with the information from the instrumentation, and initiating the monitoring of the unit.
  - Analysis of results and report to end customer before leaving permanently the solution working at site.
  - Results can be incorporated to the distributed control system (DCS) at site for better convenience and ease of access.
  - Establishing update and future improvement programs based on the methodology and results obtained.
APPENDIX A – Kalman Filters

The Kalman filtering is a popular kind of algorithm that is used to estimate the value of unknown variables given the readings from instrumentation taken over time. Part of the success of such technique resides in its simple form and the reduced computational capabilities required for its implementation. It is eminently used with non-linear problems. It is needed a process model to define the evolution of the state vectors from time (n-1), to n, in the following way:

\[ \bar{X}_n = F\bar{X}_{n-1} + BU_{n-1} + \bar{W}_{n-1} \]  \hfill (A.1)

Where \( F \) is the so-called transition matrix, which is applied to the previous state vector, \( B \) is the control input matrix applied to the control vector \( U \), and \( \bar{W} \) is typically the process noise vector. The noise is supposed to follow typical zero-mean Gaussian distribution, with associated covariance matrix \( Q \) (it can be modified accordingly with the nature of the process). The process model is supposed to go paired with the measurement model that provides the state and the measurement at the current instant:

\[ \bar{Y}_n = H\bar{X}_n + \bar{v}_n \]  \hfill (A.2)

Here \( \bar{Y} \) is the instrumentation vector, \( H \) is the measurement matrix, and \( \bar{v} \) is the noise vector associated with the instrumentation. The noise can be defined by a typical Gaussian distribution as well but making sure its covariance (matrix \( R \), in this case) is different to the one used in the process model.

The target of the algorithm is to compute \( \bar{X} \), at time \( n \), given an initial estimation \( \bar{X}_0 \), the series of readings from instrumentation, \( \bar{Y}_n \), and the information from the system given by the matrices and parameters used to define both process and measurements. It is usual to assume that such matrices and parameters are fixed with time. Some of the parameters (e.g., covariances in the noise expressions) are used as fine-tuning variables. The algorithm is articulated in 2 different steps: Prediction or Propagation, and Update or Correction. The following list of actions summarize those steps:

\[ \bar{X}_n^{-} = F\bar{X}_{n-1}^{-} + BU_{n-1} \] \hfill (A.3)  
\[ \bar{P}_n^{-} = FP_{n-1}F^T + Q \] \hfill (1b: Predicted error covariance) \hfill (A.4)  
\[ \bar{Z}_n = \bar{Y}_n - H\bar{X}_n^{-}F^T \] \hfill (2a: Measurement residual) \hfill (A.5)  
\[ K_n = \bar{P}_n^{-}H^T(R + HP_n^{-}H^T)^{-1} \] \hfill (2b: Kalman gain) \hfill (A.6)  
\[ \bar{X}_n^{+} = \bar{X}_n^{-} + K_n\bar{Z}_n \] \hfill (2c: Update estate estimate) \hfill (A.7)  
\[ \bar{P}_n^{+} = (I - K_nH)\bar{P}_n^{-} \] \hfill (2d: Update error covariance) \hfill (A.8)

Where the superscripts “-” and “+” stand for predicted and updated estimates, respectively. In the algorithm, the predicted estate estimate is obtained from the previous updated state estimate. \( P \), the state error covariance, becomes
larger at the prediction state because of the addition of $Q$, and this means the filter is more uncertain of the state after the prediction step.

$\bar{Z}$ corresponds with the measurement residual (or innovation), and it is the first thing to be computed in the update phase. It is the difference between the true measurement, $\bar{Y}_n$, and the estimated measurement $H\bar{X}_{n-1}$. The estimated residual is later multiplied by the called Kalman gain, $K_n$, to provide the correction to the predicted estimate $\bar{X}_n$. Finally, the filter calculates the new error covariance, $P$, which will be used in the next iteration. This time, $P$, is more certain of the state estimate after the measurement is used in the correction phase.

As usual, the algorithm needs an initial condition to start working, given typically by some good initial guesses of $\bar{X}$ and $P$. Those initial values, as well as $Q$ and $R$, have a crucial role in the results obtained by the algorithm. Sometimes, it is useful to select a large initial value for $P$, to get a faster convergence.

Kalman filters are developed assuming that both process and measurement models are linear (so the different matrices mentioned before, $F$, $B$ and $H$, are kept constant along the execution of the method). And that means, it would be only useful for the inverse problem of the turbofan engine when dealing with small degradations. That was one of the main reasons to avoid its incorporation to the final methodology. However, if the changes in degradations are not too big (which is expected to happen most of operational time), this technique could be reconsidered to follow soft degrading evolutions with time. In fact, when the process and measurement covariances are known, it can be proved the Kalman filter is the best possible linear estimator in the sense of the minimum mean square error sense.

Alternative versions of the classic Kalman filter algorithm, like the Extended Kalman Filter (EKF), provide some additional tools to deal with non-linear processes and measurements. In this case, it is necessary to calculate the Jacobian matrix of model's function, because the nonlinear functions in model and observations cannot be applied to covariance directly. The scheme of the algorithm is very similar to the one introduced for the linear case, but it assumes certain simplifications regarding the mean value of the model function and perform typically Taylor expansions as part of the calculation of the output's covariance. Nevertheless, those simplifications can be corrected to improve the accuracy of the method. Similar techniques have been successfully applied to navigation systems, projectile target tracking, etc.

Some other subsequent versions, like the Unscented Kalman Filter (UKF) carefully treat the Gaussian random variables by using a minimal set of previously chosen sample points that capture the true mean and variance of the Gaussian variables, and when propagated through the non-linear system, captures the posterior mean and covariance, accurately to the third order of the Taylor series expansion of the model function, for any non-linearity. This circumstance implies the obtention of better accuracies comparing with EKF.

Excepting GAs, none of the techniques used in the methodology needed of random variables, or statistical approaches whatsoever. Just the knowledge of the Objective Function, and the systematic use of the model, was enough to chase the convergence in the different solved cases (sometimes considerably challenging, given the peculiarities of the model in PROOSIS®).
The idea behind the use of conjugate directions over a quadratic form, or another function that locally counts with an approximately quadratic behavior (meaning counting with, at least, second derivatives) nearby its minimum, is using systematically the gradient of the Objective Function (OF) that must be optimized, which is equivalent to systematically consider direction of punctual maximum variation. However, contrarily to the steepest descent method, which uses only the same direction given by the gradient to advance towards the solution (adjusting the length of the step in each iteration), the Conjugate Gradient method alternates the directions by means of conjugate vectors to the gradient. Those conjugate vectors can be obtained by means of different approaches and formulas (e.g., Hestenes-Stiefel, Polak-Ribière-Polyak, Fletcher-Reeves, Liu-Storey, Dai-Yuan, etc.), fact that opens the door to certain sensitivity analysis to identify the best option for a particular kind of problem. Nevertheless, the methods associated to the two first seem to be the most efficient, as they perform a restart if a bad direction is followed.

The algorithm would be based on the following steps, based on the previous determination of OF’s gradient:

\[
\Delta \bar{X}_k = -\nabla f(\bar{X}_k) \quad \text{(Steepest descent direction)} \quad \text{(B.1)}
\]

\[
\gamma_k = \frac{\Delta \bar{X}_k^T \Delta \bar{X}_k}{(\Delta \bar{X}_{k-1}^T \Delta \bar{X}_{k-1})} \quad \text{(Conjugate factor } \gamma_k) \quad \text{(B.2)}
\]

\[
\bar{d}_k = \Delta \bar{X}_k + \gamma_k \bar{d}_{k-1} \quad \text{(Update of conjugate direction)} \quad \text{(B.3)}
\]

\[
\alpha_k = \min_{\alpha} f(\bar{X}_k + \alpha \bar{d}_k) \quad \text{(Complete a new search line with } \alpha_k) \quad \text{(B.4)}
\]

\[
\bar{X}_{k+1} = \bar{X}_k + \alpha_k \bar{d}_k \quad \text{(Position update)} \quad \text{(B.5)}
\]

Where the conjugate direction was obtained by means of the Fletcher-Reeves formula in the second step. The first step to initiate the algorithm is normally done following the steepest descent.

With purely quadratic functions, the minimum is obtained relatively quickly, in as many iterations as components has the condition vector \( \bar{X} \). However, non-quadratic functions count with a lower rate of convergence. In those cases, it is usual to need of the application of the steepest descent method instead. On the other hand, the conjugate method counts with certain advantages when dealing with ill-conditioned problems, where the steepest descent tends to follow inevitably crisscrossed trajectories. A combination of both, steepest and conjugate, could be advisable for certain problems. In this sense, the problem relative to the engine’s inverse problem would be a future candidate for such combined solution, depending on the applicable route to get to the final state from the initial state.

As it was already verified in the thesis, the methods based on Newton’s search direction (i.e., by the computation of the Hessian matrix of the function to optimize) reach convergence typically much faster, but after a remarkable increase in the complexity of the associated algorithm. Some new formulas for the conjugate directions are based in quasi-Newton methods.
APPENDIX C – Bayesian Belief Networks

The BBNs are also directly called Bayesian Networks. These are graphical models for probabilistic relationships among a set of variables. They provide an initially affordable way to apply the Bayes Theorem to complex problems, like the one treated in this thesis. The method represents a direct way to visualize the structure of the model, clarifies how the different variables are interrelated, and establishes an organized framework to run probability calculations.

The strength of this method (and one of the reasons why it was initially considered) relies on its suitability to circumstances in which one of the required inputs is not observed, or not correctly observed, as it happened with the $P_{45t}$. Most methods will generate an inaccurate prediction because as they do not incorporate the associated information, that loss avoids the full specification of the conditional dependence between variables. However, Bayesian networks provide a way to encode dependencies among variables to overcome such situations, by establishing a probabilistic model where some variables are conditionally independent.

The method uses the graph theory to represent, with nodes and edges, the relationships among variables of a specific problem. The graphs will be as complex as the relationships among variables established inside the model they represent. Nodes will be typically occupied by random variables, which will be related by means of the different linking edges. Those edges can be directed (the navigation of data is only possible in one direction, case of the Bayesian networks) or undirected (usual in Hidden Markov Models, and less restrictive). Directed edges imply that no cycles are possible (at least, in the way the directed graphs and cycles are usually defined). No loops of data are possible, avoiding getting stuck in one of them.

Bayesian probabilities are inherently subjective, based on a certain belief in an outcome (not in the past occurrence of one event). In this sense, the Bayesian networks are characterized by capturing the joint probabilities of the events represented by the model. The random variables in the problem are conditioned by one or more other variables, and those conditional relationships are defined with the edges in the graph of the network. In the model generated by this method, it is stated all the conditional independence assumptions for the known variables, defined by specific presence or lack of edges among variables, while allowing the presence of unknown variables.

The initial appealing approach to overcome a potential lack of instrumentation from a real engine with this method, was neutralized by its inherent difficult application. The method needs of defined random variables in the nodes of the associated graph, and those random variables must be interconnected in an effective way to represent the operation of a gas turbine engine. The probability distributions (and the graph structure) must be specified, and the only way to do so is by managing massive amount of data from the real system. Thus, the BBNs are models eminently data-driven, with all the drawbacks associated to them when trying to perform a quick diagnostic, prognostic, or performance estimation, in general, of a gas turbine engine.

When a BBN is ready, it can be used to make decisions, which is a valid feature for certain applications, but not particularly for monitoring purposes. It has been applied however to prognostic estimations in the field of the maintenance of gas turbine engines, as it was indicated in the thesis, but the complexity of the approaches resulted challenging when applied to such sophisticated systems.
One example of a simple BBN is shown in the next figure, when only 3 nodes are interconnected to apply a very basic form of the Bayes Theorem:

![Graph of A, B, and C nodes](image)

*Figure 140: Simplified version of a BBN. The graph shows its most relevant elements: nodes and edges.*

In this graph, A and C are dependent upon B. Those are the main relationships established, as indicated by the directed edges between nodes. So, the next conditional dependencies can be established:

\[
P(A/B) \quad \text{(A is conditionally dependent upon B)} \quad (C.1)
\]

\[
P(C/B) \quad \text{(C is conditionally dependent upon B)} \quad (C.2)
\]

It is clear in the graph that no effect is expected from A to C or vice versa. On the contrary, the conditional independencies (also crucial in this method) can be defined the following way:

\[
P(A/B,C) \quad \text{(A is conditionally independent from C)} \quad (C.3)
\]

\[
P(C/B,A) \quad \text{(C is conditionally independent from A)} \quad (C.4)
\]

In other words, either A is conditionally independent of C or A is conditionally dependent upon B in presence of C. It is also possible to obtain the probability of A and C, conditioned on B, as the following product:

\[
P(A,C/B) = P(A/B)*P(C/B) \quad \text{(joint probability of A and C, given B)} \quad (C.5)
\]

The variable B is not affected by any other, so its probability can be directly stated as P(B), as no parent nodes are upon it. After having defined such variables and their respective relationships, it is possible to obtain the joint probability:

\[
P(A,B,C) = P(A/B)*P(C/B)*P(B) \quad \text{(model joint probability)} \quad (C.6)
\]

All these simple relationships are used to generate much complex models and graphs that eventually can be used to represent the behavior of a very complex system. Several references were given in the thesis to review some examples in this sense.
APPENDIX D – Neural Networks

The NNs constitute a subset of machine learning techniques that are nowadays at the heart of the so-called deep learning algorithms. Their name and structure are inspired by the nature of the human brain and imitate the way the biological neurons interexchange signals among them.

NNs are formed with nodes, organized by different layers, between the input layer and the output layer. Each node, or artificial neuron, connects to another and has an associated weight and threshold. When the output in any node is above its specified threshold, the node gets activated, sending data to the next layer of nodes. Otherwise, no data passes to the next layer. The decision criteria in each neuron could be defined in the following way, once defined a certain weighed function for the inputs received from the previous nodes.

\[
f(X) = \begin{cases} 
1 & \text{if } \sum w_i \cdot X_i + \text{bias} \geq 0 \\
0 & \text{if } \sum w_i \cdot X_i + \text{bias} < 0 
\end{cases} 
\]  

(decision based on threshold) \hspace{1cm} (D.1)

The function and criterion in the previous expression is fully configurable. Playing with the different parameters in the previous function, node by node, it is possible to configure and calibrate the operation of the NN. It is possible to define an Objective Function with the values obtained in the output nodes to minimize its value, reaching to an optimum input.

The most usual kind of NNs is the feedforward, also known as formed by multi-layer perceptrons, and it is shown here a simplified version:

![Figure 141: Simplified version of a feedforward NN.](image)

This technique, although very popular today after 8 decades of development, needs of massive amount of data to train the network. Accurate networks for complex systems imply long training periods. Tasks in face recognition can last hours, and applications regarding gas turbine engines could mean days, or even weeks. On the contrary, the Google’s search algorithm is based on NNs. The inherent data driven philosophy of this technique, and the long training times required until getting configured, discarded its use for the purpose of the thesis.
APPENDIX E – Fuzzy Logic

The FL refers to a model of numerical computing that tries to get advantage of vague or imprecise statements, by establishing “levels of truth”. In other words, the method works with functions (so-called membership functions, defined in a way like the usual operators used in Boolean logic) that classify the inputs, potentially incomplete, ambiguous, or distorted, delivering an acceptable output, yet not accurate. Some of those operators refer to unprecise categories (e.g., very, somehow, etc.). It may result like a signal filter, but considerably more elaborated in certain cases. In fact, there is no systematic approach to create a FL-based algorithm and they are typically fully understandable only when remain simple. The methodology is very flexible and different kind of rules can be created to obtain a certain sort of output, given the expected ambiguous inputs.

The algorithms based on the use of FL are usually (but here there is a plethora of variants, ad hoc solutions, and creative alternatives) formed by the following basic steps:

- Definition of linguistic variables and terms.
- Construction of membership functions for them.
- Generation of a knowledge base of rules.
- Conversion of crisp data into fuzzy data sets by using membership functions.
- Evaluation of rules in the rule's base.
- Combination of results from each rule.
- Conversion of output data into non-fuzzy variables.

The previous steps can be summarized as a fuzzification process, followed by several actions in the so-called inference engine, to end up undoing the fuzzification and providing the required results. Since the output of a fuzzy system represents a consensus of all the inputs and all the established rules, fuzzy systems can be considered for applications where input values are not always available, or not totally trustworthy.

The main reason to discard its application for the thesis was the lack of accuracy associated to it. It can be employed as a decision-making system, but not as a solver to obtain accurately the state of degradation of a gas turbine engine. Its applicability could appear more clearly right after that engine condition has been obtained, to decide the next steps to follow based on the results of a different solver. However, the complexity associated to the kind of decisions that must be made regarding operation and maintenance with gas turbine engines predicts the need of a sophisticated fuzzification process that could result eventually useless, if the decision process associated to the implementation of the FL becomes too complex, or apparently lack of logic.
APPENDIX F – Hidden Markov Models

The Hidden Markov models are probabilistic frameworks where the observed data are modeled as a series of outputs generated by one of several internal states. HMMs handle data which can be represented as a sequence of observations over time and are based in the use of the so-called Hidden Markov chains, and in the assumption that the system being modeled can be assimilated into a Markov process. Such process counts with unobservable (i.e., hidden) states. These models are often used for computational sequencing analysis, and they rely, the same way the BBNs, in a graphical representation of the system. Each node represents a random variable.

The nature of the Markov chains’ design in these models (chains of data going through a linear statistical process based on certain rules) implies that there is an initial state, an initial observation, an initial probability distribution over states, and eventually a final state. The next step in the Markov chain will depend only in the current state. The model typically rests upon 2 main assumptions:

1. **Limited Horizon or Markov’s property**: The probability of being in a state at a time “t”, depend only on the state at the time “t-1”. So, the state in t represents a good summary of the past, to represent the future.

2. **Stationary Process Assumption**: The probability distribution over the next state, given the current state, will not change. So, states keep changing with time, but the underlying process remains stationary.

There is a state transition matrix that defines the probability of transitioning from one state to the next one, at any time. The probabilities associated to a particular set of events in a sequence are obtained as a product of the respective conditional probabilities upon the previous events in the sequence. There will be hidden states typically that cannot be observed. But the process is still a Markov process, so it is possible to infer certain information from the application of certain rules:

1. **Output independence observation**: Output observation is conditionally independent of all other hidden states and all other observations when given the current hidden state.

2. **Emission Probability**: There will be a probability for a hidden state of generating a specific output, given that state at the corresponding time was known.

As it happened with other similar techniques, the attractive of inferring hidden (not observable) states in variables, meaning the H&Q parameters from the output, which are the values form instrumentation, is balanced by the complexity associated with the model. The rules that apply to the different states in the chains are usually based on the Bayes Theorem. In this methodology, there are so many possible state sequences that cannot be enumerated. The efficient Viterbi algorithm is guaranteed to find the most probable state path given a sequence and an HMM. The Viterbi algorithm is a dynamic programming algorithm, so more complexity is added to the process. The method is based in pattern recognition upon the application of statistical inference, being not the best candidate for the application studied in the thesis.
technik.com/epartcatalo/g/productDetail.shtml?source=hitcom&pnno=3213KGB01 [Last time accessed on December 12, 2021].


[185] MTU Aero Engines, "Multiple options to keeping engine related costs low", [Content available online, released on January 19, 2018]. Available at: https://magnetsmro.com/blog/multiple-option-keeping-engine-related-costs-low/ [Last time accessed on December 12, 2021].


[197] MTU Aero Engines. [Online content]. Information available online at: https://www.mtu.de/maintenance/ [Last time accessed on December 12, 2021].

[198] MTU Aeroport, "MTU Maintenance Zhuhai’s world-class MRO shop keeps on growing" [Content available online]. Available online at: https://aeroreport.de/en/aviation/mtu-maintenance-zhuhai-s-world-class-mro-shop-keeps-on-growing/ [Last time accessed on December 12, 2021].


• Honeywell Pressure transducers: https://aerospace.honeywell.com/content/dam/aero/en/documents/learn/products/sensors/brochures/N61  
-311-000-002-LG1237_bro.pdf.

[323]. Product description from instrumentation manufacturers (oil – fuel systems) [Last time accessed on December 12th, 2021]:

[323]. Product description from instrumentation manufacturer (vibration sensors) [Last time accessed on December 12th, 2021]:


[325]. PROOSIS. [Online content]. Information available online at: https://www.ecosimpro.com/products/proosis/ [Last time accessed on December 12th, 2021].


[335]. Safran Aeroengines website. [Online content]. Information available online at: https://www.safran-group.com/companies/safran-aircraft-engines.


