Space Dynamics Group - UPM
Dpto de Física Aplicada a las Ingenierías Aeronáutica y Naval
Escuela Técnica Superior de Ingeniería Aeronáutica y del Espacio
Universidad Politécnica de Madrid

Regular propagators and other techniques in orbit determination problems

Ph.D Thesis Dissertation

Virginia Raposo Pulido
BSc in Mathematics/MSc in Astrophysics

supervised by
Jesús Peláez Álvarez
Full Professor in Aerospace Engineering
Hodei Urrutxua Cereijo
Assistant Professor in Aerospace Engineering

Madrid, July 2019

Tribunal nombrado por el Sr. Rector Magfco. de la Universidad Politécnica de Madrid, el día ..... de ..... de 2019.

Presidente: Dr. José Manuel Perales Perales

Vocal: Dr. Sebastián Ferrer Martínez

Vocal: Dr. Aaron Rosengren

Vocal: Dr. Manuel Sanjurjo Rivo

Secretario: Dr. Ana Laverón Simavilla

Suplente: Dr. Martín Lara Coira

Suplente: Dr. Juan Félix San Juan Díaz

Realizado el acto de defensa y lectura de la Tesis el día ..... de ..... de 2019 en la E.T.S.I. Aeronáutica y del Espacio.

Calificación .....

EL PRESIDENTE

LOS VOCALES

EL SECRETARIO

"¿Recuerdas aquel tipo, hará unos veinte años? No recuerdo su nombre. Subió al Everest sin oxígeno y bajó medio muerto. Le preguntaron: ¿Por qué subió allí para morir? Y él dijo: No. Subí para vivir."

ABSTRACT

The Space Surveillance Network detects and registers Earth-orbiting man-made objects of a size larger than 10 cm, currently accounting for over 24,000 objects, including nearly 2,000 active satellites and 3,000 non-operational satellites, with these numbers going up as new sensing capabilities become available in the coming years. The knowledge of their orbits is of paramount importance for two basic types of problems, namely the propagation and the determination of their orbits. In orbit propagation, the dynamical state of the satellite is known a priori at a given instant, and the aim is to compute the dynamical state at later instants. On the contrary, orbit determination is about estimating the dynamical state of the satellite at a given instant, making use of observation data from one or more tracking stations. Propagation and orbit determination are thus inverse processes and complementary to each other, since the determination process requires the propagation of the orbit, and vice versa. The application of these techniques naturally arises as a fundamental element in the field of space situational awareness, where automated computer systems routinely perform the precise orbit determination of both operational and inactive artificial satellites. Also, operational satellites require the precise knowledge of their orbits to predict the moment they fly over tracking stations, as well as to safely operate active satellites and plan collision avoidance maneuvers. Consequently, these automated systems not only need to be robust and reliable, but due the large amounts of data that need to be processed daily, they also need to be computationally efficient.

The aim of this thesis is to develop robust and efficient numerical methods to be applied in both orbit determination and orbit propagation problems.

The primary input for orbit determination are observations. For the determination of a Keplerian orbit six independent observations are required, although often many more are available. Thus, *preliminary* orbit determination is devoted to the estimation of a Keplerian orbit using only six observations. Observational data can be of a varied nature (angles only, range and range-rate, ...), leading to a wealth of well-known classical techniques, e.g. Gibbs, Herrick-Gibbs, Laplace and Gauss methods. Solving Kepler's equation is a commonplace to all of them. As a consequence, fast, accurate, robust and reliable methods for solving Kepler's equation are of a vital importance in preliminary orbit determination. This dissertation proposes one such algorithm that, based on an improved initial guess, succeeds to solve the elliptic and hyperbolic Kepler's equation more efficiently.

Another recurring element in many preliminary orbit determination methods is solving the two-body Lambert's problem. This dissertation revisits this problem from a new viewpoint, providing an innovative, insightful approach that enables a robust and efficient solution to the problem.

Real satellite orbits, however, are not Keplerian due to the many perturbing forces acting upon them, and thus six observations are not enough to realistically capture the dynamical state of the satellite. If more observations are available, a better estimate is possible; this is the focus of *statistical* orbit determination, that aims to determine the initial state of a satellite from multiple observations and under a realistic, non-Keplerian dynamical model. The corresponding techniques are usually based on a linearization around the outcomes of preliminary orbit determination, leading to a succession of values converging to the sought solution, so that the propagation with the complete dynamical model yields an orbital solution that provides the best possible fit to the actual observations. Consequently, these methods rely

on robust, accurate and efficient orbit propagation methods, so the choice of an appropriate propagation technique is fundamental. In this regard, classical methods (e.g. Cowell's method) are commonly used; these methods, although effective and simple to implement, have the drawback of an exponential error growth and singular equations of motion. Conversely, regularized orbital formulations, such as DROMO, Kustaanheimo-Stiefel, or Sperling-Bürdet, allow to reformulate the equations of motion so that the orbital dynamics is described by a system of non-singular equations, which facilitates the numerical integration and reduces the negative effects of the Lyapunov instability associated to Keplerian motion, thus providing far more accurate and efficient propagation routines. With this vision in mind, the second part of this dissertation is devoted to orbit propagation with regularized formulations, and makes new contributions to this field by developing a new reformulation of DROMO, which is specifically tailored for highly-perturbed dynamical environments.

This dissertation is organized as follows:

- The first part covers the research contributions to orbit determination. Chapter 1 introduces the basics on orbit determination and describes the classical methods; Chapters 2 and 3 present novel methods for solving the elliptic and hyperbolic Kepler's equation, respectively; Chapter 4 presents a new formulation, valid for any type of orbit, to solve the two-body Lambert's problem.
- The second part of the dissertation focuses on orbit propagation with regularized formulations. Chapter 5 introduces the basics on regularized special perturbation methods; Chapters 6 and 7 describe in detail the DROMO and ElliDROMO regularized formulations, respectively; Chapter 8 extends the two formulations to enhance their performance in strongly perturbed environments, leading to new methods referred to as DROMO-SPE and ElliDROMO-SPE; these propagators are tested in different scenarios and their performance is shown; Chapter 9 explores the application of DROMO and DROMO-SPE to the propagation of meteor trajectories and presents a sensitivity analysis of the parameters considered in the physical modeling. Finally, Chapter 10 summarizes the main conclusion of the thesis and traces potential avenues for future work.

ACTIVITIES OF THE DOCTORAL RESEARCH

As result of the doctoral research I have had the opportunity to attend international conferences, to publish journal articles and participate in several research projects. The following sections list the scientific and academic activities to which I have contributed.

PUBLISHED JOURNAL ARTICLES

1. D. de la Torre Sangrà, V. Raposo-Pulido, E. Fantino, J. Peláez; Review of Lambert's problem (in preparation).
2. E. Fantino, R. Flores, J. Peláez, V. Raposo-Pulido; Optimization of Saturn orbit insertion impulse using deep-space low thrust. *Journal of Guidance, Control, and Dynamics* (submitted 12/05/2019).
3. V. Raposo-Pulido, J. Peláez; An efficient code to solve the Kepler equation. Hyperbolic case. *Astronomy and Astrophysics* 2018; 619, A129. doi: 10.1051/0004-6361/201833563.
4. V. Raposo-Pulido, J. Peláez; An efficient code to solve the Kepler equation. Elliptic case. *Monthly Notices of the Royal Astronomical Society* 2017; 467 (2): 1702 – 1713. doi: 10.1093/mnras/ stx138.

CONFERENCE PRESENTATIONS WITH PROCEEDINGS

1. V. Raposo-Pulido, J. Peláez; DROMO propagator for highly perturbed problems, 29th AAS/AIAA Space Flight Mechanics Meeting 2019 (13 – 17/01/2019, Ka'anapali, Hawaii).
2. E. Fantino, J. Peláez, R. Flores, V. Raposo-Pulido; Gravitational capture at Saturn with low-thrust assistance, 29th AAS/AIAA Space Flight Mechanics Meeting 2019 (13 – 17/01/2019, Ka'anapali, Hawaii).
3. V. Raposo-Pulido, J. Peláez; On the resolution of the Lambert's problem with the SDG-code, 27th AAS/AIAA Space Flight Mechanics Meeting 2017 (5 – 9/02/2017, San Antonio, Texas).
4. V. Raposo-Pulido, J. Peláez; Solving the Kepler equation with the SDG-code, 27th AAS/AIAA Space Flight Mechanics Meeting 2017 (5 – 9/02/2017, San Antonio, Texas).
5. V. Raposo-Pulido, J. Peláez; An efficient code to solve the Kepler's equation for elliptic and hyperbolic orbits, 6th International Conference on Astrodynamics Tools and Techniques (14 – 17/03/2016 Darmstadt, Germany).

CONFERENCE PRESENTATIONS WITHOUT PROCEEDINGS

1. V. Raposo-Pulido, J. Peláez; The new SDG-code to solve the Kepler equation, Stardust-Asteroids and Space Debris Conference (31/10/2016 – 03/11/2016 ESTEC, Netherlands).

2. V. Raposo-Pulido, J. Peláez; An efficient code to solve the Kepler equation, Key Topics in Orbit Propagation Applied to Space Situational Awareness (28 – 30/10/2015 Toulouse, France).

REGISTERED SOFTWARE

1. V. Raposo-Pulido, J. Peláez; The EKE-SDG code to solve the elliptic Kepler equation (2018, Registro territorial de la propiedad intelectual, Comunidad de Madrid). Registre number: M-006673/2018.
2. V. Raposo-Pulido, J. Peláez; The HKE-SDG code to solve the hyperbolic Kepler equation (2018, Registro territorial de la propiedad intelectual, Comunidad de Madrid). Registre number: M-006230/2018.

ATTENDANCE AT WORKSHOPS, SEMINARS AND COURSES

1. Human Resources Strategy for Researchers (HRS4R) (03/2018, ETSIAE-UPM, Madrid, Spain).
2. Methodology and Scientific Documentation (03 – 04/2016, ETSIAE-UPM, Madrid, Spain).
3. Stability of Physical Systems I - Master in Industrial Mathematics (09/2015 – 01/2016 ETSIAE-UPM, Madrid, Spain).

TEACHING ASSIGNMENTS

1. Physics I - Degree in Aerospace Engineering (2017 – 2018 ETSIAE-UPM, Madrid, Spain) - 13h.
2. Physics I - Degree in Aerospace Engineering (2016 – 2017 ETSIAE-UPM, Madrid, Spain) - 22h.
3. Physics I - Degree in Aerospace Engineering (2015 – 2016 ETSIAE-UPM, Madrid, Spain) - 17h.

RESEARCH INTERNSHIPS

1. Space Systems Laboratory, Tokyo Metropolitan University: Trajectory of artificial meteors (04 – 06/2019, Tokyo, Japan).

RESEARCH PROJECTS

1. Dynamical analysis of complex interplanetary missions (2018 – 2020, Ministerio de Economía, Industria y Competitividad ESP2017 – 87271-P). Principal investigators: Jesús Peláez and Manuel Ruiz.
2. Dynamic analysis, advanced orbital propagation and simulation of complex space systems (2014 – 2017, Ministerio de Economía, Industria y Competitividad ESP2013 – 41634-P). Principal investigators: Jesús Peláez and Manuel Ruiz.

ACKNOWLEDGEMENTS

Han pasado 4 años, 2 meses y 25 días desde que empecé las tesis doctoral hasta la lectura de la misma. Una maratón tiene un total de 40 km, 20 hm y 195 m. Los primeros 30 km se corren con las piernas, los siguientes 10 km con la cabeza, y los últimos 2.195 m con el corazón. Así ha sido mi tesis doctoral, y como en la maratón, el triunfo en sí mismo ha sido continuar hasta el final.

Quiero dedicar esta tesis a mis padres, que me enseñaron desde pequeña el valor del esfuerzo y cuyo apoyo ha sido constante. A mi hermana, mi otra mitad, la persona que mejor me conoce y que siempre me acompaña. Y por último, a mis abuelos Antonio y Magdalena, a quienes les debo todo lo que soy y por quien siento el mayor amor inimaginable. Siempre estaréis conmigo allá donde vaya.

La tesis nunca podría haberse completado sin la ayuda de mi supervisor, Jesús Peláez. Le quiero agradecer las conversaciones, explicaciones y múltiples sesiones de skype durante estos cuatro años. No podría haber recibido mejor orientación ni mayor apoyo. Gracias por confiar en mí para realizar la tesis y por enseñarme el mundo que hay detrás de la Astrodinámica. Ha sido una maravillosa experiencia de la cual he extraído importantes lecciones. También quiero agradecer a Hodei Urrutxua, mi segundo supervisor, su ayuda y disponibilidad siempre que lo he necesitado, ha sido un pilar fundamental.

Como olvidarme de mis compañeros de doctorado y de universidad. Todo hubiera sido más arduo sin la ayuda y apoyo inestimable de Juancho, Davide, Pablo, Víctor, Jesús y Javier. Gracias por las conversaciones con café (o sin él) y animarme siempre que lo necesité. Me llevo vuestra amistad, que no es poco.

Quiero agradecer al Ministerio de Educación, Cultura y Deporte por concederme el contrato predoctoral dentro del programa "Formación de Personal Investigador (FPI)" que me ha permitido desarrollar la tesis doctoral. Gracias a Mari Carmen Velasco y a María Jesús de Andrés García por ayudarme con los múltiples formularios y documentos a lo largo del doctorado, por facilitarme las cosas y tratarme siempre con una sonrisa.

Tampoco puedo olvidarme de mis compañeros de Space Systems Laboratory en Tokyo Metropolitan University, en especial de Marin y Kei por su cálida acogida. Agradecer al Prof. Hironori Sahara que me permitiera trabajar en su laboratorio y aprender sobre la dinámica de meteoritos. Fueron tres meses que jamás olvidaré, por muchas razones. A todos ellos, Arigato gozaimasu. Agradezco al Consejo Social de la Universidad Politécnica de Madrid que me brindase la ayuda para realizar esta estancia de investigación.

Por último, y no menos importante, gracias Ainoa, David, Arantxa, Andrea, Susana, Gonzalo, Cristina y Carlos por caminar a mi lado de nuevo.

CONTENTS

I	ORBIT DETERMINATION	1
1	PRELIMINARY AND STATISTICAL ORBIT DETERMINATION	3
1.1	Determination from complete data	3
1.1.1	Spherical Earth	4
1.1.2	Ellipsoidal Earth	5
1.2	Gibbs method	7
1.2.1	Calculation of the <i>semilatus rectum</i> p	8
1.2.2	Calculation of the eccentricity e	8
1.2.3	Calculation of the perifocal trihedral	9
1.2.4	Observations	9
1.3	Herrick-Gibbs method	10
1.3.1	Observations	12
1.4	Laplace method	12
1.4.1	Determination of ρ	14
1.4.2	Determination of $\dot{\rho}$	14
1.4.3	Lagrange equation	15
1.4.4	Observations	16
1.5	Gauss method	16
1.5.1	Observations	18
1.6	The differential correction method	18
1.6.1	Observations	21
1.7	Statistical orbit determination	21
1.7.1	The least squares adjustment	24
1.7.2	Kalman filtering	25
2	THE ELLIPTIC KEPLER'S EQUATION	27
2.1	Introduction	27
2.2	The equation	29
2.2.1	Domain of the equation	30
2.3	Newton-Raphson algorithms	30
2.4	The seed value	32
2.5	Analysis in the singular corner	33
2.5.1	Inner region	35
2.5.2	Intermediate-outer region	36
2.6	The EKE-SDG algorithm	37
2.7	Analysis of results	37
2.7.1	Speed of the process	38
2.7.2	Accuracy analysis	39
2.7.3	Improving the code	41
2.8	Comparison of results	43
2.9	Applications of the EKE-SDG code	47
3	THE HYPERBOLIC KEPLER'S EQUATION	49
3.1	Introduction	49
3.2	The equation	51
3.3	Newton-Raphson algorithms	52
3.4	The seed value	53
3.4.1	Analysis in the singular corner	53
3.4.2	Inner region	54
3.4.3	Outer region	54

3.4.4	Intermediate region	56
3.4.5	Summary of the singular corner	58
3.5	Analysis outside the singular corner	58
3.5.1	Polynomial approximation	58
3.5.2	Asymptotic approximation	60
3.5.3	Improving the code	62
3.6	The HKE-SDG algorithm	63
3.7	Analysis of results	64
3.7.1	Speed of the process	64
3.7.2	Accuracy analysis	65
3.8	Comparison of results	67
3.9	Applications of the HKE-SDG code	71
4	LAMBERT'S PROBLEM	73
4.1	Introduction	73
4.2	Basic elements	74
4.3	Geometric aspects of the analysis	74
4.3.1	Minimum-energy ellipse	75
4.3.2	Fundamental ellipse	76
4.4	The free parameter χ	78
4.4.1	Eccentricity of the orbits	80
4.4.2	The plane (b, ν)	81
4.4.3	Semi-major axis and <i>semilatus rectum</i> of the orbits	83
4.4.4	The conjugate ellipse	83
4.4.5	The conjugate hyperbola	85
4.5	Parametric analysis	85
4.5.1	Family of conics with $b > 1$	87
4.5.2	Family of conics with $b < 1$	91
4.6	The time of flight equation	95
4.6.1	Elliptic case	95
4.6.2	Parabolic case	96
4.6.3	Hyperbolic case	96
4.7	Alternative formulation	96
4.7.1	Elliptic case	96
4.7.2	Geometric aspects of α and β	98
4.7.3	Hyperbolic case	100
II	ORBIT PROPAGATION	103
5	SPECIAL PERTURBATION METHODS	105
5.1	Numerical integration	106
5.1.1	Errors and integration methods	106
5.1.2	Instabilities in the Keplerian motion	108
5.2	Classical methods	108
5.2.1	Cowell's method	108
5.2.2	Encke's method	109
5.2.3	Error propagation	110
5.3	The regularized propagators	110
5.3.1	The Sperling-Bürdet (SB) regularization	111
5.3.2	The Stiefel regularization for planar motions	114
5.3.3	The Kustaanheimo-Stiefel (KS) regularization	118
5.3.4	The Palacios regularization	121
6	THE DROMO REGULARIZATION	127
6.1	Basic elements of the theory	127
6.2	Two-body problem	128

6.2.1	Position of the orbital plane	128
6.3	Kinematics of the orbital plane	131
6.3.1	Definition of the orbital plane	131
6.3.2	Composition of motions	132
6.3.3	Dimensionless quantities	133
6.3.4	Calculation of the angular velocities $\vec{\omega}_{01}$ and $\vec{\omega}_{31}$	134
6.3.5	Position of the axes Oxyz	135
6.3.6	Time evolution of the axes Oxyz	137
6.4	Dynamics of the orbital plane	137
6.4.1	Drag motion 0/1	137
6.4.2	Relative motion 2/0	138
6.4.3	Basic formulation of DROMO	143
6.4.4	A compact formulation	144
6.5	Summary of equations	147
6.5.1	Using the quaternion linked to $Ox_0y_0z_0$	147
6.5.2	Using the quaternion linked to $Ox_py_pz_p$	148
6.6	Singularities	149
6.7	Useful relations	150
6.7.1	Position vector	150
6.7.2	Velocity vector	151
6.7.3	Eccentricity vector	152
7	THE ELLIDROMO REGULARIZATION	153
7.1	Basic elements of the theory	153
7.2	The ElliDROMO Formulation	154
7.2.1	Solution of the unperturbed problem	156
7.2.2	Solution of the perturbed problem	156
7.2.3	The independent variable u	157
7.3	Relative motion	160
7.3.1	Velocity vector $\vec{v}$	160
7.3.2	Angular momentum vector $\vec{h}$	161
7.3.3	Eccentricity vector $\vec{e}$	161
7.3.4	Governing equations	162
7.4	Position of the axes Oxyz	163
7.4.1	Determination of v	163
7.4.2	Initial conditions	164
7.5	Summary of equations	165
7.5.1	Time Element	165
7.6	Useful relations	166
7.6.1	Position vector	166
7.6.2	Velocity vector	167
7.6.3	Eccentricity vector	168
8	STRONGLY PERTURBED ENVIRONMENTS	169
8.1	The DROMO-SPE formulation	169
8.1.1	Summary of equations	169
8.1.2	Useful relations	171
8.2	The ElliDROMO-SPE formulation	172
8.2.1	Summary of equations	173
8.2.2	Useful relations	174
8.3	Tsien's problem	176
8.3.1	Analysis	176
8.3.2	Comparison of results	177
8.4	Artificial satellite orbiting Pluto	180
8.4.1	Analysis	180

8.4.2	Comparison of results	183
8.5	Applicability domain of the SPE formulation	185
9	TRAJECTORY OF ARTIFICIAL METEORS	189
9.1	Introduction	190
9.2	Physical modeling	192
9.2.1	Cross-section area	192
9.2.2	Atmospheric parameters	192
9.2.3	Surface temperature	193
9.2.4	Drag coefficient	193
9.2.5	Heat transfer coefficient	194
9.3	Propagation of the orbit	194
9.3.1	Initial position	195
9.3.2	Initial velocity	195
9.3.3	Initial mass	196
9.3.4	Mathematical formulations	196
9.4	Scenarios	198
9.4.1	Sensitivity analysis	198
9.4.2	Comparison of formulations	201
10	CONCLUSIONS AND FUTURE WORK	217
III	APPENDIX	219
A	THE FIFTH DEGREE POLYNOMIAL $p_1(x, e)$	221
B	THE EKE-SDG CODE	223
B.0.1	<ellipk.h>	223
B.0.2	<polinomios24.c>	224
B.0.3	<keplere.c>	225
B.0.4	main.c	233
C	THE HKE-SDG CODE	239
C.0.1	<hyperk.h>	239
C.0.2	<polinomios25.c>	241
C.0.3	<polinomios25Q.c>	242
C.0.4	<keplerh.c>	245
C.0.5	main.c	254
	BIBLIOGRAPHY	259

Part I

ORBIT DETERMINATION

PRELIMINARY AND STATISTICAL ORBIT DETERMINATION

The preliminary orbit determination is used for the direct computation of six elements from six observations with no *a priori* knowledge of the spacecraft orbit. Preliminary orbit determination is of great importance for solar system bodies like comets and minor planets immediately following their detection. However, its significance for satellite orbits is limited by regular tracking campaigns for most satellites and orbital-element databases of reasonable accuracy. Nevertheless, a preliminary orbit determination is still required in the case of launcher injection errors or for the identification of an uncatalogued spacecraft (Montenbruck and Gill [56]).

We begin by considering the solution of the problem in a simple case, that is, when the observations provided by a single tracking station on the Earth's surface are available. To be more precise, the tracking station has a radar that is equipped with a device capable for measuring, in addition to the distance ρ to the object, its radial velocity $\dot{\rho}$ by means of the frequency offset associated with the Doppler effect. Besides, the antennas by which the object is followed, which we will call M , provide the angles $(\hat{A}, \hat{a})$, azimuth and elevation, associated with the horizontal coordinates attached to the tracking station, as well as their temporal derivatives $(\dot{\hat{A}}, \dot{\hat{a}})$.

1.1 DETERMINATION FROM COMPLETE DATA

We start from two reference systems (Urban and Kenneth Seidelmann [90]):

- The inertial system $Ex_1y_1z_1$ (Fig. 1.1), which matches with the so called Earth-Centered Inertial (ECI). Its origin (E) is the center of mass of the Earth, and the plane Ex_1y_1 is the equatorial plane with the axis Ex_1 pointing to the vernal equinox Υ . Although it is not an inertial reference, since its center is accelerated following an elliptical trajectory around the Sun, in short periods of time it can be assumed that it follows a straight line with constant velocity and, therefore, that it is inertial.
- The horizontal system $Oxyz$ (Fig. 1.1). Its origin (O) represents the position of the tracking station on the Earth's surface, with Ox pointing to the East and Oy to the North.

This way, the Earth is assumed to rotate with angular velocity $\vec{\omega}_0$ around the axis Ez_1 , that is, precession, nutation and pole movement are neglected. Likewise, the geographic coordinates (λ, φ) of the tracking station are assumed to be known, which provides observations that allow to determine the values of the position on the Earth's surface $\vec{\rho}$ and its derivative $\dot{\vec{\rho}}$ in a given instant t . However, the goal is to obtain the components of the state vector $(\vec{r}, \vec{v})$ of the satellite in that instant. Thus, the observations allow to determine the dynamical state detected from the tracking station $Oxyz$, but it is desired to determine the dynamical state of the object in its motion with respect to the inertial reference $Ex_1y_1z_1$. If (x, y, z) are the local coordinates of M , that is, the coordinates of the vector $\vec{\rho}$ on the axes $Oxyz$, their relation with the elevation and azimuth turns out to be:

$$x = \rho \cos(\hat{a}) \sin(\hat{A}) \quad (1.1a)$$

$$y = \rho \cos(\hat{a}) \cos(\hat{A}) \quad (1.1b)$$

$$z = \rho \sin(\hat{a}). \quad (1.1c)$$

Figure 1.1: Reference systems.

Depending on the considered shape model of the Earth, the estimation of the dynamical state of the satellite may present variations.

1.1.1 Spherical Earth

If the Earth is considered as a sphere with radius R_E , the relation between the two references is $[\vec{i}, \vec{j}, \vec{k}] = [\vec{i}_1, \vec{j}_1, \vec{k}_1]Q$, where the orthogonal matrix Q is given by:

$$Q = \begin{pmatrix} -\sin(\theta) & -\sin(\varphi)\cos(\theta) & \cos(\varphi)\cos(\theta) \\ \cos(\theta) & -\sin(\varphi)\sin(\theta) & \cos(\varphi)\sin(\theta) \\ 0 & \cos(\varphi) & \sin(\varphi) \end{pmatrix}, \quad (1.2)$$

where the angle φ coincides with the geocentric latitude of the station O . However, the angle θ is the local sidereal time, i.e., $\theta(t) = \theta_G(t) + \lambda$, being t the instant when the station provides the observables of the satellite (in UT₁), λ the geographic longitude of the station O and θ_G the sidereal time of Greenwich that can be obtained from the formula of Newcomb (Vallado [91]) and the angular velocity of the Earth $\omega_0 = 7.2921158553 \times 10^{-5}$ rad/s:

$$\theta_G(t) = \theta_G(t_0) + \omega_0(t - t_0) \quad (1.3)$$

$$\theta_G(t_0) = 100.4606184^\circ + 36000.77005361T_{UT1} + 0.00038793T_{UT1}^2 - 2.6 \cdot 10^{-8}T_{UT1}^3, \quad (1.4)$$

where T_{UT1} is the number of Julian centuries elapsed since the Julian date 2451545.0 JD (12h 0' 0" UT₁ January 1, 2000) until the instant t_0 (referred to 0h 0' 0" UT₁, for the same date as t):

$$T_{UT1} = \frac{JD - 2451545}{36525}, \quad (1.5)$$

such that JD is the Julian date referred to t_0 .

Although in a strict sense the formula of Newcomb provides the Greenwich mean sidereal time of Greenwich, since we are neglecting the precession, we can use it without major problem. Thus, we shall define the position vectors $\vec{r}$ and $\vec{\rho}$ as:

$$\vec{r} = x_1\vec{i}_1 + y_1\vec{j}_1 + z_1\vec{k}_1 \quad (1.6)$$

$$\vec{\rho} = x\vec{i} + y\vec{j} + z\vec{k}, \quad (1.7)$$

where $\vec{r}$ is geocentric and $\vec{\rho}$ is measured from the station. They are linked by the equation: $\vec{r} = R_E \vec{k} + \vec{\rho}$. Therefore, the relation between the two sets of coordinates of the body M will be:

$$\begin{pmatrix} x_1 \\ y_1 \\ z_1 \end{pmatrix} = Q \begin{pmatrix} x \\ y \\ R_E + z \end{pmatrix}. \quad (1.8)$$

The inertial velocity $\vec{v}$ of the body is the sum of the relative to Oxyz plus the drag velocity of this reference: $\vec{v} = \dot{\vec{\rho}} + \vec{\omega}_0 \times \vec{r}$, where:

$$\vec{v} = \dot{x}_1 \vec{i}_1 + \dot{y}_1 \vec{j}_1 + \dot{z}_1 \vec{k}_1 \quad (1.9)$$

$$\dot{\vec{\rho}} = \dot{x} \vec{i} + \dot{y} \vec{j} + \dot{z} \vec{k}. \quad (1.10)$$

Since the Oxyz axes, attached to the Earth, rotate with respect to the inertial reference, and $\vec{\omega}_0 = \omega_0 \vec{k}_1$, we have $\vec{\omega}_0 \times \vec{r} = \omega_0(-y_1 \vec{i}_1 + x_1 \vec{j}_1)$. This allows to express the inertial velocity as:

$$\begin{pmatrix} \dot{x}_1 \\ \dot{y}_1 \\ \dot{z}_1 \end{pmatrix} = Q \begin{pmatrix} \dot{x} \\ \dot{y} \\ \dot{z} \end{pmatrix} + \omega_0 \begin{pmatrix} 0 & -1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 0 \end{pmatrix} \begin{pmatrix} x_1 \\ y_1 \\ z_1 \end{pmatrix}, \quad (1.11)$$

where the values of $(\dot{x}, \dot{y}, \dot{z})$ are obtained by differentiating Eqs. (1.1a), (1.1b) and (1.1c) with respect to time.

In summary, the longitude λ and the latitude ϕ of the tracking station, and the Earth radius R_E are assumed to be known. The determination of the orbit is carried out by the following procedure:

1. At a known instant of time t the tracking station provides the values of t , ρ , $\dot{\rho}$, $\hat{a}$, $\dot{\hat{a}}$, $\hat{\Lambda}$, and $\dot{\hat{\Lambda}}$.
2. From t and λ the angle θ is determined.
3. With θ and ϕ known, the matrix Q is obtained.
4. From ρ , $\hat{a}$ and $\hat{\Lambda}$ the values of (x, y, z) are calculated by Eqs. (1.1a), (1.1b) and (1.1c).
5. From ρ , $\dot{\rho}$, $\hat{a}$, $\dot{\hat{a}}$, $\hat{\Lambda}$ and $\dot{\hat{\Lambda}}$ the values of $(\dot{x}, \dot{y}, \dot{z})$ are estimated by their expressions derived from Eqs. (1.1a), (1.1b) and (1.1c).
6. Finally, the matrix equations (1.8) and (1.11), allow to determine the dynamical state of the system in the inertial reference.

1.1.2 Ellipsoidal Earth

A more realistic model considers the Earth as an ellipsoid of revolution around the line of poles. This is defined by the equatorial radius (a) and the eccentricity (e) of the meridian ellipse or the flattening (f). In this case we assume that the tracking station is located at a point O of the Earth whose geodetic coordinates (λ , ϕ_g , h) are known. These coordinates assume that a specific reference ellipsoid is being used (WGS84¹, GRS80², ED50³ etc).

¹ https://earth-info.nga.mil/GandG/publications/tr8350.2/tr8350_2.html

² http://geoweb.mit.edu/~tah/12.221_2005/grs80_corr.pdf

³ https://georepository.com/datum_6230/European-Datum-1950.html

The coordinates (x_1^s, y_1^s, z_1^s) of the tracking station in the reference $Ex_1y_1z_1$ (Fig. 1.2) are given by:

$$x_1^s = \left(\frac{l_n}{1-e^2} + h \right) \cos(\phi_g) \cos(\theta) \quad (1.12a)$$

$$y_1^s = \left(\frac{l_n}{1-e^2} + h \right) \cos(\phi_g) \sin(\theta) \quad (1.12b)$$

$$z_1^s = (l_n + h) \sin(\phi_g), \quad (1.12c)$$

where l_n , distance between the point O and the axis Ex_1 measured over the normal in O to the ellipsoid, is known and is given by:

$$l_n = \frac{a(1-e^2)}{\sqrt{1-e^2 \sin^2(\phi_g)}}. \quad (1.13)$$

If the Earth is considered as an ellipsoid of revolution, the relation between the two references turns out to be $[\vec{i}, \vec{j}, \vec{k}] = [\vec{i}_1, \vec{j}_1, \vec{k}_1]Q$, where the orthogonal matrix Q is now given by:

$$Q = \begin{pmatrix} -\sin(\theta) & -\sin(\phi_g) \cos(\theta) & \cos(\phi_g) \cos(\theta) \\ \cos(\theta) & -\sin(\phi_g) \sin(\theta) & \cos(\phi_g) \sin(\theta) \\ 0 & \cos(\phi_g) & \sin(\phi_g) \end{pmatrix}. \quad (1.14)$$

The change of coordinates between the two references is governed by the relation: $\vec{r} = E\vec{O} + \vec{\rho}$, which in matrix form takes the following expression:

$$\begin{pmatrix} x_1 \\ y_1 \\ z_1 \end{pmatrix} = \begin{pmatrix} x_1^s \\ y_1^s \\ z_1^s \end{pmatrix} + Q \begin{pmatrix} x \\ y \\ z \end{pmatrix}. \quad (1.15)$$

The inertial velocity $\vec{v}$ of the body is the sum of the velocity relative to $Oxyz$ plus the drag velocity of this reference: $\vec{v} = \dot{\vec{\rho}} + \omega_0 \times \vec{r}$.

The analysis is identical to the one developed in the previous section, because the $Oxyz$ axes, linked to the Earth, rotate with respect to the inertial reference. Therefore, taking into account that $\vec{\omega}_0 = \omega_0 \vec{k}_1$, we get: $\vec{\omega}_0 \times \vec{r} = \omega_0(-y_1 \vec{i}_1 + x_1 \vec{j}_1)$. This relation leads to the matrix equation (1.11) but now the matrix Q is the given by Eq. (1.14). The values of $(\dot{x}, \dot{y}, \dot{z})$ are given by the same relations as in the previous section.

In summary, the longitude λ and the geodetic latitude ϕ_g of the tracking station are assumed to be known, as well as the semi-major axis and eccentricity of the reference ellipsoid of the Earth. The determination of the orbit is carried out with the following procedure:

1. In a given and known instant of time t , the tracking station provides the values of $t, \rho, \dot{\rho}, \hat{a}, \dot{\hat{a}}, \hat{A}$ y $\hat{\Lambda}$.
2. From the geodetic coordinates of the station, the values of (x_1^s, y_1^s, z_1^s) are determined by the relations (1.12a), (1.12b) and (1.12c).
3. From t and λ the angle θ is determined.
4. With θ and ϕ_g known, the matrix Q is determined from Eq. (1.14).
5. Starting from $\rho, \hat{a}$ and $\hat{A}$ the values of (x, y, z) are determined by the Eqs. (1.1a), (1.1b) and (1.1c).

Figure 1.2: Reference systems for an ellipsoidal Earth.

6. With ρ , $\dot{\rho}$, $\hat{a}$, $\dot{\hat{a}}$, $\hat{A}$ and $\dot{\hat{A}}$ the values of $(\dot{x}, \dot{y}, \dot{z})$ are determined by its expressions derived from Eqs. (1.1a), (1.1b) and (1.1c).
7. The matrix equation (1.15) allows to determine the inertial coordinates of the position vector $\vec{r}$.
8. Finally, the matrix equation (1.11) provides the inertial coordinates of the velocity vector $\vec{v}$.

1.2 GIBBS METHOD

Frequently, tracking stations do not have devices that can measure the radial velocity $\dot{\rho}$ by using the Doppler effect. In this case, only position measurements are available but not speed measurements. In this section the following problem is addressed: given three observations that allow to obtain the position vectors $\vec{r}_1$, $\vec{r}_2$ and $\vec{r}_3$, in the inertial reference $E x_1 y_1 z_1$, identify the orbit they determine. For this it is assumed that the three vectors are coplanar and independent, that is, the points they define are not aligned with each other, nor any two of them are aligned with E. These three vectors constitute the starting data of the process that is illustrated in this section. To determine the orbit, the *semilatus rectum* (p), the eccentricity (e) and the unit vectors $\vec{u}_1$, $\vec{u}_2$ and $\vec{u}_3$ of the perifocal trihedron will be found. The eccentricity vector $\vec{e}$, which defines the unit vector $\vec{u}_1$, is unknown. From its definition:

$$\vec{e} = -\frac{\vec{r}}{r} - \frac{1}{\mu}(\vec{h} \times \vec{v}), \quad (1.16)$$

in which the velocity vector $\vec{v}$ appears, which is not included in the observations, we can obtain the expression $\vec{e} \cdot \vec{r} = p - r$, with $p = \frac{h^2}{\mu}$ to be used throughout the development.

The starting data allows to determine two orbits that are different only in the direction of the angular momentum vector $\vec{h} = \vec{r} \times \vec{v}$. Therefore, additional data is needed to determine the direction of motion. The eccentricity vector $\vec{e}$ is constant upon changes in direction.

1.2.1 Calculation of the semilatus rectum p

The position vectors $\vec{r}_1$, $\vec{r}_2$, $\vec{r}_3$ are coplanar, and consequently they can be written as:

$$\vec{r}_1 \times \vec{r}_2 = \alpha \vec{u}_3 \Rightarrow \alpha = (\vec{r}_1 \times \vec{r}_2) \vec{u}_3 \quad (1.17)$$

$$\vec{r}_1 \times \vec{r}_3 = \beta \vec{u}_3 \Rightarrow \beta = (\vec{r}_1 \times \vec{r}_3) \vec{u}_3 \quad (1.18)$$

$$\vec{r}_2 \times \vec{r}_3 = \gamma \vec{u}_3 \Rightarrow \gamma = (\vec{r}_2 \times \vec{r}_3) \vec{u}_3, \quad (1.19)$$

where the scalars (α, β, γ) are known as they can be calculated in terms of known data. The unit vector $\vec{u}_3$ is chosen arbitrarily as:

$$\vec{u}_3 = \frac{\vec{r}_1 \times \vec{r}_2}{\|\vec{r}_1 \times \vec{r}_2\|}, \quad (1.20)$$

so that it will always be $\alpha = \|\vec{r}_1 \times \vec{r}_2\| > 0$. It is assumed that the direction of motion is such that the angular momentum vector has the same direction as the unit vector $\vec{u}_3$. From the independence of the three vectors it is deduced that there are non-zero scalars $(\lambda_1, \lambda_2, \lambda_3)$ which verify:

$$\lambda_1 \vec{r}_1 + \lambda_2 \vec{r}_2 + \lambda_3 \vec{r}_3 = \vec{0}. \quad (1.21)$$

Taking the dot product with the vector $\vec{e}$, and making use of the expression $\vec{e} \cdot \vec{r} = p - r$ brings:

$$p(\lambda_1 + \lambda_2 + \lambda_3) = \lambda_1 r_1 + \lambda_2 r_2 + \lambda_3 r_3. \quad (1.22)$$

If the cross product of Eq. (1.21) is taken with each of the position vectors $\vec{r}_1$, $\vec{r}_2$, $\vec{r}_3$, successively, three relations are obtained, which can be expressed in terms of λ , β , γ by Eqs. (1.17), (1.18) and (1.19), as:

$$\lambda_2 \alpha + \lambda_3 \beta = 0 \quad (1.23a)$$

$$\lambda_1 \alpha - \lambda_3 \gamma = 0 \quad (1.23b)$$

$$\lambda_1 \beta + \lambda_2 \gamma = 0. \quad (1.23c)$$

These equations constitute an indeterminate homogeneous system of order one, whose solution turns out to be: $\lambda_1 = \gamma \Lambda$, $\lambda_2 = -\beta \Lambda$, $\lambda_3 = \alpha \Lambda$, where $\Lambda \neq 0$ is not determined. If this solution is introduced in Eq. (1.22), and after dividing by Λ ($\Lambda \neq 0$), the value of the *semilatus rectum* p is obtained:

$$p = \frac{r_1 \gamma - r_2 \beta + r_3 \alpha}{\alpha - \beta + \gamma},$$

in terms of known data, where $r_i = \|\vec{r}_i\|$ for $i=1, 2, 3$.

1.2.2 Calculation of the eccentricity e

To determine the eccentricity e it is observed that, if $\vec{n}$ is any vector parallel to $\vec{u}_3$ (i.e., perpendicular to the orbital plane) the cross product $\vec{n} \times \vec{e}$ has the direction of the unit vector $\vec{u}_2$; therefore:

$$\vec{u}_2 = \pm \frac{\vec{n} \times \vec{e}}{\|\vec{n} \times \vec{e}\|}. \quad (1.24)$$

The sign (+) is taken if $\vec{n}$ has the same direction as the angular momentum $\vec{h}$. Otherwise, the minus sign (-) is taken. If it is chosen as vector $\vec{n}$ in the form $\vec{n} = (\vec{r}_1 \times \vec{r}_2 - \vec{r}_1 \times \vec{r}_3 + \vec{r}_2 \times \vec{r}_3) = (\alpha - \beta + \gamma) \vec{u}_3$, the sign (+) is taken when $(\alpha - \beta + \gamma) > 0$.

The calculation of the product $\vec{n} \times \vec{e}$ leads to double dot products, which through the relation $\vec{e} \cdot \vec{r} = p - r$, leads to:

$$(\vec{r}_1 \times \vec{r}_2) \times \vec{e} = p(\vec{r}_2 - \vec{r}_1) + r_2 \vec{r}_1 - r_1 \vec{r}_2 \quad (1.25a)$$

$$-(\vec{r}_1 \times \vec{r}_3) \times \vec{e} = p(\vec{r}_1 - \vec{r}_3) + r_1 \vec{r}_3 - r_3 \vec{r}_1 \quad (1.25b)$$

$$(\vec{r}_2 \times \vec{r}_3) \times \vec{e} = p(\vec{r}_3 - \vec{r}_2) + r_3 \vec{r}_2 - r_2 \vec{r}_3. \quad (1.25c)$$

Consequently the vector $\vec{s} = \vec{n} \times \vec{e}$:

$$\vec{s} = \vec{r}_1(r_2 - r_3) + \vec{r}_2(r_3 - r_1) + \vec{r}_3(r_1 - r_2), \quad (1.26)$$

is known in terms of data of the problem. Its magnitude $s = \|\vec{s}\|$ is known and non-null, otherwise the vectors $\vec{r}_1$, $\vec{r}_2$ y $\vec{r}_3$ would be dependent. Nevertheless, like $s = \|\vec{s}\| = \|\vec{n} \times \vec{e}\| = \|\vec{n}\|e = \|\alpha - \beta + \gamma\|e$, the eccentricity e can be solved as a function of known quantities, obtaining:

$$e = \frac{s}{\|\alpha - \beta + \gamma\|}. \quad (1.27)$$

1.2.3 Calculation of the perifocal trihedral

From the previous calculations, the estimation of the unit vectors of the perifocal trihedral is trivial. Thus, the unit vector $\vec{u}_3$ can be deduced from any of the relations (1.17), (1.18) or (1.19). The unit vector $\vec{u}_2$ is deduced from Eq. (1.24), taking into account that $\vec{s} = \vec{n} \times \vec{e}$, so that the sign is chosen according to the indications previously presented. Finally $\vec{u}_1$ is obtained from the orthogonality of the trihedral: $\vec{u}_1 = \vec{u}_2 \times \vec{u}_3$, and with it, the eccentricity vector: $\vec{e} = e \vec{u}_1$. The remaining elements of the orbit can be easily calculated. In particular, the velocities of the body at the observation points are interesting. They can be deduced from the eccentricity vector by the following expression:

$$\vec{v}_i = \frac{\mu}{h} \left(e \vec{u}_2 + \vec{u}_3 \times \frac{\vec{r}_i}{r_i} \right), \quad i = 1, 2, 3. \quad (1.28)$$

1.2.4 Observations

There are different ways to implement the Gibbs method. The one that has been followed in these lines has been adapted from Fundamentals of Astrodynamics (Bate, Mueller, and White [8]), and it differs from it in the choice of vector $\vec{n}$. It has the advantage, compared to other formulations, such as the one described in Fundamentals of Astrodynamics and Applications (Vallado [91]), that it is not necessary to assume that the vectors $\vec{r}_1$, $\vec{r}_2$ and $\vec{r}_3$ are temporarily ordered and no additional quadrant selection is required, neither prior nor later, to reach the solution. This is an important advantage because it facilitates the implementation of the code when making calculations. As indicated above, the problem proposed is purely geometric and the time evolution of the orbital motion does not intervene. Apparently in the Gibbs method there are nine conditions (three coordinates for each of the three vectors); since one orbit is determined by six independent conditions, it seems that there is an excess of conditions. Actually, this is not the case, since by forcing the three vectors to be coplanar, there are fewer conditions (three vectors contained in $z = 0$ imply six conditions). If the three vectors $\vec{r}_1$, $\vec{r}_2$ and $\vec{r}_3$ are close to each other ($\widehat{\vec{r}_i \vec{r}_j} < 1^\circ$), the Gibbs method provides less accurate results; in fact, the accuracy gets worse the closer the vectors are to each other. In such cases it is better to use the Herrick-Gibbs method, which is explained below. This method, which is based

on the appropriate use of Taylor series, is the one that must be used when any of the angles between the vectors is less than $\approx 5^\circ$. In the development of the method it has been assumed that the three vectors $\vec{r}_1$, $\vec{r}_2$ and $\vec{r}_3$ are coplanar. When the method is applied to real cases, this condition is never satisfied with accuracy, since the observations inevitably present errors; when implementing the code, this condition must be checked, to avoid big errors or to determine orbits that do not exist. However, some margin must be allowed so that one of the vectors can exit the orbital plane by a small amount, on the order of $\approx 1^\circ - 2^\circ$. Finally, we must underline that if only two vectors $\vec{r}_1$ and $\vec{r}_2$ are given, for example, an orbit can be determined if an additional data is specified: the flight time between the two positions. It is thus treated in contour problems closely related to Lambert's theorem.

1.3 HERRICK-GIBBS METHOD

When the vectors $\vec{r}_1$, $\vec{r}_2$ and $\vec{r}_3$ are close, the Gibbs method loses precision and it is necessary to resort to another procedure. The Herrick-Gibbs method differs from the Gibbs method in the fact that it includes as observation data not only the vectors $\vec{r}_1$, $\vec{r}_2$ and $\vec{r}_3$, but also the times t_1 , t_2 and t_3 corresponding to the observations. The idea of the method is to obtain the velocity vector $\vec{v}_2$ corresponding to the vector $\vec{r}_2$, which occupies the intermediate place in the series, so that once known $(\vec{r}_2, \vec{v}_2)$, the details and characteristics of the orbit are calculated.

If we consider points of the orbit in the neighborhood $\vec{r}_2$, the Taylor series expansion allows to express the position vector $\vec{r}(t)$ of a neighboring point as:

$$\vec{r}(t) = \vec{r}_2 + \vec{v}_2 \tau + \frac{1}{2!} \ddot{\vec{r}}_2 \tau^2 + \frac{1}{3!} \vec{r}_2^{(3)} \tau^3 + \frac{1}{4!} \vec{r}_2^{(4)} \tau^4 + \dots, \quad (1.29)$$

where the temporary variable $\tau = t - t_2$ is introduced and the following notation is used:

$$\vec{r}_2^{(i)} = \left. \frac{d^i \vec{r}_2}{d\tau^i} \right|_{\tau=0}, \quad i = 3, 4, \dots \quad (1.30)$$

By differentiating twice with respect to t , another expansion is obtained which provides the second derivative:

$$\ddot{\vec{r}}(t) = \ddot{\vec{r}}_2 + \vec{r}_2^{(3)} \tau + \frac{1}{2} \vec{r}_2^{(4)} \tau^2 + \dots \quad (1.31)$$

If the vectors $\vec{r}_1$, $\vec{r}_2$ and $\vec{r}_3$ are close, the values of $\vec{r}_1$ and $\vec{r}_3$ can be expressed by using the Taylor series expansion, thus yielding:

$$\vec{r}_1 = \vec{r}_2 + \vec{v}_2 \tau_1 + \frac{1}{2!} \ddot{\vec{r}}_2 \tau_1^2 + \frac{1}{3!} \vec{r}_2^{(3)} \tau_1^3 + \frac{1}{4!} \vec{r}_2^{(4)} \tau_1^4 + \dots \quad (1.32)$$

$$\vec{r}_3 = \vec{r}_2 + \vec{v}_2 \tau_3 + \frac{1}{2!} \ddot{\vec{r}}_2 \tau_3^2 + \frac{1}{3!} \vec{r}_2^{(3)} \tau_3^3 + \frac{1}{4!} \vec{r}_2^{(4)} \tau_3^4 + \dots \quad (1.33)$$

In the same way the Taylor series expansion can be used for the second derivatives which provides:

$$\ddot{\vec{r}}_1 = \ddot{\vec{r}}_2 + \vec{r}_2^{(3)} \tau_1 + \frac{1}{2} \vec{r}_2^{(4)} \tau_1^2 + \dots \quad (1.34)$$

$$\ddot{\vec{r}}_3 = \ddot{\vec{r}}_2 + \vec{r}_2^{(3)} \tau_3 + \frac{1}{2} \vec{r}_2^{(4)} \tau_3^2 + \dots \quad (1.35)$$

In these expressions the values of $\tau_1 = t_1 - t_2 < 0$ and $\tau_3 = t_3 - t_2 > 0$ are known. In Eqs. (1.32), (1.33), (1.34) and (1.35) there are two types of values:

- Known data: $\vec{r}_1$, $\vec{r}_2$, $\vec{r}_3$, τ_1 and τ_3 .

- Unknown data: $\vec{v}_2$, $\ddot{\vec{r}}_1$, $\ddot{\vec{r}}_2$, $\ddot{\vec{r}}_3$, $\vec{r}_2^{(3)}$ and $\vec{r}_2^{(4)}$.

Nevertheless, since the sought orbit is supposed to be a Keplerian orbit, the next relations are verified:

$$\ddot{\vec{r}}_1 = -\frac{\mu}{r_1^3} \vec{r}_1, \quad \ddot{\vec{r}}_3 = -\frac{\mu}{r_3^3} \vec{r}_3, \quad (1.36)$$

that allow to express such second derivatives in terms of known data. Consequently, in Eqs. (1.32), (1.33), (1.34) and (1.35) there are only four unknowns vectors, namely: $\vec{v}_2$, $\ddot{\vec{r}}_2$, $\vec{r}_2^{(3)}$ and $\vec{r}_2^{(4)}$. Of the four unknowns, only the velocity $\vec{v}_2$ is of interest. Therefore, it is necessary to eliminate the remaining unknowns ($\ddot{\vec{r}}_2$, $\vec{r}_2^{(3)}$ and $\vec{r}_2^{(4)}$) among the four equations. Thus, only one equation is obtained which is free of the unknowns eliminated and from which the value of $\vec{v}_2$ can be isolated. The elimination process is carried out in two steps:

- The derivatives $\vec{r}_2^{(3)}$ and $\vec{r}_2^{(4)}$ are solved from the Eqs. 1.34 and 1.35:

$$\vec{r}_2^{(3)} = -\frac{\tau_1 + \tau_3}{\tau_1 \tau_3} \ddot{\vec{r}}_2 + \frac{\mu \tau_3}{\tau_1 (\tau_1 - \tau_3)} \frac{\vec{r}_1}{r_1^3} - \frac{\mu \tau_1}{\tau_3 (\tau_1 - \tau_3)} \frac{\vec{r}_3}{r_3^3} \quad (1.37)$$

$$\vec{r}_2^{(4)} = \frac{2}{\tau_1 \tau_3} \ddot{\vec{r}}_2 - \frac{2\mu}{\tau_1 (\tau_1 - \tau_3)} \frac{\vec{r}_1}{r_1^3} + \frac{2\mu}{\tau_3 (\tau_1 - \tau_3)} \frac{\vec{r}_3}{r_3^3}. \quad (1.38)$$

- The derivatives $\vec{r}_2^{(3)}$ and $\vec{r}_2^{(4)}$ are replaced in Eqs. 1.32 and 1.33, so that after simplifying we obtain:

$$\vec{r}_1 = \vec{r}_2 + \vec{v}_2 \tau_1 - \frac{1}{12} \frac{\tau_1^2 (\tau_1 - 4\tau_3)}{\tau_3} \ddot{\vec{r}}_2 - \frac{1}{12} \frac{\mu \tau_1^2 (\tau_1 - 2\tau_3)}{\tau_1 - \tau_3} \frac{\vec{r}_1}{r_1^3} - \frac{1}{12} \frac{\mu \tau_1^4}{\tau_3 (\tau_1 - \tau_3)} \frac{\vec{r}_3}{r_3^3} \quad (1.39)$$

$$\vec{r}_3 = \vec{r}_2 + \vec{v}_2 \tau_3 + \frac{1}{12} \frac{\tau_3^2 (4\tau_1 - \tau_3)}{\tau_1} \ddot{\vec{r}}_2 + \frac{1}{12} \frac{\mu \tau_3^4}{\tau_1 (\tau_1 - \tau_3)} \frac{\vec{r}_1}{r_1^3} - \frac{1}{12} \frac{\mu \tau_3^2 (2\tau_1 - \tau_3)}{\tau_1 - \tau_3} \frac{\vec{r}_3}{r_3^3}. \quad (1.40)$$

Then $\ddot{\vec{r}}_2$ is eliminated between Eqs. 1.39 and 1.40 by the following combination:

$$\frac{\tau_1^2 (\tau_1 - 4\tau_3)}{\tau_3} (1.40) + \frac{\tau_3^2 (4\tau_1 - \tau_3)}{\tau_1} (1.39), \quad (1.41)$$

obtaining an equation that only contains the velocity $\vec{v}_2$, where once solved it is:

$$\vec{v}_2 = -A \vec{r}_2 + \left(B + C \frac{\mu}{r_1^3} \right) \vec{r}_1 + \left(D + E \frac{\mu}{r_3^3} \right) \vec{r}_3, \quad (1.42)$$

where the coefficients A, B, C, D and E are functions of τ_1 and τ_3 given by:

$$\chi = \tau_1^2 - 3\tau_1 \tau_3 + \tau_3^2 \quad (1.43a)$$

$$A = \frac{\tau_1 + \tau_3}{\tau_1 \tau_3} \frac{\tau_1^2 - 4\tau_1 \tau_3 + \tau_3^2}{\chi} \quad (1.43b)$$

$$B = \frac{\tau_3^2 (4\tau_1 - \tau_3)}{\tau_1 (\tau_1 - \tau_3) \chi} \quad (1.43c)$$

$$C = \frac{1}{6} \frac{\tau_1 \tau_3^2 (2\tau_1 - 3\tau_3)}{(\tau_1 - \tau_3) \chi} \quad (1.43d)$$

$$D = \frac{\tau_1^2 (\tau_1 - 4\tau_3)}{\tau_3 (\tau_1 - \tau_3) \chi} \quad (1.43e)$$

$$E = \frac{1}{6} \frac{\tau_3 \tau_1^2 (3\tau_1 - 2\tau_3)}{(\tau_1 - \tau_3) \chi}. \quad (1.43f)$$

1.3.1 Observations

The term χ , which appears in the denominators of the coefficients A, B, C, D and E, only cancels when the two values τ_1 and τ_3 are zero, situation that should not occur in a real case. The same situation happens with the difference $(\tau_1 - \tau_3)$. The solution provided by this method is approximate and it is better the closer the position vectors $\vec{r}_1$, $\vec{r}_2$ and $\vec{r}_3$ are. According to the angle between these vectors (Vallado [91]), we conclude that:

- If $\widehat{\vec{r}_i \vec{r}_j} < 1^\circ \Rightarrow$ the Herrick-Gibbs method is better than the Gibbs method.
- If $\widehat{\vec{r}_i \vec{r}_j} > 5^\circ \Rightarrow$ The Gibbs method is better than the Herrick-Gibbs method.
- If $\widehat{\vec{r}_i \vec{r}_j} \in [1^\circ, 5^\circ] \Rightarrow$ both methods provide comparable results.

This method is normally applied when the observations are obtained in the same monitoring station and in the same pass of the satellite.

1.4 LAPLACE METHOD

In a tracking station, devices are not always available to determine the radial velocity $\dot{\rho}$ of the observed satellite. In those cases, the observations only provide angular measurements. Depending on the season, different situations may arise. Thus, when observations are made with optical instruments, they usually provide the right ascension and declination with respect to the equatorial geocentric system: (α, δ) . However, the antennas usually provide the azimuth and elevation relative to the local horizontal system: $(\hat{A}, \hat{a})$. In any case, since knowing an orbit requires knowing the six orbital elements, a minimum of three independent observations are needed to be able to carry out the determination process. The method of determining orbits that only makes use of angular measurements was developed by Laplace in 1780. In what follows, a version of it is developed in a slightly different way from the classical one. Without loss of generality, it is assumed that the starting observations are the right ascension and declination with respect to the equatorial geocentric system, (α, δ) , of three different positions along the orbit. If the original observations were not these, a previous treatment would be necessary to identify such coordinates. Indeed, they constitute the starting point of the analysis and allow to unify the treatment in the cases when the observations may have originated in different tracking stations.

Thus, in this section the following problem is considered: let $Ex_1y_1z_1$ (Fig. 1.3) an inertial frame coincident with the ECI. Given three observations in the corresponding instants t_1 , t_2 and t_3 it is possible to obtain the unit vectors

$$\vec{u}_i = \cos(\delta_i) \cos(\alpha_i) \vec{i}_1 + \cos(\delta_i) \sin(\alpha_i) \vec{j}_1 + \sin(\delta_i) \vec{k}_1, \quad i = 1, 2, 3, \quad (1.44)$$

by their coordinates in the reference $Ex_1y_1z_1$. Therefore, if the inertial position vector of the station from where the satellite $\vec{R}_s$ is observed is known for the central observation, locate the orbit that they determine.

In order to apply the Laplace method, it is necessary to previously calculate a series of parameters at the time t of interest, namely: $\vec{u}$, $\dot{\vec{u}}$, $\ddot{\vec{u}}$, $\vec{R}_s$, $\dot{\vec{R}}_s$ and $\ddot{\vec{R}}_s$. To make this calculation we use the known data: $(t_i, \vec{u}_i)$, $i = 1, 2, 3$ and $\vec{R}_s$. We begin by remembering that given a function $f(x)$, of which the values $(f(x_1), f(x_2), f(x_3))$ are known in three points (x_1, x_2, x_3) of an interval $[a, b]$, it is possible to approximate the value of this function through a second degree polynomial, known as Lagrange

Figure 1.3: Reference systems and observation scenario.

interpolating polynomial. The expression of the Lagrange interpolating polynomial is:

$$P_f(x) = \frac{(x-x_1)(x-x_2)}{(x_3-x_1)(x_3-x_2)}f(x_3) + \frac{(x-x_1)(x-x_3)}{(x_2-x_1)(x_2-x_3)}f(x_2) + \frac{(x-x_2)(x-x_3)}{(x_1-x_2)(x_1-x_3)}f(x_1). \quad (1.45)$$

The Laplace method obtains approximate values of the variables of interest at any instant t of the interval $[t_1, t_3]$ by using the Lagrange interpolating polynomials. Thus, the values of $(\vec{u}, \dot{\vec{u}}, \ddot{\vec{u}})$ are obtained from the following expressions:

$$\vec{u}(t) = \frac{(t-t_2)(t-t_3)}{(t_1-t_2)(t_1-t_3)}\vec{u}_1 + \frac{(t-t_1)(t-t_3)}{(t_2-t_1)(t_2-t_3)}\vec{u}_2 + \frac{(t-t_1)(t-t_2)}{(t_3-t_1)(t_3-t_2)}\vec{u}_3 \quad (1.46)$$

$$\dot{\vec{u}}(t) = \frac{2t-t_2-t_3}{(t_1-t_2)(t_1-t_3)}\vec{u}_1 + \frac{2t-t_1-t_3}{(t_2-t_1)(t_2-t_3)}\vec{u}_2 + \frac{2t-t_1-t_3}{(t_3-t_1)(t_3-t_2)}\vec{u}_3 \quad (1.47)$$

$$\ddot{\vec{u}}(t) = \frac{2}{(t_1-t_2)(t_1-t_3)}\vec{u}_1 + \frac{2}{(t_2-t_1)(t_2-t_3)}\vec{u}_2 + \frac{2}{(t_3-t_1)(t_3-t_2)}\vec{u}_3. \quad (1.48)$$

On the other hand, to calculate the derivatives $(\dot{\vec{R}}_s, \ddot{\vec{R}}_s)$, it can be assumed without major problems that the Earth spins with constant angular velocity $\vec{\omega}_0 = \omega_0 \vec{k}_1$ around the pole line. It will therefore be:

$$\dot{\vec{R}}_s = \vec{\omega}_0 \times \vec{R}_s \quad (1.49)$$

$$\ddot{\vec{R}}_s = \vec{\omega}_0 \times (\vec{\omega}_0 \times \vec{R}_s). \quad (1.50)$$

Once all the required values have been determined, we apply the Laplace method. The basic equation that provides the position vector of the observed satellite $\vec{r}$ is:

$$\vec{r} = \vec{R}_s + \rho \vec{u}, \quad (1.51)$$

being $\vec{u}$ the unit vector of the visual that links the station with the observed satellite. Since a Keplerian orbit is being determined, the following relation is satisfied:

$$\ddot{\vec{r}} = -\frac{\mu}{r^3} \vec{r}, \quad (1.52)$$

where $\mu = 3.986004418 \cdot 10^5 \text{ km}^3/\text{s}^2$ is the gravitational parameter as defined by the reference ellipsoid WGS84. Differentiating the Eq. (1.51) with respect to time, we have:

$$\dot{\vec{r}} = \dot{\vec{R}}_s + \dot{\rho}\vec{u} + \rho\dot{\vec{u}}. \quad (1.53)$$

To determine the orbit of the satellite ($\vec{r}$, $\dot{\vec{r}}$) we must calculate the values of ρ y $\dot{\rho}$. This is done in two steps:

- ρ and $\dot{\rho}$ are calculated based on $r = \|\vec{r}\|$.
- r is calculated by an algebraic equation of eighth degree known as the Lagrange equation.

1.4.1 Determination of ρ

If Eq. (1.53) is differentiated with respect to time, the relation obtained is:

$$\ddot{\vec{r}} = \ddot{\vec{R}}_s + \ddot{\rho}\vec{u} + 2\dot{\rho}\dot{\vec{u}} + \rho\ddot{\vec{u}}. \quad (1.54)$$

If the second derivative $\ddot{\vec{r}}$ is removed using the relation (1.52) and, afterwards, the value of $\ddot{\vec{r}}$ using the relation (1.51), the expression reached is:

$$-\left[\ddot{\vec{R}}_s + \frac{\mu}{r^3}\vec{R}_s\right] = \rho\ddot{\vec{u}} + 2\dot{\rho}\dot{\vec{u}} + \left(\ddot{\rho} + \frac{\mu}{r^3}\rho\right)\vec{u}. \quad (1.55)$$

From this equation the value of ρ can be solved if the following operations are done:

- Take the cross product of the equation by $\dot{\vec{u}}$.
- Take the dot product by $\vec{u}$.

The resulting expression is:

$$-\vec{u}(\dot{\vec{u}} \times \ddot{\vec{R}}_s) - \frac{\mu}{r^3}\vec{u}(\dot{\vec{u}} \times \vec{R}_s) = \rho\vec{u}(\dot{\vec{u}} \times \ddot{\vec{u}}), \quad (1.56)$$

and the expression obtained when solving for ρ is:

$$\rho = A + B\frac{\mu}{r^3}, \quad (1.57)$$

where the coefficients A and B turn out to be:

$$A = -\frac{\vec{u}(\dot{\vec{u}} \times \ddot{\vec{R}}_s)}{D_0} \quad (1.58a)$$

$$B = -\frac{\vec{u}(\dot{\vec{u}} \times \vec{R}_s)}{D_0} \quad (1.58b)$$

$$D_0 = \vec{u}(\dot{\vec{u}} \times \ddot{\vec{u}}). \quad (1.58c)$$

1.4.2 Determination of $\dot{\rho}$

From the relation (1.55) the value of $\dot{\rho}$ can be solved by the following operations:

- Take the cross product of the equation by $\ddot{\vec{u}}$.
- Take the dot product by $\vec{u}$.

The resulting expression is:

$$\ddot{\mathbf{u}}(\ddot{\mathbf{u}} \times \ddot{\mathbf{R}}_s) + \frac{\mu}{r^3} \ddot{\mathbf{u}}(\ddot{\mathbf{u}} \times \ddot{\mathbf{R}}_s) = 2\dot{\rho} \ddot{\mathbf{u}}(\dot{\mathbf{u}} \times \ddot{\mathbf{u}}), \quad (1.59)$$

and the expression obtained when solving $\dot{\rho}$ is:

$$\dot{\rho} = C + E \frac{\mu}{r^3}, \quad (1.60)$$

where the coefficients C and E turn out to be:

$$C = \frac{\ddot{\mathbf{u}}(\ddot{\mathbf{u}} \times \ddot{\mathbf{R}}_s)}{2D_0} \quad (1.61)$$

$$E = \frac{\ddot{\mathbf{u}}(\ddot{\mathbf{u}} \times \ddot{\mathbf{R}}_s)}{2D_0}. \quad (1.62)$$

In these expressions the value of D_0 is the collected in Eq. (1.58c). It can be shown that the coefficient D_0 is proportional to the curvature of the satellite trajectory with respect to the tracking station. If this trajectory is very straight, the coefficient D_0 becomes small and the solutions found for ρ and $\dot{\rho}$ become unreliable. If the tracking station were permanently contained in the orbital plane, the coefficient D_0 would be permanently zero. Actually the tracking station cannot be permanently in the orbital plane (except exceptional cases such as a station in the north pole and a polar orbit), but if it is inside the plane at time t , then the value of D_0 is again small and the procedure loses validity.

1.4.3 Lagrange equation

In Eqs. (1.57) and (1.60) the values of $(\rho, \dot{\rho})$ depend on known data, through the coefficients A, B, C and E and the distance r , unknown. To identify the value of r we start squaring Eq. (1.51) to provide:

$$r^2 - R_s^2 - \rho^2 - 2\rho \ddot{\mathbf{u}} \ddot{\mathbf{R}}_s = 0. \quad (1.63)$$

In this relation the value of ρ given by Eq. (1.57) is substituted such that if it is multiplied by r^6 , yielding a polynomial like this:

$$f(r) = r^8 + ar^6 + br^3 + c = 0, \quad (1.64)$$

where the coefficients a, b, c are given by:

$$a = -(A^2 + R_s^2 + 2A \ddot{\mathbf{u}} \cdot \ddot{\mathbf{R}}_s) \quad (1.65a)$$

$$b = -2\mu B(A + \ddot{\mathbf{u}} \cdot \ddot{\mathbf{R}}_s) \quad (1.65b)$$

$$c = -\mu^2 B^2. \quad (1.65c)$$

The relation of Eq. (1.64), ($f(r) = 0$), is the so-called Lagrange equation that must be solved numerically, since there is no analytical solution for polynomials of eighth degree. Since it is a polynomial, the Descartes rule says that, at most, there are three positive roots (the coefficient c is always negative and the number of sign changes is three). When there are three positive roots, usually two of them can be easily discarded, because the roots that are not a solution to the problem are usually too small or too large and do not correspond to real solutions. However, this is not always the case and, sometimes, more than one valid solution can be obtained. In this case, some additional data is needed to be able to discard one of them. If additional data is not available, it should be accepted that there are

multiple candidate solutions. One of the most used techniques to solve the Lagrange equation is the Newton-Raphson method that can be easily implemented, because the derivative $f'(r)$ is simple:

$$f'(r) = 8r^7 + 6ar^5 + 3br^2. \quad (1.66)$$

The determination of the orbit is carried out by the following procedure:

1. In an instant of time t known we have as starting data the following values: t , $\vec{u}$, $\dot{\vec{u}}$, $\ddot{\vec{u}}$, $\vec{R}_s$, $\dot{\vec{R}}_s$ y $\ddot{\vec{R}}_s$.
2. The coefficients A , B , C , D_0 and E are calculated from Eqs. (1.58a), (1.58b), (1.58c), (1.61) and (1.62).
3. The coefficients a , b and c are calculated from Eqs. (1.65a), (1.65b) and (1.65c).
4. r is determined by solving the Lagrange equation, for example, with the Newton-Raphson method.
5. ρ and $\dot{\rho}$ are determined respectively through Eqs. (1.57) and (1.60).
6. $(\vec{r}, \vec{v})$ are estimated through the relations (1.51) and (1.53).

Once the dynamical state $(\vec{r}, \vec{v})$ has been calculated, the determination of the rest of characteristics of the orbit is immediate.

1.4.4 Observations

The relations (1.46), (1.47), (1.48), (1.49) and (1.50) provide the data from which to begin the process of determining the orbit by the Laplace method. The only thing left to decide is the time t in which the determination should be made. It seems reasonable to use a value that belongs to the interval $[t_1, t_3]$ where the interpolating polynomials should provide minor errors. A classic value is the midpoint t_2 , but it is not essential.

Equations (1.46), (1.47) and (1.48) lead to null derivatives when the order is greater than two. More than three observations are frequently available. In that case, it is possible to use a Lagrange interpolating polynomial of degree greater than two, which would allow to obtain the starting data with greater precision.

1.5 GAUSS METHOD

The basis of the Gauss method is to locate, given the unit vectors $\vec{u}_1$, $\vec{u}_2$ and $\vec{u}_3$ associated with three observations of the satellite in the corresponding instants t_1 , t_2 and t_3 , as well as the associated inertial position vectors $\vec{R}_{s1}$, $\vec{R}_{s2}$ and $\vec{R}_{s3}$ of the station from where the satellite is observed, the orbit they determine.

The basic equation that gives the position vector of the satellite in the corresponding instants is:

$$\vec{r}_i = \vec{R}_{si} + \rho_i \vec{u}_i, \quad i = 1, 2, 3, \quad (1.67)$$

while for the velocity vector, since a Keplerian orbit is being determined, the relation satisfied for each instant is:

$$\ddot{\vec{r}}_i = -\frac{\mu}{r_i^3} \vec{r}_i. \quad (1.68)$$

To estimate the relations given by Eq. (1.67), the calculation of ρ is required, which is *a priori* unknown. To do this, the time intervals are calculated first by subtracting the observation times:

$$\tau_1 = t_1 - t_2, \quad \tau_3 = t_3 - t_2, \quad \tau = \tau_3 - \tau_1. \quad (1.69)$$

Then we calculate the cross products of the unit vectors:

$$\vec{n}_1 = \vec{u}_2 \times \vec{u}_3, \quad \vec{n}_2 = \vec{u}_1 \times \vec{u}_3, \quad \vec{n}_3 = \vec{u}_1 \times \vec{u}_2, \quad (1.70)$$

and we determine the parameter $D_0 = \vec{u}_1 \cdot \vec{n}_1$. Similarly, we estimate the nine scalar quantities defined by $D_{ij} = \vec{R}_{s_i} \cdot \vec{n}_j$, $i, j = 1, 2, 3$ to calculate the coefficients of the scalar position given by:

$$A = \frac{1}{D_0} \left(-D_{12} \frac{\tau_3}{\tau} + D_{22} + D_{32} \frac{\tau_1}{\tau} \right) \quad (1.71a)$$

$$B = \frac{1}{6D_0} \left[D_{12}(\tau_3^2 - \tau^2) \frac{\tau_3}{\tau} + D_{32}(\tau^2 - \tau_1^2) \frac{\tau_1}{\tau} \right] \quad (1.71b)$$

$$E = \vec{R}_{s2} \cdot \vec{u}_2. \quad (1.71c)$$

The next step is to calculate the coefficients of the scalar distance polynomial for the second observation made to the satellite, so that:

$$a = -(A^2 + 2AE + R_{s2}^2) \quad (1.72a)$$

$$b = -2\mu B(A + E) \quad (1.72b)$$

$$c = -\mu^2 B^2, \quad (1.72c)$$

where $R_{s2}^2 = \vec{R}_{s2} \cdot \vec{R}_{s2}$ and $\mu = 3.986004418 \cdot 10^5 \text{ km}^3/\text{s}^2$ the gravitational parameter defined in the reference ellipsoid WGS84. The scalar distance polynomial, for the second observation, will be analogous to the polynomial obtained in the Laplace method:

$$0 = r_2^8 + ar_2^6 + br_2^3 + c, \quad (1.73)$$

and therefore, it can be solved, for example, by the Newton-Raphson method.

Once r_2 is determined, we estimate ρ_i , $i = 1, 2, 3$, that is, the distances between the station and the satellite at the respective instants:

$$\rho_1 = \frac{1}{D_0} \left[\frac{6 \left(D_{31} \frac{\tau_1}{\tau_3} + D_{21} \frac{\tau}{\tau_3} \right) r_2^3 + \mu D_{31} (\tau^2 - \tau_1^2) \frac{\tau_1}{\tau_3}}{6r_2^3 + \mu(\tau^2 - \tau_3^2)} - D_{11} \right] \quad (1.74a)$$

$$\rho_2 = A + \frac{\mu B}{r_2^3} \quad (1.74b)$$

$$\rho_3 = \frac{1}{D_0} \left[\frac{6 \left(D_{13} \frac{\tau_3}{\tau_1} + D_{23} \frac{\tau}{\tau_1} \right) r_2^3 + \mu D_{13} (\tau^2 - \tau_3^2) \frac{\tau_3}{\tau_1}}{6r_2^3 + \mu(\tau^2 - \tau_1^2)} - D_{33} \right], \quad (1.74c)$$

those with which the relations in Eq. (1.67) are determined:

$$\vec{r}_1 = \vec{R}_{s1} + \rho_1 \vec{u}_1 \quad (1.75a)$$

$$\vec{r}_2 = \vec{R}_{s2} + \rho_2 \vec{u}_2 \quad (1.75b)$$

$$\vec{r}_3 = \vec{R}_{s3} + \rho_3 \vec{u}_3. \quad (1.75c)$$

Equation (1.68) can be solved, for each instant, by a power series of τ_i ($i = 1, 3$, depending on the instant), being its coefficients functions of the constants μ , $\vec{r}_2 = \vec{r}|_{t=t_2}$ and $\dot{\vec{r}}_2 = \dot{\vec{r}}|_{t=t_2}$ (Roy [75]) such that:

$$\vec{r}_i = f_i \vec{r}_2 + g_i \dot{\vec{r}}_2, \quad i = 1, 3, \quad (1.76)$$

with the coefficients f_i and g_i given by:

$$f_i = 1 - \frac{1}{2} \frac{\mu}{r_2^3} \tau_i^2 + \dots \quad (1.77a)$$

$$g_i = \tau_i - \frac{1}{6} \frac{\mu}{r_2^3} \tau_i^2 + \dots \quad (1.77b)$$

To determine the velocity vector of the satellite, referred to the second observation, we take the approximations of second order of the series f_1 , f_3 , g_1 and g_3 and we substitute in the expression:

$$\vec{v}_2 = \frac{1}{f_1 g_3 - f_3 g_1} (-f_3 \vec{r}_1 + f_1 \vec{r}_3), \quad (1.78)$$

having finally determined the state vector, so that the determination of the rest of the characteristics of the orbit is immediate.

1.5.1 Observations

The approximation by the series f_i , g_i and the limitations of the required observation conditions cause inaccuracies in the method. One way to improve the precision is by increasing the precision of f_i , g_i by taking more elements from the series or by using more observations.

1.6 THE DIFFERENTIAL CORRECTION METHOD

When a measurement of a given magnitude x is made, the following distinction can always be established between the different values that are handled in the estimation processes and which should be known:

$$x_{obs} = x_v + \epsilon \quad (1.79a)$$

$$x_{obs} = x_e + \sigma, \quad (1.79b)$$

where:

- x_{obs} is the observed value, result of the measurement.
- x_v is the true value that we intend to measure.
- ϵ is the true error that is committed in the measure.
- x_e is the estimated value, that is, the closest value to the true value.
- σ is the residual.

The true value x_v cannot be known, and neither the true error ϵ . In the estimation processes the estimated value x_e and the residual σ are handled instead of x_v and ϵ . If the estimation technique is correct, the estimated value x_e will be very close to the true value x_v , and the residual σ very close to the true error ϵ . In an estimation problem the main objective is to find the estimated value x_e by applying estimation techniques usually based on minimizing the value of the residual σ .

The preliminary determination provides a Keplerian orbit. The knowledge of this orbit allows to know which tracking stations are going to be visited by the satellite and the instants of passage over each station. This information is essential so that, from any of the stations that are downstream from the satellite, the satellite can be acquired, that is, to enter in communication with it through some of the devices usually used for this purpose. Once a tracking station locates the satellite, it is possible to obtain new sets of observations to improve the knowledge of the orbit. The result of a preliminary determination is always approximate, since the observation data have built-in errors associated with the quality of the instruments used, the operating techniques, etc. In addition, the real orbits are not purely Keplerian, as there are perturbations that alter the dynamics of the satellite. Therefore, it is convenient to improve the data of the orbit with more observations and this can be done in different ways.

Suppose that from an additional tracking station a new set of six independent observations is obtained, for example, the values of ρ , $\dot{\rho}$, $\hat{A}$, $\hat{a}$, $\hat{\dot{A}}$ and $\hat{\dot{a}}$ are located at another time and for another tracking station. In this case it is possible to make a second determination of the orbit, independent of the first one. The mean values of the two determinations would provide more accurate results. This technique can be developed with any of the usual methods of orbit determination that have been described above. Imagine, however, that the tracking station that makes the new observations has limited capabilities and can only provide the value of χ , where χ is one of the parameters, for example $\chi = \rho$. In this case, it is also possible to improve the estimation of the orbit if the station provides six independent values of the parameter, that is, if it provides the value of χ in six different times instants: t_i , $i = 1, \dots, 6$. The process that provides a better estimate is called differential correction and consists on:

- Start from an initial estimate of the orbit.
- Have a new set of observations.
- Improve the initial estimate by making use of the new observations and the previous estimate.

Suppose, then, that the determination of an orbit has been made and, as a result of it, the values of $(t_0, \vec{r}_0, \vec{v}_0)$ which define the dynamical state of the satellite are known. If the satellite's orbit were really Keplerian and the observations used were perfect (without errors) and the calculations did not introduce any error, the value of the parameter χ could be predicted with absolute precision at any subsequent time:

$$\chi(t) = \chi(t, \vec{r}_0, \vec{v}_0). \quad (1.80)$$

If the value of t is set to t_1 , the relation (1.80) particularized to t_1 would provide as a prediction the value $\chi_1 = \chi(t_1)$. If everything were perfect and there were no perturbations nor errors of any kind, this prediction would coincide with the observed value: $\chi_1 = \chi_{obs}(t_1)$. In reality this will not happen, which implies that the estimated initial values $(\vec{r}_0, \vec{v}_0)$ are not correct and in order to accurately predict the value of χ , these values have to be corrected and instead of working with $(\vec{r}_0, \vec{v}_0)$, we have to work with the values $(\vec{r}_0 + \delta\vec{r}_0, \vec{v}_0 + \delta\vec{v}_0)$. For this we must identify the corrections $(\delta\vec{r}_0, \delta\vec{v}_0)$, choosing them so that once the initial values are corrected, there will not be discrepancies between the observed value and the prediction:

$$\chi_{obs}(t) = \chi(t, \vec{r}_0 + \delta\vec{r}_0, \vec{v}_0 + \delta\vec{v}_0). \quad (1.81)$$

Normally the function (1.80) that provides χ in terms of the initial values is strongly non-linear and imposing the condition (1.81) is not simple. However, if the initial estimate $(\vec{r}_0, \vec{v}_0)$ is not of very low quality, the corrections $(\delta\vec{r}_0, \delta\vec{v}_0)$ sought must be small. Thus, these corrections will be considered infinitesimal and we will call ϵ to their order of magnitude. Consequently, the relation (1.81) can be expanded in powers of ϵ to provide:

$$\chi_{obs}(t) = \chi(t, \vec{r}_0, \vec{v}_0) + \nabla_{\vec{r}_0} \chi \delta\vec{r}_0 + \nabla_{\vec{v}_0} \chi \delta\vec{v}_0 + \mathcal{O}(\epsilon^2), \quad (1.82)$$

where the gradients:

$$\nabla_{\vec{r}_0} \chi = \frac{\partial \chi}{\partial x_{10}} \vec{i}_1 + \frac{\partial \chi}{\partial y_{10}} \vec{j}_1 + \frac{\partial \chi}{\partial z_{10}} \vec{k}_1 \quad (1.83a)$$

$$\nabla_{\vec{v}_0} \chi = \frac{\partial \chi}{\partial \dot{x}_{10}} \vec{i}_1 + \frac{\partial \chi}{\partial \dot{y}_{10}} \vec{j}_1 + \frac{\partial \chi}{\partial \dot{z}_{10}} \vec{k}_1, \quad (1.83b)$$

are particularized for $\epsilon = 0$, that is, for the initially estimated orbit. Now, since the difference $\sigma(t) = \chi_{\text{obs}}(t) - \chi(t, \vec{r}_0, \vec{v}_0)$ is known (by definition it is the residual), if terms of order $O(\epsilon^2)$ are neglected the equation obtained is:

$$\sigma(t) \approx \nabla_{\vec{r}_0} \chi \delta \vec{r}_0 + \nabla_{\vec{v}_0} \chi \delta \vec{v}_0, \quad (1.84)$$

that relates, directly, the residual with the sought corrections. If six independent observation data are available at the times t_i , $i = 1, \dots, 6$, then six equations deduced from Eq. (1.84) are available to set out the matrix equation $[\vec{\sigma}]_{6 \times 1} = M_{6 \times 6} [\vec{z}]_{6 \times 1}$, where M , $[\vec{\sigma}]$ and $[\vec{z}]$ are the matrices given by:

$$[\vec{\sigma}] = \begin{pmatrix} \sigma_1 \\ \sigma_2 \\ \sigma_3 \\ \sigma_4 \\ \sigma_5 \\ \sigma_6 \end{pmatrix}, \quad [\vec{z}] = \begin{pmatrix} \delta x_{10} \\ \delta y_{10} \\ \delta z_{10} \\ \delta \dot{x}_{10} \\ \delta \dot{y}_{10} \\ \delta \dot{z}_{10} \end{pmatrix} \quad (1.85)$$

$$M = \begin{pmatrix} \frac{\partial \chi}{\partial x_{10}}(t_1) & \frac{\partial \chi}{\partial y_{10}}(t_1) & \frac{\partial \chi}{\partial z_{10}}(t_1) & \frac{\partial \chi}{\partial \dot{x}_{10}}(t_1) & \frac{\partial \chi}{\partial \dot{y}_{10}}(t_1) & \frac{\partial \chi}{\partial \dot{z}_{10}}(t_1) \\ \frac{\partial \chi}{\partial x_{10}}(t_2) & \frac{\partial \chi}{\partial y_{10}}(t_2) & \frac{\partial \chi}{\partial z_{10}}(t_2) & \frac{\partial \chi}{\partial \dot{x}_{10}}(t_2) & \frac{\partial \chi}{\partial \dot{y}_{10}}(t_2) & \frac{\partial \chi}{\partial \dot{z}_{10}}(t_2) \\ \frac{\partial \chi}{\partial x_{10}}(t_3) & \frac{\partial \chi}{\partial y_{10}}(t_3) & \frac{\partial \chi}{\partial z_{10}}(t_3) & \frac{\partial \chi}{\partial \dot{x}_{10}}(t_3) & \frac{\partial \chi}{\partial \dot{y}_{10}}(t_3) & \frac{\partial \chi}{\partial \dot{z}_{10}}(t_3) \\ \frac{\partial \chi}{\partial x_{10}}(t_4) & \frac{\partial \chi}{\partial y_{10}}(t_4) & \frac{\partial \chi}{\partial z_{10}}(t_4) & \frac{\partial \chi}{\partial \dot{x}_{10}}(t_4) & \frac{\partial \chi}{\partial \dot{y}_{10}}(t_4) & \frac{\partial \chi}{\partial \dot{z}_{10}}(t_4) \\ \frac{\partial \chi}{\partial x_{10}}(t_5) & \frac{\partial \chi}{\partial y_{10}}(t_5) & \frac{\partial \chi}{\partial z_{10}}(t_5) & \frac{\partial \chi}{\partial \dot{x}_{10}}(t_5) & \frac{\partial \chi}{\partial \dot{y}_{10}}(t_5) & \frac{\partial \chi}{\partial \dot{z}_{10}}(t_5) \\ \frac{\partial \chi}{\partial x_{10}}(t_6) & \frac{\partial \chi}{\partial y_{10}}(t_6) & \frac{\partial \chi}{\partial z_{10}}(t_6) & \frac{\partial \chi}{\partial \dot{x}_{10}}(t_6) & \frac{\partial \chi}{\partial \dot{y}_{10}}(t_6) & \frac{\partial \chi}{\partial \dot{z}_{10}}(t_6) \end{pmatrix}. \quad (1.86)$$

The matrix M is formed by the partial derivatives of the element χ particularized in the six observation instants t_i , $i = 1, \dots, 6$. Its calculation has to be addressed, most of the time, numerically. If the matrix M is regular, it has an inverse and the solution to the problem is obtained as:

$$[\vec{z}]_{6 \times 1} = M_{6 \times 6}^{-1} [\vec{\sigma}]_{6 \times 1}. \quad (1.87)$$

Once the correction $(\delta \vec{r}_0, \delta \vec{v}_0)$ of the initial conditions has been determined, the process can be repeated using the same set of observations, but now taking as its starting orbit the one provided by the corrected initial values. Thus, an iterative process is developed, which is continued as long as the residuals are, in each iteration, lower than those obtained in the previous step. When the residuals stops decreasing, the process stops. The matrix M plays a fundamental role in the differential correction. Its calculation has to be developed, most of the time, numerically, because depending on the nature of the parameter χ the analytical solution may not be available. However, its numerical calculation is not problematic. In effect, since the function (1.80) is available, if you want to calculate the partial derivative with respect to, for example, x_{10} , the definition of partial derivative is used:

$$\frac{\partial \chi}{\partial x_{10}}(t) = \frac{\chi(t, \vec{r}_0 + \Delta x_{10} \vec{e}_1, \vec{v}_0) - \chi(t, \vec{r}_0, \vec{v}_0)}{\Delta x_{10}}. \quad (1.88)$$

Thus, taking a reasonably small value for Δx_{10} , of the order of $0.1 x_{10}$, the difference of the numerator can be calculated numerically and the quotient provides a good approximation of the partial derivative sought. A similar calculation is made for the rest of the derivatives.

1.6.1 Observations

In the development of the method it has been assumed that all available observations are of the same parameter χ in different instants of time. However, this hypothesis can be eliminated, because the method is extended without essential modifications to the case in which observations of different elements are available, that is, if observations of the distance ρ and azimuth $\hat{A}$ are available, a similar differential correction process can be devised with minor changes. In fact, the most important change is in obtaining the matrix M whose rows will contain partial derivatives of two different elements. However, the philosophy of the procedure remains unchanged. If we consider the function (1.80), which provides the value of the element χ , in practice, this function is available numerically, that is, by means of calculation routines which provide the value of χ numerically. Consequently, the differential correction process can also be used for the determination of perturbed orbits, that is, it is not essential that the orbit to be determined be Keplerian. The only requirement is that the function (1.80), which provides the value of χ , incorporates the effects of the perturbation that needs to be taken into account. Finally, it is common to have many more observations available instead of six observations. In that case the differential correction is modified because the equation reached is of the type $[\vec{\sigma}]_{N \times 1} = M_{N \times 6}[\vec{z}]_{6 \times 1}$, where N is the total number of available observations. Now the matrix M is not square and it can not be inverted. However, this equation allows to obtain an optimal solution, in the sense of minimizing the sum of the squares of the residuals of all the observations. But the technique used, based on the method of least squares adjustment, is directly related to the statistical determination of orbits.

1.7 STATISTICAL ORBIT DETERMINATION

The orbit estimation, or differential correction, is used for the improvement of *a priori* orbital elements from a large set of tracking data. The complex mathematical formulation of orbit prediction and measurement modeling does not allow a direct determination of orbital elements and model parameters from a given set of observations of the satellite. This is only possible for the simplified case of Keplerian orbits and a coarse measurement model. Besides, the measurements employed for an orbit determination cannot be expected to be exact quantities due to inevitable measurement and model errors. In order to achieve a reliable reconstruction of a satellite orbit from actual measurements, a larger amount of tracking data than required for an initial orbit determination is necessary to smooth out these errors (Montenbruck and Gill [56]).

Two main techniques can be distinguished for the improvement of *a priori* orbit information from an arbitrary set of tracking data: the batch or least squares estimator, and the sequential estimator or filter. The first one improves an epoch state estimate by processing a whole set of observations in each run. The second one processes one measurements at a time and yields subsequent estimates of the state vector at the time of each measurement. Both batch and sequential estimators are powerful estimation methods that have successfully been applied to various types of orbit determination problems in the past. While the method of weighted least squares dates back to the end of the 18th century, where it was developed for the improvement of minor planet orbits by Gauss, the Kalman filter (Kalman [47]) was introduced during the mid 20th century. Nevertheless, it was immediately recognized as a fast and efficient method that is particularly suited for on-board and real-time applications and provides a unique way of considering process noise (Leondes [51]).

Let M be a celestial body observed from the Earth; the aim is to determine its orbit from a set of observational data. By considering the following notation:

$$\begin{aligned}\vec{x} &= (x_1, x_2, \dots, x_n)^T && \rightarrow \text{state vector} \in \mathbb{R}^n \\ \vec{f} &= (f_1, f_2, \dots, f_n)^T && \rightarrow \text{vector field (dynamic)} \in \mathbb{R}^n \\ \vec{b} &= (b_1, b_2, \dots, b_p)^T && \rightarrow \text{observations vector} \in \mathbb{R}^p \\ \vec{g} &= (g_1, g_2, \dots, g_p)^T && \rightarrow \text{relation observations-state vector} \in \mathbb{R}^p\end{aligned}$$

the system of equations describing the motion is:

$$\frac{d\vec{x}}{dt} = \vec{f}(\vec{x}, t) \quad (1.89a)$$

$$\vec{b}_i = \vec{g}(\vec{x}_i, t_i) + \vec{e}_i, \quad i = 1, \dots, l, \quad (1.89b)$$

where the unknown is the state vector $\vec{x}$ at a known instant t_s such that

$$\vec{x}_s = \vec{x}(t_s). \quad (1.90)$$

The state vector includes not only the variables defining the dynamical state of the body $(x, y, z, \dot{x}, \dot{y}, \dot{z})$, but all the coefficients taken as unknowns that are required to be determined. Consequently, the dimension of $\vec{x}$ is at least six. Equations (1.89a) govern the dynamics and contain the perturbations acting upon the orbit. There are l observations and for each observation p quantities are measured and included in a vector $\vec{b} \in \mathbb{R}^p$. The observations are done at known instants of time $t_1, t_2, \dots, t_l$ such that the number of observations p in the vector $\vec{b}$ is smaller than the dimension n of the state vector. However, the total number of observations $m = p l$ is far larger than n . If the observations would not have errors, that is, if they were done in ideal conditions, the relation (1.89b) would become $\vec{b} = \vec{g}(\vec{x}, t)$. However, this does not happen in reality and the relation between the observed magnitudes and the state vector $\vec{g}$ is in general strongly nonlinear.

The problem of determining the trajectory of the body taking into account all these factors is known as the problem of nonlinear estimation, or problem of orbit determination.

As a result of the preliminary orbit determination, explained in the previous chapter, a reference orbit described by the functions $\vec{x}_{ref}(t)$ is known. If it is admitted that the orbit to be determined $\vec{x}(t)$ and the nominal orbit $\vec{x}_{ref}(t)$ keep close during the time interval in which the orbit is estimated, then Eqs. (1.89a), (1.89b) and (1.90) can be linearized. The following deviations are introduced

$$\vec{\chi}(t) = \vec{x}(t) - \vec{x}_{ref}(t), \quad \vec{\mathcal{B}}(t) = \vec{b}(t) - \vec{b}_{ref}(t), \quad (1.91)$$

where $\vec{b}_{ref}(t) = \vec{g}(\vec{x}_{ref}(t), t)$ are the observations simulated through the reference orbit.

Applying Eq. (1.89a), referred to the true orbit, to the reference orbit and subtracting, the derivative of $\vec{\chi}$ is obtained:

$$\frac{d\vec{\chi}}{dt} = \vec{f}(\vec{x}, t) - \vec{f}(\vec{x}_{ref}, t) = \vec{f}(\vec{x}_{ref} + \vec{\chi}, t) - \vec{f}(\vec{x}_{ref}, t). \quad (1.92)$$

The right-hand side of this equation, which is exact, can be represented by a Taylor series around the reference orbit by

$$\vec{f}(\vec{x}_{ref} + \vec{\chi}, t) - \vec{f}(\vec{x}_{ref}, t) = \left(\frac{\partial \vec{f}}{\partial \vec{x}} \right) \bigg|_{ref} \vec{\chi} + \mathcal{O}(|\vec{\chi}|). \quad (1.93)$$

Neglecting the second order terms of $\vec{\chi}$, considering $|\vec{\chi}| \ll |\vec{x}_{\text{ref}}|$, the final equation is:

$$\frac{d\vec{\chi}}{dt} = A(t)\vec{\chi}, \quad A(t) = \left(\frac{\partial \vec{f}}{\partial \vec{x}} \right) \Big|_{\text{ref}}, \quad (1.94)$$

where the square matrix $A(t)$ of order n coincides with the Jacobian matrix of the function $\vec{f}$ particularized at the reference orbit $\vec{x}_{\text{ref}}(t)$.

Applying Eq. (1.89b), referred to the true orbit, to the reference orbit and subtracting, the expression of $\vec{\xi}$ is obtained:

$$\vec{B}_i = \vec{g}(\vec{x}_i, t_i) - \vec{g}(\vec{x}_{\text{ref}_i}, t_i) + \vec{e}_i, \quad i = 1, \dots, l. \quad (1.95)$$

The right-hand side of this equation, which is exact, can be represented by a Taylor series around the reference orbit by

$$\vec{g}(\vec{x}_{\text{ref}_i} + \vec{\chi}_i, t_i) - \vec{g}(\vec{x}_{\text{ref}_i}, t_i) = \left(\frac{\partial \vec{g}}{\partial \vec{x}} \right) \Big|_{\text{ref}_i} \vec{\chi}_i + \mathcal{O}(|\vec{\chi}_i|). \quad (1.96)$$

Neglecting again the second order terms of $\vec{\chi}$, considering $|\vec{\chi}| \ll |\vec{x}_{\text{ref}}|$, the final equation is:

$$\vec{B}_i = H(t_i)\vec{\chi}_i + \vec{e}_i, \quad H(t_i) = \left(\frac{\partial \vec{g}}{\partial \vec{x}} \right) \Big|_{\text{ref}_i}, \quad (1.97)$$

where the rectangular matrix $H(t_i)$ of dimension $p \times n$ coincides with the Jacobian matrix of the function $\vec{g}$ particularized at the reference orbit $\vec{x}_{\text{ref}}(t)$.

Equations (1.94) and (1.97) should be integrated from the initial conditions

$$t = t_s : \quad \vec{\chi} = \vec{\chi}_s = \vec{x}(t_s) - \vec{x}_{\text{ref}}(t_s). \quad (1.98)$$

Once the solution is obtained for $\vec{\chi}(t)$, an improved estimation of the orbit is obtained:

$$\vec{x}_e(t) = \vec{x}_{\text{ref}}(t) + \vec{\chi}(t), \quad (1.99)$$

such that this solution is adopted as reference a solution in order to repeat the algorithm and get a new estimation which is, in principle, a better approximation. This process is done while the $\vec{\chi}$ deviations are progressively smaller.

Equations (1.94) constitute a system of n linear equations with n unknowns whose associated matrix A depends on time. In general the solution will be given by numerical integration taking into account the time dependence of the matrix A , which makes impossible the estimation of analytical solution. However, the structure of the solution is known and satisfies the initial conditions

$$\vec{\chi}(t) = \Phi(t, t_s)\vec{\chi}_s, \quad (1.100)$$

where $\Phi(t, t_s)$ is a square matrix of order n named state-transition matrix or fundamental matrix. The properties of this matrix are:

- $\Phi(t_s, t_s) = I$ where I is the identity matrix of order n .
- $\Phi(t_i, t_s) = \Phi(t_i, t_j)\Phi(t_j, t_s)$.
- $\Phi^{-1}(t_i, t_s) = \Phi(t_s, t_i)$.

The n independent solutions of the system (1.94) correspond to the n independent solutions associated to the n columns of $\Phi(t, t_s)$. By differentiating Eq. (1.100) and relating with Eq. (1.94) we obtain the relation

$$A\vec{\chi} = \dot{\Phi}(t, t_s)\vec{\chi}_s. \quad (1.101)$$

If Eq. (1.100) is replaced in the previous expression, the following equation is deduced:

$$\Phi(t, t_s) = A\Phi(t, t_s), \quad (1.102)$$

which is the differential equation that satisfies the fundamental matrix. Consequently, the matrix $\Phi(t, t_s)$ is obtained by integrating Eqs. (1.102) from the initial conditions

$$t = t_s : \quad \Phi(t_s, t_s) = I. \quad (1.103)$$

The next step is to establish the relations between the observed quantities and the state vector given by Eq. (1.97) at the instant of interest $t = t_s$. If the fundamental matrix is assumed to be known, by replacing Eq. (1.100) into Eq. (1.97) the following system is obtained:

$$\vec{\mathcal{B}}_1 = H(t_1)\Phi(t_1, t_s)\vec{\chi}_s + \vec{\epsilon}_1 \quad (1.104a)$$

$$\vec{\mathcal{B}}_2 = H(t_2)\Phi(t_2, t_s)\vec{\chi}_s + \vec{\epsilon}_2 \quad (1.104b)$$

$$\vec{\mathcal{B}}_3 = H(t_3)\Phi(t_3, t_s)\vec{\chi}_s + \vec{\epsilon}_3 \quad (1.104c)$$

$$\vdots$$

$$\vec{\mathcal{B}}_l = H(t_l)\Phi(t_l, t_s)\vec{\chi}_s + \vec{\epsilon}_l, \quad (1.104d)$$

where there are $m = pl$ elements. By introducing the vectors

$$\vec{\mathcal{B}} = (\vec{\mathcal{B}}_1, \vec{\mathcal{B}}_2, \dots, \vec{\mathcal{B}}_l)^T = (\mathcal{B}_{11}, \dots, \mathcal{B}_{1p}, \mathcal{B}_{21}, \dots, \mathcal{B}_{2p}, \dots, \mathcal{B}_{l1}, \dots, \mathcal{B}_{lp})^T \quad (1.105)$$

$$\vec{\epsilon} = (\vec{\epsilon}_1, \vec{\epsilon}_2, \dots, \vec{\epsilon}_l)^T = (\epsilon_{11}, \dots, \epsilon_{1p}, \epsilon_{21}, \dots, \epsilon_{2p}, \dots, \epsilon_{l1}, \dots, \epsilon_{lp})^T, \quad (1.106)$$

and the matrix

$$Q = \begin{pmatrix} H(t_1)\Phi(t_1, t_s) \\ H(t_2)\Phi(t_2, t_s) \\ \vdots \\ H(t_l)\Phi(t_l, t_s) \end{pmatrix}, \quad (1.107)$$

the system becomes:

$$\vec{\mathcal{B}} = Q\vec{\chi}_s + \vec{\epsilon}, \quad (1.108)$$

with m equations and $m + n$ unknowns. The n components of the deviations of the state vector are unknown, as well as the m values of the error vector. Accordingly, n conditions still need to be introduced.

1.7.1 The least squares adjustment

The least squares adjustment is a model which provides the missing n equations of system (1.108) to make the problem solvable. This procedure is based on minimizing the sum of the squares of the errors, that is:

$$\min[V(\vec{\chi}_s)] = \min[\vec{\epsilon}^T \vec{\epsilon}] = \min[(\vec{\mathcal{B}} - Q\vec{\chi}_s)^T (\vec{\mathcal{B}} - Q\vec{\chi}_s)]. \quad (1.109)$$

Taking into account that the function $V(\vec{\chi}_s)$ is a quadratic function of $\vec{\chi}_s$, it reaches a minimum value if the following relations are satisfied:

$$\frac{\partial V}{\partial \vec{\chi}_s} = \vec{0}, \quad \frac{\partial^2 V}{\partial \vec{\chi}_s^2} \equiv \text{positive definite}. \quad (1.110)$$

By differentiating the expression of V with respect to $\vec{\chi}_s$ the first condition is obtained

$$\frac{\partial V}{\partial \vec{\chi}_s} = \left[\frac{\partial \vec{\epsilon}^T}{\partial \vec{\chi}_s} \vec{\epsilon} + \vec{\epsilon}^T \frac{\partial \vec{\epsilon}}{\partial \vec{\chi}_s} \right] = 2 \frac{\partial \vec{\epsilon}^T}{\partial \vec{\chi}_s} \vec{\epsilon} = \vec{0}, \quad (1.111)$$

and making use of Eq. (1.108), it is verified if

$$-Q^T(\vec{B} - Q\vec{\chi}_s) = \vec{0}. \quad (1.112)$$

The $\vec{\chi}_s$ value satisfying this equation is the best estimation given by the so called normal equation

$$Q^T \vec{B} = N \vec{\chi}_s, \quad (1.113)$$

with $N = Q^T Q$ the normal matrix of dimension $n \times n$ which is characterized to be regular. Consequently, the solution is

$$\vec{\chi}_s = N^{-1} Q^T \vec{B}. \quad (1.114)$$

In some cases the observations may be weighted if they are not equally reliable. Consequently, the function to be minimized becomes

$$\min[V(\vec{\chi}_s)] = \min[\vec{\epsilon}^T W \vec{\epsilon}], \quad (1.115)$$

with W the weight matrix of dimension $m \times m$. The normal equation is now

$$Q^T W \vec{B} = N \vec{\chi}_s, \quad (1.116)$$

with $N = Q^T W Q$ the normal matrix of dimension $n \times n$. Consequently, the solution is

$$\vec{\chi}_s = N^{-1} Q^T W \vec{B}. \quad (1.117)$$

1.7.2 Kalman filtering

The Kalman filtering is a recursive algorithm which provides real-time estimates of the orbital states $\vec{x}_i$ and the error covariances based on sequential observations. The iterative process divides each iteration in two steps: prediction and correction.

The results provided by the prediction, that is, the differential of the state vector and the covariance matrix, will be defined with the notation $\tilde{\chi}$ and $\tilde{P}$ respectively. These are the variables of the previous equations. The same variables after the correction will be defined with the notation $\hat{\chi}$ and $\hat{P}$, corresponding to the variables of the equations obtained.

By considering the reference solution good enough to not have to change it in each time ($\tilde{x}_i = \vec{x}_{ref}(t_i)$), the algorithm is, for each time $t = t_i$, as follows:

Prediction

- $\tilde{\chi}_i = \Phi(t_i, t_{i-1}) \hat{\chi}_{i-1}$ with $\hat{\chi}_0 = \vec{0}$
- $\tilde{P}_i = \Phi(t_i, t_{i-1}) \hat{P}_{i-1} \Phi^T(t_i, t_{i-1}) + Q$

Correction

- $K_i = \tilde{P}_i H_i^T (H_i \tilde{P}_i H_i^T + R_i)^{-1}$
- $\hat{\chi}_i = \tilde{\chi}_i + K_i (\vec{B}_i - H_i \tilde{\chi}_i)$
- $\hat{P}_i = \tilde{P}_i - K_i H_i \tilde{P}_i$

with R_i the covariance matrix of the observations.

In some cases the Kalman filter can be extended such that the reference trajectory is updated in each step with the information provided by the differential of the state vector.

THE ELLIPTIC KEPLER'S EQUATION

The preliminary orbit determination of terrestrial satellites constitutes the first step of the orbit determination process. Starting from the observations to the satellite, the Keplerian orbit of the body is determined by neglecting the perturbations acting upon it.

In what follows an improved preliminary orbit determination is developed. The approach is assessed by solving the elliptic and hyperbolic Kepler's equation as test cases. This chapter contains the resolution for the elliptic Kepler's equation.

2.1 INTRODUCTION

The basic goal of Kepler's equation is to provide the value of the eccentric anomaly E in terms of the mean anomaly M and the eccentricity e of the orbit:

$$M = E - e \sin(E). \quad (2.1)$$

Over the centuries the resolution of Kepler's equation has been studied by a wide variety of scientists such that in virtually every decade from 1650 to the present papers devoted to Kepler's problem and its solution have been published (Colwell [19]). Although there exists a unique solution to Kepler's equation, there are many methods to describe or approximate it. Basically it depends on the solvers' motivations and the mathematical tools available in the epoch of study.

Before 1860 the work was motivated directly by celestial mechanics, but it showed results less sensitive to the needs of astronomers. After 1860 the development and refinement of what we now call special functions of applied mathematics became the involvement of mathematicians in solving Kepler's equation more outstanding. In the case of astronomers, they were generally more interested in being able to solve Kepler's equation quickly and repeatedly, but with only modest accuracy (Rambaut [67]). From 1860 to some time after 1890, the work on Kepler's equation was mostly derivative and computational (Colwell [19]). We can highlight the work developed by Cassini (1719) providing a nonanalytic solution, the infinite series solution developed by Levi-Civita (1904) or the iterative methods based on the Newton's Principia (1686).

From mid-20th century, the classic and more recent methods focused on the development of optimal procedures where the accuracy of the algorithm and the computational effort become the main aspects to consider. Most of the algorithms are iterative and they depend on how the starting estimate for E is determined. Nevertheless, nowadays it is possible to find analytical algorithms as the method described in Pál [59] to solve Kepler's problem. In what follows we briefly describe some of the approaches currently used to solve Eq. (2.6):

- The Serafin [78] approach: the bounds for E are chosen through several inequalities to successively divide the root interval and improve the lower and upper bounds. To do that, two linear methods based on elementary properties of convex functions are used.
- The Gooding and Odell [36] approach: the starting value for E is provided by solving a cubic equation. After that, a corrected value is given through Halley's method which is used as input to approximate the evaluation of Eq. (2.6) and its derivative by their Taylor series. The final value is computed with

the Newton-Raphson method. It is an iterative method which requires several transcendental function evaluations.

- The Nijenhuis [58] approach: the starting for E is computed depending on four different regions of the (M, e) -space. In three of these regions the starting value is defined with a Halley step based on a version of Kepler's equation in which the sine function is replaced by a simple approximation. In the remaining region, which contains the singular point, the starting value is given by the solution of Mikkola's cubic equation and refined through a Newton step applied to a quintic equation. An m -th order generalized Newton method is applied thereafter to obtain the final solution, where m depends on the accuracy required. It is an iterative method which requires two trigonometric evaluations.
- The Markley [53] approach: the starter point for E is the root of a cubic equation, whose expression is derived through the Padé approximation for $\sin(E)$, which is replaced in Eq. (2.6). After that, a fifth-order refinement of the solution is applied only once. It is a non-iterative method which requires four transcendental function evaluations (a square root, a cube root and two trigonometric functions).
- The Fukushima [32] approach: the starting value for E is a trivial upper bound of the solution: π . The approximate solution is determined by transforming the Newton-Raphson method from the continuous x -space to a discretized j -space, where x represents E and j the solution index. The corrected value for E is another approximation of the Newton-Raphson method by approximating the evaluation of Eq. (2.6) and the derivative by their truncated Taylor series around the approximate solution. It is an iterative method which does not require transcendental functions.
- The Mortari and Elife [57] approach: two ranges of the mean anomaly are defined to select where the solution should be determined. The lower bound for E is obtained from two implicit functions, which are non-rational Bézier functions, linear or quadratic, depending on the derivatives of the initial bound values. The upper bound for E is estimated by applying the Newton-Raphson method with the lower bound as the starting value. After that, if the machine error accuracy is not reached, the lower and upper bounds define a new range of searching. It is an iterative method which requires several transcendental function evaluations (one quadratic iteration requires nineteen additions, twenty products and one square root, while one linear iteration requires seven additions and five products).

In the last century, with the advent of calculators and computers, there is no impediment to achieving quick solutions of great accuracy. In particular, symbolic manipulators like Maple or Mathematica are very powerful calculators which are easy to use and have a very intuitive syntax. Besides, they can be directed to provide the corresponding code in some of the standard programming languages such as C, C++ or Fortran, which are specially adapted for numerical calculus. In this chapter a procedure to solve Kepler's equation for elliptic orbits is developed taking advantage of the full potential of symbolic manipulators. The code is called Elliptic Kepler Equation-Space Dynamics Group (EKE-SDG) (Raposo-Pulido and Peláez [68]). We focus the study on the determination of an appropriate seed to initialize the numerical method for solving Kepler's equation, considering the already tested optimization of the well known Newton-Raphson method.

2.2 THE EQUATION

Kepler's equation (KE) for the elliptical case

$$x = y - e \sin(y), \quad (2.2)$$

determines a nonlinear function $y = y(x, e)$ where $y = E$ is the unknown eccentric anomaly, $M = x$ the mean anomaly which is known and $e < 1$ the eccentricity. For given values of $e \in [0, 1[$ and $M \in [0, 2\pi]$ the function $y = y(x, e)$ is univocally defined by Eq. (2.2) and this property can be deduced from the Banach fixed-point theorem.

The function $g(y)$ is contractive if there is a nonnegative real number $k \in]0, 1[$ such that $\forall y_1, y_2 \in [0, 2\pi]$:

$$|g(y_1) - g(y_2)| \leq k|y_1 - y_2|. \quad (2.3)$$

The smallest value for k is the Lipschitz constant of g . If we consider the equation $g(y) = x + e \sin(y)$, the function $g(y)$ is a contractive mapping for every value of x , since its derivative $g'(y) = e \cos(y)$ satisfies the condition $|g'(y)| < 1$ when $e \in [0, 1[$. In fact, applying the mean value theorem to the function $g(y)$, there exists an intermediate value $y^* \in [y_1, y_2]$ which verifies:

$$g(y_2) - g(y_1) = g'(y^*)(y_2 - y_1). \quad (2.4)$$

Taking into account that $g'(y) = e \cos(y)$, we have

$$|g(y_2) - g(y_1)| = |e \cos(y^*)| |y_2 - y_1| \leq e|y_2 - y_1|, \quad (2.5)$$

that is, k , the Lipschitz constant of g , verifies $k < e$, proving the property.

Finally, applying the Banach fixed-point theorem for contractive functions, it can be concluded that there is a fixed point which verifies $y = g(y)$ and this fixed point is the sought for as the solution of Kepler's equation. Starting from a given value y_0 , for example $y_0 = x$, the sequence $y_{n+1} = g(y_n)$ turns out to be convergent and its limit is the fixed point. However, this sequence is slowly convergent—specially when e is close to unity—and this way to solve KE is not effective from a numerical point of view. Thus, the interest of this procedure is much more theoretical than practical.

The solution of KE can be seen as a root finding problem of the function

$$f(y) \equiv y - e \sin(y) - x = 0, \quad (2.6)$$

for given values of e and x . Due to the transcendental character of the equation its solution is non-trivial. Nevertheless, by introducing the function

$$F(y) = y - \frac{f(y)}{f'(y)} = \frac{e(\sin(y) - y \cos(y)) + x}{1 - e \cos(y)}, \quad (2.7)$$

where $f'(y)$ is the derivative of $f(y)$, the solution of the KE reduces to the search of the fixed point of $F(y)$; note that $y = F(y) \Leftrightarrow f(y) = 0$. Again, we can form a convergent sequence $y_{n+1} = F(y_n)$ whose limit is the root of the function $f(y)$; the convergence of this new sequence is faster than the previous one (Stumpf [85]) but still insufficient from a practical point of view.

There are several analytical solutions given by expansions in terms of Bessel functions of first kind like:

$$E = M + 2 \sum_{k=1}^{\infty} J_k(ke) \frac{\sin(kM)}{k} \quad (2.8a)$$

$$J_k(ke) = \frac{1}{\pi} \int_0^{\pi} \cos(ku - ke \sin(u)) du. \quad (2.8b)$$

More analytical solutions can be found in Battin [11], Colwell [19], and Siewert and Burniston [80]. However, these expansions are not competitive among the numerical methods, as it is commented in the book Danby [21].

2.2.1 Domain of the equation

Initially, x and y range in the interval $[0, 2\pi]$; however, the transformation

$$y = 2\pi - \eta, \quad x = 2\pi - \xi, \quad (2.9)$$

converts Eq. (2.2) into:

$$\xi = \eta - e \sin(\eta). \quad (2.10)$$

As a consequence, if $x, y \in]\pi, 2\pi]$, then $\xi, \eta \in [0, \pi]$ and the mapping (2.9) reduces the problem to the interval $[0, \pi]$.

2.3 NEWTON-RAPHSON ALGORITHMS

The root sought is the solution Eq. (2.6) and we wish to establish a successive approximation method starting from a seed y_0 . In the classical Newton-Raphson (CNR) algorithm a sequence is formed whose elements become arbitrarily close to each other as the sequence progresses:

$$y_{n+1} = y_n - \frac{f(y_n)}{f'(y_n)}. \quad (2.11)$$

This Cauchy sequence converges to the solution and it is very well known that the convergence rate of this algorithm is quadratic; thus the number of correct figures is doubled, approximately, in each iteration of the sequence.

A better convergence rate is obtained with the modified Newton-Raphson (MNR) algorithm which is used in the EKE-SDG code to solve the Kepler equation. If y_n in one of the terms of the sequence, the next term will be $y_{n+1} = y_n + \Delta y_n$, where Δy_n should verify

$$f(y_{n+1}) = f(y_n + \Delta y_n) = 0, \quad (2.12)$$

with f given by Eq. (2.6). By approximating $f(y_{n+1})$ by its second order Taylor expansion about y_n it is obtained

$$f(y_{n+1}) \approx f(y_n) + f'(y_n)\Delta y_n + \frac{1}{2}f''(y_n)\Delta^2 y_n, \quad (2.13)$$

and the condition $f(y_{n+1}) = 0$ provides the value of Δy_n :

$$\Delta y_n = \frac{-f'(y_n) \pm \sqrt{f'^2(y_n) - 2f(y_n)f''(y_n)}}{f''(y_n)}. \quad (2.14)$$

Then, we multiply numerator and denominator by

$$-f'(y_n) \mp \sqrt{f'^2(y_n) - 2f(y_n)f''(y_n)}, \quad (2.15)$$

to obtain:

$$\Delta y_n = \frac{-2f(y_n)}{f'(y_n) \pm \sqrt{f'^2(y_n) - 2f(y_n)f''(y_n)}}, \quad (2.16)$$

where we have to select the sign (+) when $f'(y_n)$ is positive; if $f'(y_n) < 0$ the sign (−) must be taken.

Table 1: The right answer is $y = 2.78172231$.

Sequences for $M = 2.5$ and $e = 0.8$			
Seed $y_0 = 0.1$		Seed $y_0 = 0.3$	
CNR	MNR	CNR	MNR
12.25640804	5.82975453	10.6355865	3.95106621
-29.74030805	1.97868144	3.70452359	2.86354381
0.74394754	2.76120459	2.73142300	2.78176216
6.32984624	2.78172169	2.78194606	2.78172231
-12.55060918	2.78172231	2.78172231	2.78172231
62.72807874	2.78172231	2.78172231	2.78172231

The use of absolute value in Eq. (2.16) does not affect the convergence of the algorithm. It is introduced to prevent the algorithm from failing during the series generation when the square root is a complex number. The first and second derivatives of Eq. (2.6) are given by the relations

$$f'(y_n) = 1 - e \cos(y_n) \quad (2.17)$$

$$f''(y_n) = e \sin(y_n). \quad (2.18)$$

Equation (2.16) summarizes the MNR algorithm. It should be noticed that the algorithm becomes the CNR method, in which the function is approximated by its first order Taylor expansion, when $f''(y_n) = 0$. In general, the convergence rate of the MNR algorithm is cubic.

The MNR algorithm is a particular case of the algorithm used in the Conway's method. Bruce A. Conway applied a root-finding method of Laguerre (1834-1886) to the solution of Kepler's equation (Conway [20]). Although the method is intended for finding the roots of a polynomial, it works equally well for Kepler's equation. The algorithm associated to that method is summarized by the equation

$$\Delta y_n = \frac{-(1+p)f(y_n)}{f'(y_n) \pm \sqrt{p[p f'^2(y_n) - (1+p)f(y_n)f''(y_n)]}}, \quad (2.19)$$

with $p = m - 1$ and m is the degree of the polynomial. The convergence of the algorithm is almost independent of m ; Conway opts for $m = 5$. As we can see, the MNR method is a particular case of Conway's method for $m = 2$.

The main difference between the CNR and the Conway or the MNR methods is associated to its reliability. The three methods need an initial seed from which start the generation of the convergent sequence. The CNR method is much more affected by the quality of the initial seed than the Conway's method or the MNR method. If the quality of the initial seed is poor the CNR method could not converge to the solution.

Table 1 shows the different sequences generated by the CNR and MNR algorithms when trying to solve Kepler's equation with values $M = 2.5$ and $e = 0.8$ and two different seeds: the bad one, $y_0 = 0.1$, and another less bad, $y_0 = 0.3$. Only the first six terms of the sequence have been calculated. For the bad seed the CNR does not converge to the real solution; on the contrary, the MNR always converges to the real answer even though we start from a very poor seed. However, the behavior of the algorithms is quite different if we use a very good seed as starting point. Table 2 shows the first six terms of the sequences generated for the same values $M = 2.5$,

Table 2: The right answer is $y = 2.781722308989884$.

Sequences for $M = 2.5$ and $e = 0.8$	
Seed $y_0 = 2.8$	
CNR	MNR
2.781748270190563	2.781722746897827
2.781722309044170	2.781722308989884
2.781722308989884	2.781722308989884
2.781722308989884	2.781722308989884
2.781722308989884	2.781722308989884
2.781722308989884	2.781722308989884

$e = 0.8$ when the initial seed is very good: $y_0 = 2.8$. Both algorithms converge quickly to the real answer.

The accuracy of the Newton-Raphson algorithms is jeopardized when $M \ll 1$ and $1 - e \ll 1$; we call this region of the plane (e, M) , the singular corner. In such a region y is also small, and the expansion of the equation for small values of y provides

$$y \approx \frac{x}{1 - e}, \quad (2.20)$$

that is, the solution is ill-conditioned from a numerical point of view since it is the ratio of two infinitesimals.

The central idea, and also the goal of this work, is to generate a very good initial seed in all possible cases, also in the singular corner. Then we will use one of the Newton-Raphson algorithms to find the right solution of Kepler's equation. Roughly, we exploit the very good characteristics of the Newton-Raphson algorithms (CNR, MNR or Conway) because they assure the convergence and provide the solution with accuracy and velocity if the quality of the initial seed is high. It is much more cost-effective, from a computational point of view, to improve the initial seed and then use it to feed a Newton-Raphson algorithm than to look for new algorithms which would be able to beat the simplicity and performance of Newton-Raphson algorithms.

Moreover, the solution of KE is an issue with more than three centuries (Colwell [19]); along such a long period of time, several methods have been proposed, most of them associated with mathematical tools or techniques of its epoch. At present, one important mathematical tool which characterizes this epoch is the symbolic manipulator (Macsyma, Maxima, Derive, Maple, Mathematica, etc). We try to exploit these tools to their full usefulness, in order to generate a C or FORTRAN code that is robust, reliable, exact and fast.

2.4 THE SEED VALUE

The approach starts by discretizing the eccentric anomaly domain $[0, \pi]$ in twelve intervals of 15° in width, where the corresponding $M_i \equiv x_i$ is obtained from Eq. (2.2) (see Table 3). This way, we define twelve intervals $[x_i, x_{i+1}]$, $i = 1, \dots, 12$.

In each of these intervals we introduce a fifth degree polynomial $p_i(x, e)$, $i = 1, \dots, 12$ to interpolate the eccentric anomaly

$$p_i(x, e) = a_0^{(i)}(e) + a_1^{(i)}(e)x + a_2^{(i)}(e)x^2 + a_3^{(i)}(e)x^3 + a_4^{(i)}(e)x^4 + a_5^{(i)}(e)x^5. \quad (2.21)$$

Notice that the polynomials used in the method have coefficients that depend on the eccentricity (see Appendix A). In order to simplify the equation, we make an abuse of notation by writing the coefficients only as a function of i , that is

$$p_i(x) = a_0^{(i)} + a_1^{(i)}x + a_2^{(i)}x^2 + a_3^{(i)}x^3 + a_4^{(i)}x^4 + a_5^{(i)}x^5. \quad (2.22)$$

The polynomial $p_i(x)$ is only valid inside the interval $[x_i, x_{i+1}]$, $i = 1, \dots, 12$, where we approximate the function by the polynomial: $y(x) \approx p_i(x)$.

In order to determine the six coefficients of $p_i(x)$ the following six boundary conditions should be imposed

$$p_i(x_i) = y(x_i), \quad p_i(x_{i+1}) = y(x_{i+1}), \quad (2.23a)$$

$$p_i'(x_i) = y'(x_i), \quad p_i'(x_{i+1}) = y'(x_{i+1}), \quad (2.23b)$$

$$p_i''(x_i) = y''(x_i), \quad p_i''(x_{i+1}) = y''(x_{i+1}), \quad (2.23c)$$

where the two first derivatives of y are given by:

$$y' = \frac{dy}{dx} = \frac{1}{1 - e \cos(y)} \quad (2.24)$$

$$y'' = \frac{d^2y}{dx^2} = \frac{-e \sin(y)}{(1 - e \cos(y))^3}, \quad (2.25)$$

and they can be easily calculated at the ends of the interval. These conditions assure that the function $y(x)$ and the polynomial $p_i(x)$ share the same values and the same first and second derivatives at the boundaries of the interval $[x_i, x_{i+1}]$.

The polynomials $p_i(x)$, $i = 1, \dots, 12$, can be generated easily with the aid of a symbolic manipulator once and for all; moreover, the symbolic manipulator provides the FORTRAN or C code of the polynomials that can be appropriately stored in one module to be used later. We used the version 18.02 of Maple.

Consequently, given e and M , we determine the starting value y_0 from the following procedure:

1. First, given e the ends x_i, x_{i+1} of the intervals described in Table 3 are calculated.
2. The interval $[x_i, x_{i+1}]$ should be selected in such a way that $M \in [x_i, x_{i+1}]$. Note that the corresponding values $[y_i, y_{i+1}]$ are known and the seed y_0 satisfies: $y_0 \in [y_i, y_{i+1}]$.
3. Use the corresponding polynomial $p_i(x)$ to obtain the seed to be used with the Newton-Raphson algorithm:

$$y_0 = p_i(M). \quad (2.26)$$

Most of the times, this initial seed is a very good approximation to the solution of KE; specially when M is close to the ends of the interval $[x_i, x_{i+1}]$ since this seed is practically the solution of Kepler's equation. However, the singular behavior of KE near the singular corner is well known. In this zone the approximation of $y(x)$ by means of the polynomials $p_i(x)$ is not very effective and a special analysis should be performed.

2.5 ANALYSIS IN THE SINGULAR CORNER

To describe the solution close to the singular corner we introduce the value $\epsilon = 1 - e$ assuming that $\epsilon \ll 1$:

$$x = y - (1 - \epsilon) \sin(y) = y - \sin(y) + \epsilon \sin(y). \quad (2.27)$$

Table 3: Important values for the polynomial approach, where $\gamma_{is} = \sin((i-1)\pi/12)$ and $\gamma_{ic} = \cos((i-1)\pi/12)$.

Intervals defined in the solution				
i	$x_i = y_i - e \sin y_i$	$y_i = (i-1)\frac{\pi}{12}$	$\frac{dy}{dx} \Big _{x_i}$	$\frac{d^2y}{dx^2} \Big _{x_i}$
1	0	0	$\frac{1}{1-e}$	0
2	$\frac{\pi}{12} - \gamma_{2s}e$	$\frac{\pi}{12}$	$\frac{1}{1-\gamma_{2c}e}$	$\frac{-\gamma_{2s}e}{(1-\gamma_{2c}e)^3}$
3	$\frac{\pi}{6} - \frac{1}{2}e$	$\frac{\pi}{6}$	$\frac{1}{1-\gamma_{3c}e}$	$\frac{-e}{2(1-\gamma_{3c}e)^3}$
4	$\frac{\pi}{4} - \gamma_{4s}e$	$\frac{\pi}{4}$	$\frac{1}{1-\gamma_{4c}e}$	$\frac{-\gamma_{4s}e}{(1-\gamma_{4c}e)^3}$
5	$\frac{\pi}{3} - \gamma_{5s}e$	$\frac{\pi}{3}$	$\frac{2}{2-e}$	$\frac{-8\gamma_{5s}e}{(2-e)^3}$
6	$\frac{5\pi}{12} - \gamma_{6s}e$	$\frac{5\pi}{12}$	$\frac{1}{1-\gamma_{6c}e}$	$\frac{-\gamma_{6s}e}{(1-\gamma_{6c}e)^3}$
7	$\frac{\pi}{2} - e$	$\frac{\pi}{2}$	1	-e
8	$\frac{7\pi}{12} - \gamma_{6s}e$	$\frac{7\pi}{12}$	$\frac{1}{1+\gamma_{6c}e}$	$\frac{-\gamma_{6s}e}{(1+\gamma_{6c}e)^3}$
9	$\frac{2\pi}{3} - \gamma_{5s}e$	$\frac{2\pi}{3}$	$\frac{2}{2+e}$	$\frac{-8\gamma_{5s}e}{(2+e)^3}$
10	$\frac{3\pi}{4} - \gamma_{4s}e$	$\frac{3\pi}{4}$	$\frac{1}{1+\gamma_{4c}e}$	$\frac{-\gamma_{4s}e}{(1+\gamma_{4c}e)^3}$
11	$\frac{5\pi}{6} - \frac{1}{2}e$	$\frac{5\pi}{6}$	$\frac{1}{1+\gamma_{3c}e}$	$\frac{-e}{2(1+\gamma_{3c}e)^3}$
12	$\frac{11\pi}{12} - \gamma_{2s}e$	$\frac{11\pi}{12}$	$\frac{1}{1+\gamma_{2c}e}$	$\frac{-\gamma_{2s}e}{(1+\gamma_{2c}e)^3}$
13	π	π	$\frac{1}{1+e}$	0

The goal is to describe numerically the exact solution $y_v(x)$ of Eq. (2.27) with sufficient accuracy to be part of the seed used to start the Newton-Raphson convergent process. In order to do that, an asymptotic expansion in power of the small parameter ϵ will be obtained.

In the singular corner, when x is small, y is also small. However, it is possible to distinguish several regimes inside this region:

- The inner region, where $x \approx \mathcal{O}(\epsilon^2)$ and $y \approx \mathcal{O}(\epsilon)$. In this inner region the most important term on the right-hand side of Eq. (2.27) is: $\epsilon \sin(y)$.
- The intermediate region, where $x \approx \mathcal{O}(\epsilon^{\frac{3}{2}})$ and $y \approx \mathcal{O}(\epsilon^{\frac{1}{2}})$. In this intermediate region both terms on the right-hand side of Eq. (2.27) are of the same order: $y - \sin(y) \approx \epsilon \sin(y)$.
- The outer region, where $x \approx \mathcal{O}(\epsilon^{\frac{3}{4}})$ and $y \approx \mathcal{O}(\epsilon^{\frac{1}{4}})$. In this outer region the most important term on the right-hand side of Eq. (2.27) is: $y - \sin(y)$.

In each one of these regions it is possible to obtain an asymptotic expansion of the exact solution $y_v(x)$ of Eq. (2.27) in terms of powers of the small parameter ϵ . During the study we assessed the boundaries of the regions, obtaining the best accuracy when

- The inner region is defined, approximately, for (x, e) satisfying $x < 0.001 \epsilon^{\frac{3}{2}}$.
- The intermediate and outer region are defined in the same region for (x, e) satisfying $x \geq 0.001 \epsilon^{\frac{3}{2}}$.

As a consequence, only two regions in the singular corner are considered at the end: inner and intermediate-outer; these approximate solutions permit to generate a very good seed to feed the Newton-Raphson convergent process (Fig. 2.1).

2.5.1 Inner region

In the inner region we introduce the following change of variables:

$$x = \epsilon^2 \xi, \quad y = \epsilon \eta, \quad \text{with} \quad \xi, \eta \approx \mathcal{O}(1). \quad (2.28)$$

Note that (ξ, η) are both positive amounts. We look for a solution to Eq. (2.27) given by:

$$\xi = \eta + a_1 \epsilon + a_2 \epsilon^2 + a_3 \epsilon^3 + \dots, \quad (2.29)$$

where the coefficients a_i , $i = 1, 2, \dots$ can be obtained expanding appropriately the right-hand side of Eq. (2.27). The solution is:

$$\xi = \eta + \frac{1}{6} \eta^3 \epsilon - \frac{1}{6} \eta^3 \epsilon^2 - \frac{1}{120} \eta^5 \epsilon^3 + \frac{1}{120} \eta^5 \epsilon^4 + \dots \quad (2.30)$$

The next step is to invert this solution to obtain η as a function of ξ ; to do that we introduce the expansion

$$\eta = \xi + b_1 \epsilon + b_2 \epsilon^2 + b_3 \epsilon^3 + \dots, \quad (2.31)$$

where the coefficients b_i , $i = 1, 2, \dots$ are functions of ξ that must be calculated. Introducing this expansion in Eq. (2.30) and requiring that the different coefficients of each order match, it is possible to obtain the values of b_i , $i = 1, 2, \dots$. The result is:

$$\eta = \xi - \frac{1}{6} \xi^3 \epsilon + \frac{1}{12} \xi^3 (\xi^2 + 2) \epsilon^2 - \frac{1}{360} \xi^5 (20\xi^2 + 57) \epsilon^3 + \dots \quad (2.32)$$

2.5.2 Intermediate-outer region

In the intermediate region we introduce the following change of variables:

$$x = \epsilon^{\frac{3}{2}} \chi, \quad y = \epsilon^{\frac{1}{2}} \sigma, \quad \text{with} \quad \chi, \sigma \approx \mathcal{O}(1). \quad (2.33)$$

Note that (χ, σ) are both positive amounts. We look for a solution to Eq. (2.27) given by:

$$\chi = a_0 + a_1 \epsilon + a_2 \epsilon^2 + a_3 \epsilon^3 + \dots, \quad (2.34)$$

where the coefficients a_i , $i = 1, 2, \dots$ can be obtained expanding appropriately the right-hand side of Eq. (2.27). The solution is:

$$\chi = \frac{1}{6} \sigma^3 + \sigma - \frac{1}{120} \sigma^3 (\sigma^2 + 20) \epsilon + \frac{1}{5040} \sigma^5 (\sigma^2 + 42) \epsilon^2 + \dots \quad (2.35)$$

Let $\sigma_0 > 0$ be the real and positive solution of the cubic equation

$$f(\sigma) = \frac{1}{6} \sigma^3 + \sigma - \chi = 0 \Leftrightarrow f(\sigma) = \sigma^3 + 6\sigma - 6\chi = 0. \quad (2.36)$$

This root always exists; in fact, the function $f(\sigma)$ verifies:

$$f(0) = -6\chi < 0 \quad (2.37a)$$

$$f(\chi) = \chi^3 > 0, \quad (2.37b)$$

therefore in the interval $[0, \chi]$ there is a root of $f(\sigma) = 0$ at least. But in fact there is only one root, because the derivative

$$f'(\sigma) = 3\sigma^2 + 6 > 0, \quad (2.38)$$

is always positive in that interval. That root is given by:

$$\sigma_0 = S - \frac{2}{S}, \quad (2.39)$$

where $S = (\Lambda + 3\chi)^{\frac{1}{3}}$ and $\Lambda = \sqrt{8 + 9\chi^2}$. It should be noted that Eq. (2.39) can exhibit a rounding-error defect when $S \approx \sqrt{2}$ ($\sigma_0 \ll 1$). In order to minimize this error, we develop an alternative procedure to provide σ_0 . We rewrite the cubic equation (2.36) as

$$\sigma(\sigma^2 + 6) - 6\chi = 0 \Leftrightarrow \sigma = \frac{6\chi}{\sigma^2 + 6}, \quad (2.40)$$

such that σ is replaced by Eq. (2.39) on the right-hand side of the equation, obtaining

$$\sigma_0 = \frac{6\chi}{2 + S^2 + \frac{4}{S^2}}, \quad (2.41)$$

which provides a smaller rounding-error when $\chi \ll 1$, that is, when $\Lambda \approx 2\sqrt{2}$.

Once the value of σ_0 is known it is possible to invert the solution (2.35) to obtain σ as a function of χ ; to do that we introduce the expansion

$$\sigma = \sigma_0 + b_1 \epsilon + b_2 \epsilon^2 + b_3 \epsilon^3 + \dots, \quad (2.42)$$

where the coefficients b_i , $i = 1, 2, \dots$ are functions of σ_0 that must be calculated. Introducing this expansion in (2.35) and requiring that the different coefficients of each order match, it is possible to obtain the values of b_i , $i = 1, 2, \dots$. The result is:

$$\sigma = \sigma_0 + \frac{\sigma_0^3 (\sigma_0^2 + 20)}{60(\sigma_0^2 + 2)} \epsilon + \frac{\sigma_0^5 (\sigma_0^6 + 25\sigma_0^4 + 340\sigma_0^2 + 840)}{1400(\sigma_0^2 + 2)^3} \epsilon^2 + \dots \quad (2.43)$$

Figure 2.1: The exact solution to Kepler's equation and the asymptotic solution —up to third order— obtained in the singular corner. The agreement is excellent. Here five values of ϵ have been considered: $\epsilon = 0.001, 0.051, 0.101, 0.151, 0.201$.

2.6 THE EKE-SDG ALGORITHM

Given the values of M and e , in order to solve the elliptic Kepler's equation we use the following algorithm:

Seed estimation:

- Inside the singular corner ($M < 0.15$ and $e > 0.75$)
  1. If $x < 0.001 \epsilon^{\frac{3}{2}}$, use the sequence associated with Eq. (2.32) to estimate the starting value y_0 .
  2. If (x, e) satisfying $x \geq 0.001 \epsilon^{\frac{3}{2}}$, use the sequence associated with Eq. (2.43) to estimate the starter value y_0 .
- Outside the singular corner ($M \geq 0.15$ or $e \leq 0.75$)
  1. The ends x_k, x_{k+1} of the intervals described in Table 3 are calculated. The interval $[x_k, x_{k+1}]$ should be selected in such a way that $M \in [x_k, x_{k+1}]$. Note that the corresponding values $[y_k, y_{k+1}]$ are known and the seed y_0 satisfies: $y_0 \in [y_k, y_{k+1}]$.
  2. The seed y_0 is given by the corresponding polynomial (2.22).

Root estimation:

- The seed $y_0 = E_0$ is used to solve the equation

$$y - e \sin(y) - x = 0, \quad (2.44)$$

by using the Newton-Raphson algorithm.

- The output of this algorithm provides the desired value of the eccentric anomaly $y(x) = E$.

The EKE-SDG code is detailed in Appendix B.

2.7 ANALYSIS OF RESULTS

We performed an exhaustive numerical analysis of the algorithm that we propose in these pages. Two main aspects have been considered: 1) accuracy and 2) speed.

Figure 2.2: Results with the SDG-code (12 polynomials) using a modified Newton-Raphson (MNR) iteration scheme.

There are additional considerations that must be taken into account like, for example, the reliability of the procedure, the ease of implementation and so on. We start analyzing the speed.

2.7.1 Speed of the process

Kepler's equation for the elliptical case has been solved with the method proposed in this study by using the modified Newton-Raphson (MNR) algorithm. Kepler's equation has been solved roughly $4 \cdot 10^6$ times, using double and quadruple precision and a tolerance equal to $\varepsilon_{\text{tol}} = 1.11 \cdot 10^{-15}$ and $\varepsilon_{\text{tol}} = 1.0 \cdot 10^{-24}$ respectively. It should be noted that when working with tolerances close to machine accuracy, any numerical procedure based on successive approximations could be affected by artificial numerical chaos. By using an appropriate tolerance it is possible to escape from such a numerical chaos. In each run we count the number of iterations needed to reach a solution with a residual $\rho = |y - e \sin(y) - x|$ lower than the tolerance ε_{tol} . This number is taken as a measure of the speed of the procedure.

Figure 2.2 shows, in the plane (e, M) the number of iterations needed to reach a residual ρ lower than the tolerance ε_{tol} ; this number is always less or equal to unity. Note that in the color code of the figure, white color corresponds to one iteration and black color to zero iterations. The green color (2 iterations) and red color (3 iterations) do not appear in the figure. The averaged value obtained in the calculations is 0.9843 iterations. The reliability of this code is remarkable because after many millions of runs, the algorithm never enters in dead loops or other kind of non-convergent phases.

Two different calculations have been carried out: 1) using standard double precision and 2) using quadruple precision. In both cases the results can be summarized

Figure 2.3: The maximum total absolute error $\varepsilon_{\max}$ versus the eccentricity e , for different number of iterations. Calculations in double precision and using 12 polynomials.

in Figure 2.2 because from the point of view of the number of iterations there are no differences in both calculations.

2.7.2 Accuracy analysis

In this section the analysis is focused on the EKE-SDG code with the iteration scheme provided by the MNR algorithm. In this code the iteration ends when the residual $\rho = |y - e \sin(y) - x|$ is lower that the zero of the machine $\varepsilon_{\text{tol}} = 2.22 \cdot 10^{-16}$.

Let us consider the true solution $y_v(x, e)$ for given values of e and the mean anomaly $M = x$. Let $y_c(x, e)$ be the numerical solution provided by the EKE-SDG code for these particular values. Obviously, due to truncation and round-off errors it is possible to write

$$y_c = y_v + \varepsilon_{\text{abs}}, \quad (2.45)$$

where ε_{abs} is the total absolute error associated with the numerical solution $y_c(x, e)$. If the numerical solution $y_c(x, e)$ is a good approximation to the true solution, the value of ε_{abs} is an infinitesimal: $\varepsilon_{\text{abs}} \ll 1$. In such a case, this absolute error is closely related to the residual ρ . In fact, for the numerical solution $y_c(x, e)$ the residual is given by:

$$\rho = |y_v + \varepsilon_{\text{abs}} - e \sin(y_v + \varepsilon_{\text{abs}}) - x|. \quad (2.46)$$

Taking into account that the true solution verifies

$$y_v - e \sin(y_v) - x = 0, \quad (2.47)$$

the residual takes the form

$$\rho = |\varepsilon_{\text{abs}}| |1 - e \cos(y_v)| + \mathcal{O}(\varepsilon_{\text{abs}}^2). \quad (2.48)$$

As a consequence the total absolute error is given by:

$$|\varepsilon_{\text{abs}}| = \frac{\rho}{|1 - e \cos(y_v)|}. \quad (2.49)$$

Figure 2.4: The maximum total absolute error $\varepsilon_{\max}$ versus the eccentricity e , for different number of iterations. Calculations in quadruple precision and using 12 polynomials.

Basically, it depends on the residual ρ , just like the total relative error

$$|\varepsilon_{\text{rel}}| = \frac{|y_c - y_v|}{|y_v|} = \frac{\rho}{|y_v (1 - e \cos(y_v))|}. \quad (2.50)$$

Let us fix the value of the eccentricity e ; then we scan the whole interval $M \in [0, \pi]$ calculating the residual ρ prior to any iteration (the residual provided by the starting seed), after one iteration, two iterations and so on. Let $\rho_{\max}$ be the maximum residual that we found while scanning the whole interval $M \in [0, \pi]$ for each iteration number. These maximum residuals permit to calculate the maximum values of the absolute error $\varepsilon_{\max}$ for different number of iterations, and these absolute errors can be associated with the value of the eccentricity e .

By plotting the values of such maximum absolute errors versus the eccentricity e , the accuracy involved in the calculations after zero, one, two,... iterations can be assessed. Note that these values indicate the lowest accuracy of the numerical solution $y_c(x, e)$ provided by the EKE-SDG code; in fact, for each value of e there are hundred of values of M where the accuracy is much better than the indicated by $\varepsilon_{\max}$.

Figure 2.3 shows the decimal logarithm of the maximum absolute error versus the eccentricity e . The initial seed —0 iterations— provides an error which, in the worst case, ranges from 10^{-4} to 10^{-12} . Since the calculations associated with Figure 2.3 have been carried out in double precision, the residual obtained after one iteration practically saturates the capacity of the machine to provide the solution with 15 or 16 significant figures. It should be underlined that one additional iteration does not improve the accuracy of the numerical solution; Figure 2.3 shows that the lines corresponding to 1 and 2 iterations are, practically, the same.

Figure 2.4 is the same as Fig. 2.3 but with the calculations carried out in quadruple precision. It is clear that with one single iteration we obtain more than 20 significant figures except for values of e close to unity, where the number of significant digits is about 17 to 20. Two iterations the EKE-SDG code saturates the precision capacities of the machine, providing 34 significant figures.

Figure 2.5: Results with the EKE-SDG code (23 polynomials) using the modified Newton-Raphson (MNR) iteration scheme.

2.7.3 Improving the code

The number of polynomials used for the discretization of the interval $[0, \pi]$ of the mean anomaly $M = x$ does not affect the speed of the algorithm but it has an important impact on the accuracy. Instead of twelve, the code is modified to use more polynomials in order to obtain a better initial seed. However, the number of polynomial can not be increased arbitrarily because the intervals $[x_i, x_{i+1}]$ becomes too small and the polynomials becomes numerically unstable. Thus, when 24 polynomials are used the two first polynomials $p_1(x)$ and $p_2(x)$ associated with the two first intervals $[x_1, x_2]$ and $[x_2, x_3]$ become numerically unstable for some values of the eccentricity e . If these two polynomials are replaced by the first polynomial of the twelve family, the EKE-SDG code runs without problems and the accuracy of the algorithm is improved in a significant way.

Figure 2.5 is the same as Fig. 2.2 but now the calculations have been carried out using 23 polynomials. The figure does not change if quadruple precision is used instead of double precision. The speed of the code is not affected by the increase of polynomials. Obviously, by using double precision the code is faster than using quadruple precision.

Figure 2.6 is the same as Fig. 2.3 but now 23 polynomials have been used to provide the initial seed. The initial seed —0 iterations— provides an error which, in the worst case, ranges from 10^{-7} to 10^{-15} . Since the calculations associated with Fig. 2.6 have been carried out in double precision, the residual obtained after one iteration practically saturates the capacity of the machine to provide the solution with 15 or 16 significant figures. It should be underlined that one additional iteration does not improve the accuracy of the numerical solution; the figure shows that the lines corresponding to 1 and 2 iterations are, practically, the same. By comparing Figs. 2.3 and 2.6 a clear increase in the accuracy of the initial seed is detected when

Figure 2.6: The maximum total absolute error $\varepsilon_{\max}$ versus the eccentricity e , for different number of iterations. Calculations in double precision and using 23 polynomials.

Figure 2.7: The maximum total absolute error $\varepsilon_{\max}$ versus the eccentricity e , for different number of iterations. Calculations in quadruple precision and using 23 polynomials.

the 23-polynomials family is used. However, there are no differences in the accuracy of the results after one iteration.

Figure 2.7 is the same as Fig. 2.4 but now 23 polynomials have been used. Since the calculations associated with Fig. 2.7 have been carried out in quadruple precision, the residual obtained after one iteration practically saturates the capacity of the machine to provide the solution with 34 significant figures except in the region $0.75 < e < 1$ where the number of significant digits ranges from 27 to 34. One additional iteration improves the accuracy of the numerical solution for all values of e .

2.8 COMPARISON OF RESULTS

In the calculations, 23 approximating polynomials are used instead of 12. In fact, after checking out the behavior of the algorithm by using several number of polynomials we came to the conclusion that: 1) the differences are not important when using 12, 16, 18 or 23 polynomials, and 2) by using 23 the algorithm behavior is slightly smoother from a global point of view. All the calculations have been carried out in a workstation with Intel(R) Core(TM) i7 CPU 860 @ 2.80GHz 2.79GHz micro-processor in a Windows 7 64 bits operative system and with the same Intel C/C++ compiler.

Kepler's equation for the elliptical case has been solved with the method described previously —EKE-SDG— by using different Newton-Raphson algorithms. Moreover, the same analysis has been performed with typical solvers of Kepler's equation, namely:

- The modified Newton-Raphson (MNR).
- The Conway method.
- The classical Newton-Raphson method (CNR).
- The method described in Gooding and Odell [36].
- The method described in Nijenhuis [58].
- The method described in Fukushima [32].
- The method described in Mortari and Elife [57].

In each case, Kepler's equation has been solved as described in Sec. 2.7.

Figure 2.5 shows, in the plane (e, M) , the number of iterations associated with the EKE-SDG when using the modified Newton-Raphson (MNR) iteration scheme. Notice that in the 98.737 % of cases we reach the solution with the prescribed tolerance with only 1 iteration and in the 1.263 % no iteration is necessary (see Table 4). These results justify the option of working with quadruple precision which does not slow the calculations due to the very small number of iterations required. The CPU time invested in the nearly $4 \cdot 10^6$ times that we solved Kepler's equation was 41.012 seconds and the averaged number of iterations was 0.987. In the figure the zero iterations cases are localized in lines that, approximately, indicate the extremes of the intervals associated with the approximating polynomials.

Table 4 shows also the number of iterations associated with the EKE-SDG when using the Conway and classical Newton-Raphson (CNR) methods. Note that in a small number of cases located close to the singular corner —the 0.106 % and 0.138 % respectively— two iterations are required. The CPU time invested was 40.779 seconds with Conway and 40.591 seconds with CNR, while the averaged number of iterations was 0.988 and 0.989 respectively.

Figure 2.8: Number of significant figures, approximately, obtained with the code of Gooding and Odell [36] (upper), Nijenhuis [58] (medium) and Fukushima [32] (lower). On the left double precision; on the right quadruple precision.

Table 4: Number of iteration (percentage) obtained with the different Newton-Raphson algorithms checked in double (upper) and quadruple (lower) precision.

Scheme	$i = 0$	$i = 1$	$i = 2$	$i = 3$	$i \geq 4$
MNR	1.263	98.737	0	0	0
	0.161	99.839	0	0	0
Conway	1.263	98.631	0.106	0	0
	0.161	34.865	64.974	0	0
CNR	1.263	98.599	0.138	0	0
	0.161	35.588	66.251	0	0
Gooding	0.499	0	98.144	0	1.357
	0.499	0	98.144	0	1.357
Nijenhuis	0	0	0	50.024	49.976
	0	0	0	50.024	49.976
Fukushima	0.099	0	1	52.605	46.296
	0.099	0	1	52.605	46.296
Mortari	0.099	0.050	71.162	28.686	0.003
	0.099	0.049	0.00015	87.036	12.815

Moreover, Table 4 shows the number of iterations associated with the approaches described in Gooding and Odell [36], Nijenhuis [58], Fukushima [32] and Mortari and Elipe [57] when using double precision. Note that all of them require more than 3 iterations for some particular cases, reaching a maximum value of 18, 6, 9 and 4 iterations respectively. The CPU time invested was 41.2 seconds with Gooding's code, 42.136 seconds with Nijenhuis' code, 59.872 seconds with Fukushima's code and 44.173 seconds with Mortari's code. The averaged number of iterations was 2.205, 4.5, 3.495 and 2.284 respectively.

In quadruple precision (see Table 4) the EKE-SDG keeps again the smallest number of iterations. The methods described in Gooding and Odell [36], Nijenhuis [58] and Fukushima [32] do not experiment any change, keeping the same percentages as in double precision. The maximum number of iterations reached was the same as in double precision except for Gooding's code with 25 iterations and Mortari's code with 17 iterations. The CPU time invested was 83.190 seconds (MNR), 87.840 seconds (Conway) and 81.840 seconds (CNR) with EKE-SDG, 64.335 seconds with Gooding's code, 63.071 seconds with Nijenhuis' code, 260.926 seconds with Fukushima's code and 129.824 seconds with Mortari's code. When quadruple precision is applied, the SDG-code does not become the fastest algorithm, but provides the most precise solution with more than 30 significant digits in practically the whole plane (e, M). Indeed, Figures 2.8 and 2.9 show the number of significant digits obtained with the different codes when using the MNR method in double and quadruple precision. As it is shown, only the EKE-SDG exhibits an improvement of 20 significant figures for almost the complete plane when changing from double to quadruple precision. However, the results obtained with the remaining codes highlight the fact that they were developed to work in double precision but not in quadruple precision, specially in the case of Gooding, Nijenhuis and Fukushima codes. As a consequence, the results justify again the option to work with quadruple precision when using the EKE-SDG.

Figure 2.9: Number of significant figures, approximately, obtained with the Mortari and Elife [57] (upper) and EKE-SDG (lower). On the left double precision; on the right quadruple precision.

2.9 APPLICATIONS OF THE EKE-SDG CODE

In the introduction of the article by Wisdom and Holman [95] we can read: *Long-term integrations are playing an increasingly important role in investigations in dynamical astronomy. The reason is twofold. First, numerical exploration is an essential tool in the study of complex dynamical systems which can exhibit chaotic behavior, and there has been a growing realization of the importance of chaotic behavior in dynamical astronomy.*

In this sense, the code developed in this study is very useful in many situations, especially in the numerical integration of perturbed problems which appears in dynamical astronomy. In particular, when the integration requires to solve Kepler's equation at every step, that implies a large number of times. Not only in Keplerian perturbed problems, but in more involved situations like in N-body problems.

We must underline that this method works regardless of whether or not the eccentricity maintains a constant value. In some cases, the eccentricity is constant as for example in the traditional Encke's method (Encke and Schäfli [30]) where the reference orbit is a Keplerian one. Encke's method is used in some long-term integration codes, like the LONGSTOP project (Milani et al. [55]).

In other cases, the value of the eccentricity changes with time. This situation appears, for example, when the integration is performed with Deprit's regularized propagation method, based on a Hansen ideal frame (Deprit [27, 28]). This integration scheme is valid for elliptical orbits and requires solving Kepler's equation in every integration step.

Kepler's equation must be solved as well in symplectic integration codes as for example in the Wisdom-Holman method (Wisdom and Holman [95]), where the orbital elements exhibit small changes at every step.

To avoid the propagation of any kind of error involved in these numerical procedures it becomes necessary to solve Kepler's equation with the highest possible accuracy. The EKE-SDG algorithm, which can be used in double or quadruple precision, satisfies this condition assessing the convergence regardless of the numerical method selected.

Finally, it is important to highlight that in strongly perturbed problems it is quite easy to enter in the singular corner. Thanks to the asymptotic expansion, the EKE-SDG code is very reliable and accurate and avoids some chaotic behavior of numerical nature that appears in that region with other solvers.

THE HYPERBOLIC KEPLER'S EQUATION

The following chapter is focused on the resolution of the hyperbolic Kepler's equation continuing the work developed in the previous chapter. The procedure is extended to a new family of conics.

3.1 INTRODUCTION

The hyperbolic Kepler's equation (HKE) is

$$M = e \sinh(H) - H, \quad (3.1)$$

and it must be solved to obtain the value of the hyperbolic anomaly H in terms of the mean anomaly M and the eccentricity e of the orbit: $H = H(M, e)$ or $y = y(x, e)$. In this study we use the following notation: the hyperbolic anomaly is denoted by H or y without distinction. Equally, the mean anomaly is denoted by M or x .

Solving Kepler's equation—in its elliptic or hyperbolic version—is not a new problem. For approximately the last three centuries many scientists have been interested in this resolution and many papers related to this topic have been published in each decade from the mid-seventeenth century. By 1960 problems of celestial mechanics and orbit determination were being reformulated, and indeed new fields, including astronautics and astrodynamics, were being built to study them (Colwell [19]). The introduction of rocket thrusters for space flight presented possibilities of encountering trajectories with osculating orbits whose character changes significantly during the period of interest. In that sense, a low-acceleration escape trajectory might start in a nearly circular orbit and finish on a hyperbolic path. This aspect motivated researchers to employ a formulation free from the possibilities of indetermination, no matter what form the orbit takes, be it elliptic, parabolic, hyperbolic, or possibly a circular or linear limit (Pitkin [64]).

As a consequence, during that time there were efforts in several directions to reformulate the laws of orbital motion, and in particular Kepler's equation, into universal variables which incorporated all eccentricities and allow rapid, accurate computation of positions and velocities. Some of these approaches can be found in Battin [10], Goodyear [37], Stumpff [86], Herrick [43], Gooding [34], Sarnecki [77], Sharaf and Sharaf [79], Beust et al. [15], or Wisdom and Hernandez [94]. Even analytical solutions have been found using universal variables.

However, the interest for the hyperbolic version of Kepler's equation is founded on many different reasons. First of all, the solution of any highly transcendental equation is interesting, in itself, from the point of view of applied mathematics. In the field of numerical analysis some powerful techniques have been devoted to solving transcendental equations; for example, the homotopy continuation methods (Alhindi and Sharaf [2]).

Obviously HKE is most important in the fields of astronomy, celestial mechanics, and astrodynamics. In their paper, Wisdom and Holman [95] strongly underlined the important role that long-term integrations play in investigations on dynamical astronomy. A clear example is the LONGSTOP project (Milani et al. [55]), an international cooperative program with the purpose of investigating the stability of our solar system. In this kind of integrations, it is of paramount importance to take care of the errors involved (Milani and Nobili [54]). The imperfect convergence of an iterative algorithm—especially when we solve transcendental equations such as

Kepler's equation— is a well known source of error. Minimizing such errors is always an interesting issue that helps to achieve the main objective of the research. Kepler's equation must be solved in symplectic integration codes as for example in the Wisdom-Holman method (Wisdom and Holman [95]), where the orbital elements experience small changes at every step. It can be used for long-term gravitational simulations (Rein and Tamayo [71]), or for the collisional gravitational N-body problem (Dehnen and Hernandez [26] and Hernandez and Bertschinger [42]). Beyond this, the correct determination of the position and velocity in perturbed cometary orbits, which depends on the fast and accurate solution of the HKE, became an important issue. The introduction of computing machines gave impetus to the generation of algorithms that help in tasks of this kind from the very beginning. One example can be found in Davidson [22, 23] and Smiley [82] where a method is introduced in order to calculate the true anomaly and the time of perihelion passage for hyperbolic orbits. The authors gave an approximate solution of the HKE without explicitly mentioning it. Another example of algorithms designed to be used in a computing machine can be found in Sitarski [81].

Some comets are originated outside of the solar system. Such comets enter the solar system coming from the interstellar space and may exit the solar system following hyperbolic trajectories. Recently the object 1I/2017 U1 reached its perihelion (0.2483 ua) following one of these hyperbolic trajectories. Frequently, the HKE is used to describe the hyperbolic anomaly of the comet. The determination of the radial distance—or the Cartesian coordinates— of such a comet requires somewhat accurate knowledge of the hyperbolic anomaly. Faintich [31] shows how a near-parabolic comet in the Oort cloud can be perturbed by a passing star. The description of the perturbed comet orbit requires us to consider the hyperbolic motion of both stars around its common center of mass and the time evolution of the whole system leads to a solution of the HKE.

There are, however, other fields in which the HKE plays a significant role. For example, Rauh and Parisi [70] faced the problem of deriving macroscopic properties from the Hamiltonian of the hydrogen atom considering possible hyperbolic orbits. By using the Kustaanheimo-Stiefel (KS) transformation, a pseudo-time σ is introduced into the problem. However, the relation between σ and the real time t is given by a particular version of the HKE. Thus in some quantum mechanics problems the HKE appears naturally.

Stellar dynamics is another field where the HKE also appears (see, for example, Kurth [49]). In Yabushita [96] an investigation is exposed on the possibility of a capture process in which a planet of negligible mass, initially in Keplerian orbit about a star, is perturbed by a passing star with hyperbolic velocity at great distance and becomes a planet of the star escaping to infinity with it. In order to integrate the governing equations, the HKE needs to be solved for each integration step. Since they did not have a fast and reliable solver the authors opted to integrate an additional differential equation.

Thus in many integration processes the solution of Kepler's equation—elliptic or hyperbolic— could jeopardize the procedure as indicated in Balaraman and Vrinceanu [7]. It follows, from here, the need for an efficient, fast and accurate calculation method to solve the HKE.

The HKE has been studied by several authors seeking to develop optimal methods. In this regard, the main aspects to take into account are the accuracy and the computational performance, which are related to the method for determining a starting estimate for H . The next approaches summarize some of the current methods used to solve the HKE as in Eq. (3.2):

- The Gooding and Odell [36] approach: the starting value is based on Lagrange's theorem, where the equation is rewritten as a function of $S = \sinh H$

instead of H . First, Halley's method is applied to obtain the corrector of S as well as of the function and its derivative. With these values, the Newton-Raphson method is applied to determine the solution of the equation. It is an iterative method which requires several transcendental function evaluations.

- The Fukushima [33] approach: several intervals of $L = M/e$ are defined to select where the solution should be determined. The different methods and approximations chosen for each interval depend on the expected magnitude of the solution. First, four cases in which the solution is large are selected. For these cases an asymptotic form of the main equation is considered and an approximate solution is found. Next, four other cases are selected for which the solution is small. In these latter cases an iterative procedure is applied to solve the approximate forms of Kepler's equation. In the remaining case, for which the solution interval becomes finite, a discretized Newton method is applied as well as a Newton method in which the functions are evaluated by Taylor series expansions. In that case, the starting value is the minimum of some upper bound of the solution prepared by using the Newton correction formula. This approach, depending on the case, uses or does not use an iterative method that requires several transcendental function evaluations.
- The Avendano, Martín-Molina, and Ortigas-Galindo [5] and Avendano, Martín-Molina, and Ortigas-Galindo [6] approach: the equation is rewritten as a function of $S = \sinh(H)$ instead of H , such that the starting value of S is a piecewise-defined function involving several linear expressions and one with cubic and square roots. Once the seed is estimated, a Newton's method is applied, by using Smale's α -theory to decide whether the starting value gives the claimed convergence rate. It is an iterative method which requires several transcendental function evaluations.

The goal of the present study is the determination of a suitable seed to initialize the numerical method for solving the hyperbolic Kepler's equation. For that we consider the already tested and optimized Newton-Raphson method and extend the method described in Raposo-Pulido and Peláez [68] for the elliptical Kepler's equation to the HKE. As in the aforementioned work, the Maple symbolic manipulator is chosen to obtain the code in the C programming language. The code developed is called Hyperbolic Kepler Equation-Space Dynamics Group (HKE-SDG) (Raposo-Pulido and Peláez [69]). As a result we obtain a simple way to provide fast and accurate solutions.

3.2 THE EQUATION

Kepler's equation for the hyperbolic case

$$x = e \sinh(y) - y, \quad x \in]-\infty, \infty[, \quad y \in]-\infty, \infty[, \quad e > 1, \quad (3.2)$$

determines a nonlinear function $y = y(x, e)$ where y is the hyperbolic anomaly related to the true anomaly θ by means of

$$\cos \theta = \frac{e - \cosh(y)}{e \cosh(y) - 1}, \quad \cosh(y) = \frac{e + \cos(\theta)}{e \cos(\theta) + 1}, \quad (3.3a)$$

$$\sin(\theta) = \frac{\sqrt{e^2 - 1} \sinh(y)}{e \cosh(y) - 1}, \quad \sinh(y) = \frac{\sqrt{e^2 - 1} \sin(\theta)}{e \cos(\theta) + 1}, \quad (3.3b)$$

and x is equivalent to the mean anomaly in the elliptic motion:

$$x = \sqrt{\frac{\mu}{a^3}}(t - \tau). \quad (3.4)$$

The eccentricity is larger than one. As a consequence x and y have the same sign, that is, $y > 0 \Leftrightarrow x > 0$. Since the right-hand side of Eq. (3.2) is an odd function, for negative values of x the following change of variables leads to

$$x = -\mathfrak{x}, \quad y = -\eta \quad \Rightarrow \quad \mathfrak{x} = e \sinh(\eta) - \eta, \quad (3.5)$$

where $\mathfrak{x}$ and η are positive. Therefore, we focus the analysis in the resolution of Eq. (3.2) for positive values: $(x, y) \in [0, \infty[\times [0, \infty[$.

The solution of Kepler's equation can be seen as a root finder problem for the function

$$f(y) \equiv e \sinh(y) - y - x = 0, \quad (3.6)$$

for given values of e and x . In such a case, Kepler's equation univocally determines a bijective function $y = y(x, e)$ which is defined on $\mathbb{R}_+$. This property can be deduced from Bolzano's theorem and the fact that function $f(y)$ in Eq. (3.6) is monotonically increasing. Indeed, $f(y)$ is a continuous function on the closed interval $[y_1, y_2] = [0, \frac{x}{e-1}]$ and it takes values of the opposite sign at the extremes

$$f(0) = -x < 0 \quad (3.7)$$

$$\begin{aligned} f\left(\frac{x}{e-1}\right) &= e \sinh\left(\frac{x}{e-1}\right) - \frac{x}{e-1} - x = \\ &= e \left(\frac{1}{3!} \left(\frac{x}{e-1}\right)^3 + \frac{1}{5!} \left(\frac{x}{e-1}\right)^5 + \dots \right) > 0. \end{aligned} \quad (3.8)$$

Applying Bolzano's theorem to the function $f(y)$, it exists an intermediate value $y^* \in [y_1, y_2]$ which verifies $f(y^*) = 0$. Therefore $f(y)$ vanishes at least once in the stated interval. However, the derivative of $f(y)$, $f'(y)$, is always positive

$$\frac{df}{dy} = e \cosh(y) - 1 \geq e - 1 > 0, \quad (3.9)$$

that is, $\forall y_2 > y_1$ with $y_1, y_2 \in [0, \infty[$, $f(y_2) \geq f(y_1)$ and $f(y)$ is strictly increasing. Consequently, $f(y)$ is zero for only one value of $y \in [y_1, y_2]$. In summary, for given x , $\exists ! y$ such that $f(y) = 0$.

3.3 NEWTON-RAPHSON ALGORITHMS

Following the procedure developed and validated by Raposo-Pulido and Peláez [68], we use Newton-Raphson algorithms to solve the HKE. The root sought is the solution to Eq. (3.6). To find this root we wish to establish a successive approximations method starting from a seed y_0 . If y_n is one of the terms of the sequence, the next term will be $y_{n+1} = y_n + \Delta y_n$, where Δy_n is given by applying a root-finding method of Laguerre (1834-1886) to the solution of the HKE (Conway [20])

$$\Delta y_n = \frac{-mf(y_n)}{f'(y_n) \pm \sqrt{|(m-1)[(m-1)f'^2(y_n) - mf(y_n)f''(y_n)]|}}, \quad (3.10)$$

in which the sign (+) is selected when $f'(y_n)$ is positive; if $f'(y_n) < 0$ the sign (−) must be chosen. The introduction of the absolute value in Eq. (3.10) is useful to avoid failures when the square root is a complex number. It does not affect the convergence of the algorithm which is practically independent from m , the degree of the polynomial.

Three different algorithms are considered in this study: the classical Newton-Raphson (CNR) algorithm ($m = 1$); the modified Newton-Raphson (MNR) algorithm ($m = 2$); and the Conway algorithm ($m = 5$).

All of the aforementioned three algorithms use an initial seed to initialize the convergent sequence. However, its behavior, related to the reliability, is going to mark the difference. While the MNR method always converges to the real solution regardless of the quality of the seed, the CNR is less stable. Therefore CNR method is more correlated with the initial value than the Conway or the MNR methods. A way to escape from this disadvantage and assure the reliability of the method is by using a very good seed as initial point. In that case the CNR and the MNR algorithms converge quickly to the real solution, as it is shown in Raposo-Pulido and Peláez [68] for the elliptic Kepler's equation.

3.4 THE SEED VALUE

As we briefly explained in the introduction, the core of this work is to generate a very good initial seed in all possible cases. If this is satisfied, then we can benefit from the convergence properties of Newton-Raphson algorithms (CNR, MNR, or Conway) to determine the right solution to Kepler's equation. This approach has been used with success in the elliptic case (see Raposo-Pulido and Peláez [68]). In this work, we show that in the hyperbolic case too (see Raposo-Pulido and Peláez [69]), improving the initial seed to feed a Newton-Raphson algorithm is much more cost-effective than searching for new algorithms better than the classic Newton-Raphson algorithms.

We must take into account that the accuracy of Newton-Raphson algorithms is jeopardized when $x = M \ll 1$ and $e - 1 \ll 1$; This region of the plane (e, M) is called the singular corner. In such a region y is also small, and the expansion of the equation for small values of y provides

$$y \approx \frac{x}{e-1}, \quad (3.11)$$

that is, the solution is ill-determined from a numerical point of view since it is the ratio of two infinitesimals.

In order to determine the optimal seed, we need to introduce two different scenarios, depending on whether we are inside or outside of the singular corner. From a practical point of view we define the singular corner as the region in which $x = M < 0.25$ and $e < 1.5$. These values have been chosen arbitrarily: firstly inside the singular corner ($x = M < 0.25$ and $e < 1.5$) three regions (inner, intermediate and outer) are defined in terms of the magnitude of x and y . In these regions we estimate the starting value through an asymptotic expansion. And secondly outside the singular corner ($x = M \geq 0.25$ or $e \geq 1.5$) two regions are defined in terms of the infinite y -domain $[0, \infty[$ to estimate the starting value. In the finite interval $[0, 5[$ the seed is determined through a fifth degree polynomial, while in the infinite interval $[5, \infty[$ the seed is estimated through an asymptotic approximation.

As we see below, the polynomial approximation works very well in most cases, especially when $x = M$ is close to the boundaries of the interval where the polynomial is defined. In such a case the seed practically leads to the solution of Kepler's equation. However, the singular behavior of Kepler's equation makes the approximation of $y(x)$ through this procedure ineffective and a special analysis should be implemented.

3.4.1 Analysis in the singular corner

To describe the solution inside the singular corner we introduce the value $\epsilon = e - 1$ assuming that $\epsilon \ll 1$:

$$x = (1 + \epsilon) \sinh(y) - y = (\sinh(y) - y) + \epsilon \sinh(y). \quad (3.12)$$

The objective is to obtain an approximation to the exact solution $y_v(x)$ of Eq. (3.12) with sufficient accuracy to be the seed used to start the Newton-Raphson process. To this end, an asymptotic expansion in powers of the small parameter ϵ is introduced.

However we must distinguish between several regimes inside the singular corner:

- The inner region, where $x \approx \mathcal{O}(\epsilon^2)$ and $y \approx \mathcal{O}(\epsilon)$. In this inner region the most important term on the right-hand side of Eq. (3.12) is $\epsilon \sinh(y)$.
- The intermediate region, where $x \approx \mathcal{O}(\epsilon^{\frac{3}{2}})$ and $y \approx \mathcal{O}(\epsilon^{\frac{1}{2}})$. In this intermediate region both terms on the right-hand side of Eq. (3.12) are of the same order: $\sinh(y) - y \approx \epsilon \sinh(y)$.
- The outer region, where $x \approx \mathcal{O}(\epsilon^{\frac{3}{4}})$ and $y \approx \mathcal{O}(\epsilon^{\frac{1}{4}})$. In this outer region the most important term on the right-hand side of Eq. (3.12) is $\sinh(y) - y$.

In each of these regions we obtain an asymptotic expansion of the exact solution $y_v(x)$ of Eq. (3.12) in terms of powers of the small parameter ϵ ; the change of variable defined in Eq. (3.42) provides a very good seed S_0 to feed the Newton-Raphson process.

3.4.2 Inner region

In the inner region we use new variables given by the following relations:

$$x = \epsilon^2 \xi, \quad y = \epsilon \eta, \quad \text{with} \quad \xi, \eta \approx \mathcal{O}(1), \quad (3.13)$$

where (ξ, η) are both positive quantities. In Eq. (3.12) we introduce the expansion

$$\xi = \eta + a_1 \epsilon + a_2 \epsilon^2 + a_3 \epsilon^3 + \dots \quad (3.14)$$

The coefficients a_i , $i = 1, 2, \dots$ can be obtained expanding appropriately the right-hand side of Eq. (3.12). The solution is

$$\xi = \eta + \frac{1}{6} \eta^3 \epsilon + \frac{1}{6} \eta^3 \epsilon^2 + \frac{1}{120} \eta^5 \epsilon^3 + \frac{1}{120} \eta^5 \epsilon^4 + \dots \quad (3.15)$$

We must invert this solution to obtain η as a function of ξ ; to this end we introduce the expansion

$$\eta = \xi + b_1 \epsilon + b_2 \epsilon^2 + b_3 \epsilon^3 + \dots \quad (3.16)$$

To calculate the coefficients b_i , $i = 1, 2, \dots$, which are functions of ξ , we introduce this expansion in Eq. (3.15) and require that the different coefficients of each order match. The result is

$$\eta = \xi - \frac{1}{6} \xi^3 \epsilon + \frac{1}{12} \xi^3 (\xi^2 - 2) \epsilon^2 - \frac{1}{360} \xi^5 (20\xi^2 - 57) \epsilon^3 + \dots \quad (3.17)$$

Figure 3.1 shows the behavior of the asymptotic solution (Eqs. (3.13)-(3.17)) in the inner region. The agreement with the exact solution is excellent for very small values of x .

3.4.3 Outer region

In the outer region we introduce the following change of variables:

$$x = \epsilon^{\frac{3}{4}} \nu, \quad y = \epsilon^{\frac{1}{4}} \mu, \quad \text{with} \quad \nu, \mu \approx \mathcal{O}(1), \quad (3.18)$$

Figure 3.1: Exact solution to Kepler's equation and the asymptotic solution —up to sixth order— obtained in the inner region of the singular corner. The agreement is excellent for very small values of ϵ . Here four values of ϵ have been considered: $\epsilon = 0.001, 0.101, 0.201, 0.301$.

where (ν, μ) are both positive quantities. We look for a solution to Eq. (3.12) given by

$$\nu = a_0 + a_1 \epsilon^{\frac{1}{2}} + a_2 \epsilon + a_3 \epsilon^{\frac{3}{2}} + \dots, \quad (3.19)$$

where the coefficients a_i , $i = 1, 2, \dots$ can be obtained by appropriately expanding the right-hand side of Eq. (3.12). The solution is

$$\nu = \frac{1}{6} \mu^3 + \frac{1}{120} \mu (\mu^4 + 120) \epsilon^{\frac{1}{2}} + \frac{1}{5040} \mu^3 (\mu^4 + 840) \epsilon + \dots \quad (3.20)$$

Let $\mu_0 > 0$ be the real and positive solution of the cubic equation

$$\frac{1}{6} \mu^3 - \nu = 0, \quad (3.21)$$

this root always exists and is univocally defined because ν is always positive. It takes the value

$$\mu_0 = (6\nu)^{\frac{1}{3}}. \quad (3.22)$$

Once the value of μ_0 is known it is possible to invert the solution given in Eq. (3.20) to obtain μ as a function of ν ; to do that we introduce the expansion

$$\mu = \mu_0 + b_1 \epsilon^{\frac{1}{2}} + b_2 \epsilon + b_3 \epsilon^{\frac{3}{2}} + \dots, \quad (3.23)$$

where the coefficients b_i , $i = 1, 2, \dots$ are functions of μ_0 that must be calculated. Introducing this expansion into Eq. (3.20) and enforcing that the different coefficients of each order match it is possible to obtain the values of b_i , $i = 1, 2, \dots$. The result is

$$\begin{aligned} \mu = \mu_0 - \frac{\mu_0^4 + 120}{60 \mu_0} \epsilon^{\frac{1}{2}} + \frac{\mu_0 (\mu_0^4 - 280)}{1400} \epsilon \\ - \frac{\mu_0^{12} - 162 \mu_0^8 - 25200 \mu_0^4 - 67200}{25200 \mu_0^5} \epsilon^{\frac{3}{2}} + \dots \end{aligned} \quad (3.24)$$

3.4.4 Intermediate region

In the intermediate region we introduce the following change of variables:

$$x = \epsilon^{\frac{3}{2}} \chi, \quad y = \epsilon^{\frac{1}{2}} \sigma, \quad \text{with} \quad \chi, \sigma \approx \mathcal{O}(1), \quad (3.25)$$

where (χ, σ) are both positive quantities. We look for a solution to Eq. (3.12) given by

$$\chi = a_0 + a_1 \epsilon + a_2 \epsilon^2 + a_3 \epsilon^3 + \dots, \quad (3.26)$$

where the coefficients $a_i, i = 1, 2, \dots$ can be obtained expanding appropriately the right-hand side of Eq. (3.12). The solution is

$$\chi = \frac{1}{6} \sigma^3 + \sigma + \frac{1}{120} \sigma^3 (\sigma^2 + 20) \epsilon + \frac{1}{5040} \sigma^5 (\sigma^2 + 42) \epsilon^2 + \dots \quad (3.27)$$

The next step is to invert this solution to obtain σ as a function of χ . Let $\sigma_0 > 0$ be the real and positive solution of the cubic equation

$$f(\sigma) = \frac{1}{6} \sigma^3 + \sigma - \chi = 0, \quad (3.28)$$

this root always exists; in fact, the function $f(\sigma)$ verifies

$$f(0) = -\chi < 0 \quad (3.29)$$

$$f(\chi) = \frac{\chi^3}{6} > 0. \quad (3.30)$$

Therefore in the interval $[0, \chi]$ there is a root of $f(\sigma) = 0$ at least. But in fact there is only one single root, because the derivative

$$f'(\sigma) = \frac{\sigma^2}{2} + 1 > 0, \quad (3.31)$$

is always positive within that interval. That root is given by

$$\sigma_0 = T - \frac{2}{T}, \quad (3.32)$$

where $T = (\Lambda + 3\chi)^{\frac{1}{3}}$ and $\Lambda = \sqrt{8 + 9\chi^2}$. It should be noted that the expression in Eq. (3.32) can exhibit a rounding-error defect when $T \approx \sqrt{2}$ ($\sigma_0 \ll 1$). In order to minimize this error, we develop an alternative procedure to provide σ_0 . We rewrite the cubic equation (3.28) as

$$\sigma \left(\frac{\sigma^2}{6} + 1 \right) - \chi = 0 \Leftrightarrow \sigma = \frac{6\chi}{\sigma^2 + 6}, \quad (3.33)$$

such that σ is replaced by Eq. (3.32) on the right-hand side of the equation, yielding

$$\sigma_0 = \frac{6\chi}{2 + T^2 + \frac{4}{T^2}}, \quad (3.34)$$

which provides a smaller rounding-error when $\chi \ll 1$, that is, when $\Lambda \approx 2\sqrt{2}$.

Once the value of σ_0 is known, it is possible to invert the solution of Eq. (3.27) to obtain σ as a function of χ ; to do that we introduce the expansion

$$\sigma = \sigma_0 + b_1 \epsilon + b_2 \epsilon^2 + b_3 \epsilon^3 + \dots, \quad (3.35)$$

where the coefficients $b_i, i = 1, 2, \dots$ are functions of σ_0 that must be calculated. Introducing this expansion into Eq. (3.27) and enforcing that the different coefficients of each order match it is possible to obtain the values of $b_i, i = 1, 2, \dots$. The result is

$$\sigma = \sigma_0 - \frac{\sigma_0^3 (\sigma_0^2 + 20)}{60(\sigma_0^2 + 2)} \epsilon + \frac{\sigma_0^5 (\sigma_0^6 + 25\sigma_0^4 + 340\sigma_0^2 + 840)}{1400(\sigma_0^2 + 2)^3} \epsilon^2 + \dots \quad (3.36)$$

Figure 3.2: Exact solution to Kepler's equation and the asymptotic solution—up to the fourth order—obtained in the intermediate region of the singular corner. The agreement is excellent for whole range. Here four values of ϵ have been considered: $\epsilon = 0.001, 0.101, 0.201, 0.301$.

3.4.4.1 Matching with the inner region

A detailed insight of the solution in this region shows that when (χ, σ) take small values the asymptotic solution tends to the inner solution obtained previously. In fact, when

$$\chi \approx \epsilon^{\frac{1}{2}} \xi, \quad \sigma \approx \epsilon^{\frac{1}{2}} \eta, \quad (3.37)$$

the solution enters in the domain of the inner solution. For those values the parameter σ_0 given by Eq. (3.34) takes the value

$$\sigma_0 \approx \epsilon^{\frac{1}{2}} \left(\xi - \frac{1}{6} \xi^3 \epsilon + \frac{1}{12} \xi^5 \epsilon^2 - \frac{1}{18} \xi^7 \epsilon^3 + \dots \right). \quad (3.38)$$

If this value is introduced into Eq. (3.36) this yields

$$\sigma = \epsilon^{\frac{1}{2}} \eta = \epsilon^{\frac{1}{2}} \left(\xi - \frac{1}{6} \xi^3 \epsilon + \frac{1}{12} \xi^3 (\xi^2 - 2) \epsilon^2 + \dots \right), \quad (3.39)$$

and this expression coincides with Eq. (3.17) corresponding to the inner solution. This is the reason why the asymptotic solution given by Eq. (3.36) also provides the solution in the inner region where x is in the domain of the inner solution.

3.4.4.2 Matching with the outer region

When (χ, σ) take large values the asymptotic solution tends to the outer solution obtained previously. In fact, when

$$\chi \approx \frac{\nu}{\epsilon^{\frac{3}{4}}}, \quad \sigma \approx \frac{\mu}{\epsilon^{\frac{1}{4}}}, \quad (3.40)$$

the solution enters in the domain of the outer solution. For those values the parameter σ given by Eq. (3.36) takes the value

$$\sigma = \frac{\mu}{\epsilon^{\frac{1}{4}}} = \frac{1}{\epsilon^{\frac{1}{4}}} \left\{ \mu_0 - \frac{\mu_0^4 + 120}{60 \mu_0} \epsilon^{\frac{1}{2}} + \frac{\mu_0 (\mu_0^4 - 280)}{1400} \epsilon - \frac{\mu_0^{12} - 162 \mu_0^8 - 25200 \mu_0^4 - 67200}{25200 \mu_0^5} \epsilon^{\frac{3}{2}} + \dots \right\}, \quad (3.41)$$

and this expression coincides with Eq. (3.24) corresponding to the outer solution. This is the reason why the asymptotic solution given by Eq. (3.36) also provides the solution in the outer region where x is in the domain of the outer solution.

3.4.5 Summary of the singular corner

The analysis carried out above shows that the asymptotic expansion of Eq. (3.36) corresponding to the intermediate region also describes the solution in the inner and outer regions of the singular corner. Figure 3.2 shows the exact solution of the HKE and the asymptotic solution —up to the fourth order— obtained in the intermediate region of the singular corner. The agreement is remarkable in the whole range of values of $x = M$. These numerical results confirm the theoretical analysis. The main consequence is that, from a practical point of view, the asymptotic expansion of Eq. (3.36) can be used in the whole domain of the singular corner.

Therefore, only one asymptotic expansion is considered in HKE-SDG, i.e. that corresponding to the intermediate region. This approximate solution permits us to generate a very good seed to feed the Newton-Raphson process for values of (e, M) inside the singular corner.

3.5 ANALYSIS OUTSIDE THE SINGULAR CORNER

We begin by performing the following change of variable

$$S = \sinh(y) \quad \Leftrightarrow \quad y = \ln|S + \sqrt{1 + S^2}|, \quad (3.42)$$

which transforms the hyperbolic Kepler's equation into the relation

$$x = eS - \ln|S + \sqrt{1 + S^2}|, \quad (3.43)$$

with the variable S defined in the interval $[0, \infty[$. Due to the infinite domains of both variables (x, y) , the range of y is divided in two intervals: $[0, 5[$ and $[5, \infty[$ such that for each interval a different treatment will be applied. The limit value of five has been chosen arbitrarily; we note that the corresponding value of $S \approx 74.2$ is large compared with unity. In the interval $[0, 5[$ the seed is generated with polynomials. In the interval $[5, \infty[$ the seed is generated with the help of an asymptotic solution of Eq. (3.43) for large values of $S \gg 1$ which correspond, basically, to large values of M .

3.5.1 Polynomial approximation

If $y \in [0, 5[$, the interval is divided into 25 evenly spaced intervals; the lower bounds of such intervals are given by

$$y_i = \frac{i-1}{5}, \quad i = 1, \dots, 26, \quad (3.44)$$

where the corresponding S_i are obtained from Eq. (3.42) and $M_i \equiv x_i$ are obtained from Eq. (3.43) (see Table 5). This way, we define twenty five intervals $[x_i, x_{i+1}]$, $i = 1, \dots, 25$. In each of these intervals we introduce a fifth degree polynomial $p_i(x, e)$, $i = 1, \dots, 25$ to interpolate the hyperbolic sine of the hyperbolic anomaly, that is, S .

The coefficients of the polynomials considered in the method depend on the eccentricity. In order to simplify the reading of the equation, we make an abuse of notation by writing the coefficients only as a function of i

$$p_i(x) = a_0^{(i)} + a_1^{(i)}x + a_2^{(i)}x^2 + a_3^{(i)}x^3 + a_4^{(i)}x^4 + a_5^{(i)}x^5. \quad (3.45)$$

Table 5: Important values for the polynomial approach $-p_i(x)-$ (columns 1-5) and for the polynomial approach $-q_j(x)-$ (columns 6-10). In real calculations the ends on these intervals are determined with more significant figures.

Intervals defined in the solution - $p_i(x)$					Intervals defined in the solution - $q_j(x)$				
i	S_i	S_{i+1}	x_i	x_{i+1}	j	S_j	S_{j+1}	x_j	x_{j+1}
1	0	0.201	0	$-0.200 + 0.201 e$	1	0.100	0.305	$0.100 e - 0.100$	$0.305 e - 0.300$
2	0.201	0.411	$-0.200 + 0.201 e$	$-0.400 + 0.411 e$	2	0.305	0.521	$0.305 e - 0.300$	$0.521 e - 0.500$
3	0.411	0.637	$-0.400 + 0.411 e$	$-0.600 + 0.637 e$	3	0.521	0.759	$0.521 e - 0.500$	$0.759 e - 0.700$
4	0.637	0.888	$-0.600 + 0.637 e$	$-0.800 + 0.888 e$	4	0.759	1.027	$0.759 e - 0.700$	$1.027 e - 0.900$
5	0.888	1.175	$-0.800 + 0.888 e$	$-1.000 + 1.175 e$	5	1.027	1.336	$1.027 e - 0.900$	$1.336 e - 1.100$
6	1.175	1.509	$-1.000 + 1.175 e$	$-1.200 + 1.509 e$	6	1.336	1.698	$1.336 e - 1.100$	$1.698 e - 1.300$
7	1.509	1.904	$-1.200 + 1.509 e$	$-1.400 + 1.904 e$	7	1.698	2.129	$1.698 e - 1.300$	$2.129 e - 1.500$
8	1.904	2.376	$-1.400 + 1.904 e$	$-1.600 + 2.376 e$	8	2.129	2.646	$2.129 e - 1.500$	$2.646 e - 1.700$
9	2.376	2.942	$-1.600 + 2.376 e$	$-1.800 + 2.942 e$	9	2.646	3.268	$2.646 e - 1.700$	$3.268 e - 1.900$
10	2.942	3.627	$-1.800 + 2.942 e$	$-2.000 + 3.627 e$	10	3.268	4.022	$3.268 e - 1.900$	$4.022 e - 2.100$
11	3.627	4.457	$-2.000 + 3.627 e$	$-2.200 + 4.457 e$	11	4.022	4.937	$4.022 e - 2.100$	$4.937 e - 2.300$
12	4.457	5.467	$-2.200 + 4.457 e$	$-2.400 + 5.467 e$	12	4.937	6.050	$4.937 e - 2.300$	$6.050 e - 2.500$
13	5.467	6.695	$-2.400 + 5.467 e$	$-2.600 + 6.695 e$	13	6.050	7.406	$6.050 e - 2.500$	$7.406 e - 2.700$
14	6.695	8.192	$-2.600 + 6.695 e$	$-2.800 + 8.192 e$	14	7.406	9.060	$7.406 e - 2.700$	$9.060 e - 2.900$
15	8.192	10.018	$-2.800 + 8.192 e$	$-3.000 + 10.018 e$	15	9.060	11.077	$9.060 e - 2.900$	$11.076 e - 3.100$
16	10.018	12.246	$-3.000 + 10.018 e$	$-3.200 + 12.246 e$	16	11.077	13.538	$11.076 e - 3.100$	$13.538 e - 3.300$
17	12.246	14.965	$-3.200 + 12.246 e$	$-3.400 + 14.965 e$	17	13.538	16.543	$13.538 e - 3.300$	$16.543 e - 3.500$
18	14.965	18.285	$-3.400 + 14.965 e$	$-3.600 + 18.285 e$	18	16.543	20.211	$16.543 e - 3.500$	$20.211 e - 3.700$
19	18.285	22.339	$-3.600 + 18.285 e$	$-3.800 + 22.339 e$	19	20.211	24.691	$20.211 e - 3.700$	$24.691 e - 3.900$
20	22.339	27.290	$-3.800 + 22.339 e$	$-4.000 + 27.290 e$	20	24.691	30.162	$24.691 e - 3.900$	$30.162 e - 4.100$
21	27.290	33.336	$-4.000 + 27.290 e$	$-4.200 + 33.336 e$	21	30.162	36.843	$30.162 e - 4.100$	$36.843 e - 4.300$
22	33.336	40.719	$-4.200 + 33.336 e$	$-4.400 + 40.719 e$	22	36.843	45.003	$36.843 e - 4.300$	$45.003 e - 4.500$
23	40.719	49.737	$-4.400 + 40.719 e$	$-4.600 + 49.737 e$	23	45.003	54.969	$45.003 e - 4.500$	$54.969 e - 4.700$
24	49.737	60.751	$-4.600 + 49.737 e$	$-4.800 + 60.751 e$	24	54.969	67.141	$54.969 e - 4.700$	$67.141 e - 4.900$
25	60.751	74.203	$-4.800 + 60.751 e$	$-5.00 + 74.203 e$					
26	74.203	∞	$-5.00 + 74.203 e$	∞					

The polynomial $p_i(x)$ is only valid inside the interval $[x_i, x_{i+1}]$, $i = 1, \dots, 25$, where the function is approximated by the polynomial: $S(x) \approx p_i(x)$. To do that we follow the procedure developed in Raposo-Pulido and Peláez [68] by imposing six boundary conditions to determine the six coefficients of $p_i(x)$

$$p_i(x_i) = S(x_i), \quad p_i(x_{i+1}) = S(x_{i+1}), \quad (3.46a)$$

$$p_i'(x_i) = S'(x_i), \quad p_i'(x_{i+1}) = S'(x_{i+1}), \quad (3.46b)$$

$$p_i''(x_i) = S''(x_i), \quad p_i''(x_{i+1}) = S''(x_{i+1}), \quad (3.46c)$$

where two first derivatives of S are given by:

$$S' = \frac{dS}{dx} = 1 - \frac{1}{e\sqrt{S^2 + 1}} \quad (3.47)$$

$$S'' = \frac{d^2S}{dx^2} = \frac{S}{e\sqrt{(S^2 + 1)^3}}, \quad (3.48)$$

and they can be easily calculated at the boundaries of each interval. These conditions assure that the function $S(x)$ and the polynomial $p_i(x)$ share the same values and the same first and second derivatives at the ends of the interval $[x_i, x_{i+1}]$.

Modern symbolic manipulators such as Maple or Mathematica allow us to obtain the polynomials $p_i(x)$, $i = 1, \dots, 25$ easily once and for all. Moreover, they provide the FORTRAN or C code of the polynomials that can be appropriately stored in a module to be used later. In HKE-SDG we use the version 18.02 of Maple (Raposo-Pulido and Peláez [68]) to perform this task.

3.5.2 Asymptotic approximation

We note that the HKE defined in Eq. (3.43) can be expressed as follows:

$$S = f(S, x, e), \quad f(S, x, e) = \frac{1}{e} \ln(S + \sqrt{1 + S^2}) + \frac{x}{e}, \quad (3.49)$$

where f is a function of the variable S ; the values of (x, e) appear as parameters. The derivative of f turns out to be

$$\frac{df}{dS} = \frac{1}{e\sqrt{1 + S^2}}. \quad (3.50)$$

If $y = H \in [5, \infty[$, the corresponding interval for S is given by $[74.203, \infty[$, where the value of x could be very large. In this range x and S tend to infinite, in a first approximation.

The derivative given by Eq. (3.50) is, obviously, lower than unity, when $y = H \in [5, \infty[$ because $S \gg 1$ and $e > 1$. As a consequence the function $f(S, x, e)$ given by Eq. (3.49) is a contraction mapping. The Banach fixed-point theorem can be applied and $f(S, x, e)$ admits a unique fixed-point S^* which verifies $S^* = f(S^*, x, e)$. This fixed-point is the root we are looking for.

The mathematical analysis shows that the following sequence S_n defined by

$$S_{n+1} = f(S_n, x, e), \quad (3.51)$$

which starts in the value $S_0 = \frac{x}{e}$, is convergent and it tends to the limit S^* , that is, to the unique fixed-point.

The speed of convergence of this sequence is larger than usual because of the large values of S in the interval $y \in [5, \infty[$. However, a detailed analysis shows that there are faster algorithms which are also reliable. In fact, the second term of the right-hand side of Eq. (3.43) becomes, with respect to the first term, very

small. As a consequence we can consider the logarithm as a small parameter $\epsilon = \ln|S + \sqrt{1 + S^2}|/x$, obtaining S by applying an asymptotic expansion in power of ϵ . Equation (3.43) takes now the expression

$$x = e S - \epsilon x. \quad (3.52)$$

In the case $\epsilon = 0$ the solution to Eq. (3.52) is $s_0 = \frac{x}{e}$. This function is known by the *a priori* data and therefore it is possible to obtain an asymptotic solution to Eq. (3.52) in the limit when ϵ tends to zero but for $\epsilon \neq 0$

$$S = s_0 + s_1 \epsilon + s_2 \epsilon^2 + \dots, \quad (3.53)$$

where the remaining coefficients s_i , $i = 1, 2, \dots$ can be easily obtained by introducing this expansion into Eq. (3.52) and enforcing that the resulting series vanishes for every order in ϵ . The asymptotic solution obtained is

$$S = \frac{x}{e}(1 + \varphi), \quad \varphi = \frac{\epsilon}{x} \ll 1. \quad (3.54)$$

We have now to obtain φ as a function of the known parameters x and e . Replacing Eq. (3.54) into Eq. (3.43)

$$x = e \left(\frac{x}{e}(1 + \varphi) \right) - \ln \left(\frac{x}{e}(1 + \varphi) + \sqrt{1 + \frac{x^2}{e^2}(1 + \varphi)^2} \right), \quad (3.55)$$

and taking into account that $\varphi \ll 1$ when x tends to infinite, we compute the first order series expansion of the previous expression with respect to the variable φ

$$0 = -\ln(x + \sqrt{e^2 + x^2}) + \ln(e) + \frac{(\sqrt{e^2 + x^2} - 1)x\varphi}{\sqrt{e^2 + x^2}}, \quad (3.56)$$

such that

$$\varphi = \frac{\sqrt{e^2 + x^2} (\ln(x + \sqrt{e^2 + x^2}) - \ln(e))}{(\sqrt{e^2 + x^2} - 1)x}. \quad (3.57)$$

Replacing Eq. (3.57) into Eq. (3.54) we obtain S as a function of x and e

$$S = \frac{\sqrt{e^2 + x^2} (\ln(x + \sqrt{e^2 + x^2}) - \ln(e) + x) - x}{(\sqrt{e^2 + x^2} - 1)e}. \quad (3.58)$$

In order to increase the accuracy of S , we can expand Eq. (3.54) by

$$S = \frac{x}{e}(1 + \varphi + \xi), \quad \xi \ll 1, \quad \xi \ll \varphi, \quad (3.59)$$

where ξ should now be determined as a function of the known parameters x , e and φ . Replacing Eq. (3.59) into Eq. (3.43)

$$x = e \left(\frac{x}{e}(1 + \varphi + \xi) \right) - \ln \left(\frac{x}{e}(1 + \varphi + \xi) + \sqrt{1 + \frac{x^2}{e^2}(1 + \varphi + \xi)^2} \right), \quad (3.60)$$

and repeating the process for φ , we obtain

$$\xi = -\frac{1}{2x^3} \left[\ln \left(\frac{e}{2x} \right) \right]^2. \quad (3.61)$$

Table 6: Order of magnitude of φ and ξ as a function of M when $e = 1$.

M	50	75	100	200	500
$\varphi \cdot 10^2$	9.40	6.77	5.35	3.01	1.38
$\xi \cdot 10^5$	-8.48	-2.98	-1.40	-0.224	-0.0191

Figure 3.3: Residual ρ versus the hyperbolic anomaly y , when the solution is determined by the polynomials $p_{16}(x)$, $p_{17}(x)$ and $q_{16}(x)$.

The final expression for S , replacing Eq. (3.61), is given by

$$\begin{aligned}
 S = & \frac{1}{2ex^2(\sqrt{e^2+x^2}-1)} \left[\sqrt{e^2+x^2} \left(2x^2 \ln(x + \sqrt{e^2+x^2}) \right. \right. \\
 & \left. \left. - 2x^2 \ln(e) \right) + (\sqrt{e^2+x^2}-1) \left(2x^3 - (\ln(e) - \ln(x))^2 \right. \right. \\
 & \left. \left. + \ln(2)(2\ln(e) - 2\ln(x) - \ln(2)) \right) \right]. \quad (3.62)
 \end{aligned}$$

Now we can assess the accuracy of this asymptotic solution just studying the order of magnitude of the variables we have approximated. We note that φ and ξ decrease when e increases. Therefore $e = 1$ marks the lowest accuracy of the solution in Eq. (3.62), being better when $e > 1$ (see Table 6). If $M \geq 50$, then φ is smaller than 10^{-1} and ξ is smaller in absolute value than 10^{-4} . If $y \geq 5$, we obtain $M \geq 69.2$ when $e = 1$, which assures small values for φ and ξ .

3.5.3 Improving the code

The interval $[0, 5[$ of the hyperbolic anomaly $H \equiv y$ has been discretized in 25 intervals. However, this number, related with the number of polynomials, can be modified having a significant impact on the accuracy of the algorithm but not jeopardizing the speed of the code. By plotting the residual ρ caused by the polynomial $p_i(x)$ associated to the interval $[y_i, y_{i+1}]$, we observe that it follows a Gaussian-like function reaching the maximum, approximately, at the center of the interval (see Fig. 3.3). One option, to decrease the residual and obtain a better initial seed, is to increase the number of polynomials. However, by decreasing the size of the intervals they become too small and the polynomials, which depend on the eccentricity e ,

can become numerically unstable. An alternative is to define a new group of polynomials $q_j(x)$ whose ends are defined by the mean points of the initial intervals $[y_i, y_{i+1}]$, $i = 1, \dots, 25$, by keeping the same size (see Fig. 3.3), that is

$$y_j = \frac{2j-1}{10}, \quad j = 1, \dots, 25, \quad (3.63)$$

where the corresponding S_j is obtained from Eq. (3.42) and $M_j \equiv x_j$ is obtained from Eq. (3.43) (see Table 5). This way, we define 24 intervals $[x_j, x_{j+1}]$, $j = 1, \dots, 24$ and in each one of these intervals we introduce a fifth-degree polynomial $q_j(x)$, $j = 1, \dots, 24$ to interpolate the hyperbolic sine of the hyperbolic anomaly

$$q_j(x) = a_0^{(j)} + a_1^{(j)}x + a_2^{(j)}x^2 + a_3^{(j)}x^3 + a_4^{(j)}x^4 + a_5^{(j)}x^5. \quad (3.64)$$

The introduction of these new polynomials $q_j(x)$ is not necessary, that is, the algorithm works normally with the original polynomials $p_i(x)$. However, the tools elaborated to create the polynomial $p_i(x)$ permit to generate the new polynomial $q_j(x)$ without additional effort. The benefit is that the accuracy and reliability of the code is increased.

3.6 THE HKE-SDG ALGORITHM

Given the values of M and e , to solve the hyperbolic Kepler equation we use the following algorithm:

Seed estimation:

- Inside the singular corner ($M < 0.25$ and $e < 1.5$)
  1. Use the sequence associated with Eq. (3.36) to estimate the starting value y_0 .
  2. The seed S_0 is obtained from the relation: $S_0 = \sinh(y_0)$.
- Outside the singular corner ($M \geq 0.25$ or $e \geq 1.5$)
  1. The value of M permits to detect the ends x_k, x_{k+1} of the right interval—one of the intervals related to the polynomials $p_i(x)$ — which contains M , that is, $M \in [x_k, x_{k+1}]$. Note that the corresponding values $[S_k, S_{k+1}]$ are known and the seed S_0 satisfies: $S_0 \in [S_k, S_{k+1}]$.
  2. If the right interval is the last one: $[x_{26}, x_{27}]$, the algorithm uses the sequence associated with Eq. (3.62) to estimate the starting value S_0 (asymptotic approximation).
  3. If the right interval is not the last one, the algorithm identifies a new interval from the intervals related to the polynomials $q_j(x)$ containing M . Then it chooses the interval that has M closest to one of their ends. The seed S_0 is given by the corresponding polynomial (3.45) or (3.64) (polynomial approximation).

Root estimation:

- The seed S_0 is used to solve the equation

$$S - \frac{1}{e} \ln|S + \sqrt{1 + S^2}| - \frac{1}{e}x = 0, \quad (3.65)$$

by using the Newton-Raphson algorithm.

- The output of this algorithm provides, through the change in variable defined in Eq. (3.42), the desired value of the hyperbolic anomaly $y(x) = H$.

The HKE-SDG code is detailed in Appendix C.

Figure 3.4: Results with HKE-SDG using the MNR iteration scheme in double precision.

3.7 ANALYSIS OF RESULTS

A full analysis of the algorithm was performed by considering two main aspects: speed and accuracy. The numerical analysis of HKE-SDG takes into account the ease of implementation and the robustness of the procedure too. All these considerations are detailed below. The calculations have been carried out in a computer with an Intel(R) Xeon(R) E3-1535M v6 @ 3.10Ghz microprocessor and the executables have been prepared with the version 17.0 of the Intel® C++ Compiler, under the Windows 10 operative system 64 bits.

3.7.1 Speed of the process

Kepler's equation for the hyperbolic case has been solved roughly $4 \cdot 10^6$ times with the algorithm previously explained by using a tolerance ε_{tol} . The eccentricity ranges in the interval $e \in [1, 10]$ and the values of $M \equiv x$ range in the interval $M \in [0, 100]$ (rad). By applying the MNR algorithm we have carried out the calculations in standard double precision and quadruple precision with a tolerance equal to: $\varepsilon_{tol} = 2.22 \cdot 10^{-16}$ and $\varepsilon_{tol} = 10^{-32}$ respectively. The method used in this study is a numerical procedure based on successive approximations. It means that, if the tolerance considered is close to the machine round-off, the method could experience artificial numerical chaos. A way to avoid this problem is by using an appropriate tolerance.

In each run we count the number of iterations needed to reach a solution with a residual $\rho = |e \sinh(y) - y - x|$ lower than the tolerance ε_{tol} . This number is taken as a gross measure of the speed of the procedure. Figures 3.4 and 3.5 show, in the plane (e, M) , the number of iterations associated with HKE-SDG when using the MNR scheme for double and quadruple precision. Note that, in the color code of the figures, green color corresponds to two iterations, white color to one iteration and black color to zero iterations. The averaged value of iterations is 1.582 and 1.010 for double and quadruple precision respectively. The red color (three iterations) does

Figure 3.5: Results with HKE-SDG using the MNR iteration scheme in quadruple precision.

not appear in the figures. The algorithm never enters dead loops or any other kind of nonconvergent phase after many millions of runs.

3.7.2 Accuracy analysis

This section focuses the analysis in the HKE-SDG code with the MNR algorithm, such that the iteration ends when: i) the residual $\rho = |e \sinh(y) - y - x|$ is lower than the tolerance ε_{tol} , or ii) the increment $\Delta S_n/S_n$ in the n iteration of the MNR algorithm is lower than machine round-off.

Let $y_v(x, e)$ be the true solution for given values of e and the mean anomaly $M \equiv x$. Let $y_c(x, e)$ be the numerical solution provided by HKE-SDG for these particular values. Due to truncation and round-off errors, both solutions are related by the equation

$$y_c = y_v + \varepsilon_{abs}, \quad (3.66)$$

where ε_{abs} is the total absolute error associated with the numerical solution $y_c(x, e)$. This absolute error is closely related with the residual ρ . In fact, for the numerical solution $y_c(x, e)$ the residual is given by

$$\rho = |e \sinh(y_v + \varepsilon_{abs}) - (y_v + \varepsilon_{abs}) - x|. \quad (3.67)$$

Since ε_{abs} is small and taking into account that the true solution verifies

$$e \sinh(y_v) - y_v - x = 0, \quad (3.68)$$

the residual takes the form

$$\rho = |\varepsilon_{abs}| |e \cosh(y_v) - 1| + \mathcal{O}(\varepsilon_{abs}^2). \quad (3.69)$$

Therefore the absolute error can be approximated by

$$|\varepsilon_{abs}| = \frac{\rho}{|e \cosh(y_v) - 1|}, \quad (3.70)$$

Figure 3.6: Maximum total absolute error $\varepsilon_{\max}$ versus the eccentricity e , for different number of iterations. Calculations in double precision.

Figure 3.7: Maximum total absolute error $\varepsilon_{\max}$ versus the eccentricity e , for different number of iterations. Calculations in quadruple precision.

and depends on the residual ρ . Something similar happens with the total relative error:

$$|\varepsilon_{\text{rel}}| = \frac{|y_c - y_v|}{|y_v|} = \frac{\rho}{|y_v(e \cosh(y_v) - 1)|}. \quad (3.71)$$

Let us fix the value of the eccentricity e , in the interval $e \in]1, 10]$; then we scan the whole interval $M \in [0, 100]$ calculating the residual ρ without iteration (the residual provided by the starting seed), after one iteration, after two iterations and so on. Let $\rho_{\max}$ be the maximum residual that we find in the scanning of the whole interval $M \in [0, 100]$ for each iteration number. These maximum residuals allow us to calculate the maximum values of the absolute error $\varepsilon_{\max}$ for different number of iterations, and these absolute errors can be associated with the value of the eccentricity e .

The values of such maximum absolute errors can be plotted versus the eccentricity e . This picture provides a very good idea of the accuracy involved in the calculations after zero, one, two, ... iterations. Note that the values plotted indicate the worst

accuracy of the numerical solution $y_c(x, e)$; in fact, for each value of e there are hundreds of values of M where the accuracy is much better than the indicated by $\varepsilon_{\max}$.

Figure 3.6 shows the decimal logarithm of the maximum absolute error versus the eccentricity e . The initial seed —0 iterations— provides an error which, in the worst case, ranges from 10^{-5} to 10^{-8} . Since the calculations associated to Figure 3.6 have been carried out in double precision, the residual obtained after one iteration practically saturates the capacity of the machine to provide the solution with 14 or 15 significant figures. We emphasize that one additional iteration does not improve the accuracy of the numerical solution; this figure shows that the lines corresponding to one, two, and three iterations are, practically, the same.

Figure 3.7 is similar to Fig. 3.6 but with the calculations carried out in quadruple precision. It is clear that with one iteration we obtain more than 31 significant figures except for values of e close to unity, where the number of significant digits is about 24 to 31. With two iterations HKE-SDG saturates the precision capacities of the machine, providing more than 31 significant figures. It should be underlined that one additional iteration does not improve the accuracy of the numerical solution; figure shows that the lines corresponding to two and three iterations are, practically, the same.

3.8 COMPARISON OF RESULTS

Kepler's equation for the hyperbolic case has been solved with the method proposed in this study —HKE-SDG— by using different Newton-Raphson algorithms. Besides, the same analysis has been performed with typical solvers of Kepler's equation which have been compared with HKE-SDG when using the MNR method:

- The modified Newton-Raphson (MNR).
- The Conway method.
- The classical Newton-Raphson method (CNR).
- The method described in Gooding and Odell [36].
- The method described in Fukushima [33].
- The method described in Avendano, Martín-Molina, and Ortigas-Galindo [5].

In each case, Kepler's equation has been solved as described in Sec. 3.7.

Figure 3.4 shows, in the plane (e, M) , the number of iterations associated with the SDG-code when using the modified Newton-Raphson (MNR) scheme and double precision. Table 7 summarizes the behavior of the HKE-SDG using different Newton-Raphson algorithms. When the MNR was used, in the 99.874 % of cases we reach the solution with only 1 (white color) or 2 iterations (green color) and in the 0.126 % no iteration is necessary. The averaged number of iterations was 1.582. The main result from Table 7 is to confirm that the HKE-SDG is almost insensitive to the Newton-Raphson algorithm. With the Conway or the CNR in a very small number of cases 3 iterations are needed; the averaged number of iterations was 1.583 for both methods. The robustness of the HKE-SDG is due to the strong optimization of the initial seed performed in the algorithm; this optimization avoids the convergence problems which could appear in the Newton-Raphson iteration when the initial seeds is far from the real root.

Table 7 also shows the number of iterations associated with the approaches described in Gooding and Odell [36], Fukushima [33] and Avendano, Martín-Molina, and Ortigas-Galindo [5] when using double precision. Note that all of them require

Table 7: Number of iteration (percentage) and computational time obtained with HKE-SDG by using different Newton-Raphson methods and different algorithms checked in double (upper) and quadruple (lower) precision.

Scheme	$i = 0$	$i = 1$	$i = 2$	$i \geq 3$	CPU time
MNR	0.126	41.597	58.277	0	1.398
	0.050	98.899	1.051	0	34.267
Conway	0.126	41.498	58.370	0.006	1.297
	0.050	0.455	99.416	0.079	38.677
CNR	0.126	41.481	58.385	0.008	1.145
	0.050	0.430	99.427	0.093	35.733
Gooding	0.050	0	99.862	0.088	1.808
	0.050	0	99.862	0.088	24.684
Fukushima	0	0	$2 \cdot 10^{-4}$	100	28.669
Avendaño	0	0.058	36.799	63.143	1.657
	0	0	7.163	92.837	31.422

more than 3 iterations for some particular cases, reaching a maximum value of 43, 20 and 19 iterations the Gooding, Fukushima and Avendaño codes respectively. Table 7 shows the computational time invested by the approaches described in Gooding and Odell [36], Fukushima [33] and Avendano, Martín-Molina, and Ortigas-Galindo [5] when using double precision. Gooding's code requires 1.808 seconds, 28.669 seconds with Fukushima's code and 1.657 seconds with Avendaño's code. The averaged number of iterations was 2.014, 6.117 and 2.672 for Gooding, Fukushima and Avendaño respectively. With any of the Newton-Raphson algorithms and using double precision, the HKE-SDG took 1.3 seconds approximately to solve the whole set. The CPU time was 1.398 seconds when using the most reliable scheme: the MNR (see Table 7). The reason why the Fukushima's code takes much more time than the rest of the codes is the very strict tolerance that we use: the machine round-off. Fukushima's algorithm has been developed to work in double precision by evaluating Kepler's equation by a Taylor series expansion with the appropriate number of terms. Due to that truncation, to reach the precision associated to the ε_{tol} turns out to be difficult and it requires more time than usual. With not-so-strict tolerances the differences are much smaller.

Table 7 provides as well the number of iterations and computational time in quadruple precision. Note that Fukushima's code does not include quadruple precision. The use of quadruple precision would imply to extend the original series included in Fukushima [33]. When using the MNR iteration scheme the HKE-SDG reaches the solution with only 1 or 2 iterations in the 99.950 % of cases and in the 0.050 % no iteration is necessary (Fig. 3.5). Notice that the HKE-SDG keeps again the smallest number of iterations. The CPU time invested was 34.267 seconds (MNR), 38.677 seconds (Conway) and 35.733 seconds (CNR), while the averaged number of iterations was 1.010, 1.995 and 1.996 respectively. In quadruple precision the HKE-SDG reaches a maximum number of 2 iterations when applying the MNR method and 3 iterations (≤ 0.1 %) when working with the Conway and CNR methods. The method described in [36] does not exhibit almost any change, keeping the same percentages as in double precision. In quadruple precision Gooding's code reaches

Figure 3.8: Number of significant figures, approximately, obtained with the code of Gooding and Odell [36] (upper left), Fukushima [33] (upper right), Avendaño, Martín-Molina, and Ortigas-Galindo [5] (lower left) and HKE-SDG (lower right) in double precision.

a maximum number of 94 iterations, while Avendaño's code reaches a maximum of 21 iterations. The CPU time invested was 24.684 seconds Gooding's code and 31.422 seconds Avendaño's code. The averaged number of iterations was 2.035 and 2.976 for Gooding and Avendaño respectively.

Figure 3.8 shows the number of significant digits obtained with the different codes when using the MNR method in double precision. Darker colors are associated with less precision, whereas lighter colors represent a better precision. At first sight the four pictures are quite similar and only small differences can be appreciated. If we compare HKE-SDG to the other approaches, the results obtained with Gooding's code show a lower precision in a small region inside the singular corner, while in the case of Avendaño this happens for large values of eccentricity ($e > 8$) and small values of mean anomaly ($M \leq 5$). The number of digits in Fukushima is very similar to the HKE-SDG, however, for some specific points in the yellow region ($M \leq 5$) Fukushima's code reaches less precision.

Figures 3.9 and 3.10 show the number of significant digits when quadruple precision is used in calculations. Note that Gooding's approach displays again a significant deterioration in the singular corner, while Avendaño shows radial beams

Figure 3.9: Number of significant figures, approximately, obtained with the code of Aven-dano, Martín-Molina, and Ortigas-Galindo [5] (left) and HKE-SDG (right) in quadruple precision.

Figure 3.10: Number of significant figures, approximately, obtained with the code of Gooding and Odell [36] in quadruple precision.

with less precision than the HKE-SDG. In fact, the HKE-SDG provides the most precise solution with more than 33 significant digits in the whole plane (e, M) when quadruple precision is applied. As it is shown, HKE-SDG exhibits an improvement of 20 significant figures for almost the complete plane when changing from double to quadruple precision.

The results obtained with the SDG-code when using different numerical methods are very similar and all of them assure the reliability and quality of the solution. In summary, when double precision is used, HKE-SDG turns out to be the fastest method and it provides the most precise results. When quadruple precision is needed, HKE-SDG is the most accurate, stable and reliable even if it turns out to be a little slower. Gooding's method, in quadruple precision, should be tuned to eliminate the lack of precision in the neighborhood of the singular corner.

3.9 APPLICATIONS OF THE HKE-SDG CODE

The code developed in this study could be very useful in many different situations. Some of them have been described in the Introduction. It is especially appropriate in the numerical integration of perturbed problems which appears in dynamical astronomy and stellar dynamics. In particular, when the integration requires to solve Kepler's equation at every step, i.e., a large number of times. Not only in Keplerian perturbed problems, but in more involved situations like in N-body problems.

The HKE-SDG method, as it happens with the EKE-SDG method, works regardless of whether or not the eccentricity maintains constant. In some cases, the eccentricity will keep a constant value like for example in the traditional Encke's method (Encke and Schäfl [\[30\]](#)). In other cases, the value of the eccentricity changes with time like in Keplerian perturbed problems because the perturbations change the value of the eccentricity. The analysis of perturbed Keplerian motion is most often formulated in terms of classical orbit elements. There are many different sets of coordinates that have been used in the numerical integration of perturbed problems. Reference Hintz [\[44\]](#) summarizes many of them.

With a certain frequency we find articles that devote some pages to describe how to solve Kepler's equation in particular situations. The solution of the HKE turns out to be an important step in the course of the research described in the study (see, for example, Roy and Moran [\[76\]](#)). In those cases, and in other situations, having a good solver for the HKE at our disposal could improve drastically the results of the work we are doing.

In the work by Wisdom and Holman [\[95\]](#) it is strongly underlined the important role that long-term integrations are playing in investigations in dynamical astronomy. A clear example is the LONGSTOP project (Milani et al. [\[55\]](#)), an international cooperative program with the purpose of investigating the stability of our Solar System. In this kind of integrations to take care of the errors involved is of paramount importance (see Milani and Nobili [\[54\]](#)). The imperfect convergence of an iterative algorithm—especially when we solve transcendental equations like Kepler's equation—is a well known source of error. Minimizing these kind of errors is always an interesting issue that aids to reach the main objective of the research.

In Yabushita [\[96\]](#), for example, the integration of the governing equations requires to solve the HKE in each integration step. With a fast solver for the HKE, like the one presented in this study, it would not have been necessary to integrate one more equation.

Kepler's equation must be solved as well in symplectic integration codes, as for example in the Wisdom-Holman method (Wisdom and Holman [\[95\]](#)), where the orbital elements exhibit small changes at every step. It can be used for long-term gravitational simulations (Rein and Tamayo [\[71\]](#)), or for the collisional gravitational

N-body problem (Dehnen and Hernandez [26] and Hernandez and Bertschinger [42]).

Another example is the work by Wahde [93], where the orbital parameters of interacting galaxies are determined by using a genetic algorithm. In this case, when the orbital parameters have been decoded from the chromosome, the positions of the two galaxies are integrated backwards in time. Kepler's equation is solved at each step during the backwards integration, where the galaxies are represented by mass-point particles moving on a two-body orbit.

Finally, it is important to highlight, as with the EKE-SDG method, that in strongly perturbed problems it is quite easy to enter in the singular corner. Thanks to the asymptotic expansion, the HKE-SDG code is very reliable and accurate and it avoids some chaotic behavior of numerical nature that appears in that region with other solvers.

LAMBERT'S PROBLEM

The validation of the SDG code allows us to consider more complex orbit determination problems, as for example Lambert's problem. To do that, Lambert's problem is reformulated by describing the family of orbits with a new free parameter. This parameter identifies univocally the different families of orbits providing additional information of the orbits in a very simple way.

4.1 INTRODUCTION

Lambert's orbital boundary-value problem can be stated as follows: *determine the Keplerian orbit passing through two known points - given by their position vectors $\vec{r}_1$ and $\vec{r}_2$ - knowing the time of flight needed to go from one to the other is preassigned and its value is t_f .*

Euler, in 1744, approached a previous version when determining the time employed for a comet to describe an arc of its orbit; he showed that the time is a function of the sum $r_1 + r_2$ and the cord $\lambda_c = |\vec{r}_1 - \vec{r}_2|$, that is, the distance between both positions. It was Lambert (1728-1777) who extended Euler's results to the remaining genders of conics. His studies can be summarized in Lambert's theorem, which states that: *the time of flight t_f needed for going from $\vec{r}_1$ to $\vec{r}_2$ is only function of the sum $r_1 + r_2$, the cord λ_c and the semi-major axis a of the conic defining the transfer: $t_f = t_f(r_1 + r_2, \lambda_c, a)$.*

Due to the development of Astronautics, Lambert's problem, initially associated to orbit determination, has acquired a great importance in fields like rendezvous of satellites, guidance and navigation. In fact, it arises naturally in rendezvous maneuvers between spaceborn objects. In that case, $\vec{r}_1$ represents the starting position of the spacecraft performing the maneuver; $\vec{r}_2$ represents the coordinates of the point where the encounter will happen. It is said that an 'orbital transfer' has been made, or simply 'transfer', from the initial position $\vec{r}_1$ to the final position $\vec{r}_2$. It also shows up in interplanetary missions, when it is required to reach a certain planet at a certain time. If in the contact only the matching of the spacecraft vector position is required, then we speak of "interception"; if, in addition, requires that the speeds are the same, we speak of "rendezvous" or "space encounter".

In the present analysis we suppose that the first point $\vec{r}_1$ is before the second point $\vec{r}_2$, i.e., if the angle between $\vec{r}_2$ and $\vec{r}_1$ is denoted by ν , it follows that $\nu < \pi$. Under this hypothesis the unit vector $\vec{k}$, normal to the orbital plane and defined by

$$\vec{k} = \frac{\vec{r}_1 \times \vec{r}_2}{|\vec{r}_1 \times \vec{r}_2|}, \quad (4.1)$$

would take the direction of the angular momentum: $\vec{h} = h \vec{k}$. If this condition is not satisfied, the roles of $\vec{r}_1$ and $\vec{r}_2$ would have to be interchanged.

Let P_1 and P_2 be the positions represented by the vectors $\vec{r}_1$ and $\vec{r}_2$ respectively. There is a uni-parametric family of Keplerian orbits verifying the geometric restrictions of the problem, that is: 1) the conics pass through P_1 and P_2 (two conditions) and, 2) the conics have their focus in the origin O (two conditions). In order to select a particular conic of this family, an additional condition has to be imposed. In Lambert's problem this condition is given by the time of flight t_f . Consequently, the orbits of the family whose time of flight from P_1 to P_2 is the same than the prefixed time t_f , will verify the kinematic condition and will become the aim of the analysis.

The procedure is based on expressing all Keplerian orbits passing through P_1 and P_2 as a function of a single parameter and then selecting those which verify the kinetic condition. There are different options to choose this parameter based on the geometric properties involved in Lambert's problem Arora and Russell [3], Avanzini [4], Battin and Vaughan [9], Battin [11], Gooding [35], and Izzo [46]. The most direct way is to consider the semi-major axis or the eccentricity. In what follows a new parameter χ is defined, presenting similarity with the parameter taken by Avanzini [4].

4.2 BASIC ELEMENTS

We start by introducing the unit vectors $\vec{u}_1$ and $\vec{u}_2$ defined as:

$$\vec{u}_1 = \frac{\vec{r}_1}{r_1}, \quad \vec{u}_2 = \frac{\vec{r}_2}{r_2}. \quad (4.2)$$

The angle ν , called transfer angle, is a known parameter given by the relations:

$$\cos(\nu) = \frac{\vec{r}_1 \cdot \vec{r}_2}{r_1 r_2} = \vec{u}_1 \cdot \vec{u}_2, \quad \sin(\nu) = \frac{|\vec{r}_1 \times \vec{r}_2|}{r_1 r_2} = |\vec{u}_1 \times \vec{u}_2|. \quad (4.3)$$

We admit that $\nu \in]0, \pi]$, although the situation $\nu = \pi$ presents singularities that will be studied later. The auxiliary axes Ox_0y_0 are introduced, based on the bisectrix of the vectors $\vec{u}_1$ and $\vec{u}_2$ and the perpendicular direction. The unit basis vectors $(\vec{i}_0, \vec{j}_0)$ of the reference Ox_0y_0 are related with $(\vec{u}_1, \vec{u}_2)$ through the equations:

$$\vec{i}_0 = \frac{\vec{u}_1 - \vec{u}_2}{|\vec{u}_1 - \vec{u}_2|}, \quad \vec{j}_0 = \frac{\vec{u}_1 + \vec{u}_2}{|\vec{u}_1 + \vec{u}_2|}. \quad (4.4)$$

Taking into account that $|\vec{u}_1 + \vec{u}_2|^2 = 4 \cos^2(\frac{\nu}{2})$ and $|\vec{u}_1 - \vec{u}_2|^2 = 4 \sin^2(\frac{\nu}{2})$, it follows:

$$\vec{i}_0 = \frac{\vec{u}_1 - \vec{u}_2}{2 \sin(\frac{\nu}{2})}, \quad \vec{j}_0 = \frac{\vec{u}_1 + \vec{u}_2}{2 \cos(\frac{\nu}{2})}, \quad (4.5)$$

or equivalently:

$$\vec{u}_1 = \sin\left(\frac{\nu}{2}\right) \vec{i}_0 + \cos\left(\frac{\nu}{2}\right) \vec{j}_0 \quad (4.6a)$$

$$\vec{u}_2 = -\sin\left(\frac{\nu}{2}\right) \vec{i}_0 + \cos\left(\frac{\nu}{2}\right) \vec{j}_0. \quad (4.6b)$$

The eccentricity vector $\vec{e}$ is provided by the unit vector pointing to the pericenter such that:

$$\vec{e} = e \vec{i}_p. \quad (4.7)$$

4.3 GEOMETRIC ASPECTS OF THE ANALYSIS

The family of conics passing through $\vec{r}_1$ and $\vec{r}_2$ depends on a single parameter; we could choose the semi-major axis a , so that changing the value of a , all the conics of the family are obtained. If we consider ellipses, the attractive center O is one of the foci: the occupied focus. The other focus O' , called empty focus, is involved in the metric definition of an ellipse: *the locus of points whose sum of distances to two fixed points, the foci, remains constant and equal to $2a$* . If the semi-major axis value is fixed, the relations

$$|\vec{O} \vec{P}_1| + |\vec{O}' \vec{P}_1| = r_1 + |\vec{O}' \vec{P}_1| = 2a \quad (4.8a)$$

$$|\vec{O} \vec{P}_2| + |\vec{O}' \vec{P}_2| = r_2 + |\vec{O}' \vec{P}_2| = 2a, \quad (4.8b)$$

determine the position of O' such that: $|\overrightarrow{O'P_1}| = 2a - r_1$ and $|\overrightarrow{O'P_2}| = 2a - r_2$. This means that the empty focus belongs to two circumferences with centers P_1 and P_2 and radii $2a - r_1$ and $2a - r_2$ respectively. As a consequence, O' will be one of the intersection points of both circumferences. In general there are two symmetric solutions with respect to the cord λ_c , although there can be just one or none. When the solution is double, the intersection points of the two circumferences will define the empty foci of two ellipses passing through P_1 and P_2 with the same semi-major axis. These are called conjugate orbits or conjugate ellipses. By decreasing the semi-major axis, the mechanical energy of the ellipse decreases, since:

$$E = -\frac{\mu}{2a}. \quad (4.9)$$

The successive foci obtained by changing the parameter a describe a locus which corresponds to a hyperbola with foci P_1 and P_2 . The metric definition of a hyperbola is *the locus of points whose distance difference to two fixed points, the foci, remains constant*. If this is applied member by member to the equations $|\overrightarrow{O'P_1}|$ and $|\overrightarrow{O'P_2}|$, the constant value obtained is $r_2 - r_1$.

4.3.1 Minimum-energy ellipse

As it has been previously mentioned, the minimum-energy ellipse is the ellipse with minimum semi-major axis. When decreasing a , the two solutions given by the empty focus approach until they coincide. This double root corresponds to the minimum of a , such that below this value there is no solution to the problem. Indeed, in the presence of an attractive center the position P_2 is never reached from the position P_1 if the energy applied to the satellite is smaller than the minimum-energy.

The double root is located on the cord connecting P_1 and P_2 , that is:

$$2a_* - r_1 + 2a_* - r_2 = \lambda_c, \quad (4.10)$$

which implies:

$$a_* = \frac{s}{2}, \quad (4.11)$$

being s the semiperimeter of the spatial triangle $\overline{OP_1P_2}$ defined by

$$s = \frac{1}{2}(r_1 + r_2 + \lambda_c). \quad (4.12)$$

An elliptic orbit verifies the condition $p = a(1 - e^2)$ and consequently the following expression holds:

$$4a^2e^2 = 4a(a - p). \quad (4.13)$$

Let p_* be the *semilatus rectum* of the minimum-energy orbit, by introducing the semiperimeter s , the previous equation becomes

$$d^2 = 4a_*^2e_*^2 = s^2 - 2sp_*. \quad (4.14)$$

Consider again the spatial triangle $\overline{OP_1P_2}$ of the transfer (Fig. 4.1). In the orbit of minimum-energy the empty focus O' is located on the cord P_1P_2 , separated from P_1 and P_2 the distances:

$$|\overrightarrow{O'P_1}| = 2a_* - r_1 = s - r_1, \quad |\overrightarrow{O'P_2}| = s - r_2. \quad (4.15)$$

Let $\widehat{OQO'}$ be the rectangular triangle and α the angle defined by $\vec{r}_2$ and the cord P_1P_2 . The parameter d is defined as

$$d^2 = |\overrightarrow{OQ}| + |\overrightarrow{O'Q}| = (r_2 - (s - r_2)\cos(\alpha))^2 + (s - r_2)^2\sin^2(\alpha), \quad (4.16)$$

Figure 4.1: Spatial triangle and ellipse of minimum-energy.

that is:

$$d^2 = s^2 - 2r_2(s - r_2)(1 + \cos(\alpha)). \quad (4.17)$$

Applying the cosine theorem to the spatial triangle $\overline{OP_1P_2}$ it follows

$$r_1^2 = \lambda_c^2 + r_2^2 - 2\lambda_c r_2 \cos(\alpha) \Rightarrow \cos(\alpha) = \frac{\lambda_c^2 + r_2^2 - r_1^2}{2\lambda_c r_2}, \quad (4.18)$$

such that replacing this expression into Eq. (4.17):

$$d^2 = s^2 - 2r_2(s - r_2) \frac{2s(s - r_1)}{\lambda_c r_2} = s^2 - \frac{4s}{\lambda_c}(s - r_1)(s - r_2). \quad (4.19)$$

Matching this expression with Eq. (4.14), and taking into account the value of the semiperimeter s , the following expression for p_* is obtained:

$$p_* = \frac{r_1 r_2}{\lambda_c} (1 - \cos(\nu)). \quad (4.20)$$

The eccentricity of the minimum-energy orbit is directly derived from the condition $p = a(1 - e^2)$, yielding

$$e_* = \sqrt{1 - \frac{2p_*}{s}}. \quad (4.21)$$

Equation (4.21) bears out the fact that the minimum-energy orbit can only be an ellipse. On the one hand an hyperbolic orbit is not possible because neither p_* nor s can be negative and on the other hand a parabolic orbit would require $p_* = 0$ which can not happen.

4.3.2 Fundamental ellipse

There is another remarkable ellipse inside the family of ellipses associated to the spatial triangle $\overline{OP_1P_2}$. It is called the fundamental ellipse and it is characterized for having the minimum eccentricity.

The hodograph of the eccentricity vector $\vec{e}$, that is, the locus of the space determined by the end of the eccentricity vector of the family of orbits from the common origin O , describes a straight line (Fig. 4.2). This property can be deduced from the focal equation:

$$r = \frac{p}{1 + e \cos(\theta)} \Rightarrow e \cos(\theta) = \frac{p - r}{r}. \quad (4.22)$$

Figure 4.2: Locus of the end points of the eccentricity vector.

Considering the true anomalies θ_1 and θ_2 of the radio vectors, we have respectively:

$$\vec{e} \cdot \vec{r}_1 = r_1 e \cos(\theta_1) = p - r_1 \quad (4.23a)$$

$$\vec{e} \cdot \vec{r}_2 = r_2 e \cos(\theta_2) = p - r_2, \quad (4.23b)$$

and subtracting both equations:

$$\vec{e} \cdot (\vec{r}_2 - \vec{r}_1) = r_1 - r_2. \quad (4.24)$$

The constant vector $\vec{r}_2 - \vec{r}_1$ is expressed in the form:

$$\vec{r}_2 - \vec{r}_1 = \lambda_c \sigma \vec{u}_c \Leftrightarrow \vec{u}_c = \frac{1}{\sigma} \frac{\vec{p}_1 \vec{p}_2}{|\vec{p}_1 \vec{p}_2|}, \quad (4.25)$$

being $\vec{u}_c$ a unit vector in the direction of the cord and $\sigma = \pm 1$ a factor to fix the sense of $\vec{u}_c$. This factor is chosen as follows:

- If $b = \frac{r_2}{r_1} < 1 \Rightarrow \sigma = 1$.
- If $b = \frac{r_2}{r_1} > 1 \Rightarrow \sigma = -1$.

The value of σ agrees with the sign of $1 - b$, that is:

$$\sigma = \frac{1 - b}{|1 - b|}. \quad (4.26)$$

The vector $\vec{u}_c$ chosen in that way always points from the farthest position P_i of the attractive center O to the nearest.

From Eqs. (4.24) and (4.25) we obtain the relation:

$$\vec{e} \vec{u}_c = \frac{r_1 - r_2}{\sigma \lambda_c} = \frac{|r_2 - r_1|}{\lambda_c}, \quad (4.27)$$

which shows that the end of the eccentricity vector describes a straight line perpendicular to the cord. Because $\vec{u}_c$ has a norm equal to unity, the eccentricity reaches a minimum value given by:

$$\vec{e}_{\odot} = \vec{e}_{\min} = \frac{|r_2 - r_1|}{\lambda_c}, \quad (4.28)$$

Figure 4.3: Fundamental ellipse.

when the vector $\vec{e}$ has the direction and sense of $\vec{u}_c$, that is, parallel to the cord and a sense from the farthest point to the nearest with respect to the occupied focus O . In this case the eccentricity is named as fundamental eccentricity.

Among all the orbits of the family associated with the spatial triangle $\overline{OP_1P_2}$, there is one of them characterized with the minimum eccentricity (4.28) and it is named fundamental ellipse. In this orbit the semi-major axis is parallel to the cord because $\vec{e}_\odot$ has the direction of $\vec{u}_c$. As a consequence, and taking into account that P_1 and P_2 are points of the ellipse, P_1 and P_2 should be symmetric with respect to the semi-minor axis (Fig. 4.3). This symmetry implies that $|\vec{OP}_1| = |\vec{OP}_2| = r_1$ and $|\vec{O'P}_2| = |\vec{O'P}_1| = r_2$, which together with Eq. (4.8a) lead to:

$$\vec{OP}_1 + \vec{O'P}_1 = 2\vec{a} \Rightarrow \vec{OP}_1 + \vec{OP}_2 = 2\vec{a} \Rightarrow r_1 + r_2 = 2a, \quad (4.29)$$

and the fundamental semi-major axis is expressed as:

$$a_\odot = \frac{1}{2}(r_1 + r_2). \quad (4.30)$$

Once Eqs. (4.28) and (4.30) are defined, the fundamental *semilatus rectum* is provided just by applying its classic expression and the equation $\lambda_c^2 = r_1^2 + r_2^2 - 2r_1r_2\cos(\nu)$:

$$p_\odot = a_\odot(1 - e_\odot^2) = \frac{r_1r_2}{\lambda_c^2}(r_1 + r_2)(1 - \cos(\nu)). \quad (4.31)$$

4.4 THE FREE PARAMETER χ

In order to describe and parameterize the family of orbits passing through P_1 and P_2 , a free parameter should be chosen. The selection of the semi-major axis is not recommended due to the existence of conjugate orbits for the same value of a , which implies duplicity in the results. The analysis described in the following lines introduces a new parameter χ which presents important advantages. It is defined as the true anomaly of the bisectrix determined by the unit vectors $\vec{u}_1$ and $\vec{u}_2$. The true anomalies of the points are related with χ as follows (see Fig. 4.4):

$$\theta_1 = \chi - \frac{\nu}{2}, \quad \theta_2 = \chi + \frac{\nu}{2}. \quad (4.32)$$

The fundamental parameters described in the previous section are all known. Thus, the goal is to express the orbit elements of the conic we are looking for as a function of the parameter χ . To do that we have to define first an additional parameter, the fundamental parameter $\chi_\odot$.

Figure 4.4: Geometric meaning of χ and true anomalies.

The unit vector pointing to the pericenter $\vec{i}_p$, which provides the direction of the eccentricity, has components in the auxiliary axes:

$$\vec{i}_p = \sin(\chi) \vec{i}_0 + \cos(\chi) \vec{j}_0. \quad (4.33)$$

If we are in the fundamental ellipse, $\chi = \chi_\odot$ and Eq. (4.33) coincides with $\vec{u}_c$. The unit vector in the direction of the cord is:

$$\vec{u}_c = \sin(\chi_\odot) \vec{i}_0 + \cos(\chi_\odot) \vec{j}_0. \quad (4.34)$$

Taking into account Eqs. (4.7), (4.27) and (4.28) it follows that:

$$\vec{e} = e \vec{i}_p \Rightarrow \vec{e} \cdot \vec{u}_c = e \vec{i}_p \cdot \vec{u}_c \Rightarrow e_\odot = e \vec{i}_p \cdot \vec{u}_c, \quad (4.35)$$

and since $\vec{i}_p \cdot \vec{u}_c = \cos(\chi - \chi_\odot)$ we get:

$$e = \frac{e_\odot}{\cos(\chi - \chi_\odot)}, \quad (4.36)$$

which provides the eccentricity as a function of the fundamental eccentricity and $\chi_\odot$.

The following step is to express the difference $\vec{r}_2 - \vec{r}_1$ in the base $(\vec{i}_0, \vec{j}_0)$. It is done by a change of base given by the Eqs. (4.6), such that:

$$\vec{r}_2 - \vec{r}_1 = -(r_2 + r_1) \sin\left(\frac{\nu}{2}\right) \vec{i}_0 + (r_2 - r_1) \cos\left(\frac{\nu}{2}\right) \vec{j}_0. \quad (4.37)$$

Introducing this expression into Eq. (4.25) for $\vec{u}_c$:

$$\vec{u}_c = -\frac{r_2 + r_1}{\lambda_c \sigma} \sin\left(\frac{\nu}{2}\right) \vec{i}_0 + \frac{r_2 - r_1}{\lambda_c \sigma} \cos\left(\frac{\nu}{2}\right) \vec{j}_0. \quad (4.38)$$

In order to obtain $\chi_\odot$, we relate Eqs. (4.34) and (4.38) obtaining:

$$\sin(\chi_\odot) = -\frac{r_2 + r_1}{\lambda_c \sigma} \sin\left(\frac{\nu}{2}\right) = -\kappa \frac{1+b}{1-b} \sin\left(\frac{\nu}{2}\right) \quad (4.39a)$$

$$\cos(\chi_\odot) = \frac{r_2 - r_1}{\lambda_c \sigma} \cos\left(\frac{\nu}{2}\right) = -\kappa \cos\left(\frac{\nu}{2}\right), \quad (4.39b)$$

with $\kappa = |1-b| \frac{r_1}{\lambda_c} > 0$. Dividing both expressions in Eqs. (4.39) we finally get:

$$\tan(\chi_\odot) = \tan\left(\frac{\nu}{2}\right) \frac{1+b}{1-b} = \zeta_1, \quad (4.40)$$

Figure 4.5: $\vec{e}_{\odot}$ has the sense $\vec{r}_1 - \vec{r}_2$ when $b > 1$ (left) and $\vec{r}_2 - \vec{r}_1$ when $b < 1$ (right).

which provides the value of $\chi_{\odot} = \chi_{\odot}(\nu, b)$, that is, the direction of the semi-major axis of the fundamental ellipse.

Instead of using the parameter χ , Eq. (4.40) can be rewritten in terms of the parameter ϕ defined by

$$\chi + \phi = \pi \Rightarrow \tan(\chi) = -\tan(\phi), \quad (4.41)$$

such that it adopts the form:

$$\tan(\phi_{\odot}) = \tan\left(\frac{\nu}{2}\right) \frac{b+1}{b-1} = \zeta_2. \quad (4.42)$$

The two angles, χ and ϕ , can be employed to identify the eccentricity vector and any of them can be selected as the parameter of the family of Keplerian conics passing through P_1 and P_2 . Both angles are measured from the bisectrix of vectors $\vec{r}_1$ and $\vec{r}_2$, but from different rays. The angle χ is measured from the inner ray to the triangle of the transfer and in clockwise direction. The angle ϕ is measured from the opposite ray and in counterclockwise direction.

4.4.1 Eccentricity of the orbits

Once the parameters $\chi_{\odot}$ and $\phi_{\odot}$ are defined, the fundamental eccentricity (4.28) can be expressed as a function of these parameters (see Fig. 4.5). We start by calculating the quotient between r_1 and r_2 through the focal equation to obtain an expression for the eccentricity:

$$e = \frac{r_2 - r_1}{r_1 \cos(\theta_1) - r_2 \cos(\theta_2)}. \quad (4.43)$$

Then, by introducing the values of the true anomalies given by Eq. (4.32) and removing the parameter b through Eq. (4.40) it is obtained:

$$e = -\frac{\cos(\chi_{\odot})}{\cos\left(\frac{\nu}{2}\right) \cos(\chi - \chi_{\odot})} = \frac{\cos(\phi_{\odot})}{\cos\left(\frac{\nu}{2}\right) \cos(\phi - \phi_{\odot})}, \quad (4.44)$$

whose minimum is reached when $\cos(\chi - \chi_{\odot}) = 1$ such that:

$$e_{\odot} = -\frac{\cos(\chi_{\odot})}{\cos\left(\frac{\nu}{2}\right)} = \frac{\cos(\phi_{\odot})}{\cos\left(\frac{\nu}{2}\right)}. \quad (4.45)$$

Taking into account that Eq. (4.45) should be necessarily positive and smaller than one ($e_{\odot} = 1$ implies $\chi_{\odot} = \frac{\gamma}{2} + \pi$ and $b = 0$ in Eq. (4.40), which is not possible), it requires the study of the domain of Eq. (4.40) depending on the sign of b .

If $b \neq 1$, Eq. (4.40) has two roots which differ from each other by π radians, but only one of them provides a positive eccentricity:

- If $b < 1 \Rightarrow \tan(\chi_{\odot}) > 0$
  - $\chi_{\odot} \in [0, \frac{\pi}{2}[\Rightarrow \cos(\chi_{\odot}) > 0 \Rightarrow e_{\odot} < 0.$
  - $\chi_{\odot} \in [\pi, \frac{3\pi}{2}[\Rightarrow \cos(\chi_{\odot}) < 0 \Rightarrow e_{\odot} > 0.$
- If $b > 1 \Rightarrow \tan(\chi_{\odot}) < 0$
  - $\chi_{\odot} \in]\frac{3\pi}{2}, 2\pi] \Rightarrow \cos(\chi_{\odot}) > 0 \Rightarrow e_{\odot} < 0.$
  - $\chi_{\odot} \in]\frac{\pi}{2}, \pi] \Rightarrow \cos(\chi_{\odot}) < 0 \Rightarrow e_{\odot} > 0.$

The correct solution is obtained when the root of Eq. (4.40) is in the interval $] \frac{\pi}{2}, \frac{3\pi}{2}[$, that is, when $\cos(\chi_{\odot}) < 0$ and thus, $e_{\odot} > 0$. Taking into account how the direction and sense of the fundamental eccentricity have been defined, we can conclude that:

- If $b < 1 \Rightarrow r_2 < r_1 \Rightarrow \frac{\vec{e}_{\odot}}{|\vec{e}_{\odot}|} = \frac{\overrightarrow{P_1 P_2}}{|P_1 P_2|}.$
- If $b > 1 \Rightarrow r_2 > r_1 \Rightarrow \frac{\vec{e}_{\odot}}{|\vec{e}_{\odot}|} = \frac{\overrightarrow{P_2 P_1}}{|P_2 P_1|}.$

If $b = 1$ then Eq. (4.28) is zero and the fundamental ellipse is a circumference. In this case $\chi_{\odot}$ can be $\frac{\pi}{2}$ or $\frac{3\pi}{2}$ and $e_{\odot} = 0$.

Summarizing, only the root belonging to the interval $[\frac{\pi}{2}, \frac{3\pi}{2}]$ provides the correct solution. The root $\chi_{\odot} \in [\frac{\pi}{2}, \frac{3\pi}{2}]$ has an associated root $\phi_{\odot} \in [-\frac{\pi}{2}, \frac{\pi}{2}]$ such that when $\chi_{\odot} \leq \pi$ then $\phi_{\odot} < 0$ (Fig. 4.6).

4.4.2 The plane (b, ν)

Two parameters, $e_{\odot}$ and $\chi_{\odot}$, have been introduced. Both are related with the fundamental ellipse and only dependent on the transfer angle ν and the quotient $b = \frac{r_2}{r_1}$. These functions are given by Eqs. (4.40) and (4.28) where the last equation can be rewritten in terms of (b, ν) as:

$$e_{\odot} = \frac{|1 - b|}{\sqrt{1 + b^2 - 2b \cos(\nu)}}. \quad (4.46)$$

The equations of the inverse bijective correspondence between the parameters (ν, b) and $(e_{\odot}, \chi_{\odot})$ are:

$$\nu = 2 \arccos \left(-\frac{\cos(\chi_{\odot})}{e_{\odot}} \right) \quad (4.47a)$$

$$b = \frac{\sin(\chi_{\odot}) + \sqrt{e_{\odot}^2 - \cos^2(\chi_{\odot})}}{\sin(\chi_{\odot}) - \sqrt{e_{\odot}^2 - \cos^2(\chi_{\odot})}}. \quad (4.47b)$$

The parameters $(e_{\odot}, \chi_{\odot})$ are independent. However, they can not take arbitrary values. In the first equation of (4.47) the values of $\chi_{\odot}$ and $e_{\odot}$ should be chosen such that $\cos(\frac{\gamma}{2})$ is smaller than one. In the second equation $\cos(\chi_{\odot})$ should be smaller than $e_{\odot}$ to obtain a positive value of b . As a consequence, the relations (4.47) work as restrictions that should verify $(e_{\odot}, \chi_{\odot})$.

Figure 4.6: Domain of the free parameters $\chi_{\odot}$ (upper) and $\phi_{\odot}$ (lower).

4.4.3 Semi-major axis and semilatus rectum of the orbits

In what follows the semi-major axis and *semilatus rectum* are described as a function of the fundamental parameters and the free parameter χ . The focal equation allows to obtain the expressions:

$$\frac{p}{r_1} = 1 + e \cos(\theta_1), \quad \frac{p}{r_2} = 1 + e \cos(\theta_2), \quad (4.48)$$

and adding up both equations:

$$p \frac{r_1 + r_2}{r_1 r_2} = 2 + e (\cos(\theta_1) + \cos(\theta_2)). \quad (4.49)$$

The expressions (4.32) are replaced into (4.44) and this one in the previous expression:

$$p = \frac{2 r_1 r_2}{r_1 + r_2} \frac{\sin(\chi) \sin(\chi_\odot)}{\cos(\chi - \chi_\odot)}. \quad (4.50)$$

From this equation the fundamental *semilatus rectum* is derived just taking $\chi = \chi_\odot$:

$$p_\odot = \frac{2 r_1 r_2}{r_1 + r_2} \sin^2(\chi_\odot), \quad (4.51)$$

and the *semilatus rectum* can be expressed as a function of Eqs. (4.51), (4.44) and (4.45):

$$p = \frac{\sin(\chi)}{\sin(\chi_\odot)} \frac{p_\odot}{\cos(\chi - \chi_\odot)} = p_\odot \frac{e \sin(\chi)}{e_\odot \sin(\chi_\odot)}. \quad (4.52)$$

The semi-major axis is obtained from the classic relation: $p = \pm a(1 - e^2)$ such that:

$$e < 1 \Rightarrow a(\chi) = \frac{p(\chi)}{1 - e^2(\chi)} \quad (4.53a)$$

$$e > 1 \Rightarrow a(\chi) = \frac{p(\chi)}{e^2(\chi) - 1}. \quad (4.53b)$$

4.4.4 The conjugate ellipse

Let χ be the value of the parameter associated to one of the ellipses of the family, and $\tilde{\chi}$ the parameter related to the conjugate ellipse. The definition of conjugate orbit consists in imposing the same semi-major axis on the two ellipses:

$$a(\chi) = \frac{p(\chi)}{1 - e^2(\chi)} = \frac{p(\tilde{\chi})}{1 - e^2(\tilde{\chi})} = a(\tilde{\chi}). \quad (4.54)$$

Taking into account the value of $p(\chi)$ in Eq. (4.52):

$$\frac{e(\chi) \sin(\chi)}{1 - e^2(\chi)} = \frac{e(\tilde{\chi}) \sin(\tilde{\chi})}{1 - e^2(\tilde{\chi})}, \quad (4.55)$$

and introducing Eq. (4.36) in the equation we get:

$$\frac{\sin(\chi) \cos(\chi - \chi_\odot)}{\cos^2(\chi - \chi_\odot) - e_\odot^2} = \frac{\sin(\tilde{\chi}) \cos(\tilde{\chi} - \chi_\odot)}{\cos^2(\tilde{\chi} - \chi_\odot) - e_\odot^2}. \quad (4.56)$$

Since $(\chi, \chi_\odot, e_\odot)$ are known, the quotient of the first member of the equation is known, while in the second member is found the unknown $\tilde{\chi}$. If we name the following quantity as ρ :

$$\rho(\chi) = \frac{\sin(\chi) \cos(\chi - \chi_\odot)}{\cos^2(\chi - \chi_\odot) - e_\odot^2}, \quad (4.57)$$

the equation we have to solve is:

$$\sin(\tilde{\chi}) \cos(\tilde{\chi} - \chi_{\odot}) - \rho(\cos^2(\tilde{\chi} - \chi_{\odot}) - e_{\odot}^2) = 0. \quad (4.58)$$

Doing a change of variable: $\tilde{\chi} = \chi_{\odot} + \zeta$, the equation takes the form:

$$(\rho - \sin(\chi_{\odot})) \cos^2(\zeta) - \cos(\chi_{\odot}) \sin(\zeta) \cos(\zeta) - \rho e_{\odot}^2 = 0. \quad (4.59)$$

Dividing by $\cos^2(\zeta)$ and applying the change of variable $u = \tan(\zeta)$ we obtain the following second order equation to be solved:

$$\rho e_{\odot}^2 u^2 + \cos(\chi_{\odot}) u + \sin(\chi_{\odot}) - \rho(1 - e_{\odot}^2) = 0, \quad (4.60)$$

that, in general, has two real roots: one of them u_1 belongs to χ and the other, u_2 belongs to $\tilde{\chi}$. Applying the Cardano relation:

$$u_1 + u_2 = -\frac{\cos(\chi_{\odot})}{\rho e_{\odot}^2}, \quad (4.61)$$

we finally arrive at the solution:

$$\tan(\tilde{\chi} - \chi_{\odot}) = -\frac{\cos(\chi_{\odot})}{\rho(\chi) e_{\odot}^2} - \tan(\chi - \chi_{\odot}), \quad (4.62)$$

which provides the value $\tilde{\chi}$ with $\rho(\chi)$ given by Eq. (4.57).

The ellipse of minimum-energy is the conjugate of itself and consequently the value of the parameter χ corresponding to that ellipse verifies the condition:

$$\chi_* = \chi = \tilde{\chi}. \quad (4.63)$$

In this case Eq. (4.62) provides the relation

$$2 \tan(\chi_* - \chi_{\odot}) = -\frac{\cos(\chi_{\odot})}{\rho(\chi_*) e_{\odot}^2}, \quad (4.64)$$

and by introducing Eq. (4.57) for the value of $\rho(\chi_*)$ it becomes

$$2 \sin(\chi_*) \sin(\chi_* - \chi_{\odot}) + \frac{\cos(\chi_{\odot})}{e_{\odot}^2} \cos^2(\chi_* - \chi_{\odot}) = \cos(\chi_{\odot}). \quad (4.65)$$

The change of variable $\chi_* = \chi_{\odot} + \zeta$ provides the expression

$$\cos(\chi_{\odot}) \sin^2(\zeta) + \sin(\chi_{\odot}) \sin(\zeta) \cos(\zeta) + \frac{\cos(\chi_{\odot})}{2 e_{\odot}^2} (\cos^2(\zeta) - 1) = 0, \quad (4.66)$$

and the change of variable $u = \tan(\zeta)$ gives the second degree equation

$$\cos(\chi_{\odot}) u^2 + 2 \sin(\chi_{\odot}) u + \cos(\chi_{\odot}) \frac{1 - e_{\odot}^2}{e_{\odot}^2} = 0, \quad (4.67)$$

with roots

$$\tan(\chi_* - \chi_{\odot}) = -\tan(\chi_{\odot}) \pm \frac{\sqrt{e_{\odot}^2 - \cos^2(\chi_{\odot})}}{e_{\odot} \cos(\chi_{\odot})}. \quad (4.68)$$

One of the roots of Eq. (4.68) is the one sought while the remaining is a spurious solution, that is, it does not correspond to an elliptic orbit. As it will be explained

later, it is spurious because does not verify the condition $\chi_* \in]\chi_\odot - \delta, \chi_\odot + \delta]$ with δ defined by $\cos(\delta) = e_\odot$.

If δ is now introduced into Eq. (4.67), it adopts the form

$$u^2 + 2 \tan(\chi_\odot) u + \tan^2(\delta) = 0, \quad (4.69)$$

and the roots, defined in Eq. (4.68), are:

$$\tan(\chi_* - \chi_\odot) = -\tan(\chi_\odot) \pm \sqrt{\tan^2(\chi_\odot) - \tan^2(\delta)}. \quad (4.70)$$

The condition $\chi_* \in]\chi_\odot - \delta, \chi_\odot + \delta]$ becomes $\chi_* - \chi_\odot \in]-\delta, \delta]$ such that if $\tan(\chi_\odot) > 0$ ($b < 1$) the root defined with the sign (+) is in the interval and provides the value of χ_* . If $\tan(\chi_\odot) < 0$ ($b > 1$) then the remaining root provides the value of χ_* .

4.4.5 The conjugate hyperbola

Let χ be the value of the parameter associated to one of the hyperbolas of the family with semi-major axis a . There is another hyperbola with the same semi-major axis and parameter $\tilde{\chi}$ referred to as conjugate hyperbola. By imposing the same semi-major axis on the two hyperbolas the equation obtained is:

$$a(\chi) = \frac{p(\chi)}{e^2(\chi) - 1} = \frac{p(\tilde{\chi})}{e^2(\tilde{\chi}) - 1} = a(\tilde{\chi}). \quad (4.71)$$

The procedure followed for the conjugate ellipse is applied to finally obtain the same expressions as in Eqs. (4.57) and (4.62) for $\rho(\chi)$ and $\tilde{\chi}$ respectively. Similarly to the family of ellipses, where a minimum-energy ellipse exists which is the conjugate of itself, the question arising is the existence or not of a self-conjugate hyperbola. Let $\chi = \chi_*$ be value characterizing this orbit; it should verify the equation

$$2 \tan(\chi_* - \chi_\odot) = -\frac{\cos(\chi_\odot)}{\rho(\chi_*) e_\odot^2}. \quad (4.72)$$

The solution is obtained by following the same procedure as in the elliptic case: the two changes of variables $\chi_* = \chi_\odot + \zeta$ and $u = \tan(\zeta)$ as well as the introduction of the parameter δ . The final expression of the second degree equation is that of Eq. (4.69) and the roots are:

$$\tan(\chi_* - \chi_\odot) = -\tan(\chi_\odot) \pm \sqrt{\tan^2(\chi_\odot) - \tan^2(\delta)}. \quad (4.73)$$

The sought solution, valid for the self-conjugate hyperbola, is the root defined out of the interval, that is: $|\chi_* - \chi_\odot| > \delta$. However, as it will be shown later, there are not roots providing hyperbolas of the family and consequently there are no self-conjugate hyperbolas.

4.5 PARAMETRIC ANALYSIS

In what follows the evolution of the orbits of the family as a function of the parameter χ is analyzed. As it was deduced, the fundamental parameter $\chi_\odot$ is defined in the interval $[\frac{\pi}{2}, \frac{3\pi}{2}]$ which imposes $\chi \in]\chi_\odot - \frac{\pi}{2}, \chi_\odot + \frac{\pi}{2}[$ (see Fig. 4.7).

As it is known, the eccentricity vector describes a straight line. Plotting this line in the plane Ox_0y_0 and the circumference with center the origin and unit radius (see Fig. 4.8), a diagram with the different subfamilies of orbits inside the family is obtained. Inside the circumference there are elliptic orbits, on the circumference there

Figure 4.7: Evolution of the eccentricity as a function of χ for the orbits of the family ($\chi_{\odot} = 100^\circ$ and $e_{\odot} = 0.5$).

are parabolic orbits, and outside the circumference there are hyperbolic orbits. The domain of these subfamilies will be different depending on whether b is greater or smaller than unity (see Fig. 4.8). Consequently, by defining the parameter $\delta = \arccos(e_{\odot}) \in [0, \frac{\pi}{2}]$, the situation when $b > 1$ is:

- If $\chi \in]\chi_{\odot} - \frac{\pi}{2}, \chi_{\odot} - \delta[\Rightarrow$ Subfamily of hyperbolas I ($e > 1$).
- If $\chi = \chi_{\mathcal{P}_1} = \chi_{\odot} - \delta \Rightarrow$ Parabola $\mathcal{P}_1$ ($e = 1$).
- If $\chi \in]\chi_{\odot} - \delta, \chi_{\odot} + \delta[\Rightarrow$ Subfamily of ellipses ($e < 1$).
- If $\chi = \chi_{\mathcal{P}_2} = \chi_{\odot} + \delta \Rightarrow$ Parabola $\mathcal{P}_2$ ($e = 1$).
- If $\chi \in]\chi_{\odot} + \delta, \pi[\Rightarrow$ Subfamily of hyperbolas II ($e > 1$).

The domain for hyperbolas II is initially defined by $\chi \in]\chi_{\odot} + \delta, \chi_{\odot} + \frac{\pi}{2}[$ but not the entire interval is valid. Indeed, the semilatus rectum p given by Eq. (4.52) is zero when $\chi = \pi$ (the solution $\chi = 0$ is never reached for the conics of this family). When $b > 1$ then $\chi_{\odot} > \frac{\pi}{2}$ and the value $\chi = \pi$ is in the interval $] \chi_{\odot} + \delta, \chi_{\odot} + \frac{\pi}{2}[$ corresponding to the subfamily of hyperbolas II. The family of conics finishes in $\chi = \pi$ and in the remaining interval $] \pi, \chi_{\odot} + \frac{\pi}{2}[$ the expressions previously derived result in negative values for p and a . These are spurious solutions where the origin O is the empty focus of the hyperbola and not the occupied focus.

The situation when $b < 1$ is:

- If $\chi \in]\chi_{\odot} + \delta, \chi_{\odot} - \frac{\pi}{2}[\Rightarrow$ Subfamily of hyperbolas I ($e > 1$).
- If $\chi = \chi_{\mathcal{P}_1} = \chi_{\odot} + \delta \Rightarrow$ Parabola $\mathcal{P}_1$ ($e = 1$).
- If $\chi \in]\chi_{\odot} - \delta, \chi_{\odot} + \delta[\Rightarrow$ Subfamily of ellipses ($e < 1$).
- If $\chi = \chi_{\mathcal{P}_2} = \chi_{\odot} - \delta \Rightarrow$ Parabola $\mathcal{P}_2$ ($e = 1$).
- If $\chi \in]\pi, \chi_{\odot} - \delta[\Rightarrow$ Subfamily of hyperbolas II ($e > 1$).

The domain for hyperbolas II is initially defined by $\chi \in]\chi_{\odot} - \frac{\pi}{2}, \chi_{\odot} - \delta[$ but not the entire interval is valid. Indeed, when $b < 1$ then $\chi_{\odot} < \pi$ and the value $\chi = \pi$ is in the interval $] \chi_{\odot} - \frac{\pi}{2}, \chi_{\odot} - \delta[$ corresponding to the subfamily of hyperbolas II. The family of conics finishes in $\chi = \pi$ and in the remaining interval $] \chi_{\odot} - \frac{\pi}{2}, \pi$

Figure 4.8: Evolution of the hodograph of the eccentricity vector as a function of χ for the orbits of the family. Upper $b > 1$, lower $b < 1$.

the expressions previously derived result in negative values for p and a , and consequently spurious solutions where the origin O is the empty focus of the hyperbola.

This explains the lack of self-conjugate hyperbolas. The domain where they are defined is $|\chi_* - \chi_\odot| > \delta$ and taking into account the domain of both subfamilies of hyperbolas, it is only satisfied within the spurious domain.

The next examples describe the parametric evolution of different subfamilies involved in a family of conics. For this issue, dimensionless variables are used by introducing characteristic values for longitude and time such that the gravitational parameter μ is unitary.

4.5.1 Family of conics with $b > 1$

Let $\vec{r}_1 = \frac{1}{2}\vec{l}_1$ and $\vec{r}_2 = 2\vec{j}_1$ be the position vectors, such that the transfer angle is $\nu = 90^\circ$ and the parameter $b = 4 > 1$. The value of $\chi_\odot$, which only depends on ν and b , takes the value $\chi_\odot \approx 120.96^\circ$. The fundamental eccentricity is $e_\odot \approx 0.72761$ and the parameter characterizing the parabolic orbits is given by $\delta \approx 43.31^\circ$. A

Figure 4.9: Subfamily of hyperbolas I when $b > 1$.

scanning of the variable χ , starting in the lower end of its interval of existence, is developed and presented in the plane Ox_0y_0 .

- Subfamily of hyperbolas I: when χ changes within the interval $\chi \in]\chi_{\odot} - \frac{\pi}{2}, \chi_{\odot} - \delta[$ the hyperbolas obtained are shown in Fig. 4.9. At the lower end of the interval, $\chi = \chi_{\odot} - \frac{\pi}{2}$, it has a degenerate hyperbola given by a double straight line (the line connecting the extreme positions P_1 and P_2 of the space triangle). This degenerate hyperbola is a singularity with $a = 0$ and $e = \infty$. When the value of χ increases, the semi-major axis a increases and the eccentricity decreases. This subfamily ends in the parabola $\mathcal{P}_1$ when $\chi \Rightarrow \chi_{\odot} - \delta$, such that $e = 1$ and $a = \infty$. The transfers done with the orbits of this family are prograde or direct, that is, they have positive angular momentum (according to $\vec{k}_0$). They are the fastest to go from P_1 to P_2 with a prograde transfer. Indeed, close to the degenerate hyperbola the velocity tends to infinity, while it decreases when approaching to the limit parabola $\mathcal{P}_1$. The conjugate hyperbola associated to each of the hyperbolas of this subfamily belongs to the subfamily of hyperbolas II and vice versa.
- Subfamily of parabolas: there are only two parabolas in the family (see Fig. 4.10). In this case the parameter χ takes the values $\chi = \chi_{\mathcal{P}_1} = \chi_{\odot} - \delta$ (parabola $\mathcal{P}_1$) and $\chi = \chi_{\mathcal{P}_2} = \chi_{\odot} + \delta$ (parabola $\mathcal{P}_2$). For the parabolas (see Fig. 4.10) the angular momentum has to be pointed according to $\vec{k}_0$; as a consequence, in the parabola $\mathcal{P}_1$ the transfer from $\vec{r}_1$ to $\vec{r}_2$ is direct. However, in the parabola $\mathcal{P}_2$ the prograde transfer from $\vec{r}_1$ to $\vec{r}_2$ is done through the infinite point of the parabola and consequently requires an infinite time. However, it is possible to pass from P_1 to P_2 through a retrograde transfer with a finite time of flight. The χ values of both parabolas verify Eq. (4.62). Indeed, when $\chi \Rightarrow \chi_{\odot} \pm \delta$ the variable $\rho \Rightarrow \infty$ and the equation is reduced to the identity $\tan(\delta) \equiv \tan(\delta)$. This means that the parabola $\mathcal{P}_2$ is the conjugate parabola of $\mathcal{P}_1$ and vice versa. The parabola $\mathcal{P}_1$ is the limit orbit where the subfamily of hyperbolas I tends when $\chi \Rightarrow \chi_{\odot} - \delta$. From this parabola arises the subfamily of ellipses.
- Subfamily of ellipses: when χ varies within the interval $] \chi_{\odot} - \delta, \chi_{\odot} + \delta [$, the subfamily of ellipses is obtained (see Fig. 4.11). This subfamily starts from the parabola $\mathcal{P}_1$ plotted in Fig. 4.10, while the parabola $\mathcal{P}_2$ is the limit orbit where the ellipses of the subfamily tend when $\chi \Rightarrow \chi_{\odot} + \delta$. The ellipses can be divided in two subfamilies:

Figure 4.10: Parabolas of the family when $b > 1$. Figure 4.11: Family of elliptic orbits when $b > 1$.

- If $\chi \in]\chi_{\odot} - \delta, \chi_*[\Rightarrow$ Subfamily of ellipses I.
- If $\chi \in]\chi_*, \chi_{\odot} + \delta[\Rightarrow$ Subfamily of ellipses II.

By increasing the value of χ from the lower end $\chi_{\odot} - \delta$ the semi-major axis $a(\chi)$ monotonically decreases until it reaches the minimum value a_* corresponding to the self-conjugate ellipse. From the value χ_* it enters in the subfamily of ellipses II, such that the semi-major axis $a(\chi)$ monotonically increases when increasing the value of χ . Both subfamilies are separated by the ellipse of minimum-energy (self-conjugate ellipse), corresponding to the minimum value of a in the family. Consequently, the parabola $\mathcal{P}_1$ separates the subfamily I of hyperbolas and ellipses and the parabola $\mathcal{P}_2$ separates the subfamily II of hyperbolas and ellipses. Besides, every ellipse of the subfamily I has its conjugate in the subfamily II and vice versa.

- Subfamily of hyperbolas II: this subfamily starts from the parabola $\mathcal{P}_2$. The parameter χ takes values within the interval $\chi \in]\chi_{\odot} + \delta, \pi[$ (see Fig. 4.12). This subfamily corresponds to transfers through the infinite right line (considering the hypothesis that the angular momentum has the sense of $\vec{k}_0$). Thus, to go from $\vec{r}_1$ to $\vec{r}_2$ the infinite point of the hyperbola should be visited. As it was explained before, the upper limit of the parameter χ for this family is the minimum of the values $(\chi_{\odot} + \frac{\pi}{2}, \pi)$; when $\chi = \pi$ a degenerate hyperbola, given by the product of the straight lines of the radio vectors $\vec{r}_1$ and $\vec{r}_2$, is obtained.

Figure 4.12: Subfamily of hyperbolas II when $b > 1$.

Table 8: Transfers allowed depending on the family of orbits considered.

Transfers		
	Prograde	Retrograde
Subfamily of hyperbolas I	Yes	No
Parabola $\mathcal{P}_1$	Yes	No
Subfamily of ellipses	Yes	Yes
Parabola $\mathcal{P}_2$	No	Yes
Subfamily of hyperbolas II	No	Yes

If the value of χ increases above π the solutions computed are spurious due to the origin O , the attractive center, would be the empty focus of these spurious solutions.

Summarizing, the prograde transfer can be made with the subfamily of hyperbolas I, the parabola $\mathcal{P}_1$ and the subfamily of ellipses. The retrograde transfer can be made with the subfamily of ellipses, the parabola $\mathcal{P}_2$ and the subfamily of hyperbolas II (see Table 8). The knowledge of the parameter χ related to the prograde transfer allows to locate the value of $\tilde{\chi}$ related to the conjugate orbit through Eq. (4.62) and the flight times of both transfers. Indeed, if one hyperbola is selected from the subfamily of hyperbolas I, the conjugate hyperbola defined in the subfamily of hyperbolas II provides at the same time the retrograde transfer. The same conclusion is obtained for the transfer through the parabolas, a particular case of the transfer through the hyperbolas of the subfamily I ($\mathcal{P}_1$ is the limit case when $\chi \Rightarrow \chi_{\odot} - \delta$). In the subfamily of ellipses, the location of χ related to the prograde transfer ($\chi < \chi_*$) allows to obtain not only the value of $\tilde{\chi}$ related to the conjugate ellipse ($\tilde{\chi} > \chi_*$), but the flight times of the four possible transfers: the arcs (1) and (3) corresponding to the prograde and retrograde transfers respectively in the ellipse of subfamily I, and the arcs (2) and (4) related to the prograde and retrograde transfers respectively in the conjugate ellipse belonging to the subfamily II.

Figure 4.13 shows the time of flight related to direct transfers from P_1 to P_2 and $b > 1$ as a function of the parameter χ of the conic. Later it will be shown how the flight times T_f are determined. From the lowest time $T_f = 0$ the transfer is done through the straight line $P_1 P_2$ (degenerate hyperbola) which implies a transfer with an infinite velocity. Increasing the value of χ the subfamily of hyperbolas I is

Figure 4.13: Time of flight T_f versus χ when $b > 1$ (prograde transfers).

Figure 4.14: Evolution of the eccentricity and semi-major axis (dimensionless values).

traveled and the value of T_f monotonically increases while the velocities involved in the transfer decreases. This region of the curve finishes when the time of flight reaches the value $T_f^{P_1}$, related to the parabola $\mathcal{P}_1$. After that, the transfer takes place through an ellipse of the subfamily I ($\chi \leq \chi_*$) until the ellipse of minimum-energy is reached ($\chi = \chi_*$). The time T_f^* , associated to the parameter χ_* , is maximum in this subfamily (arc (1)) while the velocity is minimum. Values of T_f larger than T_f^* do not allow transfers through the arc (1) but through the arc (2) related to the conjugate ellipse in the subfamily II. In this region T_f increases without limits due to $T_f \Rightarrow \infty$ when approaching the parabola $\mathcal{P}_2$ ($\chi \Rightarrow \chi_{\mathcal{P}_2}$). Figure 1.25 reflects the monotonically increasing behavior of the function $t_f = T_f(\chi)$ for all the regions. The variations in eccentricity and semi-major axis of the orbits of the family are described in Figure 4.14.

4.5.2 Family of conics with $b < 1$

Let $\vec{r}_1 = 4\vec{i}_1$ and $\vec{r}_2 = 2\vec{j}_1$ be the position vectors, which the transfer angle is $\nu = 90^\circ$ and the parameter $b = \frac{1}{2} > 1$. The value of $\chi_\odot$, which only depends on ν and b , takes the value $\chi_\odot \approx 251.57^\circ$. The fundamental eccentricity is $e_\odot \approx 0.4472$ and the parameter characterizing the parabolic orbits is given by $\delta \approx 63.43^\circ$. A

Figure 4.15: Subfamily of hyperbolas I when $b < 1$. Figure 4.16: Parabolas of the family when $b < 1$.

scanning of the variable χ , starting in the upper end of its interval of existence, is developed and presented in the plane Ox_0y_0 .

- Subfamily of hyperbolas I: when χ changes in the interval $\chi \in]\chi_{\odot} + \delta, \chi_{\odot} + \frac{\pi}{2}[$ the hyperbolas obtained are shown in Fig. 4.15. At the upper end of the interval, $\chi = \chi_{\odot} + \frac{\pi}{2}$, it has a degenerate hyperbola given by a double straight line (the line connecting the extreme positions P_1 and P_2 of the space triangle). This degenerate hyperbola is a singularity with $a = 0$ and $e = \infty$. When the value of χ decreases, the semi-major axis a increases and the eccentricity decreases. This subfamily ends in the parabola $\mathcal{P}_1$ when $\chi \Rightarrow \chi_{\odot} + \delta$, such that $e = 1$ and $a = \infty$. The transfers with the orbits of this family are prograde, that is, they have positive angular momentum (according to $\vec{k}_0$). They are the fastest to go from P_1 to P_2 . Indeed, close to the degenerate hyperbola the velocity tends to infinity, while it decreases approaching to the limit parabola $\mathcal{P}_1$. The conjugate hyperbola associated to each of the hyperbolas of this subfamily belongs to the subfamily of hyperbolas II and vice versa.
- Subfamily of parabolas: there are only two parabolas in the family (see Fig. 4.16). In this case the parameter χ takes the values $\chi = \chi_{\mathcal{P}_1} = \chi_{\odot} + \delta$ (parabola $\mathcal{P}_1$) and $\chi = \chi_{\mathcal{P}_2} = \chi_{\odot} - \delta$ (parabola $\mathcal{P}_2$). For the parabolas (see Fig. 4.10) the angular momentum has to be pointed according to $\vec{k}_0$; as a consequence, in the parabola $\mathcal{P}_1$ the transfer from $\vec{r}_1$ to $\vec{r}_2$ is direct. However, in the parabola

Figure 4.17: Family of elliptic orbits when $b < 1$.

$\mathcal{P}_2$ the transfer from $\vec{r}_1$ to $\vec{r}_2$ is retrograde. The parabola $\mathcal{P}_2$ is the conjugate parabola of $\mathcal{P}_1$ and vice versa. Both parabolas define the endings of the region where the subfamily of ellipses is.

- Subfamily of ellipses: when χ varies within the interval $]\chi_{\odot} - \delta, \chi_{\odot} + \delta[$, the subfamily of ellipses is obtained (see Fig. 4.17). This subfamily starts from the upper end of the interval with the parabola $\mathcal{P}_1$ (see Fig. 4.16). The parabola $\mathcal{P}_2$ is the limit orbit where the ellipses of the subfamily tend when approaching to the lower end, that is $\chi \Rightarrow \chi_{\odot} - \delta$. The ellipses can be divided in two subfamilies:
  - If $\chi \in]\chi_*, \chi_{\odot} + \delta[\Rightarrow$ Subfamily of ellipses I.
  - If $\chi \in]\chi_{\odot} - \delta, \chi_*[\Rightarrow$ Subfamily of ellipses II.

By decreasing the value of χ from the upper end $\chi_{\odot} + \delta$ the semi-major axis $a(\chi)$ monotonically decreases until it reaches the minimum value a_* corresponding to the self-conjugate ellipse. From the value χ_* it enters in the subfamily of ellipses II, such that the semi-major axis $a(\chi)$ monotonically increases when decreasing the value of χ . Both subfamilies are separated by the ellipse of minimum-energy (self-conjugate ellipse), corresponding to the minimum value of a in the family. Consequently, the parabola $\mathcal{P}_1$ separates the subfamily I of hyperbolas and ellipses and the parabola $\mathcal{P}_2$ separates the subfamily II of hyperbolas and ellipses. Besides, every ellipse of the subfamily I has its conjugate in the subfamily II and vice versa.

- Subfamily of hyperbolas II: this subfamily starts from the parabola $\mathcal{P}_2$. The parameter χ takes values in the interval $\chi \in]\pi, \chi_{\odot} - \delta[$ (see Fig. 4.18). This subfamily corresponds to transfers through the infinite right line (considering the hypothesis that the angular momentum has the sense of $\vec{k}_0$). Thus, to pass from $\vec{r}_1$ to $\vec{r}_2$ the infinite point of the hyperbola should be visited. As it was explained before, the lower limit of the parameter χ for this family is the maximum of the values $(\chi_{\odot} - \frac{\pi}{2}, \pi)$; when $\chi = \pi$ a degenerate hyperbola, given by the product of the straight lines of the radio vectors $\vec{r}_1$ and $\vec{r}_2$, is obtained. If the value of χ decreases below π the solutions computed are spurious due to the origin O, and the attractive center, would be the empty focus of these spurious solutions.

The prograde transfer from P_1 to P_2 can be done along different conics: the hyperbola of the subfamily I, the parabola $\mathcal{P}_1$ and the ellipse of the subfamily I or

Figure 4.18: Subfamily of hyperbolas II when $b < 1$. Figure 4.19: Time of flight T_f versus χ when $b < 1$ (prograde transfers).

II. As it was explained for the family of conics with $b > 1$, the flight times of the transfers from P_1 to P_2 can be obtained just locating the time of flight of the direct transfers because the remaining are estimated from them.

Figure 4.19 shows the time of flight related to direct transfers from P_1 to P_2 and $b < 1$ as a function of the parameter χ of the conic. From the lowest time $T_f = 0$ the transfer is done through the straight line P_1P_2 (degenerate hyperbola) which implies a transfer with an infinite velocity. Decreasing the value of χ the subfamily of hyperbolas I is traveled and the value of T_f monotonically increases while the velocities involved in the transfer decreases. This region of the curve finishes when the time of flight reaches the value $T_f^{\mathcal{P}_1}$, related to the parabola $\mathcal{P}_1$. After that, the transfer takes place through an ellipse of the subfamily I ($\chi \leq \chi_*$) until the ellipse of minimum-energy is reached ($\chi = \chi_*$). The time T_f^* , associated to the parameter χ_* , is maximum in this subfamily (arc (1)) while the velocity is minimum. Values of T_f larger than T_f^* do not allow transfers through the arc (1) but through the arc (2) related to the conjugate ellipse in the subfamily II. In this region T_f increases without limits due to $T_f \Rightarrow \infty$ when approaching the parabola $\mathcal{P}_2$ ($\chi \Rightarrow \chi_{\mathcal{P}_2}$). Figure 1.33 reflects the monotonically decreasing behavior of the function $t_f = T_f(\chi)$ for all the regions. The variations in eccentricity and semi-major axis of the orbits of the family are described in Figure 4.20.

Figure 4.20: Evolution of the eccentricity and semi-major axis (dimensionless values)

4.6 THE TIME OF FLIGHT EQUATION

The next step is to determine, as a function of the free parameter χ , the time of flight needed to pass from P_1 to P_2 . Notice that the equations described below are the ones corresponding to direct transfers for the elliptic, parabolic and hyperbolic cases.

4.6.1 Elliptic case

Kepler's equation for the elliptic case

$$n(t - \tau) = u - e \sin(u), \quad (4.74)$$

governs the time law. Therefore, if the orbit is elliptic ($e < 1$), the time of flight is given by the equation

$$T_f = \frac{1}{n}(u_2 - u_1 - e[\sin(u_2) - \sin(u_1)]), \quad (4.75)$$

where $n = \sqrt{\frac{\mu}{a^3}}$ and the eccentric anomaly u is related with the true anomaly θ through the relations

$$\sin(\theta) = \frac{\sqrt{1-e^2} \sin(u)}{1 - e \cos(u)}, \quad \cos(\theta) = \frac{\cos(u) - e}{1 - e \cos(u)}, \quad (4.76)$$

or by the inverse relations

$$\sin(u) = \frac{\sqrt{1-e^2} \sin(\theta)}{1 + e \cos(\theta)}, \quad \cos(u) = \frac{\cos(\theta) + e}{1 + e \cos(\theta)}. \quad (4.77)$$

Equation (4.75) describes the time of flight for a transfer of one revolution. In the case of a transfer requiring more than one revolution it implies that the time of flight is not going to be minimum but $T_f = T_{f_{\min}} + kT$ such that $T = \frac{2\pi}{n}$. Consequently, the general problem for solving the multiple-revolution Lambert's problem becomes

$$T_{f_{\min}} = \frac{1}{n}(u_2 - u_1 - e[\sin(u_2) - \sin(u_1)] - 2\pi k). \quad (4.78)$$

The difference with respect to the initial problem is the introduction of a new parameter k such that for each set of values $(\vec{r}_1, \vec{r}_2, T_{f_{\min}}, k)$ a solution of χ is obtained. The selection of k determines the number of revolutions.

4.6.2 Parabolic case

The time law is governed by Barker's equation

$$2\sqrt{\frac{\mu}{p^3}}(t - \tau) = \tan\left(\frac{\theta}{2}\right) + \frac{1}{3}\tan^3\left(\frac{\theta}{2}\right). \quad (4.79)$$

Thus, if the orbit is parabolic ($e = 1$), the time of flight is given by the equation

$$\begin{aligned} T_f = \frac{1}{2}\sqrt{\frac{p^3}{\mu}} & \left(\left[\tan\left(\frac{\theta_2}{2}\right) - \tan\left(\frac{\theta_1}{2}\right) \right] \right. \\ & \left. + \frac{1}{3} \left[\tan^3\left(\frac{\theta_2}{2}\right) - \tan^3\left(\frac{\theta_1}{2}\right) \right] \right). \end{aligned} \quad (4.80)$$

4.6.3 Hyperbolic case

The time law is governed by the equation

$$\sqrt{\frac{\mu}{|a|^3}}(t - \tau) = e \sinh(u) - u. \quad (4.81)$$

Therefore, if the orbit is hyperbolic ($e > 1$), the time of flight is given by the equation

$$T_f = \frac{1}{n} (e[\sinh(u_2) - \sinh(u_1)] - (u_2 - u_1)), \quad (4.82)$$

where $n = \sqrt{\frac{\mu}{|a|^3}}$ and the hyperbolic anomaly u is related with the true anomaly θ through the relations

$$\sin(\theta) = \frac{\sqrt{e^2 - 1} \sinh(u)}{e \cosh(u) - 1}, \quad \cos(\theta) = \frac{e - \cosh(u)}{e \cosh(u) - 1}, \quad (4.83)$$

or by the inverse relations

$$\sinh(u) = \frac{\sqrt{e^2 - 1} \sin(\theta)}{1 + e \cos(\theta)}, \quad \cosh(u) = \frac{\cos(\theta) + e}{1 + e \cos(\theta)}. \quad (4.84)$$

4.7 ALTERNATIVE FORMULATION

Lambert's theorem states that: *the time of flight t_f needed for going from $\vec{r}_1$ to $\vec{r}_2$ is only a function of the sum $r_1 + r_2$, the cord λ_c and the semi-major axis a of the conic connecting the two points: $t_f = t_f(r_1 + r_2, \lambda_c, a)$.*

It was Lagrange who first demonstrated it for the case of elliptic orbits. The procedure, in broad strokes, is explained in the next lines.

4.7.1 Elliptic case

Let u_1 and u_2 be the values of the eccentric anomaly related to the points P_1 and P_2 respectively. Kepler's equation is given by

$$n(t_2 - \tau) = u_2 - e \sin(u_2) \quad (4.85a)$$

$$n(t_1 - \tau) = u_1 - e \sin(u_1), \quad (4.85b)$$

and subtracting both equations yields

$$n t_f = u_2 - u_1 - e(\sin(u_2) - \sin(u_1)), \quad (4.86)$$

where $t_f = t_2 - t_1$ is the time employed in the transfer and $n = \sqrt{\frac{\mu}{a^3}}$ is the mean angular velocity of the motion.

By introducing the values

$$\Psi_s = \frac{1}{2}(u_1 + u_2) \quad (4.87a)$$

$$\Psi_r = \frac{1}{2}(u_2 - u_1), \quad (4.87b)$$

the eccentric anomalies become $u_1 = \Psi_s - \Psi_r$ and $u_2 = \Psi_s + \Psi_r$ such that Eq. (4.86) is replaced by

$$n t_f = 2(\Psi_r - e \cos(\Psi_s) \sin(\Psi_r)). \quad (4.88)$$

Taking into account that $|e \cos(\Psi_s)| < 1$ for elliptic orbits, we define

$$\cos(\xi) = e \cos(\Psi_s) \quad (4.89a)$$

$$\alpha = \xi + \Psi_r \quad (4.89b)$$

$$\beta = \xi - \Psi_r, \quad (4.89c)$$

with $\xi = \frac{1}{2}(\alpha + \beta)$ and $\Psi_r = \frac{1}{2}(\alpha - \beta)$. If they are introduced into Eq. (4.88), then the following expression is obtained

$$n t_f = \alpha - \beta - (\sin(\alpha) - \sin(\beta)). \quad (4.90)$$

This relation is termed as Lambert's equation and the angles α and β depend on $r_1 + r_2$, λ_c and a . Indeed, the relation $r = a(1 - e \cos(u))$ allows to express

$$r_1 + r_2 = 2a - a e(\cos(u_1) + \cos(u_2)), \quad (4.91)$$

which in terms of Ψ_s and Ψ_r becomes

$$r_1 + r_2 = 2a(1 - \cos(\xi) \cos(\Psi_r)). \quad (4.92)$$

In the axes of the ellipse the positions of P_1 and P_2 are given by

$$x_1 = a \cos(u_1), \quad y_1 = b \sin(u_1) \quad (4.93a)$$

$$x_2 = a \cos(u_2), \quad y_2 = b \sin(u_2), \quad (4.93b)$$

which introduced in the expression of the cord results in $\lambda_c^2 = (x_2 - x_1)^2 + (y_2 - y_1)^2 = 4a^2 \sin^2(\xi) \sin^2(\Psi_r)$ such that

$$\lambda_c = 2a \sin(\xi) \sin(\Psi_r). \quad (4.94)$$

Let s be the semiperimeter of the spatial triangle, in terms of Eqs. (4.92) and (4.94) it turns out to be $2s = (r_1 + r_2 + \lambda_c) = 2a(1 - \cos(\alpha))$. That is, the following relation is obtained:

$$\sin\left(\frac{\alpha}{2}\right) = \pm \sqrt{\frac{s}{2a}}. \quad (4.95)$$

Likewise it has $2(s - \lambda_c) = 2a(1 - \cos(\beta))$ and consequently

$$\sin\left(\frac{\beta}{2}\right) = \pm \sqrt{\frac{s - \lambda_c}{2a}}. \quad (4.96)$$

In summary, the values α and β only depend on $r_1 + r_2$, λ_c and a . Taking into account this, the time of flight t_f provided by Eq. (4.90), only depends on $r_1 + r_2$, λ_c and a . This demonstrates Lambert's theorem.

Figure 4.21: Conjugate elliptic orbits.

4.7.2 Geometric aspects of α and β

Once the value of a is fixed, and greater than the minimum value a_* given by Eq. (4.11), the solution of Eqs. (4.95) and (4.96) is not univocal. Indeed, depending on the signs $\pm$ of α and β four different possibilities are present, which are associated to the four possible transfers to go from P_1 to P_2 through two conjugate orbits with the same value of a : the arcs (1) and (2) describing the prograde transfers the arcs (3) and (4) describing the retrograde transfers (see Fig. 4.21).

It starts considering the case where α and β are positive:

$$\alpha = \alpha_0 = 2 \arcsin \left(\sqrt{\frac{s}{2a}} \right), \quad \alpha_0 \in [0, \pi] \quad (4.97a)$$

$$\beta = \beta_0 = 2 \arcsin \left(\sqrt{\frac{s - \lambda_c}{2a}} \right), \quad \beta_0 \in [0, \pi]. \quad (4.97b)$$

According to Lambert's theorem, the time of flight t_f does not depend on the eccentricity of the transfer orbit. Consequently, Lambert's equation (4.90) is still true if the eccentricity of the ellipse is modified without changing the values of $r_1 + r_2$, λ_c and a . This can be done by changing the foci f and f^* of the ellipse until the value $e = 1$ when a degenerate ellipse (rectilinear) is obtained. A rectilinear ellipse with semi-major axis a is a line segment of longitude $2a$ with ends the foci f and f_* of the ellipse (see Fig. 4.22). In this scenario $r_1 = q_1$ and $r_2 = q_1 + \lambda_c$ and since their sum remains constant, it yields:

$$r_1 + r_2 = 2q_1 + \lambda_c \Rightarrow q_1 = s - \lambda_c, \quad (4.98)$$

and considering the relation $|ff^*| = 2a$ it follows

$$q_2 + \lambda_c + q_1 = 2a \Rightarrow q_2 = 2a - s. \quad (4.99)$$

The center C of the ellipse is located at a distance d from P_2 given by $d = a - q_2 = s - a$.

Equations (4.86) and (4.90) have the same structure. If the time of flight t_f is the same for both, the second equation corresponds to a rectilinear orbit with $e = 1$ such that α and β become the eccentric anomalies of the departure and arrival points of an orbital transfer. By considering the auxiliary circumference associated

Figure 4.22: Rectilinear ellipse.

Figure 4.23: Prograde transfer (1) (upper left), prograde transfer (2) (upper right), retrograde transfer (3) (lower left) and retrograde transfer (4) (lower right).

to the rectilinear ellipse, the values α_0 and β_0 are the eccentric anomalies of the points Q_2 and Q_1 respectively (see Fig. 4.23). Therefore, the transfer from P_1 to P_2 in the rectilinear orbit is associated to Lambert's equation (4.90). Indeed, from the figure the following is deduced:

$$\cos(\beta_0) = \frac{a - q_1}{a} \Rightarrow \sin^2\left(\frac{\beta_0}{2}\right) = \frac{s - \lambda_c}{2a} \quad (4.100)$$

$$\cos(\pi - \alpha_0) = \frac{a - q_2}{a} \Rightarrow \sin^2\left(\frac{\alpha_0}{2}\right) = \frac{s}{2a}. \quad (4.101)$$

The difference $\alpha - \beta$ is always $u_2 - u_1$ regardless of where the angles α and β are placed. In what follows the time of flight for the arcs (1), (2), (3) and (4) is determined.

- Prograde transfer (1): the transfer angle ν is smaller than π . The values α and β are α_0 and β_0 respectively given by Eqs. (4.97). Figure 4.23 shows the situation of the points Q_1 and Q_2 of the rectilinear ellipse. The time of flight turns out to be:

$$n t_f^{(1)} = \alpha_0 - \beta_0 - \sin(\alpha_0) + \sin(\beta_0). \quad (4.102)$$

- Prograde transfer (2): the transfer angle ν is smaller than π . The angle β is given by the solution (4.96) with sign (+) and consequently $\beta = \beta_0$. In the case of α the sign (-) is taken in Eq. (4.95). Taking into account that $\alpha - \beta = u_2 - u_1$ is positive, the value of α can not be negative and $\alpha = 2\pi - \alpha_0$. Figure 4.23 shows the situation of the points Q_1 and Q_2 of the rectilinear ellipse. The time of flight turns out to be:

$$n t_f^{(2)} = 2\pi - \alpha_0 - \beta_0 + \sin(\alpha_0) + \sin(\beta_0). \quad (4.103)$$

- Retrograde transfer (3): the transfer angle ν is larger than π . The angle β is given by the solution (4.96) with sign (-) and consequently $\beta = -\beta_0$. In the case of α the sign (-) is taken in Eq. (4.95) and $\alpha = 2\pi - \alpha_0$. Figure 4.23 shows the situation of the points Q_1 and Q_2 of the rectilinear ellipse. The time of flight turns out to be:

$$n t_f^{(3)} = 2\pi - \alpha_0 + \beta_0 + \sin(\alpha_0) - \sin(\beta_0). \quad (4.104)$$

- Retrograde transfer (4): the transfer angle ν is larger than π . The angle β is given by the solution (4.96) with sign (-) and consequently $\beta = -\beta_0$. In the case of α the sign (+) is taken in Eq. (4.95) and $\alpha = \alpha_0$. Figure 4.23 shows the situation of the points Q_1 and Q_2 of the rectilinear ellipse. The time of flight turns out to be:

$$n t_f^{(4)} = \alpha_0 + \beta_0 - \sin(\alpha_0) - \sin(\beta_0). \quad (4.105)$$

4.7.3 Hyperbolic case

Let u_1 and u_2 be the values of the hyperbolic anomaly related to the points P_1 and P_2 respectively. Kepler's equation for hyperbolic orbits is given by

$$n(t_2 - \tau) = e \sinh(u_2) - u_2 \quad (4.106a)$$

$$n(t_1 - \tau) = e \sinh(u_1) - u_1, \quad (4.106b)$$

and subtracting both equations yields

$$n t_f = e(\sinh(u_2) - \sinh(u_1)) - (u_2 - u_1), \quad (4.107)$$

where $t_f = t_2 - t_1$ is the time employed in the transfer and $n = \sqrt{\frac{\mu}{a^3}}$ is the mean angular velocity of the motion.

By introducing the values

$$\Psi_s = \frac{1}{2}(u_1 + u_2) \quad (4.108a)$$

$$\Psi_r = \frac{1}{2}(u_2 - u_1), \quad (4.108b)$$

the hyperbolic anomalies become $u_1 = \Psi_s - \Psi_r$ and $u_2 = \Psi_s + \Psi_r$ such that Eq. (4.107) is replaced by

$$n t_f = 2(e \cosh(\Psi_s) \sinh(\Psi_r) - \Psi_r). \quad (4.109)$$

Taking into account that $|e \cosh(\Psi_s)| > 1$ for hyperbolic orbits, $\cosh(\xi) = e \cosh(\Psi_s)$ is defined and the values

$$\alpha = \xi + \Psi_r \quad (4.110a)$$

$$\beta = \xi - \Psi_r, \quad (4.110b)$$

Figure 4.24: Conjugate hyperbolic orbits.

with $\xi = \frac{1}{2}(\alpha + \beta)$ and $\Psi_r = \frac{1}{2}(\alpha - \beta)$ are introduced into Eq. (4.109). This yields

$$n t_f = (\sinh(\alpha) - \sinh(\beta)) - (\alpha - \beta). \quad (4.111)$$

This relation is termed as Lambert's equation and the angles α and β depend on $r_1 + r_2$, λ_c and a . Indeed, the relation $r = a(e \cosh(u) - 1)$ allows to express

$$r_1 + r_2 = a e (\cosh(u_1) + \cosh(u_2)) - 2a, \quad (4.112)$$

which in terms of Ψ_s and Ψ_r becomes

$$r_1 + r_2 = 2a (\cosh(\xi) \cosh(\Psi_r) - 1). \quad (4.113)$$

In the axes of the hyperbola the positions of P_1 and P_2 are given by

$$x_1 = a \cosh(u_1), \quad y_1 = b \sinh(u_1), \quad (4.114a)$$

$$x_2 = a \cosh(u_2), \quad y_2 = b \sinh(u_2), \quad (4.114b)$$

which introduced in the expression of the cord results in $\lambda_c^2 = (x_2 - x_1)^2 + (y_2 - y_1)^2$. By operating to obtain in terms of α and β it becomes

$$\lambda_c = 2a (\cosh(\alpha) - \cosh(\beta)), \quad (4.115)$$

where it is admitted that $\alpha > \beta$ because $\lambda_c > 0$.

Let s be the semiperimeter of the spatial triangle, in terms of α and β it turns out to be $s = a(\cosh(\alpha) - 1)$. That is, the following relation is obtained:

$$\cosh\left(\frac{\alpha}{2}\right) = \sqrt{\frac{s + 2a}{2a}}. \quad (4.116)$$

Likewise it has $s - \lambda_c = a(\cosh(\beta) - 1)$ and consequently

$$\cosh\left(\frac{\beta}{2}\right) = \sqrt{\frac{s + 2a - \lambda_c}{2a}}. \quad (4.117)$$

Equation (4.116) has two roots: $\alpha_0 > 0$ and the opposite. The same happens for Eq. (4.117). This provides four possible solutions:

$$(\alpha_0, \beta_0), \quad (\alpha_0, -\beta_0), \quad (-\alpha_0, \beta_0), \quad (-\alpha_0, -\beta_0). \quad (4.118)$$

Only the first two solutions verify the condition $\alpha > \beta$, which provide two flight times

$$n t_{f_1} = \sinh(\alpha_0) - \sinh(\beta_0) - (\alpha_0 - \beta_0) \quad (4.119a)$$

$$n t_{f_2} = \sinh(\alpha_0) + \sinh(\beta_0) - (\alpha_0 + \beta_0). \quad (4.119b)$$

The time t_{f_1} is smaller than t_{f_2} and corresponds to the time of flight needed to go from P_1 to P_2 with a prograde transfer through the arc (1) of an hyperbola of the subfamily I. The time t_{f_2} is the time of flight needed to go from P_1 to P_2 with a retrograde transfer through the arc (4) of the conjugate hyperbola of the previous hyperbola and defined in the subfamily II. Figure 4.24 shows these transfers. The second transfer is slower than the first because the trajectory should circle the attractive center.

Part II

ORBIT PROPAGATION

In Astrodynamics, the typical scenario for the basic perturbed problem is the following: a particle M with mass m moves with respect to an inertial frame $Ox_1y_1z_1$ with O the center of mass of a solar system body (the Earth, the Sun, a planet, etc) and such that the particle M is acted upon by: 1) the main term of the gravitational attraction of that body; and 2) the remaining forces, which are accounted for as a perturbing force $\vec{F}_p$. Instead of the resulting perturbing force $\vec{F}_p$ the perturbing acceleration $\vec{a}_p = \frac{\vec{F}_p}{m}$ is usually used. The motion is then governed by the vector differential equation:

$$\frac{d^2\vec{x}}{dt^2} = -\frac{\mu}{|\vec{x}|^3}\vec{x} + \vec{a}_p, \quad (5.1)$$

that should be integrated from known initial conditions:

$$\text{at } t = t_0 : \quad \vec{x}(t_0) = \vec{x}_0, \quad \frac{d\vec{x}}{dt}(t_0) = \vec{v}_0, \quad (5.2)$$

where t_0 is the date when the propagation starts, $\vec{x}_0$ is the initial position in inertial frame coordinates and $\vec{v}_0$ is the initial velocity in inertial frame coordinates.

The integration of the perturbed problem can be carried out by three different techniques:

- Analytical integration: it seeks to obtain an analytical solution of the problem; the solution is general and therefore is valid regardless of the given initial conditions. It is the least frequent, as often problems are not analytically integrable and thus need to be approximated in series expansions. Usually, additional procedures are used to obtain approximate analytical solutions such as asymptotic techniques, the averaging technique etc.
- Numerical integration: it approaches the problem from a pure numerical point of view, such that the equations of motions are numerically integrated from the prescribed initial conditions and accounting for given perturbing accelerations. The obtained solution is particular to the given initial conditions, and therefore changing the initial conditions requires that a new numerical solution must be computed.
- Semianalytical integration: it combines a partial analytical integration with a numerical integration of those equations that could not be analytical integrated.

In the field of orbital dynamics and celestial mechanics, the procedures employed to numerically integrate the equations of a perturbed problem are called special perturbation methods while the procedures used in the analytical and semianalytical integration are called general perturbation methods. The advantage of the general methods is the possibility to draw conclusions of general validity. However, some of the hypothesis used to simplify the equations are not rigorous and the accuracy is deteriorated. In general it is limited in time. In the case of the special methods the main advantage is the the precision they are able to achieve. However, they do not allow to draw conclusions of general validity. If the perturbing acceleration is a complicated function, the analytical integration of the Eqs. (5.1) becomes impossible and consequently, the numerical integration from the known initial conditions should be applied.

The variety of special perturbation methods is abundant and the choice of the appropriate method to be used in each case depends on the following factors:

- The type of orbit to integrate: circular, low eccentricity, high eccentricity, parabolic or hyperbolic, weakly perturbed or perturbed.
- The operational requirements of the mission: depending on the mission the precision and accuracy involved in the calculation can vary as well as the extension of the integration.
- The formulation of the equations of the motion: the order of the system can vary from sixth order to a higher order system.
- The routines and algorithms employed in the integration process: if all the equations are first order differential equations, an extensive number of fixed or variable stepsize/order routines and single-step or multi-step routines can be used. However, for second order equations it can be more appropriate to use specific routines like Runge-Kutta-Nyström or Störmer-Cowell.
- The benefits of the computer equipment where the simulation is going to be performed.

5.1 NUMERICAL INTEGRATION

The numerical integration looks for a numerical solution to a physical-mathematical problem where the error involved needs to be known in order to accept the solution. The area of numerical calculus is continually moving forwards thanks to the advances in computers and the increase of computing power.

5.1.1 Errors and integration methods

Lets us assume, without loss of generality, that the dynamical problem is described by a second order differential equation:

$$\frac{d^2\vec{x}}{dt^2} = \vec{f}(\vec{x}, t), \quad \vec{x}(0) = \vec{x}_0, \quad \frac{d\vec{x}}{dt}(0) = \vec{v}_0, \quad (5.3)$$

where $\vec{x}$ is the state vector and $\vec{v}_0$ the initial velocity. Any numerical integration algorithm can be expressed by a sequence of values $\vec{x}_n$ of the state vector approximating the solution given by Eq. (5.3) at the times t_n with $t_n = t_{n-1} + h_n$. The term h_n is the integration stepsize which can be fixed or variable. For each step the numerical integration algorithm is a difference scheme:

$$\vec{x}_{n+1} = F(\vec{x}_n, \vec{x}_{n-1}, \vec{x}_{n-2}, \dots, \vec{x}_{n-m}, t_n), \quad (5.4)$$

where m is an integer defined in $[0, n]$ and F is the characteristic function of the algorithm.

The numerical integration requires to repeat Eq. (5.4) iteratively until the end. However, this procedure is affected by a set of integration errors (Milani and Nobili [54]) whose sources are the following:

- Truncation error: this error is the difference between the exact solution $\vec{x}(t_{n+1})$ of Eq. (5.3) and the approximate value $\vec{x}_{n+1}$ as computed by Eq. (5.4). The local truncation error is the error resulting from the use of Eq. (5.4) even in the assumption that the previous values used are exact, i.e. $\vec{x}_j = \vec{x}(t_j)$ for $j \leq n$.

- Round-off error: it is the error done by the computer in computing the function F of Eq. (5.4). Conceptually, a numerical integration works with real numbers, i.e. with infinite decimal figures. However, a computer approximate a real number by a rational number with a finite number of decimal digits. The round-off error is of the order of the machine accuracy in relative terms for each floating point operation.
- Coefficient error: this error arises because the function F of Eq. (5.4) has an analytic expression containing not only the previous values $\bar{x}_{n-s}$, the function f of Eq. (5.3) and the stepsize h , but also some constant coefficients which have to be chosen in such a way that the local truncation error is minimized for a given h . This amounts to a form of round-off but its effect is more like truncation due to the use of wrong coefficients making the truncation error not minimal.
- Time advance error: the time t must be updated after each step. If it is considered by a sum, the operation $t_{n+1} = t_n + h$ may introduce a relative error because of round-off. If ϵ is the machine accuracy, after N steps the accumulated error is

$$\Delta t_N = \sum_{n=1}^N \frac{-\epsilon t_n}{2} = -\frac{\epsilon h}{4} N(N+1). \quad (5.5)$$

- Errors by coordinate changes: in order to reduce other sources of error the integration can be performed in a different way than Eq. (5.4). The set of variables to be propagated from one step to the next does not include only the state vector $\bar{x}$ and the time t , but also additional hidden variables $\bar{y}$. If the transformation between both have to be computed at each step, then the computation contains errors. This errors come from the round-off or imperfect convergence effects, or both.
- Errors in the initial conditions: if the accumulated effect of the error in the initial conditions is larger than the accuracy sought, then the prediction of the real position of the body becomes useless. However, in most of the cases the aim is to investigate the stability of the system by a very long integration. In this case, unless the stability domains are very small, a stable orbit remains stable even after a small change in initial conditions.
- Errors in the physical model: the equation of motion (5.3) are only an approximation to the real physical system. The reason may be due to the existence of further conservative perturbations which have been neglected, to the poor knowledge of some system parameters or to true non-conservative perturbations which have been discounted.

The optimal integration stepsize h_0 should be chosen taking into account that the truncation error usually decreases with the integration stepsize h , while the round-off error increases with the number of evaluations, that is, when h decreases. Accordingly, h_0 will be the stepsize where the sum of these errors is minimal. Numerical integration methods can be traditionally divided into single-step and multi-step categories. The multi-step methods usually require a single-step method to initialize them at the beginning. Some of these methods are Milne, Adams-Moulton, Gauss-Jackson, Obrechhoff and Adams-Bashforth. In the case of single-step methods we can highlight Runge-Kutta, Gill, Euler-Cauchy and Bowie among others.

There is no a general rule to deduce h_0 , however, there are good procedures that enable to select an appropriate integration stepsize. The basic formulation of the equations of motion is going to play an important role in this point.

5.1.2 Instabilities in the Keplerian motion

The errors committed in the determination of the future position of a satellite depend on the truncation and round-off errors, that is, the errors produced in the determination of the initial position. As a consequence, the errors follow a monotonous growth with time in the direction tangent to the orbit. Thus, although the numerical integration process is stable, the determination of the position of the satellite is affected.

The Keplerian motion is unstable in the sense of Lyapunov, that is, small differences in the semi-major axis of two orbits result in a separation that grows in time because of having different periods (Roa [72]). However, if the Keplerian orbit is slightly perturbed, the resulting orbit is a Keplerian orbit close to the given, that is, it shows orbital stability. It follows that the instability in the sense of Lyapunov is time-varying but does not affect the trajectory. Nevertheless, depending on the equations to be integrated, the instability can also affect the orbit.

5.2 CLASSICAL METHODS

5.2.1 Cowell's method

This method was developed by P. H. Cowell (Bate, Mueller, and White [8]) in the early 20th century to estimate the orbit of the known eight satellites of Jupiter as well as the Halley comet. It performs the numerical integration of the perturbed equations by using cartesian coordinates, usually referred to an inertial frame. To do that, the second order equation (5.1) is reduced to first order differential equations:

$$\frac{d\vec{x}}{dt} = \vec{v}, \quad \frac{d\vec{v}}{dt} = -\frac{\mu}{|\vec{x}|^3} \vec{x} + \vec{a}_p, \quad (5.6)$$

where $\vec{x} = (x_1, y_1, z_1)$ and $\vec{v} = (v_{x_1}, v_{y_1}, v_{z_1})$ are the position and the velocity of a satellite with respect to the central body.

This vector system is now broken down into the scalar components

$$\frac{dx_1}{dt} = v_{x_1}, \quad \frac{dv_{x_1}}{dt} = -\frac{\mu}{r^3} x_1 + a_{p_{x_1}}, \quad (5.7a)$$

$$\frac{dy_1}{dt} = v_{y_1}, \quad \frac{dv_{y_1}}{dt} = -\frac{\mu}{r^3} y_1 + a_{p_{y_1}}, \quad (5.7b)$$

$$\frac{dz_1}{dt} = v_{z_1}, \quad \frac{dv_{z_1}}{dt} = -\frac{\mu}{r^3} z_1 + a_{p_{z_1}}, \quad (5.7c)$$

where $r = \sqrt{x_1^2 + y_1^2 + z_1^2}$ and $\vec{a}_p = (a_{p_{x_1}}, a_{p_{y_1}}, a_{p_{z_1}})$ is the perturbing acceleration. Applying one of the available numerical integration schemes to the system of equations (5.7a), (5.7b) and (5.7c), the state vector is obtained.

Nowadays, the Störmer-Cowell method, a combination of the Cowell formulation with a Störmer-Cowell integration routine, is employed to integrate second order equations. In this case the equation system is the following

$$\frac{d^2 x_1}{dt^2} = -\frac{\mu}{r^3} x_1 + a_{p_{x_1}} \quad (5.8a)$$

$$\frac{d^2 y_1}{dt^2} = -\frac{\mu}{r^3} y_1 + a_{p_{y_1}} \quad (5.8b)$$

$$\frac{d^2 z_1}{dt^2} = -\frac{\mu}{r^3} z_1 + a_{p_{z_1}}. \quad (5.8c)$$

This algorithm reduces the number of function calls which is very convenient when we have massive propagations.

Figure 5.1: Osculating orbit of the Encke method.

5.2.2 Encke's method

This method was introduced by J. F. Encke (Bate, Mueller, and White [8]) at the beginning of the 21st century to calculate orbits of comets with eccentricity close to unit.

Let M be the body whose trajectory we want to calculate. If the perturbing acceleration $\vec{a}_p$ is eliminated at t_0 , M will follow the osculating orbit Γ_0 determined by the values $(t_0, \vec{x}_0, \vec{v}_0)$ (see Fig. 5.1). The position of the particle on the osculating orbit is analytically determined at $t > t_0$ without truncation errors. From that instant two particles are distinguished: the particle B , moving along the osculating orbit Γ_0 and located by the position vector $\vec{x}_b$, and the particle M , moving along the real trajectory which is intended to be obtained and located by the position vector $\vec{x}$ (see Fig. 5.1). Encke's method is based on the numerical integration of the equation governing the time evolution of $\vec{\rho}$, that is, the vector placing M with respect to B : $\vec{\rho} = \vec{x} - \vec{x}_b$.

The evolution of the particles M and B is governed, respectively, by the equations:

$$\frac{d^2 \vec{x}}{dt^2} = -\frac{\mu}{|\vec{x}|^3} \vec{x} + \vec{a}_p \quad (5.9a)$$

$$\frac{d^2 \vec{x}_b}{dt^2} = -\frac{\mu}{|\vec{x}_b|^3} \vec{x}_b. \quad (5.9b)$$

Therefore, the time evolution of $\vec{\rho}$ is given by

$$\frac{d^2 \vec{\rho}}{dt^2} = -\frac{\mu}{r_b^3} [\vec{\rho} + f(\Lambda) \vec{x}] + \vec{a}_p(t, \vec{x}, \vec{v}), \quad (5.10)$$

where $r_b = |\vec{x}_b|$, $r = |\vec{x}|$ and $f(\Lambda) = (r_b^3/r^3 - 1)$. To minimize numerical errors that can appear when calculating $f(\Lambda)$, it is rewritten as follows

$$f(\Lambda) = \Lambda \frac{3 + 3\Lambda + \Lambda^2}{(1 + \Lambda)^{\frac{3}{2}} + 1}, \quad \Lambda = \frac{\vec{\rho}(\vec{\rho} - 2\vec{x})}{\vec{x} \cdot \vec{x}}. \quad (5.11)$$

The integration scheme consists in the following steps:

1. The values $(\vec{x}_b, \vec{v}_b)$ are estimated at the instant t with the analytic solution and the known initial conditions $(t_0, \vec{x}_0, \vec{v}_0)$.

2. Equation (5.10) is numerically integrated from $t = t_0$ with the initial conditions $\vec{p} = \vec{0}$, $\dot{\vec{p}} = \vec{0}$.
3. The rectification process is applied when $|\vec{p}| \gg |\vec{a}_p|$. In this case the integration is stopped and the osculating orbit is rectified, such that the reference orbit Λ_0 is replaced by the new orbit Λ_1 .
4. The scheme is repeated as many times as necessary to finish the simulation.

5.2.3 Error propagation

Lets us consider a known orbit $\vec{x} = \vec{x}_R(t)$ and we are going to assume that we make a small error in the initial conditions ($t = t_0$) given by:

$$\vec{x} = \vec{x}_0 + \delta\vec{x}_0, \quad \vec{v} = \vec{v}_0 + \delta\vec{v}_0. \quad (5.12)$$

The trajectory corresponding to these new initial conditions will be close to the Keplerian orbit. It means, the solution can be considered like

$$\vec{x} = \vec{x}_R(t) + \delta\vec{x}, \quad (5.13)$$

where $\delta\vec{x}$ is a small value compared with $\vec{x}_R(t)$.

The variational equations that govern the time evolution of $\delta\vec{x}$ are rewritten as a system of first order differential equations by introducing the state vector

$$\vec{y} = (\delta x_1, \delta y_1, \delta z_1, \delta \dot{x}_1, \delta \dot{y}_1, \delta \dot{z}_1)^T, \quad (5.14)$$

such that its time evolution is given by

$$\frac{d\vec{y}}{dt} = A(t)\vec{y}, \quad (5.15)$$

where A is a square matrix 6×6 .

The solution of the linear variational equations for small period of time is:

$$\vec{y}(\Delta t) = e^{A \Delta t} \vec{y}(0). \quad (5.16)$$

As a result of this, the eigenvalues of the matrix A govern the time evolution of the errors. In the case of Cowell's method this propagation error is exponential and it is related to the unstable character in the Lyapunov sense which appears not only in Keplerian orbits but also in the propagation of perturbed problems. A similar situation arises in Encke's method.

5.3 THE REGULARIZED PROPAGATORS

The regularization process is fundamental for the precise integration of the equations of orbital motion. Indeed, if we consider the Eq. (5.1), it has a singularity in the origin $\vec{x} = \vec{0}$. The problem is not in the solution itself, but in the equation itself. This situation jeopardizes the numerical integration of the perturbed problem, which can be avoided by introducing a regularization.

The regularization of the equations is a process that consists in changing the dependent and independent variables such that the dynamics analyzed is described by a set of equations free of singularities. The advantages of regularization are twofold: the numerical integration of the equations is facilitated; and the negative effects of the instability related to Keplerian orbits decrease. The linearization, a particular case of regularization, is a process to describe the dynamics through exact linear equations by a change of dependent and independent variables.

The first regularization method was Sundman's transformation (Sundman [87]), introduced by Karl F. Sundman in 1912. It was developed to attempt to solve the three-body problem by applying a time transformation of the type

$$dt = r ds \quad (5.17)$$

by means of introducing a new independent variable s , or fictitious time, replacing the physical time t . This transformation converts the Eq. (5.1) into

$$\frac{d^2 \vec{x}}{ds^2} = \frac{1}{|\vec{x}|} \frac{d|\vec{x}|}{ds} \frac{d\vec{x}}{ds} - \frac{\mu}{r} \vec{x} + |\vec{x}|^2 \vec{a}_p, \quad (5.18)$$

which, although is still singular when $r = |\vec{x}|$ tends to zero, the order of such singularity is now reduced from three to one, improving the numerical performance. With time new regularizations were developed to provide equations completely free of singularities where more than one change of variable was considered. In the next lines some of them are described.

5.3.1 The Sperling-Bürdet (SB) regularization

In 1961 Sperling carried out, for the first time, a joint regularization and linearization of the equations by means of a Sundman transformation and the use of first integrals of the two-body problem, such as the integral of the energy and the eccentricity vector (Sperling [83]). In 1968 Bürdet, following the developments of Sperling, published a theory of perturbations based on the regularized and linearized equations of Sperling (Bürdet [18]). By using the variation of constants method Bürdet developed perturbation equations for the elements involved in the Sperling equations.

5.3.1.1 Sperling regularization

Let $\vec{e}$ be the eccentricity vector

$$\vec{e} = -\frac{\vec{x}}{r} - \frac{1}{\mu} (\vec{h} \times \vec{v}). \quad (5.19)$$

If the dot product of right-hand side is expanded:

$$\vec{e} = \frac{\vec{x}}{\mu} \left(v^2 - \frac{\mu}{r} \right) - \frac{1}{\mu} \vec{v} (\vec{v} \cdot \vec{x}) \quad (5.20)$$

and we consider the integral of the energy:

$$v^2 - \frac{2\mu}{r} = -\frac{\mu}{a}, \quad (5.21)$$

the term v^2 can be replaced in Eq. (5.19) obtaining the following expression:

$$\vec{e} = \vec{x} \left(\frac{1}{r} - \frac{1}{a} \right) - \frac{1}{\mu} \vec{v} (\vec{v} \cdot \vec{x}), \quad (5.22)$$

where a is supposed to be negative for hyperbolic orbits. If the term $\frac{\vec{x}}{r}$ is replaced in Eq. (5.1), it adopts the form:

$$\frac{d^2 \vec{x}}{dt^2} = -\frac{1}{r} \frac{dr}{dt} \frac{d\vec{x}}{dt} - \frac{\mu}{a} \frac{\vec{x}}{r^2} - \mu \frac{\vec{e}}{r^3} + \vec{a}_p. \quad (5.23)$$

The Sundman transformation:

$$ds = \frac{\sqrt{\mu}}{r} dt, \quad t - t_0 = \int_{s_0}^s \frac{r}{\sqrt{\mu}} ds \quad (5.24)$$

is now applied to establish s as independent variable of the Eq. (5.23):

$$\frac{d^2 \vec{x}}{ds^2} = -\frac{1}{a} \vec{x} - \vec{e} + \frac{r^2}{\mu} \vec{a}_p, \quad (5.25)$$

where a and $\vec{e}$ are constants in the unperturbed motion. However, in the perturbed motion they are functions of s .

The next step is to obtain a similar relation to Eq. (5.25) for the variable $r = |\vec{x}|$. We start defining the eccentricity vector $\vec{e}$ in terms of s . Note that:

$$\vec{x} \cdot \frac{d\vec{x}}{ds} = \frac{1}{2} \frac{d(\vec{x} \cdot \vec{x})}{ds} = r \frac{dr}{ds} \quad (5.26)$$

and consequently

$$\vec{v}(\vec{v} \cdot \vec{x}) = \frac{\mu}{r^2} \frac{d\vec{x}}{ds} \left(\vec{x} \cdot \frac{d\vec{x}}{ds} \right) = \frac{\mu}{r} \frac{dr}{ds} \frac{d\vec{x}}{ds}. \quad (5.27)$$

Thus, the expression (5.22) takes the form:

$$\vec{e} = \vec{x} \left(\frac{1}{r} - \frac{1}{a} \right) - \frac{1}{r} \frac{dr}{ds} \frac{d\vec{x}}{ds} \quad (5.28)$$

and

$$\vec{e} \cdot \vec{x} = r \left(1 - \frac{1}{a} \right) - \left(\frac{dr}{ds} \right)^2. \quad (5.29)$$

If we differentiate the expression (5.26) with respect s , it becomes

$$\frac{d\vec{x}}{ds} \cdot \frac{d\vec{x}}{ds} + \vec{x} \cdot \frac{d^2 \vec{x}}{ds^2} = \left(\frac{dr}{ds} \right)^2 + r \frac{d^2 r}{ds^2}. \quad (5.30)$$

Furthermore, by using the integral of the energy the value of the first term of Eq. (5.30) can be deduced:

$$\frac{d\vec{x}}{ds} \cdot \frac{d\vec{x}}{ds} = \frac{r^2}{\mu} v^2 = r \left(2 - \frac{r}{a} \right) \quad (5.31)$$

and Eq. (5.30) becomes, by isolating its second term of the left-hand side:

$$\vec{x} \cdot \frac{d^2 \vec{x}}{ds^2} = \left(\frac{dr}{ds} \right)^2 + r \frac{d^2 r}{ds^2} - r \left(2 - \frac{r}{a} \right). \quad (5.32)$$

If the dot product is applied to both sides of Eq. (5.25) with $\vec{x}$, by using Eq. (5.29) and Eq. (5.32) the following relation is obtained:

$$\frac{d^2 r}{ds^2} = -\frac{1}{a} r + 1 + \frac{r}{\mu} \vec{x} \cdot \vec{a}_p \quad (5.33)$$

with an identical structure to Eq. (5.25).

5.3.1.2 The unperturbed motion

We study the unperturbed motion for the elliptic case. If there is no perturbation the equations governing the dynamics take the form:

$$t - t_0 = \int_{s_0}^s \frac{r}{\sqrt{\mu}} ds \quad (5.34a)$$

$$\frac{d^2 \vec{x}}{ds^2} = -\frac{1}{a} \vec{x} - \vec{e} \quad (5.34b)$$

$$\frac{d^2 r}{ds^2} = -\frac{1}{a} r + 1, \quad (5.34c)$$

where the values of a and $\vec{e}$ are constants. Obviously, these equations should be integrated from known initial conditions at $s = s_0$:

$$\vec{x}(s_0) = \vec{x}_0, \quad \frac{d\vec{x}}{ds}(s_0) = \frac{r_0}{\sqrt{\mu}} \vec{v}_0. \quad (5.35)$$

Note that Eqs. (5.34a), (5.34b) and (5.34c) (Sperling equations) are linear and completely regular such that the singularity $r = 0$ no longer appears. In the elliptic motion $0 \leq a < \infty$ and the Eq. (5.34b) corresponds to a three-dimensional oscillator of frequency $1/\sqrt{a}$, in the variable s , vibrating around the stationary solution $\vec{x}_e = -a\vec{e}$. This frequency of oscillation is determined by the initial conditions and is, therefore, perfectly known before starting the integration process. Thus, the calculation of the frequency is not affected by the truncation and roundoff errors committed in the numerical integration process.

Equation (5.34c) can be analytically integrated providing the following solution:

$$r = a + C_1 \cos\left(\frac{s}{\sqrt{a}}\right) + C_2 \sin\left(\frac{s}{\sqrt{a}}\right), \quad (5.36)$$

where C_1 and C_2 are integration constants. Introducing this solution in Eq. (5.34a) the next solution is obtained:

$$t = \frac{a}{\sqrt{\mu}} s + \sqrt{\frac{a}{\mu}} \left[C_1 \sin\left(\frac{s}{\sqrt{a}}\right) - C_2 \cos\left(\frac{s}{\sqrt{a}}\right) \right] + C_3, \quad (5.37)$$

where the time t grows linearly with the fictitious time s . The constant a should be determined with the maximum precision because any error in its determination will produce and increasing error in the time t and consequently in the position of the particle in the trajectory. The dynamical instability of the equations has been removed, however, it reappears in the time expression. This is logical taking into account that it is not possible to change the reality of a physical nature. Nevertheless, the advantage of the regularization carried out is the absence of problems of dynamic stability during the numerical integration in terms of the variable s and, consequently, the numerical errors accumulated during the process are minimal.

5.3.1.3 The perturbed motion

If the particle is perturbed by the acceleration $\vec{a}_p$, the equations governing the motion are:

$$t - t_0 = \int_{s_0}^s \frac{r}{\sqrt{\mu}} ds \quad (5.38a)$$

$$\frac{d^2 \vec{x}}{ds^2} = -\frac{1}{a} \vec{x} - \vec{e} + \frac{r^2}{\mu} \vec{a}_p \quad (5.38b)$$

$$\frac{d^2 r}{ds^2} = -\frac{1}{a} r + 1 + \frac{r}{\mu} \vec{x} \cdot \vec{a}_p, \quad (5.38c)$$

where the values of a and $\vec{e}$ are no longer constants but changing with s . Obviously, these equations should be integrated from known initial conditions (5.35).

If the perturbing acceleration $\vec{a}_p$ does not present any singularity, these equations are completely regular. In order to integrate them numerically, the $a(s)$ and $\vec{e}(s)$ values, whose evolution is determined by the perturbation $\vec{a}_p$, should be provided.

We start by determining the eccentricity vector. Its time derivative is given by:

$$\frac{d\vec{e}}{dt} = \frac{1}{\mu} [\vec{a}_p \times \vec{h} + \vec{v} \times (\vec{x} \times \vec{a}_p)]. \quad (5.39)$$

If the terms of the right-hand side are developed we get the expression

$$\frac{d\vec{e}}{dt} = \frac{1}{\mu} [2\vec{x}(\vec{v} \cdot \vec{a}_p) - \vec{v}(\vec{a}_p \cdot \vec{x}) - \vec{a}_p(\vec{x} \cdot \vec{v})], \quad (5.40)$$

that in terms of the variable s it becomes:

$$\frac{d\vec{e}}{ds} = \frac{1}{\mu} \left[2\vec{x} \left(\frac{d\vec{x}}{ds} \cdot \vec{a}_p \right) - \frac{d\vec{x}}{ds} (\vec{a}_p \cdot \vec{x}) - \vec{a}_p \left(\vec{x} \cdot \frac{d\vec{x}}{ds} \right) \right]. \quad (5.41)$$

The energy equation adopts the form:

$$\frac{dE}{dt} = \vec{v} \cdot \vec{a}_p, \quad (5.42)$$

where E is the total Keplerian energy of the particle. In terms of the variable s it turns out to be

$$\frac{dE}{ds} = \frac{d\vec{x}}{ds} \cdot \vec{a}_p. \quad (5.43)$$

Taking into account that

$$E = -\frac{\mu}{2a}, \quad (5.44)$$

the following equation is obtained

$$\frac{d}{ds} \left(\frac{1}{a} \right) = -\frac{2}{\mu} \frac{d\vec{x}}{ds} \cdot \vec{a}_p, \quad (5.45)$$

which completes the set of equations to integrate.

In summary, the equations that govern the motion of the particle are Eqs. (5.38a), (5.38b) and (5.38c) such that the order of the system is thirteen. This regularization procedure gives rise to seven redundant variables when it is compared to the original equations. It is the price that must be paid to obtain a system of equations that allows a numerical integration practically free of dynamic instabilities and minimizing the calculation errors. The necessity to obtain numerically the value of $1/a$ makes the characteristic dynamic instability of the Keplerian motion still present in the integration process, but its negative effects are much smaller in the regular scheme than in the original Eq. (5.1) of the problem.

5.3.2 The Stiefel regularization for planar motions

The regularized method described here emerges from the analysis of Tullio Levi-Civita about the regularization of the planar two-body problem. It was later generalized to the three dimensional case by Stiefel and Scheifele [84].

Let the fictitious time s given by a Sundman transformation

$$dt = r ds, \quad (5.46)$$

the relations between the time derivatives and the derivatives with respect to s for a certain quantity ψ are given by the equations

$$\frac{d\psi}{dt} = \frac{1}{r} \frac{d\psi}{ds} \quad (5.47a)$$

$$\frac{d^2\psi}{dt^2} = \frac{1}{r^3} \left(r \frac{d^2\psi}{ds^2} - \frac{dr}{ds} \frac{d\psi}{ds} \right). \quad (5.47b)$$

By applying the change of independent variable to the Eq. (5.1) it adopts the form:

$$\frac{d^2\vec{x}}{ds^2} - \frac{1}{r} \frac{dr}{ds} \frac{d\vec{x}}{ds} + \frac{\mu}{r} \vec{x} = r^2 \vec{a}_p, \quad (5.48)$$

which still preserves the singularity in the origin $r = 0$.

Figure 5.2: Transformation in the complex plane.

5.3.2.1 The Levi-Civita transformation

If we fix the semi-major axis of an ellipse and make the eccentricity tend to one, the ellipse becomes a line segment described by an oscillatory motion in the limit, that is, a degenerated ellipse. In this scenario each time the particle reaches the origin (collision) it is done with infinite velocity, whereas an infinite velocity of opposite sign is necessary if the particle leaves the origin. The argument of the position vector increments 2π for each passing in the origin where the singularity resides.

In order to avoid this singularity, Levi Civita proposed a transformation of the inertial plane Ox_1y_1 , where the motion takes place, to a plane with parameters Ou_1u_2 (see Fig. 5.2) such that for each collision the particle crosses the singularity instead of bouncing. For that he considered the complex plane and established a conformal transformation governed by the following relation:

$$u_1 + i u_2 = \sqrt{x_1 + i y_1}, \quad (5.49)$$

where i represents the imaginary unit $i = \sqrt{-1}$.

By taking the square of Eq. (5.49) and decomposing in its real and imaginary part, the next equations are obtained:

$$x_1 = u_1^2 - u_2^2 \quad (5.50a)$$

$$y_1 = 2 u_1 u_2. \quad (5.50b)$$

We denote $\vec{x} = (x_1, y_1) = x_1 + i y_1$ and $\vec{u} = (u_1, u_2) = u_1 + i u_2$. Equations (5.50a) and (5.50b) adopt the matrix form:

$$\begin{pmatrix} x_1 \\ y_1 \end{pmatrix} = \begin{pmatrix} u_1 & -u_2 \\ u_2 & u_1 \end{pmatrix} \begin{pmatrix} u_1 \\ u_2 \end{pmatrix}, \quad (5.51)$$

which can be expressed in compact form:

$$\vec{x} = \mathcal{L}(\vec{u})\vec{u}, \quad (5.52)$$

where the Levi-Civita matrix is given by:

$$\mathcal{L}(\vec{u}) = \begin{pmatrix} u_1 & -u_2 \\ u_2 & u_1 \end{pmatrix}. \quad (5.53)$$

If Eqs. (5.50a) and (5.50b) are derived with respect to the variable s we obtain:

$$\frac{dx_1}{ds} = 2u_1 \frac{du_1}{ds} - 2u_2 \frac{du_2}{ds} \quad (5.54a)$$

$$\frac{dy_1}{ds} = 2u_2 \frac{du_1}{ds} + 2u_1 \frac{du_2}{ds}, \quad (5.54b)$$

which can be written as follows

$$\frac{d\vec{x}}{ds} = 2\mathcal{L}(\vec{u}) \frac{d\vec{u}}{ds}. \quad (5.55)$$

5.3.2.2 Regularized equations

The aim is to express the equations governing the dynamics in terms of the (u_1, u_2) parameters. We start by differentiating the expression (5.55) with respect to s :

$$\frac{d^2\vec{x}}{ds^2} = 2\mathcal{L}(\vec{u}) \frac{d^2\vec{u}}{ds^2} + 2\mathcal{L}\left(\frac{d\vec{u}}{ds}\right) \frac{d\vec{u}}{ds}, \quad (5.56)$$

where the homogeneity property of the Levi-Civita matrix has been considered, that is

$$\frac{d}{ds}\mathcal{L}(\vec{u}) = \mathcal{L}\left(\frac{d\vec{u}}{ds}\right). \quad (5.57)$$

Then, the expression (5.56) is replaced in Eq. (5.48) which is multiplied by r and divided by 2. By making use of the relations (5.52) and (5.55) it adopts the form:

$$\mathcal{L}(\vec{u}) \left(r \frac{d^2\vec{u}}{ds^2} - \frac{dr}{ds} \frac{d\vec{u}}{ds} + \frac{\mu}{ds} \vec{u} \right) + r \mathcal{L}\left(\frac{d\vec{u}}{ds}\right) \frac{d\vec{u}}{ds} = \frac{r^3}{2} \vec{a}_p. \quad (5.58)$$

Taking into account that

$$r = \vec{u} \cdot \vec{u}, \quad \vec{v} = \frac{d\vec{u}}{ds}, \quad (5.59)$$

and by making use of the interchangeability property of the Levi-Civita matrix:

$$(\vec{u} \cdot \vec{u})\mathcal{L}(\vec{v})\vec{v} + (\vec{v} \cdot \vec{v})\mathcal{L}(\vec{u})\vec{u} = 2(\vec{u} \cdot \vec{v})\mathcal{L}(\vec{u})\vec{v}, \quad (5.60)$$

if we premultiply by the inverse of $\mathcal{L}(\vec{u})$ we obtain the equation governing the evolution of $\vec{u}$ with respect to the fictitious time s :

$$\frac{d^2\vec{u}}{ds^2} + \frac{\frac{\mu}{2} - \left(\frac{d\vec{u}}{ds} \cdot \frac{d\vec{u}}{ds}\right)}{(\vec{u} \cdot \vec{u})} \vec{u} = \frac{(\vec{u} \cdot \vec{u})}{2} \mathcal{L}^T(\vec{u}) \vec{a}_p, \quad (5.61)$$

where the perturbing acceleration $\vec{a}_p$ should be expressed in terms of s , $\vec{u}$ and its derivatives.

However, Eq. (5.61) is still singular because the second term becomes infinite when $r = 0$. In order to avoid this singularity, the Keplerian energy needs to be introduced through the semi-major axis a which is defined by

$$\frac{-\mu}{2a} = \frac{1}{2}v^2 - \frac{\mu}{r}, \quad (5.62)$$

or in terms of the (u_1, u_2) parameters by

$$\frac{\mu}{2a} = 2 \frac{\frac{\mu}{2} - \left(\frac{d\vec{u}}{ds} \cdot \frac{d\vec{u}}{ds}\right)}{(\vec{u} \cdot \vec{u})}. \quad (5.63)$$

Consequently, Eq.(5.61) can be written as follows

$$\frac{d^2 \vec{u}}{ds^2} + \frac{\mu}{4a} \vec{u} = \frac{(\vec{u} \cdot \vec{u})}{2} \mathcal{L}^T(\vec{u}) \vec{a}_p, \quad (5.64)$$

which is regular.

While in the unperturbed motion the value of a remains constant, in the perturbed motion a is a function of s which should be determined numerically. Therefore, Eq. (5.64) needs to be complemented with an additional equation providing the value of a . In order to determine the evolution equation of a we have to consider the energy equation (5.42) as we did in the Sperling-Bürdet regularization. Accordingly the equation is the given by (5.45), which in terms of the (u_1, u_2) parameters adopts the form:

$$\frac{d}{ds} \left(\frac{1}{a} \right) = -\frac{4}{\mu} \left[\mathcal{L}(\vec{u}) \frac{d\vec{u}}{ds} \right] \vec{a}_p. \quad (5.65)$$

In summary, the equations that govern the motion of the particle are

$$\frac{dt}{ds} = \vec{u} \cdot \vec{u} \quad (5.66a)$$

$$\frac{d^2 \vec{u}}{ds^2} + \frac{\mu}{4a} \vec{u} = \frac{(\vec{u} \cdot \vec{u})}{2} \mathcal{L}^T(\vec{u}) \vec{a}_p \quad (5.66b)$$

$$\frac{d}{ds} \left(\frac{1}{a} \right) = -\frac{4}{\mu} \left[\mathcal{L}(\vec{u}) \frac{d\vec{u}}{ds} \right] \vec{a}_p, \quad (5.66c)$$

which should be integrated from appropriate initial conditions

$$s = s_0 : \quad \vec{u}(s_0) = \vec{u}_0, \quad \frac{d\vec{u}}{ds}(s_0) = \frac{d\vec{u}}{ds} \Big|_{(0)}, \quad a(s_0) = a_0. \quad (5.67)$$

The order of the resulting system is six. This regularization procedure increases the order of the system in two units. This is the price to obtain a regular integration scheme that minimizes the effects of the dynamics instabilities.

5.3.2.3 Determination of the initial conditions

The numerical integration should start from known initial values. It is admitted that at the initial instant the dynamical state of the system is known:

$$t = t_0 : \quad \vec{x} = \vec{x}_0, \quad \vec{v} = \vec{v}_0. \quad (5.68)$$

From the initial state $(\vec{x}_0, \vec{v}_0)$ the initial conditions in the Stiefel variables $(s, \vec{u})$ have to be obtained. Although the Eqs. (5.50a) and (5.50b) univocally determine (x_1, y_1) , the inverse transformation provides two solutions for (u_1, u_2) satisfying the Eq. (5.49). One option is to consider the next procedure:

$$x_1 > 0 : \quad u_1 = \sqrt{\frac{1}{2}(r + x_1)}, \quad u_2 = \frac{y_1}{\sqrt{2(r + x_1)}} \quad (5.69a)$$

$$x_1 < 0 : \quad u_1 = \frac{y_1}{\sqrt{2(r - x_1)}}, \quad u_2 = \sqrt{\frac{1}{2}(r - x_1)}, \quad (5.69b)$$

where $r = \sqrt{x_1^2 + y_1^2}$.

Once the vector $u_0 = (u_1(0), u_2(0))$ is known by Eqs. (5.69a) or (5.69b), the initial condition for the derivative is given by inverting the Eq. (5.55):

$$\frac{d\vec{u}}{ds}(0) = \frac{1}{2r} \mathcal{L}^T(\vec{u}_0) \frac{d\vec{x}}{ds} = \frac{1}{2} \mathcal{L}^T(\vec{u}_0) \vec{v}_0. \quad (5.70)$$

5.3.3 The Kustaanheimo-Stiefel (KS) regularization

The Stiefel regularization previously described is referred to planar motions. By considering again the fictitious time s given by Eq. (5.46), the idea arising now is if it can be extended to the three-dimensional case. In 1933 A. Hurwitz proved that the generalization of Levi-Civita transformation to three dimensions is not possible (Hurwitz [45]), but it can be extended to four dimensions. Following this idea, Kustaanheimo proposed to generalize the Levi-Civita transformation to the four-dimensional space by introducing two complex numbers (Kustaanheimo et al. [50]) and showed how it is possible to use that generalization to regularize the equations of orbital motion. This procedure is known as Kustaanheimo-Stiefel regularization, or KS method for short.

The method starts by introducing the extension to the Levi-Civita matrix:

$$\mathcal{K}(\vec{u}) = \begin{pmatrix} u_1 & -u_2 & -u_3 & u_4 \\ u_2 & u_1 & -u_4 & -u_3 \\ u_3 & u_4 & u_1 & u_2 \\ u_4 & -u_3 & u_2 & -u_1 \end{pmatrix}. \quad (5.71)$$

Then, a transformation between two spaces of four dimensions is introduced by the equation

$$\vec{x} = \mathcal{K}(\vec{u})\vec{u}, \quad (5.72)$$

where $\vec{u} = (u_1, u_2, u_3, u_4)$ and $\vec{x} = (x_1, x_2, x_3, x_4)$.

If we develop the Eq. (5.72) we obtain a system where the fourth component is null, that is:

$$x_1 = u_1^2 - u_2^2 - u_3^2 + u_4^2 \quad (5.73a)$$

$$x_2 = 2(u_1 u_2 - u_3 u_4) \quad (5.73b)$$

$$x_3 = 2(u_1 u_3 + u_2 u_4) \quad (5.73c)$$

$$x_4 = 0, \quad (5.73d)$$

such that $r = |\vec{x}|$

The transformation governed by Eq. (5.72) is defined as the KS transformation and the image of $\mathbb{R}^4$ by this transformation is a three-dimensional subspace immersed in $\mathbb{R}^4$. In the KS transformation the first three components of $\vec{x}$ are identified with the inertial coordinates of the position vector of the particle.

Some of the properties of the Levi-Civita matrix $\mathcal{L}$ are extended to the matrix $\mathcal{K}$. This matrix is r -orthogonal, linear and homogeneous, that is, the following properties hold:

$$\mathcal{K}^T(\vec{u})\mathcal{K}(\vec{u}) = rI \quad (5.74a)$$

$$\mathcal{K}(\vec{u} + \vec{v}) = \mathcal{K}(\vec{u}) + \mathcal{K}(\vec{v}) \quad (5.74b)$$

$$\frac{d}{ds}[\mathcal{K}(\vec{u})] = \mathcal{K}\left(\frac{d\vec{u}}{ds}\right). \quad (5.74c)$$

However, the rest of properties of the Levi-Civita matrix are not directly transferred to the $\mathcal{K}$ matrix unless additional conditions are introduced. This is the case of the interchangeability property:

$$\mathcal{K}(\vec{u})\vec{v} = \mathcal{K}(\vec{v})\vec{u}, \quad (5.75)$$

which requires the next bilinear relation to be satisfied:

$$\rho(\vec{u}, \vec{v}) = u_1 v_4 - u_2 v_3 + u_3 v_2 - u_4 v_1 = 0. \quad (5.76)$$

If the relations (5.73a), (5.73b), (5.73c) and (5.73d) are differentiated with respect to the variable s , we obtain:

$$\frac{d\vec{x}}{ds} = \mathcal{K} \left(\frac{d\vec{u}}{ds} \right) \vec{u} + \mathcal{K}(\vec{u}) \frac{d\vec{u}}{ds}, \quad (5.77)$$

which becomes, if the vectors $\vec{u}$ and $\frac{d\vec{u}}{ds}$ verify the bilinear relation (5.76), the equation:

$$\frac{d\vec{x}}{ds} = 2 \mathcal{K}(\vec{u}) \frac{d\vec{u}}{ds}. \quad (5.78)$$

Equation (5.78) is the extension to the three-dimensional case of the planar case (5.55). Consequently, this relation is true when $\vec{u}$ and its derivative verify the bilinear relation, which ensures that the fourth component of the left-hand side of Eq. (5.78) is canceled.

5.3.3.1 The regularized equations

The aim is to express the equations governing the dynamics in terms of the (u_1, u_2, u_3, u_4) parameters. We start by differentiating the expression (5.78) with respect to s by using the homogeneity property:

$$\frac{d^2\vec{x}}{ds^2} = 2 \mathcal{K}(\vec{u}) \frac{d^2\vec{u}}{ds^2} + 2 \mathcal{K} \left(\frac{d\vec{u}}{ds} \right) \frac{d\vec{u}}{ds}. \quad (5.79)$$

Then, the second derivative given by Eq. (5.79) is replaced in Eq. (5.48) by taking into account that

$$\frac{dr}{ds} = 2 \vec{u} \cdot \frac{d\vec{u}}{ds} \quad (5.80)$$

and it is multiplied by $r = (\vec{u} \cdot \vec{u})$.

By applying the interchangeability property by taking as vector $\vec{v}$ the derivative of $\vec{u}$ in

$$(\vec{u} \cdot \vec{u}) \mathcal{K}(\vec{v}) \vec{v} + (\vec{v} \cdot \vec{v}) \mathcal{K}(\vec{u}) \vec{u} = 2 (\vec{u} \cdot \vec{v}) \mathcal{K}(\vec{u}) \vec{v}, \quad (5.81)$$

if we multiply both sides of the equation by $\mathcal{K}^{-1}(\vec{u})$ and the inverse of $(\vec{u} \cdot \vec{u})$, we obtain the equation governing the evolution of $\vec{u}$ as a function of the fictitious time s :

$$\frac{d^2\vec{u}}{ds^2} + \frac{\mu}{2} \left(\frac{d\vec{u}}{ds} \cdot \frac{d\vec{u}}{ds} \right) \vec{u} = \frac{(\vec{u} \cdot \vec{u})}{2} \mathcal{K}^T(\vec{u}) \vec{a}_p. \quad (5.82)$$

However, Eq. (5.82) is still singular because the second term becomes infinite when $r = 0$. In order to avoid this singularity, the Keplerian energy is introduced as we did in the two-dimensional problem. Consequently, Eq. (5.82) can be written as follows

$$\frac{d^2\vec{u}}{ds^2} + \frac{\mu}{4a} \vec{u} = \frac{(\vec{u} \cdot \vec{u})}{2} \mathcal{K}^T(\vec{u}) \vec{a}_p, \quad (5.83)$$

which is regular.

In summary, the equations that govern the motion of the particle are

$$\frac{dt}{ds} = \vec{u} \cdot \vec{u} \quad (5.84a)$$

$$\frac{d^2\vec{u}}{ds^2} + \frac{\mu}{4a} \vec{u} = \frac{(\vec{u} \cdot \vec{u})}{2} \mathcal{K}^T(\vec{u}) \vec{a}_p \quad (5.84b)$$

$$\frac{d}{ds} \left(\frac{1}{a} \right) = -\frac{4}{\mu} \left[\mathcal{K}(\vec{u}) \frac{d\vec{u}}{ds} \right] \vec{a}_p, \quad (5.84c)$$

which should be integrated from appropriate initial conditions

$$s = s_0 : \quad \vec{u}(s_0) = \vec{u}_0, \quad \left. \frac{d\vec{u}}{ds} \right|_{(0)} = \frac{d\vec{u}}{ds} \Big|_{(0)}, \quad a(s_0) = a_0. \quad (5.85)$$

The order of the resulting system is ten. This regularization procedure increases the order of the system in four units. This is the price to obtain a regular integration scheme that minimizes the effects of the dynamics instabilities.

5.3.3.2 Determination of the initial conditions

The main idea of Stiefel's scheme for planar motions is to describe the dynamics in terms of the variables $(\vec{u}, s)$ instead of $(\vec{x}, t)$. The same idea is considered in the KS method but with a consideration. In both the two-dimensional case and three-dimensional case the transformation involved is univocal when considering the estimation of $\vec{x}$ from $\vec{u}$. That is, given a value of $\vec{u}$ the Eqs. (5.52) (two dimensions) and (5.72) (three dimensions) provide a unique value for $\vec{x}$. However, it does not happen with the inverse transformation. For the two-dimensional case the transformation from $\vec{x}$ to $\vec{u}$ has two possible solutions because it derives from the square root of a complex number. In the three-dimensional case the transformation from $\vec{x}$ to $\vec{u}$ is more complex because for a given value of $\vec{x}$ there is a curve in the parameter space and all the values of $\vec{u}$ in that curve are transformed into $\vec{x}$ by means of the KS transformation (5.72).

It can be proved that if $r + x_1 \neq 0$ then all the points of the circumference of $\mathbb{R}^4$ given by

$$u_1 = \frac{\sqrt{2}}{2} \sqrt{r + x_1} \sin(\theta) \quad (5.86a)$$

$$u_2 = \frac{\sqrt{2}}{2} \frac{1}{\sqrt{r + x_1}} (x_2 \sin(\theta) - x_3 \cos(\theta)) \quad (5.86b)$$

$$u_3 = \frac{\sqrt{2}}{2} \frac{1}{\sqrt{r + x_1}} (x_2 \cos(\theta) + x_3 \sin(\theta)) \quad (5.86c)$$

$$u_4 = -\frac{\sqrt{2}}{2} \sqrt{r + x_1} \cos(\theta), \quad (5.86d)$$

being $r = |\vec{x}| = \sqrt{x_1^2 + x_2^2 + x_3^2}$ and $\theta \in [0, 2\pi]$, transform into $\vec{x}$. Analogously, if $r - x_1 \neq 0$ then all the points of the circumference of $\mathbb{R}^4$ given by

$$u_1 = \frac{\sqrt{2}}{2} \frac{1}{\sqrt{r - x_1}} (x_2 \sin(\theta) - x_3 \cos(\theta)) \quad (5.87a)$$

$$u_2 = \frac{\sqrt{2}}{2} \sqrt{r - x_1} \sin(\theta) \quad (5.87b)$$

$$u_3 = -\frac{\sqrt{2}}{2} \sqrt{r - x_1} \cos(\theta) \quad (5.87c)$$

$$u_4 = \frac{\sqrt{2}}{2} \frac{1}{\sqrt{r - x_1}} (x_2 \cos(\theta) + x_3 \sin(\theta)), \quad (5.87d)$$

transform into $\vec{x}$.

The numerical integration should start from known initial values. It is admitted that at the initial instant the dynamical state of the system is known:

$$t = t_0 : \quad \vec{x} = \vec{x}_0, \quad \vec{v} = \vec{v}_0. \quad (5.88)$$

From the initial state $(\vec{x}_0, \vec{v}_0)$ the initial conditions in the KS variables $(s, \vec{u})$ have to be obtained (Roa, Urrutxua, and Peláez [74]). First, $s = 0$ is usually chosen as

starting value for the integration process. The critical value to be determined is the vector $\vec{u}(0)$ from which the process starts. For simplicity we call (x_0, y_0, z_0) the inertial coordinates of the vector $\vec{x}_0$, that is, $x_0 \equiv x_0(0)$.

If $x_0 > 0$ the vector $\vec{u}(0)$ is usually chosen from the values defined in Eqs. (5.86a), (5.86b), (5.86c) and (5.86d). Any value of θ would be possible, but it seems reasonable to choose a simple value like $\theta = 0^\circ$. In that case the initial condition would be:

$$u_1(0) = 0, \quad u_2(0) = -\frac{\sqrt{2}}{2} \frac{z_0}{\sqrt{r+x_0}}, \quad (5.89a)$$

$$u_3(0) = \frac{\sqrt{2}}{2} \frac{y_0}{\sqrt{r+x_0}}, \quad u_4(0) = -\frac{\sqrt{2}}{2} \sqrt{r+x_0}. \quad (5.89b)$$

If $x_0 < 0$ the vector $\vec{u}(0)$ is usually chosen from the values defined in Eqs. (5.87a), (5.87b), (5.87c) and (5.87d). Any value of θ would be possible, but it seems reasonable to choose a simple value like $\theta = 0$. In that case the initial condition would be:

$$u_1(0) = -\frac{\sqrt{2}}{2} \frac{z_0}{\sqrt{r-x_0}}, \quad u_2(0) = 0, \quad (5.90a)$$

$$u_3(0) = -\frac{\sqrt{2}}{2} \sqrt{r-x_0}, \quad u_4(0) = \frac{\sqrt{2}}{2} \frac{y_0}{\sqrt{r-x_0}}. \quad (5.90b)$$

In this way, the initial condition $\vec{u}(0)$ is obtained in every situation except in the singular limit case in which initially $\vec{x} = \vec{0}$ ($r = 0$). However, this situation is very unrealistic and is not considered. In addition, the value of $\vec{u}(0)$ selected in this way becomes, under the KS transformation, in the vector $\vec{x}_0$.

Once the initial value of $\vec{u}$ has been determined, the value of its derivative should be determined. To do this, the relation (5.78) is inverted obtaining:

$$\frac{d\vec{u}}{ds}(0) = \frac{1}{2r} \mathcal{K}^T(\vec{u}_0) \frac{d\vec{x}}{ds} = \frac{1}{2} \mathcal{K}^T(\vec{u}_0) \vec{v}_0. \quad (5.91)$$

With this selection the vector $\vec{u}$ and its derivative satisfy, at the initial instant, the bilinear relation:

$$\rho\left(\vec{u}_0, \frac{d\vec{u}}{ds}(0)\right) = 0. \quad (5.92)$$

5.3.4 The Palacios regularization

The Palacios regularized formulation, based on Hansen's ideal frame (Hansen [40]) and quaternions for rotation, makes use of the true anomaly as a time-like variable and a constant stepsize for the numerical integration (Palacios and Calvo [60]).

Let $\mathcal{S}$ be the space frame defined by three fixed orthonormal vectors $(\vec{s}_1, \vec{s}_2, \vec{s}_3)$, the selection of the frames and the set of variables is done by following Deprit [27]. Let $\mathcal{O}(\vec{u}, \vec{v}, \vec{n})$ be the orbital plane, the orthonormal basis is given by

$$\vec{u} = \frac{\vec{x}}{\|\vec{x}\|} = \frac{\vec{x}}{r} \quad (5.93a)$$

$$\vec{n} = \frac{\vec{x} \times \vec{X}}{\|\vec{x} \times \vec{X}\|} = \frac{\vec{G}}{G} \quad (5.93b)$$

$$\vec{v} = \vec{n} \times \vec{u}, \quad (5.93c)$$

where $\vec{x}$ and $\vec{X}$ stand for radial and velocity vectors, respectively, and $\vec{G}$ for the angular momentum vector.

In particular, at an initial instant t_0 , the orbital frame defines a fixed orthonormal reference system $(\vec{u}_0, \vec{v}_0, \vec{n}_0)$, named as the departure frame and denoted by $\mathcal{D}$. The motion of the orbital frame with respect to the fixed frame may be viewed as the rotation given by

$$\dot{\vec{u}} = \vec{\omega} \times \vec{u} \quad \dot{\vec{v}} = \vec{\omega} \times \vec{v} \quad \dot{\vec{n}} = \vec{\omega} \times \vec{n}, \quad (5.94)$$

where the angular velocity vector is

$$\vec{\omega} = \frac{G}{r^2} \vec{n} + \frac{r}{G} (\vec{F} \cdot \vec{n}) \vec{u}. \quad (5.95)$$

Since the inclination of the orbit of an artificial satellite experiences small variations, the equations of motion are formulated in a frame that keeps the inclination always near zero avoiding thereby the loss of significant digits.

The ideal frame $\mathcal{I}(\vec{u}_I, \vec{v}_I, \vec{n}_I)$ is defined by the second term of the angular velocity

$$\vec{\omega}_I = \frac{r}{G} (\vec{F} \cdot \vec{n}) \vec{u} \quad (5.96)$$

as the departure frame rotated by the rotation defined by vector $\vec{\omega}_I$. This frame is called ideal because it has the same velocity than the departure frame (Hansen [40]). When the perturbation magnitude is small or nearly along the orbital plane, the vector $\vec{\omega}_I$ is also small, which means a slow rotation. In this case the motion may be composed of two parts: 1) a slow rotation of the orbital (or ideal) plane and 2) the motion in this plane which is a two degrees of freedom problem. Since the departure frame $\mathcal{D}$ (and ideal frame too) may be regarded as the image of the space frame $\mathcal{S}$ by some rotation, the relation between both systems may be given either in matrix form or by using the corresponding quaternion. The quaternions present the advantage of having no inherent geometrical singularities, and consequently, they are chosen to express the relation.

5.3.4.1 Formulation of the method

Several formulations of the equations of motion of orbital propagation are now introduced (see Fig. 5.3). The first consists in the classical model using Cartesian coordinates in the inertial space frame. The other four are introduced with the idea of improving the efficiency of the numerical integration. They all use more variables than degrees of freedom, and the corresponding constraints are used to control the numerical performance. Besides, projective coordinates are employed in order to represent in a better form the dynamical behavior of the problem.

Cartesian coordinates referred to the space frame: in the space frame $\mathcal{S}$, a particle with position vector $\vec{x}$ and velocity vector $\dot{\vec{x}}$ (rectangular coordinates) is acted upon by a perturbing force $\vec{F}$. The motion is described by the first order differential system

$$\dot{\vec{x}} = \vec{X} \quad (5.97a)$$

$$\dot{\vec{X}} = -\frac{\mu}{r^3} \vec{x} + \vec{F}, \quad (5.97b)$$

with the initial conditions $\vec{x}(t_0) = \vec{x}_0$ and $\dot{\vec{x}}(t_0) = \dot{\vec{x}}_0$. The dynamical model described by Eqs. (5.97a) and (5.97b) is denoted as *cs*.

Projective coordinates in the departure frame: the departure frame $\mathcal{D}$, coincides, by definition, with the orbital frame $\mathcal{O}(\vec{u}(t), \vec{v}(t), \vec{n}(t))$ at the initial instant t_0 and coincides with the rotated space frame by a fixed rotation defined by $(\vec{u}(t_0), \vec{v}(t_0), \vec{n}(t_0))$.

Figure 5.3: Scheme of Palacios formulations.

From Eqs. (5.97a) and (5.97b) and the definition of angular momentum the corresponding expression and the derivatives with respect to the independent variable t are obtained:

$$\vec{G} = \vec{x} \times \vec{X}, \quad \dot{\vec{G}} = \vec{x} \times \vec{F}, \quad \dot{G} = r \vec{F} \cdot \vec{v}. \quad (5.98)$$

The remaining orbital parameters that define the orbit are given by the following equation:

$$\ddot{r} = \frac{G^2}{r^3} - \frac{\mu}{r^2} + \vec{F} \cdot \vec{u}. \quad (5.99)$$

The Newtonian equations (5.97a) and (5.97b) are equivalent to the system of dimension 12 made of the nine equations given in (5.94) defining the orbital frame and Eq. (5.99) where the unknowns $\vec{u}$, $\vec{v}$ and $\vec{n}$ have to satisfy the invariant orthonormality relations. By considering the departure frame, the following initial conditions are taken:

$$\vec{u}(t_0) = (1, 0, 0), \quad \vec{v}(t_0) = (0, 1, 0), \quad \vec{n}(t_0) = (0, 0, 1), \quad (5.100a)$$

$$r(t_0) = r_0, \quad \dot{r}(t_0) = \dot{r}_0, \quad \dot{G}(t_0) = \dot{G}_0. \quad (5.100b)$$

The dynamical model described by the equations above is denoted as pd .

Projective coordinates in the ideal frame: the ideal frame, referred to the departure frame, may be viewed as the rotation

$$\dot{\vec{u}}_I = \vec{\omega}_I \times \vec{u}_I, \quad \dot{\vec{v}}_I = \vec{\omega}_I \times \vec{v}_I, \quad \dot{\vec{n}}_I = \vec{\omega}_I \times \vec{n}_I, \quad (5.101)$$

with the initial conditions

$$\vec{u}_I(t_0) = (1, 0, 0), \quad \vec{v}_I(t_0) = (0, 1, 0), \quad \vec{n}_I(t_0) = (0, 0, 1). \quad (5.102)$$

Notice that the vector $\vec{n}$ is an integral of the motion and the ideal frame plane is fixed in time when the perturbation is in the orbital plane. Considering the acceleration relative to the ideal frame and taking into account the equality of the velocities in both frames, the following equation is deduced

$$\frac{\partial \dot{\vec{x}}}{\partial t} = -\frac{\mu}{r^3} \vec{x} + (\vec{F} \cdot \vec{u}) \vec{u} + (\vec{F} \cdot \vec{v}) \vec{v}, \quad (5.103)$$

where vectors are referred to the ideal frame and $\frac{\partial}{\partial t}$ stands for the time derivative in the ideal frame. Equations (5.101) and (5.103) define the ideal frame (or the instantaneous orbital plane) and the motion in the ideal plane, respectively. But Eqs. (5.101) may be substituted equivalently by the equations for the corresponding quaternion defining the rotation from the departure frame to the ideal frame. Taking into account the equation governing the temporal variation of a quaternion, for $\vec{\omega}_I$ it becomes:

$$\dot{\vec{q}} = \frac{1}{2} \vec{\omega}_I \vec{q}, \quad (5.104)$$

or, equivalently, in components once the value of $\vec{\omega}_I$ has been substituted:

$$\dot{q}_0 = -\frac{1}{2} \frac{r}{G} (\vec{F} \cdot \vec{n}) (q_1 \cos(\theta) + q_2 \sin(\theta)) \quad (5.105a)$$

$$\dot{q}_1 = -\frac{1}{2} \frac{r}{G} (\vec{F} \cdot \vec{n}) (-q_0 \cos(\theta) - q_3 \sin(\theta)) \quad (5.105b)$$

$$\dot{q}_2 = -\frac{1}{2} \frac{r}{G} (\vec{F} \cdot \vec{n}) (q_3 \cos(\theta) - q_0 \sin(\theta)) \quad (5.105c)$$

$$\dot{q}_3 = -\frac{1}{2} \frac{r}{G} (\vec{F} \cdot \vec{n}) (-q_2 \cos(\theta) + q_1 \sin(\theta)), \quad (5.105d)$$

where θ is the angle between $\vec{u}_I$ of the ideal plane and the radial direction $\vec{u}$ of the departure frame. The angle θ varies with time. Since $\vec{u}$ and its instantaneous variation in the ideal frame are given by

$$\vec{u} = (\cos(\theta), \sin(\theta), 0) \quad (5.106a)$$

$$\frac{\partial \vec{u}_I}{\partial t} = \vec{\omega}_I \times \vec{u}_I = \frac{G}{r^2} \vec{v}_I = \frac{\partial \vec{u}_I}{\partial \theta} \frac{d\theta}{dt}, \quad (5.106b)$$

the variation of theta is given by

$$\frac{d\theta}{dt} = \frac{G}{r^2}. \quad (5.107)$$

The vectorial equation (5.103) may be substituted equivalently by Eqs. (5.99) and (5.107). Therefore, we have an initial value problem composed by a system of eight equations (5.105a), (5.105b), (5.105c), (5.105d), (5.99) and (5.107) with eight unknowns and the initial conditions

$$q_0(t_0) = 1, \quad q_1(t_0) = 0, \quad q_2(t_0) = 0, \quad q_3(t_0) = 0, \quad (5.108)$$

$$r(t_0) = r_0, \quad \dot{r}(t_0) = \dot{r}_0, \quad G(t_0) = G_0, \quad \theta(t_0) = \theta_0, \quad (5.109)$$

However, the system is redundant since the unknowns should satisfy the constraint of normalization for the quaternion, $|\vec{q}| = 1$, which will be employed to monitor the propagation and truncation errors during the numerical integration. The dynamical model described by the equations above is denoted as *piq*.

Projective coordinates and regularization in the departure frame: taking into account the Eqs. (5.99) and (5.107), the independent variable t and the distance r are replaced, respectively, by the dimensionless variable θ and the quantity ρ such that

$$r^2 d\theta = G dt, \quad \rho = \frac{1}{r}. \quad (5.110)$$

Rescaling the force and the angular velocity to

$$\vec{F}^* = \frac{r^3}{G^2} \vec{F}, \quad \vec{\omega}^* = \frac{r^2}{G} \vec{\omega} = \vec{n} + (\vec{F}^* \cdot \vec{n}) \vec{u}, \quad (5.111)$$

the orbital equations adopt the following form:

$$\vec{u}' = \omega^* \times \vec{u} \quad (5.112a)$$

$$\vec{v}' = \omega^* \times \vec{v} \quad (5.112b)$$

$$\vec{n}' = \omega^* \times \vec{n} \quad (5.112c)$$

$$G' = G(\vec{F}^* \cdot \vec{v}) \quad (5.112d)$$

$$\rho'' + \rho = -\rho(\vec{F}^* \cdot \vec{u}) - \rho'(\vec{F}^* \cdot \vec{v}). \quad (5.112e)$$

The Newtonian equations (5.97a) and (5.97b) are equivalent to the system of dimension 13 comprised by Eqs. (5.112a), (5.112b), (5.112c), (5.112d) and (5.112e) along with Eq. (5.110) for θ , where the unknowns $\vec{u}$, $\vec{v}$ and $\vec{n}$ satisfy the invariant orthonormality relations. The dynamical model described by the equations above is denoted as *pdr*.

Projective coordinates and regularization in the ideal frame: the preceding formulation can be posed as for the *piq* formulation. Thus, considering $\vec{F}^*$ and $\vec{\omega}^*$ and the angle θ as independent variable, the following equations are obtained:

$$q_0' = -\frac{1}{2}(\vec{F}^* \cdot \vec{n})(q_1 \cos(\theta) + q_2 \sin(\theta)) \quad (5.113a)$$

$$q_1' = -\frac{1}{2}(\vec{F}^* \cdot \vec{n})(-q_0 \cos(\theta) - q_3 \sin(\theta)) \quad (5.113b)$$

$$q_2' = -\frac{1}{2}(\vec{F}^* \cdot \vec{n})(q_3 \cos(\theta) - q_0 \sin(\theta)) \quad (5.113c)$$

$$q_3' = -\frac{1}{2}(\vec{F}^* \cdot \vec{n})(-q_2 \cos(\theta) + q_1 \sin(\theta)) \quad (5.113d)$$

$$G' = G(\vec{F}^* \cdot \vec{v}) \quad (5.113e)$$

$$\rho'' + \rho = -(\vec{F}^* \cdot \vec{u})\rho - (\vec{F}^* \cdot \rho' \vec{v}) \quad (5.113f)$$

$$t' = \frac{1}{G \rho^2}. \quad (5.113g)$$

The Newtonian equations (5.97a) and (5.97b) are equivalent to the system of dimension eight made of Eqs. (5.113a), (5.113b), (5.113c), (5.113d), (5.113e), (5.113f) and (5.113g), where the unknowns should satisfy the constraint of normalization for the quaternion, $|\vec{q}| = 1$. From a theoretical point of view, this last formulation has a simple physical interpretation because of its inherent linearity. The dynamical model described by the equations above is denoted as *pirq*.

5.3.4.2 Characteristics of the formulations

Formulation *piq* has the property that singularities in inclination have been removed from the equations. Besides, the orbital frame defining the position and velocity vectors is determined by means of the quaternion defining the ideal frame, which in orbital problems varies very slowly, such that the quaternion is nearly constant in time near unity.

Formulations *pdr* and *pirq* use projective coordinates and include regularization, allowing thereby the use of constant stepsize for numerical integration. Furthermore, in formulation *pirq* quaternions and the ideal frame have been considered, thereby removing all singularities.

THE DROMO REGULARIZATION

DROMO is an orbital propagator developed in 2000 by the Space Dynamics Group at the Technical University of Madrid (Peláez, Hedo, and Andres [62]). This special perturbation method, based on the Hansen's ideal frame concept (Hansen [40]), is characterized by eight ordinary differential equations. In the next lines the classic theory of DROMO is developed, the fundamentals of the method are explained and its equations are derived.

6.1 BASIC ELEMENTS OF THE THEORY

A particle M with mass m moves with respect to the inertial frame $Ox_1y_1z_1$. The reference $Ox_1y_1z_1$ is defined as rigid body (1). It is assumed that O is the center of mass of a solar system body (the Earth, the Sun, a planet, etc) and the particle M is acted upon by: 1) the main term of the gravitational attraction of that body and, 2) the remaining forces, which are considered as a perturbation whose net result is $\vec{F}_p$; this force includes all orbital perturbations that may be acting on the particle (gravitational perturbations, solar radiation pressure, perturbation of a third body, propulsive systems, etc).

The motion is governed by the equation:

$$m \frac{d^2 \vec{x}}{dt^2} = -\frac{m\mu}{|\vec{x}|^3} \vec{x} + \vec{F}_p. \quad (6.1)$$

The integration of (6.1) should be carried out from known initial conditions

$$\text{at } t = 0: \quad \vec{x}(0) = \vec{x}_0, \quad \vec{v}(0) = \vec{v}_0, \quad (6.2)$$

where $\vec{x}$ is the position vector, $\vec{v} = \frac{d\vec{x}}{dt}$ the velocity vector and μ the standard gravitational parameter.

Let $Mxyz$ the reference frame which accompanies the particle in its motion; it is called orbital frame and it is defined as follows:

- The axis Mx is the prolongation of the radius vector $\vec{x}$ of the particle.
- The plane Mxy is defined by the velocity vector $\dot{\vec{x}} = \vec{v}$ of the particle and it is oriented such that $\vec{v} \cdot \vec{j} \geq 0$.
- The axis Mz is orthogonal to the previous ones and it is chosen to define a right-handed frame.

The plane Mxy is the orbital plane; in the pure Keplerian motion ($\vec{F}_p = \vec{0}$) it is fixed to the inertial frame $Ox_1y_1z_1$ and it contains the trajectory of M which is a planar curve. However, if due to the existence of perturbations the motion of the particle M is not Keplerian, the trajectory will not be flat; in general, it will be a warped curve.

For several reasons it is more convenient to use, instead of the orbital frame $Mxyz$ the frame $Oxyz$ (see Fig. 6.1) such that the only difference between both is the origin of coordinates. Obviously, the plane Oxy matches geometrically with Mxy and, thus, we can also call it orbital plane; however, kinematically, they are different planes because there is a relative motion between them.

Figure 6.1: Oxyz frame.

6.2 TWO-BODY PROBLEM

The two-body problem serves as the base for the DROMO formulation; it is presented when there are no perturbations, that is, when $\vec{F}_p = \vec{0}$. The motion, then, is governed by the equation

$$\frac{d^2 \vec{x}}{dt^2} = -\frac{\mu}{|\vec{x}|^3} \vec{x}, \quad (6.3)$$

which admits an analytical solution.

In the classic formulation of DROMO, the orbital frame at the initial instant plays an important role. However, the reference system $Ox_0y_0z_0$, defined as the rigid body (0), is more useful; its axes have the same directions and senses than the axes of the initial orbital frame. In the two-body problem, the axes $Ox_0y_0z_0$ are also inertial, since the orbital plane Oxy is fixed. In fact, some authors choose the inertial frame $Ox_1y_1z_1$ so that it coincides with the axes $Ox_0y_0z_0$ since some advantages are derived from it. In this analysis, however, and due to its intent for numerical propagation of orbits, the two references are distinguished, that is, it is assumed that the references $Ox_1y_1z_1$ and $Ox_0y_0z_0$ are different. This is convenient because it is often necessary to study the behavior of the orbit when changing the initial conditions and it is essential, in such a case, to refer the results to the same inertial frame.

6.2.1 Position of the orbital plane

In the two-body problem the orbital plane has a fixed position in the inertial space. Traditionally, the orbital plane is placed by the classic elements of the orbit: inclination i , longitude of the ascending node Ω and argument of pericenter ω , which adopt constant values in the two-body problem; nevertheless, if a perturbation acts upon M , the classic elements (i, Ω, ω) will evolve with time. Consequently, these angles are related to the frame $Ox_0y_0z_0$ at the initial instant.

Alternatively, the position of the frame $Ox_0y_0z_0$ can be fixed by a unit quaternion $\mathbf{q}_0$ of coordinates

$$\mathbf{q}_0 = \varepsilon_4^0 + (\varepsilon_1^0, \varepsilon_2^0, \varepsilon_3^0). \quad (6.4)$$

Figure 6.2: Perifocal frame.

The matrix Q_0 , expressed in terms of the quaternion $\mathbf{q}_0$, fixes the position of the frame $Ox_0y_0z_0$ linked to the orbital plane with respect to the inertial frame $Ox_1y_1z_1$. It is given by:

$$Q_0 = \begin{pmatrix} 1 - 2((\epsilon_2^0)^2 + (\epsilon_3^0)^2) & 2\epsilon_1^0 \epsilon_2^0 - 2\epsilon_4^0 \epsilon_3^0 & 2\epsilon_1^0 \epsilon_3^0 + 2\epsilon_4^0 \epsilon_2^0 \\ 2\epsilon_1^0 \epsilon_2^0 + 2\epsilon_4^0 \epsilon_3^0 & 1 - 2((\epsilon_1^0)^2 + (\epsilon_3^0)^2) & 2\epsilon_2^0 \epsilon_3^0 - 2\epsilon_4^0 \epsilon_1^0 \\ 2\epsilon_1^0 \epsilon_3^0 - 2\epsilon_4^0 \epsilon_2^0 & 2\epsilon_3^0 \epsilon_2^0 + 2\epsilon_4^0 \epsilon_1^0 & 1 - 2((\epsilon_1^0)^2 + (\epsilon_2^0)^2) \end{pmatrix}, \quad (6.5)$$

which in terms of the classical elements is defined by:

$$Q_0 = \begin{pmatrix} +\cos \Omega \cos(\theta_0 + \omega) - \cos i \sin \Omega \sin(\theta_0 + \omega) & -\cos \Omega \sin(\theta_0 + \omega) - \cos i \sin \Omega \cos(\theta_0 + \omega) & +\sin i \sin \Omega \\ +\sin \Omega \cos(\theta_0 + \omega) + \cos i \cos \Omega \sin(\theta_0 + \omega) & -\sin \Omega \sin(\theta_0 + \omega) + \cos i \cos \Omega \cos(\theta_0 + \omega) & -\sin i \cos \Omega \\ +\sin i \sin(\theta_0 + \omega) & +\sin i \cos(\theta_0 + \omega) & \cos i \end{pmatrix}, \quad (6.6)$$

where θ_0 is the initial true anomaly. This matrix relates the bases of the frames $Ox_0y_0z_0$ and $Ox_1y_1z_1$ by means of

$$[\vec{i}_0, \vec{j}_0, \vec{k}] = [\vec{i}_1, \vec{j}_1, \vec{k}_1] Q_0. \quad (6.7)$$

Besides, it is possible to change from one formulation to another just taking into account the relations:

$$\epsilon_1^0 = \sin\left(\frac{i}{2}\right) \cos\left(\frac{\Omega - \theta_0 - \omega}{2}\right) \quad (6.8a)$$

$$\epsilon_2^0 = \sin\left(\frac{i}{2}\right) \sin\left(\frac{\Omega - \theta_0 - \omega}{2}\right) \quad (6.8b)$$

$$\epsilon_3^0 = \cos\left(\frac{i}{2}\right) \sin\left(\frac{\Omega + \theta_0 + \omega}{2}\right) \quad (6.8c)$$

$$\epsilon_4^0 = \cos\left(\frac{i}{2}\right) \cos\left(\frac{\Omega + \theta_0 + \omega}{2}\right). \quad (6.8d)$$

Another reference system that naturally intervenes in the problem is the perifocal frame $Ox_p y_p z_p$ whose unit vectors are denoted by $(\vec{u}_1, \vec{u}_2, \vec{k})$ and defined as follows:

$$\vec{u}_1 = \frac{\vec{e}}{e}, \quad \vec{k} = \frac{\vec{h}}{h}, \quad \vec{u}_2 = \vec{k} \times \vec{u}_1, \quad (6.9)$$

where $\vec{e}$ is the eccentricity vector and $\vec{h}$ the angular momentum vector.

In the unperturbed two-body problem the perifocal frame is fixed in the inertial space and it is very useful because the first axis points to the perigee of the orbit (see Fig. 6.2). Alternatively, the position of the perifocal frame $Ox_p y_p z_p$ can be fixed by a unit quaternion $\mathbf{n}$ of coordinates

$$\mathbf{n} = \eta_4 + (\eta_1, \eta_2, \eta_3). \quad (6.10)$$

The matrix P_0 expressed in terms of the quaternion $\mathbf{n}$, fixes the position of the perifocal frame $Ox_p y_p z_p$ linked to the orbital plane with respect to the inertial frame $Ox_1 y_1 z_1$. It is given by:

$$P_0 = \begin{pmatrix} 1 - 2(\eta_2^2 + \eta_3^2) & 2\eta_1 \eta_2 - 2\eta_4 \eta_3 & 2\eta_1 \eta_3 + 2\eta_4 \eta_2 \\ 2\eta_1 \eta_2 + 2\eta_4 \eta_3 & 1 - 2(\eta_1^2 + \eta_3^2) & 2\eta_2 \eta_3 - 2\eta_4 \eta_1 \\ 2\eta_1 \eta_3 - 2\eta_4 \eta_2 & 2\eta_2 \eta_3 + 2\eta_4 \eta_1 & 1 - 2(\eta_1^2 + \eta_2^2) \end{pmatrix}, \quad (6.11)$$

which in terms of the classical elements is defined by:

$$P_0 = \begin{pmatrix} +\cos \Omega \cos \omega - \cos i \sin \Omega \sin \omega & -\cos \Omega \sin \omega - \cos i \sin \Omega \cos \omega & +\sin i \sin \Omega \\ +\sin \Omega \cos \omega + \cos i \cos \Omega \sin \omega & -\sin \Omega \sin \omega + \cos i \cos \Omega \cos \omega & -\sin i \cos \Omega \\ +\sin i \sin \omega & +\sin i \cos \omega & \cos i \end{pmatrix}. \quad (6.12)$$

In this case, their elements only depend on the classic elements of the orbit (i, Ω, ω). This matrix relates the bases of the frames $Ox_p y_p z_p$ and $Ox_1 y_1 z_1$ by means of

$$[\vec{u}_1, \vec{u}_2, \vec{k}] = [\vec{i}_1, \vec{j}_1, \vec{k}_1] P_0. \quad (6.13)$$

Besides, it is possible to change from one formulation to another just taking into account the relations:

$$\eta_1 = \sin\left(\frac{i}{2}\right) \cos\left(\frac{\Omega - \omega}{2}\right) \quad (6.14a)$$

$$\eta_2 = \sin\left(\frac{i}{2}\right) \sin\left(\frac{\Omega - \omega}{2}\right) \quad (6.14b)$$

$$\eta_3 = \cos\left(\frac{i}{2}\right) \sin\left(\frac{\Omega + \omega}{2}\right) \quad (6.14c)$$

$$\eta_4 = \cos\left(\frac{i}{2}\right) \cos\left(\frac{\Omega + \omega}{2}\right). \quad (6.14d)$$

Note that the matrix P_0 matches with the matrix that results from substituting $\theta_0 = 0$ in the matrix Q_0 . In other words, if the perifocal axes $Ox_p y_p z_p$ are rotated an angle θ_0 around the axis $Oz_p \equiv Oz_0$ we make them coincide with the axes $Ox_0 y_0 z_0$; thus, it turns out to be:

$$[\vec{i}_0, \vec{j}_0, \vec{k}] = [\vec{u}_1, \vec{u}_2, \vec{k}] R_3(\theta_0) \quad (6.15)$$

with

$$R_3(\theta_0) = \begin{pmatrix} +\cos(\theta_0) & -\sin(\theta_0) & 0 \\ +\sin(\theta_0) & +\cos(\theta_0) & 0 \\ 0 & 0 & 1 \end{pmatrix}. \quad (6.16)$$

Therefore, the relation between the matrix Q_0 that fixes the position of the axes $Ox_0 y_0 z_0$ and the matrix P_0 that fixes the position of the axes $Ox_p y_p z_p$ is:

$$Q_0 = P_0 R_3(\theta_0). \quad (6.17)$$

Figure 6.3: Kinematic elements of the orbital plane.

6.3 KINEMATICS OF THE ORBITAL PLANE

We consider again the perturbed motion, where an additional perturbing force $\vec{F}_p \neq \vec{0}$ is considered to act upon M . Its presence has a fundamental consequence: the orbital plane begins to move, whereas in the two-body problem it occupied an immobile position in the inertial frame. The way it evolves depends on $\vec{F}_p$ such that two different situations can arise:

- $\vec{F}_p$ contained in the orbital plane ($\vec{F}_p \cdot \vec{k} = 0$). In this case, the orbital plane remains motionless in the inertial space. The trajectory of M is flat, although it will not be Keplerian.
- $\vec{F}_p$ with a component normal to the orbital plane ($\vec{F}_p \cdot \vec{k} \neq 0$). In this case, the orbital plane will rotate in the inertial space.

The basic kinematic elements (see Fig. 8.2) to highlight in the problem are:

- The inertial frame $Ox_1y_1z_1$; it is defined as rigid body (1).
- The frame $Ox_0y_0z_0$ linked to the orbital plane and fixed by the initial conditions; it is defined as rigid body (0) and referred to as departure frame.
- The perifocal frame $Ox_py_pz_p$ linked to the orbital frame and fixed by the initial conditions; it is also rigid body (0). It is also called perifocal departure frame. Its first vector $\vec{u}_1$ points to the direction of $\vec{e}_0$.
- The frame $Oxyz$ that follows the particle M in its motion and whose plane Oxy matches with the orbital plane Ox_0y_0 ; It is denoted as rigid body (3).

The particle M is defined as rigid body (2). Note that the motion of interest is the 2/1 motion, that is, the motion of the particle M with respect to the inertial frame.

6.3.1 Definition of the orbital plane

When the orbital plane moves and occupies different positions in the inertial space, a difficulty arises in the very definition of the orbital plane itself. From a geometrical point of view, the definition of the orbital plane does not cause problems and is uniquely determined. However, from a kinematic point of view a small difficulty arises.

For instance, consider N sheets of paper, all of them contained in the orbital plane but sliding relative to each other; geometrically they all define a single plane and are, therefore, equivalent. However, from a kinematic point of view, each of the sheets is a different rigid body with a different motion from the others. To distinguish one sheet from another, what is done is to number the sheets with an index $i = 1, \dots, N$ and to link to each of them a reference system $C_i x_i y_i z_i$, $i = 1, \dots, N$; thus, there will be N different reference systems. The relative motion between the sheets would be the composition of: 1) one translation parallel to the orbital plane; and 2) one rotation within the orbital plane.

Let us think that all the reference systems have in common the origin O , such that the relative motion between the sheets would only be a planar rotation within the orbital plane. We would have, in this case, N reference systems $Ox_i y_i z_i$, $i = 1, \dots, N$ whose planes $Ox_i y_i$ would coincide with the orbital plane; all of them would have the same unit vector along the axis Oz_i , namely the unit vector $\vec{k}$.

As a consequence, if $\vec{\omega}_i$ is the angular velocity of the reference $Ox_i y_i z_i$, $i = 1, \dots, N$ it is verified that:

$$\frac{d\vec{k}}{dt} = \vec{\omega}_i \times \vec{k}, \quad \forall i = 1, \dots, N. \quad (6.18)$$

Applying the cross product with $\vec{k}$ yields:

$$\vec{\omega}_i = \underbrace{\vec{k} \times \dot{\vec{k}}}_{\parallel \text{ to the orbital plane}} + \underbrace{\vec{k} (\vec{\omega}_i \cdot \vec{k})}_{\perp \text{ to the orbital plane}}. \quad (6.19)$$

That is, all angular velocities $\vec{\omega}_i$, $i = 1, \dots, N$ have the same component parallel to the plane and differ only in the component perpendicular to the orbital plane. To identify, from a kinematic point of view, the sheet that coincides with the orbital plane, we choose the one whose angular velocity is contained in the orbital plane itself. This reference system is the rigid body (0) that can be described through the axes $Ox_0 y_0 z_0$, or through the axes $Ox_p y_p z_p$. Usually, the departure frame is used; Hansen referred to this reference system as ideal frame. Thus, the orbital plane is the plane $Ox_0 y_0$ linked to the axes $Ox_0 y_0 z_0$. But since the axes of the perifocal reference $Ox_p y_p z_p$, fixed by the initial conditions, are also rigidly linked to the rigid body (0), these axes can also be used instead of the $Ox_0 y_0 z_0$.

Regardless of the axes that are chosen, the angular velocity of the rigid body (0) is:

$$\vec{\omega}_{01} = \vec{k} \times \dot{\vec{k}} \quad (6.20)$$

and as we will see, it depends on the perturbation force $\vec{F}_p$.

6.3.2 Composition of motions

The theory of composition of motions allows to write:

$$\vec{v}_{21}^M = \vec{v}_{20}^M + \vec{v}_{01}^M, \quad (6.21)$$

where $\vec{v}_{20}^M$ is the velocity of M relative to the orbital plane and $\vec{v}_{01}^M$ the drag velocity of such plane.

Note that the motion of the orbital plane is a pure rotation, since the origin O is fixed. In this rotation the essential element is the angular velocity $\vec{\omega}_{01}$, which always points along the axis Ox , that is

$$\vec{\omega}_{01} = \omega_{01} \vec{i} \quad (6.22)$$

and its intensity ω_{01} depends on the perturbing force $\vec{F}_p$.

Therefore, the instantaneous axis of the drag motion 0/1 is the axis Ox and the drag velocity $\vec{v}_{01}^M$ of the point M turns out to be null. Consequently, for the particle M it is verified that:

$$\vec{v}_{21}^M = \vec{v}_{20}^M. \quad (6.23)$$

In Fig. 8.2 the angle that the axis Ox forms with the vector $\vec{e}_0$ (initial eccentricity vector) is denoted as σ , and is different from the true anomaly θ . The true anomaly is the angle that the axis Ox forms with the eccentricity vector $\vec{e}$, which changes with time due to the perturbation $\vec{F}_p$. Nevertheless, the value σ_0 matches with the true anomaly at the initial instant:

$$\sigma_0 = \theta_0. \quad (6.24)$$

Note that σ_0 and θ_0 values are determined by the initial conditions regardless of whether or not there is a perturbation. However, the values of $\sigma(t)$ and $\theta(t)$ depend on the existence of the perturbation. If there is no perturbation then $\sigma(t) \equiv \theta(t)$, $\forall t \geq 0$, that is, $\sigma(t)$ matches the true anomaly during the motion. If there is perturbation, however, $\sigma(t)$ and $\theta(t)$ evolve differently.

The axes $Oxyz$ share with the $Ox_0y_0z_0$ the origin O and the plane $Oxy \equiv Ox_0y_0$. The motion 3/0 consists of a rotation around the common normal (axis Oz) with angular velocity and angular acceleration given by:

$$\vec{\omega}_{30} = \dot{\sigma} \vec{k}, \quad \vec{\alpha}_{30} = \ddot{\sigma} \vec{k}. \quad (6.25)$$

Obviously, the theory of composition of motions allows to determine the inertial angular velocity of the axes $Oxyz$:

$$\vec{\omega}_{31} = \vec{\omega}_{30} + \vec{\omega}_{01} = \dot{\sigma} \vec{k} + \omega_{01} \vec{i}. \quad (6.26)$$

At the initial instant $t = 0$ the initial values $\vec{x}_0$ and $\vec{v}_0$ define the classical elements of the orbit and the reference systems $Ox_0y_0z_0$ and $Ox_py_pz_p$. For later instants, $t > 0$, the reference systems $Ox_0y_0z_0$ and $Ox_py_pz_p$ will move with a pure rotation $\vec{\omega}_{01}$ around the axis Ox and the particle M will move through the orbital plane due to all the forces acting upon it. Following the particle, the axes $Oxyz$ rotate with respect to the orbital plane with an angular velocity $\vec{\omega}_{30} = \dot{\sigma} \vec{k}$. The eccentricity vector $\vec{e}$ evolves within the orbital plane. As a consequence, the perifocal reference at time $t > 0$ has the first axis pointing in the direction of $\vec{e}$, while the initial perifocal reference $Ox_py_pz_p$ has the first axis pointing in the direction of $\vec{e}_0$.

6.3.3 Dimensionless quantities

For the subsequent analysis, the characteristic variables L_c and ω_c are introduced as follows:

longitudes	→	dimensionless with	L_c
times	→	dimensionless with	$1/\omega_c$
velocities	→	dimensionless with	$\omega_c L_c$
angular momentum	→	dimensionless with	$\omega_c L_c^2$
accelerations	→	dimensionless with	$\omega_c^2 L_c$

In the original version of DROMO the following characteristic values are used

$$L_c = |\vec{x}_0|, \quad \omega_c = \sqrt{\frac{\mu}{L_c^3}} \quad (6.27)$$

and the dimensionless time $\tau = \omega_c t$ is introduced.

From this moment the variables used will be dimensionless variables, unless otherwise indicated. Thus, $(\vec{x}, \vec{v})$ will be the dimensionless quantities fixing the position and velocity of M in the inertial frame.

Once the dimensionless variables are introduced, the Eq. (6.1) governing the motion takes the form

$$\frac{d^2 \vec{x}}{d\tau^2} = -\frac{1}{r^3} \vec{x} + \vec{a}_p(\vec{x}, \vec{v}, \tau), \quad (6.28)$$

where $r = |\vec{x}|$, and $\vec{a}_p = \frac{\vec{F}_p}{mL_c \omega_c^2}$. The vector $\vec{a}_p$ represents the perturbing acceleration acting upon M (in dimensionless form) and, in the most general possible case, it is a function of the dynamical state of the system: $\vec{a}_p(\vec{x}, \vec{v}, \tau)$.

The equation governing the time evolution of the angular momentum, per unit of mass, of the particle must be added to Eq. (6.28):

$$\frac{d\vec{h}}{d\tau} = \vec{x} \times \vec{a}_p. \quad (6.29)$$

6.3.4 Calculation of the angular velocities $\vec{\omega}_{01}$ and $\vec{\omega}_{31}$

We start by estimating the angular velocity of the rigid body (3), that is, of the axes $Oxyz$ moving with M . Note that the unit vectors $(\vec{i}, \vec{j}, \vec{k})$ of these axes are defined by:

$$\vec{i} = \frac{\vec{x}}{|\vec{x}|} = \frac{\vec{x}}{r}, \quad \vec{k} = \frac{\vec{x} \times \vec{v}}{|\vec{x} \times \vec{v}|} = \frac{\vec{h}}{h}, \quad \vec{j} = \vec{k} \times \vec{i}. \quad (6.30)$$

Differentiating the unit vectors $(\vec{i}, \vec{k})$ of this frame with respect to the time, we obtain:

$$\frac{d\vec{i}}{d\tau} = \frac{1}{r} \left[\vec{v} - \frac{dr}{d\tau} \vec{i} \right], \quad \frac{d\vec{k}}{d\tau} = \frac{1}{h} \left[\frac{d\vec{h}}{d\tau} - \frac{dh}{d\tau} \vec{k} \right]. \quad (6.31)$$

However, if $(\omega_1, \omega_2, \omega_3)$ are the components of $\vec{\omega}_{31}$ in the body (3) frame coordinates

$$\vec{\omega}_{31} = \omega_1 \vec{i} + \omega_2 \vec{j} + \omega_3 \vec{k} \quad (6.32)$$

and taking into account that $Oxyz$ is a rigid body, the following relations are verified:

$$\frac{d\vec{i}}{d\tau} = \vec{\omega}_{31} \times \vec{i} = \begin{vmatrix} \vec{i} & \vec{j} & \vec{k} \\ \omega_1 & \omega_2 & \omega_3 \\ 1 & 0 & 0 \end{vmatrix} = \omega_3 \vec{j} - \omega_2 \vec{k} \quad (6.33a)$$

$$\frac{d\vec{k}}{d\tau} = \vec{\omega}_{31} \times \vec{k} = \begin{vmatrix} \vec{i} & \vec{j} & \vec{k} \\ \omega_1 & \omega_2 & \omega_3 \\ 0 & 0 & 1 \end{vmatrix} = \omega_2 \vec{i} - \omega_1 \vec{j}. \quad (6.33b)$$

Identifying terms with the previously obtained derivatives of $(\vec{i}, \vec{k})$, the components of $\vec{\omega}_{31}$ are deduced:

$$\frac{1}{r} \left[\vec{v} - \frac{dr}{d\tau} \vec{i} \right] = \omega_3 \vec{j} - \omega_2 \vec{k} \Rightarrow \begin{cases} \omega_3 = \frac{1}{r} (\vec{v} \cdot \vec{j}) \\ \omega_2 = 0 \end{cases} \quad (6.34a)$$

$$\frac{1}{h} \left[\frac{d\vec{h}}{d\tau} - \frac{dh}{d\tau} \vec{k} \right] = \omega_2 \vec{i} - \omega_1 \vec{j} \Rightarrow \begin{cases} \omega_2 = \frac{1}{h} \left(\frac{d\vec{h}}{d\tau} \cdot \vec{i} \right) = 0 \\ \omega_1 = -\frac{1}{h} \left(\vec{j} \cdot \frac{d\vec{h}}{d\tau} \right). \end{cases} \quad (6.34b)$$

With the help of the Eq. (6.29), the ω_1 and ω_3 components of the angular velocity can be rewritten as follows:

$$\omega_1 = -\frac{1}{h} \left(\vec{j} \cdot \frac{d\vec{h}}{d\tau} \right) = -\frac{1}{h} \vec{j} \cdot (\vec{x} \times \vec{a}_p) = \frac{r}{h} (\vec{a}_p \cdot \vec{k}) \quad (6.35a)$$

$$\omega_3 = \frac{1}{r} (\vec{v} \cdot \vec{j}) = -\frac{1}{r} \vec{v} \cdot (\vec{i} \times \vec{k}) = \frac{1}{r^2} (\vec{k} \cdot \vec{h}) = \frac{h}{r^2}. \quad (6.35b)$$

The derivatives of the unit vectors turn out to be:

$$\frac{d\vec{i}}{d\tau} = \omega_3 \vec{j} \quad (6.36a)$$

$$\frac{d\vec{j}}{d\tau} = -\omega_3 \vec{i} + \omega_1 \vec{k} \quad (6.36b)$$

$$\frac{d\vec{k}}{d\tau} = -\omega_1 \vec{j}, \quad (6.36c)$$

where $\omega_3 = \frac{h}{r^2}$ and $\omega_1 = \frac{r}{h} (\vec{a}_p \cdot \vec{k})$.

The calculations made provide the following expression for $\vec{\omega}_{31}$:

$$\vec{\omega}_{31} = \frac{r}{h} (\vec{a}_p \cdot \vec{k}) \vec{i} + \frac{h}{r^2} \vec{k}. \quad (6.37)$$

If it is compared with the relation

$$\vec{\omega}_{31} = \vec{\omega}_{30} + \vec{\omega}_{01} = \dot{\sigma} \vec{k} + \omega_{01} \vec{i}, \quad (6.38)$$

the following relations are deduced:

$$\frac{d\sigma}{d\tau} = \frac{h}{r^2}, \quad \vec{\omega}_{01} = \frac{r}{h} (\vec{a}_p \cdot \vec{k}) \vec{i}, \quad (6.39)$$

which provide the time derivative of the angle σ and the angular velocity $\vec{\omega}_{01}$ of the orbital plane. As already mentioned, the orbital plane remains fixed if the perturbation $\vec{a}_p$ is parallel to the orbital plane.

6.3.5 Position of the axes Oxyz

The study of the evolution of the particle M implies the study of the evolution of the axes Oxyz that follow it. The axes Oxyz are easily located with respect to the axes $Ox_0y_0z_0$:

$$[\vec{i}, \vec{j}, \vec{k}] = [\vec{i}_0, \vec{j}_0, \vec{k}_0] R_3(\sigma - \sigma_0), \quad (6.40)$$

where the matrix $R_3(\sigma - \sigma_0)$ is given by:

$$R_3(\sigma - \sigma_0) = \begin{pmatrix} \cos(\sigma - \sigma_0) & -\sin(\sigma - \sigma_0) & 0 \\ \sin(\sigma - \sigma_0) & \cos(\sigma - \sigma_0) & 0 \\ 0 & 0 & 1 \end{pmatrix}. \quad (6.41)$$

Since the axes $Ox_0y_0z_0$ are located, with respect to the inertial frame $Ox_1y_1z_1$, through the matrix Q_0 defined in Eq. (6.6) and (6.5):

$$[\vec{i}_0, \vec{j}_0, \vec{k}_0] = [\vec{i}_1, \vec{j}_1, \vec{k}_1] Q_0, \quad (6.42)$$

the axes Oxyz are located, with respect to the inertial frame $Ox_1y_1z_1$, through the matrix $Q = Q_0 R_3(\sigma - \sigma_0)$:

$$[\vec{i}, \vec{j}, \vec{k}] = [\vec{i}_1, \vec{j}_1, \vec{k}_1] Q. \quad (6.43)$$

Therefore, since $\mathbf{q}_0$ is the quaternion associated to the matrix Q_0 defined in Eq. (6.5) and $\mathbf{r}_3$ the quaternion associated to the matrix $R_3(\sigma - \sigma_0)$:

$$\mathbf{r}_3 = \cos\left(\frac{\sigma - \sigma_0}{2}\right) + \left(0, 0, \sin\left(\frac{\sigma - \sigma_0}{2}\right)\right), \quad (6.44)$$

the matrix $Q = Q_0 R_3(\sigma - \sigma_0)$ is related to the quaternion product $\mathbf{q} = \mathbf{q}_0 \cdot \mathbf{r}_3$.

If:

$$\mathbf{q} = \varepsilon_4 + (\varepsilon_1, \varepsilon_2, \varepsilon_3) \quad (6.45)$$

are the components of such quaternion, once the product is made, we get:

$$\varepsilon_4 = \varepsilon_4^0 \cos\left(\frac{\sigma - \sigma_0}{2}\right) - \varepsilon_3^0 \sin\left(\frac{\sigma - \sigma_0}{2}\right) \quad (6.46a)$$

$$\varepsilon_1 = \varepsilon_1^0 \cos\left(\frac{\sigma - \sigma_0}{2}\right) + \varepsilon_2^0 \sin\left(\frac{\sigma - \sigma_0}{2}\right) \quad (6.46b)$$

$$\varepsilon_2 = \varepsilon_2^0 \cos\left(\frac{\sigma - \sigma_0}{2}\right) - \varepsilon_1^0 \sin\left(\frac{\sigma - \sigma_0}{2}\right) \quad (6.46c)$$

$$\varepsilon_3 = \varepsilon_3^0 \cos\left(\frac{\sigma - \sigma_0}{2}\right) + \varepsilon_4^0 \sin\left(\frac{\sigma - \sigma_0}{2}\right). \quad (6.46d)$$

The process followed to place the axes $Oxyz$ has been based on the axes $Ox_0y_0z_0$. A similar scheme is obtained if the axes $Ox_p y_p z_p$ are used instead of $Ox_0y_0z_0$. The axes $Oxyz$ are obtained from the axes $Ox_p y_p z_p$ through a turn of an angle σ around Oz ; as the axes $Ox_p y_p z_p$ are located, with respect to the inertial frame $Ox_1y_1z_1$, by means of the matrix P_0 collected in Eqs. (6.11) and (6.12), in this case it has:

$$[\vec{i}, \vec{j}, \vec{k}] = [\vec{u}_1, \vec{u}_2, \vec{k}] R_3(\sigma) \quad (6.47a)$$

$$[\vec{u}_1, \vec{u}_2, \vec{k}] = [\vec{i}_1, \vec{j}_1, \vec{k}_1] P_0, \quad (6.47b)$$

where the matrix $R_3(\sigma)$ is given by:

$$R_3(\sigma) = \begin{pmatrix} \cos(\sigma) & -\sin(\sigma) & 0 \\ \sin(\sigma) & \cos(\sigma) & 0 \\ 0 & 0 & 1 \end{pmatrix}. \quad (6.48)$$

Consequently, the axes $Oxyz$ are placed, with respect to the inertial frame $Ox_1y_1z_1$, through the matrix $Q = P_0 R_3(\sigma)$:

$$[\vec{i}, \vec{j}, \vec{k}] = [\vec{i}_1, \vec{j}_1, \vec{k}_1] Q. \quad (6.49)$$

Therefore, since $\mathbf{n}$ is the quaternion associated to the matrix P_0 of Eq. (6.11), if we define $\mathbf{v}_3$ as the quaternion related to the matrix $R_3(\sigma)$:

$$\mathbf{v}_3 = \cos\left(\frac{\sigma}{2}\right) + \left(0, 0, \sin\left(\frac{\sigma}{2}\right)\right), \quad (6.50)$$

the matrix $Q = P_0 R_3(\sigma)$ is associated to the product quaternion $\mathbf{q} = \mathbf{n} \cdot \mathbf{v}_3$.

If (6.45) are the components of such quaternion $\mathbf{q}$, once the product is made, we obtain:

$$\varepsilon_4 = \eta_4 \cos\left(\frac{\sigma}{2}\right) - \eta_3 \sin\left(\frac{\sigma}{2}\right) \quad (6.51a)$$

$$\varepsilon_1 = \eta_1 \cos\left(\frac{\sigma}{2}\right) + \eta_2 \sin\left(\frac{\sigma}{2}\right) \quad (6.51b)$$

$$\varepsilon_2 = \eta_2 \cos\left(\frac{\sigma}{2}\right) - \eta_1 \sin\left(\frac{\sigma}{2}\right) \quad (6.51c)$$

$$\varepsilon_3 = \eta_3 \cos\left(\frac{\sigma}{2}\right) + \eta_4 \sin\left(\frac{\sigma}{2}\right). \quad (6.51d)$$

6.3.6 Time evolution of the axes Oxyz

The time evolution of the axes Oxyz (rigid body (3)) can be proposed directly. If $2\Omega_{31}$ is the vector quaternion associated to the angular velocity $\vec{\omega}_{31}$, and $\mathbf{q}$ the quaternion that fixes the position of the axes Oxyz with respect to the inertial axes, the time evolution of $\mathbf{q}$ is given by:

$$\frac{d\mathbf{q}}{d\tau} = \frac{1}{2}\mathbf{q}(2\Omega_{31}), \quad (6.52)$$

which developed it leads to:

$$\frac{d\varepsilon_1}{d\tau} = \frac{1}{2} \left\{ \varepsilon_4 \omega_{01} + \varepsilon_2 \frac{d\sigma}{d\tau} \right\} \quad (6.53a)$$

$$\frac{d\varepsilon_2}{d\tau} = \frac{1}{2} \left\{ \varepsilon_3 \omega_{01} - \varepsilon_1 \frac{d\sigma}{d\tau} \right\} \quad (6.53b)$$

$$\frac{d\varepsilon_3}{d\tau} = \frac{1}{2} \left\{ -\varepsilon_2 \omega_{01} + \varepsilon_4 \frac{d\sigma}{d\tau} \right\} \quad (6.53c)$$

$$\frac{d\varepsilon_4}{d\tau} = \frac{1}{2} \left\{ -\varepsilon_1 \omega_{01} - \varepsilon_3 \frac{d\sigma}{d\tau} \right\}. \quad (6.53d)$$

6.4 DYNAMICS OF THE ORBITAL PLANE

The goal is to determine the time evolution of the particle M with respect to the inertial frame $Ox_1y_1z_1$. The motion of interest is, then, what we call 2/1 with the numbering of rigid bodies introduced previously. To obtain this motion in DROMO we decompose the motion in:

$$2/1 \rightarrow 2/0 + 0/1$$

taking advantage of the fact that each of these motions (2/0 and 0/1) are simpler than the original absolute motion.

6.4.1 Drag motion 0/1

The motion 0/1 is the motion of the orbital plane. The goal is to locate the position of the orbital plane in a generic instant and to place the axes Oxyz that follow the particle M along its trajectory.

For the description of the motion 0/1 there are two alternatives available: use the axes $Ox_0y_0z_0$, or use the axes $Ox_p y_p z_p$ as axes rigidly attached to the rigid body (0).

Using the axes $Ox_0y_0z_0$: in this case the position of the axes $Ox_0y_0z_0$ with respect to the inertial frame is described by the quaternion $\mathbf{q}_0$ whose time evolution is governed by the equations:

$$\frac{d\varepsilon_1^0}{d\tau} = \frac{1}{2}\omega_{01} \{ \varepsilon_4^0 \cos(\sigma - \sigma_0) - \varepsilon_3^0 \sin(\sigma - \sigma_0) \} \quad (6.54a)$$

$$\frac{d\varepsilon_2^0}{d\tau} = \frac{1}{2}\omega_{01} \{ \varepsilon_3^0 \cos(\sigma - \sigma_0) + \varepsilon_4^0 \sin(\sigma - \sigma_0) \} \quad (6.54b)$$

$$\frac{d\varepsilon_3^0}{d\tau} = \frac{1}{2}\omega_{01} \{ -\varepsilon_2^0 \cos(\sigma - \sigma_0) + \varepsilon_1^0 \sin(\sigma - \sigma_0) \} \quad (6.54c)$$

$$\frac{d\varepsilon_4^0}{d\tau} = \frac{1}{2}\omega_{01} \{ -\varepsilon_1^0 \cos(\sigma - \sigma_0) - \varepsilon_2^0 \sin(\sigma - \sigma_0) \} \quad (6.54d)$$

and should be integrated from the initial conditions determined by the initial dynamic state (in $\tau = 0$) of the particle M : $\vec{x}_0$ and $\vec{v}_0$. In them the angular velocity ω_{01} is:

$$\omega_{01} = \frac{r}{h} (\vec{a}_p \cdot \vec{k}). \quad (6.55)$$

The components of the quaternion $\mathbf{q}_0$ verify the following useful relations to control the quality of the integration:

$$\varepsilon_1^0 \frac{d\varepsilon_1^0}{d\tau} + \varepsilon_2^0 \frac{d\varepsilon_1^0}{d\tau} + \varepsilon_3^0 \frac{d\varepsilon_3^0}{d\tau} + \varepsilon_4^0 \frac{d\varepsilon_4^0}{d\tau} = 0 \quad (6.56a)$$

$$-\varepsilon_3^0 \frac{d\varepsilon_4^0}{d\tau} + \varepsilon_2^0 \frac{d\varepsilon_1^0}{d\tau} - \varepsilon_1^0 \frac{d\varepsilon_2^0}{d\tau} + \varepsilon_4^0 \frac{d\varepsilon_3^0}{d\tau} = 0. \quad (6.56b)$$

Once the quaternion $\mathbf{q}_0$ is known, the axes $Oxyz$ are placed, with respect to the inertial frame, by means of the quaternion $\mathbf{q} = \mathbf{q}_0 \cdot \mathbf{r}_3$ according to the Eqs. (6.46a), (6.46b), (6.46c) and (6.46d).

Using the axes $Ox_p y_p z_p$: in this case the position of the axes $Ox_p y_p z_p$ with respect to the inertial frame is described by the quaternion $\mathbf{n}$ whose time evolution is governed by the equations:

$$\frac{d\eta_1}{d\tau} = \frac{1}{2} \omega_{01} \{ \eta_4 \cos(\sigma) - \eta_3 \sin(\sigma) \} \quad (6.57a)$$

$$\frac{d\eta_2}{d\tau} = \frac{1}{2} \omega_{01} \{ \eta_3 \cos(\sigma) + \eta_4 \sin(\sigma) \} \quad (6.57b)$$

$$\frac{d\eta_3}{d\tau} = \frac{1}{2} \omega_{01} \{ -\eta_2 \cos(\sigma) + \eta_1 \sin(\sigma) \} \quad (6.57c)$$

$$\frac{d\eta_4}{d\tau} = \frac{1}{2} \omega_{01} \{ -\eta_1 \cos(\sigma) - \eta_2 \sin(\sigma) \}, \quad (6.57d)$$

where ω_{01} is the given by Eq. (6.55). These equations should be integrated from the initial conditions determined by the initial dynamic state (in $\tau = 0$) of the particle M : $\vec{x}_0$ and $\vec{v}_0$.

The components of the quaternion $\mathbf{n}$ verify the following useful relations to control the quality of the integration:

$$\eta_1 \frac{d\eta_1}{d\tau} + \eta_2 \frac{d\eta_1}{d\tau} + \eta_3 \frac{d\eta_3}{d\tau} + \eta_4 \frac{d\eta_4}{d\tau} = 0 \quad (6.58a)$$

$$\eta_2 \frac{d\eta_1}{d\tau} - \eta_1 \frac{d\eta_2}{d\tau} + \eta_4 \frac{d\eta_3}{d\tau} - \eta_3 \frac{d\eta_4}{d\tau} = 0 \quad (6.58b)$$

Once the quaternion $\mathbf{n}$ is known, the axes $Oxyz$ are placed, with respect to the inertial frame, through the quaternion $\mathbf{q} = \mathbf{n} \cdot \mathbf{v}_3$ according to the Eqs. (6.51a), (6.51b), (6.51c) and (6.51d).

The initial conditions for the quaternions $\mathbf{q}_0$ and $\mathbf{n}$ are different. The initial value of $\mathbf{n}$ is obtained from the matrix $P_0(0)$ where only the classical elements (i, Ω, ω) of the initial osculating orbit, determined by $\vec{x}_0$ and $\vec{v}_0$, intervene. However, the initial value of $\mathbf{q}_0$ is obtained from the matrix $Q_0(0)$ where the classical elements of the initial osculating orbit and $\theta_0 = \sigma_0$ intervene.

6.4.2 Relative motion 2/0

In the relative motion 2/0 the forces acting upon the particle are: 1) the main gravitational term; 2) the acceleration of perturbation $\vec{a}_p$; and 3) the inertial accelerations $-\vec{\gamma}_{01}^M$ and $-2\vec{\omega}_{01} \times \vec{v}_{20}^M$. The momentum equation takes the form:

$$\vec{\gamma}_{20}^M = -\frac{1}{r^2} \vec{i} + \vec{a}_p + \{ -\vec{\gamma}_{01}^M - 2\vec{\omega}_{01} \times \vec{v}_{20}^M \}. \quad (6.59)$$

The drag acceleration $\vec{\gamma}_{01}$ takes the value:

$$\vec{\gamma}_{01}^M = \vec{\gamma}_{01}^O + \vec{\alpha}_{01} \times \vec{x} + \vec{\omega}_{01} \times (\vec{\omega}_{01} \times \vec{x}). \quad (6.60)$$

Besides, since $\vec{\omega}_{01} = \omega_{01} \vec{i}$, the angular acceleration $\vec{\alpha}_{01}$ is:

$$\vec{\alpha}_{01} = \dot{\omega}_{01} \vec{i} + \omega_{01} \omega_3 \vec{j}. \quad (6.61)$$

Consequently, taking into account that:

$$\omega_{01} = \omega_1 = \frac{r}{h} (\vec{a}_p \cdot \vec{k}), \quad \omega_3 = \frac{d\sigma}{d\tau} = \frac{h}{r^2}, \quad (6.62)$$

the value obtained for the drag acceleration is:

$$\vec{\gamma}_{01}^M = -r \omega_{01} \omega_3 \vec{k} = -(\vec{a}_p \cdot \vec{k}) \vec{k}. \quad (6.63)$$

The Coriolis acceleration takes the value:

$$2\vec{\omega}_{01} \times \vec{v}_{20}^M = 2\omega_{01} \vec{i} \times \left\{ \frac{dr}{d\tau} \vec{i} + \frac{h}{r} \vec{j} \right\}, \quad (6.64)$$

this is:

$$2\vec{\omega}_{01} \times \vec{v}_{20}^M = 2(\vec{a}_p \cdot \vec{k}) \vec{k}. \quad (6.65)$$

In summary, the resultant of the inertial accelerations is:

$$-\vec{\gamma}_{01}^M - 2\vec{\omega}_{01} \times \vec{v}_{20}^M = -(\vec{a}_p \cdot \vec{k}) \vec{k}. \quad (6.66)$$

The momentum equation turns out to be:

$$\vec{\gamma}_{20}^M = -\frac{1}{r^2} \vec{i} + \vec{a}_p - (\vec{a}_p \cdot \vec{k}) \vec{k}, \quad (6.67)$$

this is

$$\vec{\gamma}_{20}^M = -\frac{1}{r^2} \vec{i} + \vec{a}_{\parallel p}, \quad (6.68)$$

where the perturbing acceleration parallel to the orbital plane has been introduced:

$$\vec{a}_{\parallel p} = \vec{a}_p - (\vec{a}_p \cdot \vec{k}) \vec{k} = (\vec{a}_p \cdot \vec{i}) \vec{i} + (\vec{a}_p \cdot \vec{j}) \vec{j}. \quad (6.69)$$

In Eq. (6.68) the normal components of the orbital plane have disappeared. That is, the relative motion 2/0 is a planar motion. Thus, the normal component of the perturbing acceleration governs the evolution of the orbital plane, but in the relative motion the orbital plane plays no role.

Finally, taking into account that within the orbital plane (r, σ) are polar coordinates of M with origin of angles in $\vec{u}_1$, the relative acceleration $\vec{\gamma}_{20}^M$ takes the value:

$$\vec{\gamma}_{20}^M = \left[\frac{d^2 r}{d\tau^2} - r \left(\frac{d\sigma}{d\tau} \right)^2 \right] \vec{i} + \frac{1}{r} \frac{d}{d\tau} \left(r^2 \frac{d\sigma}{d\tau} \right) \vec{j}. \quad (6.70)$$

The developed momentum equation takes the form:

$$\left[\frac{d^2 r}{d\tau^2} - r \left(\frac{d\sigma}{d\tau} \right)^2 \right] \vec{i} + \frac{1}{r} \frac{d}{d\tau} \left(r^2 \frac{d\sigma}{d\tau} \right) \vec{j} = -\frac{1}{r^2} \vec{i} + (\vec{a}_p \cdot \vec{i}) \vec{i} + (\vec{a}_p \cdot \vec{j}) \vec{j} \quad (6.71)$$

and provides the relations:

$$\frac{d^2 r}{d\tau^2} = -\frac{1}{r^2} + \frac{h^2}{r^3} + (\vec{a}_p \cdot \vec{i}) \quad (6.72a)$$

$$\frac{dh}{d\tau} = r (\vec{a}_p \cdot \vec{j}). \quad (6.72b)$$

The integration of the Eqs. 6.72a and 6.72b from the initial conditions determines the relative motion. However, the formulation of DROMO is based on a more sophisticated approach that will be discussed in the following sections. To do this, the time evolution of some of the important vector magnitudes of the problem will be studied.

6.4.2.1 Evolution of the vector $\vec{h}$

The angular momentum evolves according to the equation:

$$\frac{d\vec{h}}{d\tau} = \vec{x} \times \vec{a}_p \quad (6.73)$$

expressed in dimensionless variables. Taking into account that $\vec{h} = h \vec{k}$, this equation adopts the following form:

$$\frac{dh}{d\tau} \vec{k} + h \frac{d\vec{k}}{d\tau} = \frac{dh}{d\tau} \vec{k} - h\omega_1 \vec{j} = r(\vec{i} \times \vec{a}_p). \quad (6.74)$$

Applying the dot product to Eq. (6.74) and $\vec{i}$ we obtain a triviality; if it is done with $\vec{j}$ we recover the value of ω_1 calculated previously. Finally, if we multiply with respect to $\vec{k}$ we get the equation

$$\frac{dh}{d\tau} = r(\vec{a}_p \cdot \vec{j}) \quad (6.75)$$

which takes the form

$$\frac{dh}{d\tau} = r(\vec{a}_{\parallel p} \cdot \vec{j}), \quad (6.76)$$

that shows that there is only a change in the modulus of the angular momentum if the perturbation force has a non-zero transversal component.

Equation (6.73) of the angular momentum reflects the time variation of $\vec{h}$ in the inertial space. If we are interested in obtaining the time evolution of the motion relative to the orbital plane, the Coriolis theorem should be used. If $\vec{A}$ is a time varying vector according to this theorem we have:

$$\left. \frac{d\vec{A}}{d\tau} \right|_{(1)} = \left. \frac{d\vec{A}}{d\tau} \right|_{(0)} + \vec{\omega}_{01} \times \vec{A}. \quad (6.77)$$

Applied to the angular momentum it yields:

$$\left. \frac{d\vec{h}}{d\tau} \right|_{(0)} = \left. \frac{d\vec{h}}{d\tau} \right|_{(1)} - \vec{\omega}_{01} \times \vec{h}. \quad (6.78)$$

The term $\vec{\omega}_{01} \times \vec{h}$ adopts the form:

$$\omega_{01} h \vec{i} \times \vec{k} = r(\vec{a}_p \cdot \vec{k}) \vec{i} \times \vec{k} = \vec{x} \times [(\vec{a}_p \cdot \vec{k}) \vec{k}] \quad (6.79)$$

and consequently:

$$\left. \frac{d\vec{h}}{d\tau} \right|_{(0)} = \vec{x} \times \vec{a}_p - \vec{x} \times [(\vec{a}_p \cdot \vec{k}) \vec{k}], \quad (6.80)$$

that is

$$\left. \frac{d\vec{h}}{d\tau} \right|_{(0)} = \vec{x} \times \vec{a}_{\parallel p}. \quad (6.81)$$

This equation again shows that in the motion relative to the orbital plane only the perturbing acceleration $\vec{a}_{\parallel p}$ influences.

6.4.2.2 Evolution of the vector $\vec{e}$

By definition the eccentricity vector $\vec{e}$ is given by:

$$\vec{e} = -\frac{\vec{x}}{|\vec{x}|} - \frac{1}{\mu}(\vec{h} \times \vec{v}). \quad (6.82)$$

A simple calculation shows that its time derivative is:

$$\frac{d\vec{e}}{dt} = \frac{1}{\mu} \left(\vec{a}_p \times \vec{h} + \vec{v} \times (\vec{x} \times \vec{a}_p) \right) \quad (6.83)$$

and, as expected, it is canceled when the perturbing acceleration $\vec{a}_p$ vanishes.

If dimensionless variables are used, the eccentricity vector $\vec{e}$ turns out to be:

$$\vec{e} = -\vec{i} - \vec{h} \times \vec{v}. \quad (6.84)$$

From this relation, applying the cross product with $\vec{k}$, the following equation is obtained:

$$\vec{v} = \frac{1}{h}(\vec{j} + \vec{k} \times \vec{e}), \quad (6.85)$$

which is known as the equation of the hodograph and provides the velocity vector $\vec{v}$.

Equation (6.83) rewritten in dimensionless variables turns out to be

$$\frac{d\vec{e}}{d\tau} = \vec{a}_p \times \vec{h} + \vec{v} \times (\vec{x} \times \vec{a}_p). \quad (6.86)$$

The components of the eccentricity vector in the axes Oxyz are easy to obtain. From the definition of Eq. (6.84), if the next relations are taken into account

$$\vec{x} = r\vec{i}, \quad \vec{v} = \frac{dr}{d\tau}\vec{i} + \frac{h}{r}\vec{j}, \quad (6.87)$$

it is obtained:

$$\vec{e} = \left(\frac{h^2}{r} - 1 \right) \vec{i} - h \frac{dr}{d\tau} \vec{j}. \quad (6.88)$$

However, to handle the eccentricity vector different coordinates are used:

$$\vec{e} = \zeta_1 \vec{u}_1 + \zeta_2 \vec{u}_2. \quad (6.89)$$

The unit vectors $(\vec{u}_1, \vec{u}_2)$ are rigidly attached to the orbital plane Ox_0y_0 ; in fact, the vector $\vec{u}_1$ points in the direction of the initial eccentricity vector $\vec{e}_0$. Note that the axis Ox forms an angle σ (variable with time) with $\vec{u}_1$ while the axis Ox_0 forms an angle σ_0 (constant) with $\vec{u}_1$ (see Fig. 6.4).

The relation between $(\vec{i}, \vec{j})$ and the unit vectors $(\vec{u}_1, \vec{u}_2)$ is:

$$\vec{i} = \cos(\sigma) \vec{u}_1 + \sin(\sigma) \vec{u}_2 \quad (6.90a)$$

$$\vec{j} = -\sin(\sigma) \vec{u}_1 + \cos(\sigma) \vec{u}_2. \quad (6.90b)$$

A simple change of coordinates allows to obtain the following relations:

$$\zeta_1 = \left(\frac{h^2}{r} - 1 \right) \cos(\sigma) + h \frac{dr}{d\tau} \sin(\sigma) \quad (6.91a)$$

$$\zeta_2 = \left(\frac{h^2}{r} - 1 \right) \sin(\sigma) - h \frac{dr}{d\tau} \cos(\sigma), \quad (6.91b)$$

Figure 6.4: Basis $(\vec{u}_1, \vec{u}_2)$.

that can be inverted to obtain

$$\frac{h^2}{r} = 1 + \zeta_1 \cos(\sigma) + \zeta_2 \sin(\sigma) \quad (6.92a)$$

$$\frac{dr}{d\tau} = \frac{1}{h} [\zeta_1 \sin(\sigma) - \zeta_2 \cos(\sigma)]. \quad (6.92b)$$

Obviously, from Eq. (6.89) and taking into account that the vectors $(\vec{u}_1, \vec{u}_2)$ are unit vectors, it is deduced

$$e = \sqrt{\zeta_1^2 + \zeta_2^2}. \quad (6.93)$$

If θ is the true anomaly, i.e. the angle that the position vector $\vec{r}$ forms with the eccentricity vector, it is easy to obtain the following expressions:

$$\cos(\theta) = \frac{\vec{e} \cdot \vec{r}}{e} = \frac{1}{e} \{\zeta_1 \cos(\sigma) + \zeta_2 \sin(\sigma)\} \quad (6.94a)$$

$$\sin(\theta) = \vec{r} \cdot \frac{\vec{e} \times \vec{i}}{e} = \frac{1}{e} \{\zeta_1 \sin(\sigma) - \zeta_2 \cos(\sigma)\}. \quad (6.94b)$$

Introducing the angle β through the relations

$$\cos(\beta) = \frac{\zeta_1}{e}, \quad \sin(\beta) = \frac{\zeta_2}{e}, \quad (6.95)$$

equations 6.94a and 6.94b can be written as:

$$\left. \begin{aligned} \cos(\theta) &= \cos(\sigma - \beta) \\ \sin(\theta) &= \sin(\sigma - \beta) \end{aligned} \right\} \Rightarrow \theta = \sigma - \beta. \quad (6.96)$$

Note that in the unperturbed motion the angles θ and σ coincide and $\beta = 0$ at all times. Accordingly, $\zeta_1(\tau) \equiv e_0$ and $\zeta_2(\tau) \equiv 0$. In other words, the quantities ζ_1 and ζ_2 are first integrals of the motion in the unperturbed two-body problem. However, in the perturbed motion the angle σ does not match the true anomaly θ and the difference $\sigma - \theta = \beta$ is non-zero and changes over time. The angle β (see Fig. 6.5) is the one formed by the eccentricity vector $\vec{e}$ at a generic time with the eccentricity vector $\vec{e}_0$ determined by the initial conditions; the latter is a fixed vector in the axes $Ox_0y_0z_0$ while the first moves with respect to them.

Figure 6.5: Angle β .

6.4.3 Basic formulation of DROMO

In the motion relative to the orbital plane the dynamical state is described by the variables:

$$\left(r, \frac{dr}{d\tau}, \sigma, \frac{d\sigma}{d\tau}, \tau \right) \quad (6.97)$$

and, since $r^2 \dot{\sigma} = h$, the angular momentum h can be used instead of the last derivative:

$$\left(r, \frac{dr}{d\tau}, \sigma, h, \tau \right). \quad (6.98)$$

The basic formulation of DROMO consists in using as state variables the following quantities:

$$\left(\zeta_1, \zeta_2, \zeta_3 = \frac{1}{h}, \sigma, \tau \right) \quad (6.99)$$

to describe the motion relative to the orbital plane; (ζ_1, ζ_2) are the components of the eccentricity vector defined in Eq. (6.89) and σ is the independent variable in lieu of the physical time τ which is now considered as dependent variable.

The next step is to determine the derivatives with respect to the parameter σ of the unit vectors $(\vec{i}, \vec{j}, \vec{k}, \vec{u}_1, \vec{u}_2)$; the derivative of the unit vector $\vec{i}$ is:

$$\frac{d\vec{i}}{d\sigma} = \frac{d\vec{i}}{d\tau} \frac{d\tau}{d\sigma} = \omega_3 \vec{j} \frac{r^2}{h} = \frac{h}{r^2} \vec{j} \frac{r^2}{h} = \vec{j}. \quad (6.100)$$

A similar procedure leads to the following results:

$$\frac{d\vec{i}}{d\sigma} = \vec{j}, \quad \frac{d\vec{u}_1}{d\sigma} = -\Lambda \sin(\sigma) \vec{k}, \quad \frac{d\vec{k}}{d\sigma} = -\Lambda \vec{j}, \quad (6.101a)$$

$$\frac{d\vec{j}}{d\sigma} = -\vec{i} + \Lambda \vec{k}, \quad \frac{d\vec{u}_2}{d\sigma} = \Lambda \cos(\sigma) \vec{k}, \quad \Lambda = \frac{r^3}{h^2} (\vec{a}_p \cdot \vec{k}). \quad (6.101b)$$

The process starts from the equation of the hodograph (6.85); taking into account the definition of Eq. (6.89) we obtain the following expression for the hodograph:

$$\zeta_1 \vec{u}_2 - \zeta_2 \vec{u}_1 = -\vec{j} + h \vec{v}, \quad (6.102)$$

that provides the following components of the velocity vector in the basis $(\vec{u}_1, \vec{u}_2)$:

$$\vec{v} \cdot \vec{u}_1 = -\frac{1}{h} (\zeta_2 + \sin(\sigma)) \quad (6.103a)$$

$$\vec{v} \cdot \vec{u}_2 = \frac{1}{h} (\zeta_1 + \cos(\sigma)). \quad (6.103b)$$

Differentiating the two members of Eq. (6.102) with respect to σ and taking into account the previously calculated derivatives of the unit vectors, we get:

$$-\frac{d\zeta_2}{d\sigma}\vec{u}_1 + \frac{d\zeta_1}{d\sigma}\vec{u}_2 + \Lambda s = r^2 \vec{a}_p + \frac{dh}{d\sigma}\vec{v}, \quad (6.104)$$

where s is the quantity:

$$s = 1 + \zeta_1 \cos \sigma + \zeta_2 \sin \sigma. \quad (6.105)$$

The dot product is applied to Eq. (6.104) with respect to the unit vectors $(\vec{u}_1, \vec{u}_2, \vec{k}_0)$ to obtain:

$$\frac{d\zeta_2}{d\sigma} = -r^2(\vec{a}_p \cdot \vec{u}_1) - \frac{dh}{d\sigma}(\vec{v} \cdot \vec{u}_1) \quad (6.106a)$$

$$\frac{d\zeta_1}{d\sigma} = r^2(\vec{a}_p \cdot \vec{u}_2) + \frac{dh}{d\sigma}(\vec{v} \cdot \vec{u}_2) \quad (6.106b)$$

$$\Lambda s = r^2(\vec{a}_p \cdot \vec{k}_0). \quad (6.106c)$$

The derivative:

$$\frac{dh}{d\sigma} = \frac{dh}{d\tau} \frac{r^2}{h} = \frac{r^3}{h}(\vec{a}_p \cdot \vec{j}) \Rightarrow \frac{d\zeta_3}{d\sigma} = -\frac{1}{s^3 \zeta_3^3}(\vec{a}_p \cdot \vec{j}) \quad (6.107)$$

provides the following equations for (ζ_1, ζ_2) :

$$\frac{d\zeta_1}{d\sigma} = \frac{1}{s^3 \zeta_3^4} [s \sin(\sigma)(\vec{a}_p \cdot \vec{i}) + (\zeta_1 + (1+s) \cos(\sigma))(\vec{a}_p \cdot \vec{j})] \quad (6.108a)$$

$$\frac{d\zeta_2}{d\sigma} = \frac{1}{s^3 \zeta_3^4} [-s \cos(\sigma)(\vec{a}_p \cdot \vec{i}) + (\zeta_2 + (1+s) \sin(\sigma))(\vec{a}_p \cdot \vec{j})]. \quad (6.108b)$$

6.4.4 A compact formulation

In this section a more elegant and compact formulation of DROMO is presented. It is based on the analysis of the motion relative to the orbital plane. For this, the Coriolis theorem is used. If $\vec{A}$ is a variable vector with time according to this theorem, we have:

$$\left. \frac{d\vec{A}}{d\tau} \right|_{(1)} = \left. \frac{d\vec{A}}{d\tau} \right|_{(0)} + \vec{\omega}_{01} \times \vec{A}. \quad (6.109)$$

If the time of DROMO σ is used instead of the real time τ , the following relation between time derivatives holds:

$$\left. \frac{d\vec{A}}{d\sigma} \right|_{(1)} = \left. \frac{d\vec{A}}{d\sigma} \right|_{(0)} + \vec{\Omega}_{01} \times \vec{A}, \quad (6.110)$$

where $\vec{\Omega}_{01}$ is the vector:

$$\vec{\Omega}_{01} = \vec{\omega}_{01} \frac{r^2}{h} = \frac{r^3}{h^2} a_{pz} \vec{i}. \quad (6.111)$$

The derivative calculated in the rigid body (0) assumes that the unit vectors $(\vec{i}_0, \vec{j}_0, \vec{k}_0)$ do not change with time. Note that the unit vectors $(\vec{u}_1, \vec{u}_2)$ are fixed to the rigid body (0) and, therefore these do not change either if $(\vec{i}_0, \vec{j}_0, \vec{k}_0)$ do not evolve over time.

In the following, time derivatives, made in the rigid body (0) of different vectors of interest, are calculated.

6.4.4.1 Velocity vector $\vec{v}$

It is known that:

$$\left. \frac{d\vec{v}}{d\tau} \right|_{(1)} = -\frac{\vec{i}}{r^2} + \vec{a}_p \Rightarrow \left. \frac{d\vec{v}}{d\sigma} \right|_{(1)} = \frac{1}{h}(-\vec{i} + r^2 \vec{a}_p). \quad (6.112)$$

On the other hand we have:

$$\vec{\Omega}_{01} \times \vec{v} = \Omega_{01} \vec{i} \times \vec{v} = \frac{\Omega_{01}}{r} \vec{h} = \frac{r^2}{h} a_{pz} \vec{k}. \quad (6.113)$$

From these expressions and the Coriolis theorem given by :

$$\left. \frac{d\vec{v}}{d\sigma} \right|_{(0)} = \left. \frac{d\vec{v}}{d\sigma} \right|_{(1)} - \vec{\Omega}_{01} \times \vec{v}, \quad (6.114)$$

the following value of the derivative in the rigid body (0) is deduced:

$$\left. \frac{d\vec{v}}{d\sigma} \right|_{(0)} = \frac{1}{h}(-\vec{i} + r^2 \vec{a}_{p\parallel}), \quad (6.115)$$

where $\vec{a}_{p\parallel}$ refers to the component of perturbing acceleration parallel to the orbital plane:

$$\vec{a}_{p\parallel} = \vec{a}_p - a_{pz} \vec{k}. \quad (6.116)$$

6.4.4.2 Angular momentum vector $\vec{h}$

Regarding the angular momentum $\vec{h}$ we know:

$$\left. \frac{d\vec{h}}{d\sigma} \right|_{(1)} = \frac{r^2}{h} \vec{x} \times \vec{a}_p = \frac{r^3}{h} \vec{i} \times \vec{a}_p \quad (6.117)$$

and since:

$$\vec{\Omega}_{01} \times \vec{h} = \Omega_{01} h \vec{i} \times \vec{k} = -\frac{r^3}{h} a_{pz} \vec{j}, \quad (6.118)$$

applying the Coriolis theorem, yields:

$$\left. \frac{d\vec{h}}{d\sigma} \right|_{(0)} = \frac{r^3}{h} a_{py} \vec{k} = \frac{r^3}{h} \vec{i} \times \vec{a}_{p\parallel}. \quad (6.119)$$

6.4.4.3 Eccentricity vector $\vec{e}$

The eccentricity vector is given by:

$$\vec{e} = -\vec{i} - \vec{h} \times \vec{v}. \quad (6.120)$$

Therefore, it has:

$$\left. \frac{d\vec{e}}{d\sigma} \right|_{(0)} = -\left. \frac{d\vec{i}}{d\sigma} \right|_{(0)} - \left. \frac{d\vec{h}}{d\sigma} \right|_{(0)} \times \vec{v} - \vec{h} \times \left. \frac{d\vec{v}}{d\sigma} \right|_{(0)}. \quad (6.121)$$

The equation of the hodograph (6.85) allows to calculate the terms of the second member of the Eq. (6.121), being:

$$\left. \frac{d\vec{i}}{d\sigma} \right|_{(0)} = \vec{j} \quad (6.122a)$$

$$\left. \frac{d\vec{h}}{d\sigma} \right|_{(0)} \times \vec{v} = -\frac{r^3}{h^2} a_{py} (\vec{i} + \vec{e}) \quad (6.122b)$$

$$\vec{h} \times \left. \frac{d\vec{v}}{d\sigma} \right|_{(0)} = -\vec{j} + r^2 \vec{k} \times \vec{a}_{p\parallel}. \quad (6.122c)$$

Thus, the evolution of the eccentricity vector is described as follows:

$$\left. \frac{d\vec{e}}{d\sigma} \right|_{(0)} = \frac{r^3}{h^2} \left(a_{py} [\vec{i} + \vec{e}] - \frac{h^2}{r} \vec{k} \times \vec{a}_{p\parallel} \right). \quad (6.123)$$

By introducing the vector $\vec{b}_{p\parallel}$:

$$\vec{b}_{p\parallel} = \frac{r^3}{h^2} \vec{a}_{p\parallel}, \quad (6.124)$$

which has the direction and sense of the perturbing acceleration $\vec{a}_{p\parallel}$ but different module, and defining $\vec{a}_f$ by the quantity

$$\vec{a}_f = s \vec{b}_{p\parallel} + b_{py} \vec{j}, \quad (6.125)$$

the notation of the Eq. (6.123) is simplified and takes the form:

$$\left. \frac{d\vec{e}}{d\sigma} \right|_{(0)} = \vec{a}_f \times \vec{k} + b_{py} \vec{e}. \quad (6.126)$$

Note that the term $a_{py} \vec{i}$ can be expressed as $-\vec{k} \times (a_{py} \vec{j})$.

6.4.4.4 Governing equations

The vector equation (6.126) can be decomposed into coordinates in different bases. If the following relations are taken into account:

$$\vec{i} = +\cos(\sigma) \vec{u}_1 + \sin(\sigma) \vec{u}_2 \quad (6.127a)$$

$$\vec{j} = -\sin(\sigma) \vec{u}_1 + \cos(\sigma) \vec{u}_2 \quad (6.127b)$$

$$\vec{e} = \zeta_1 \vec{u}_1 + \zeta_2 \vec{u}_2 \quad (6.127c)$$

and it is projected on the basis $(\vec{u}_1, \vec{u}_2)$, fixed to rigid body (0), it has:

$$\left. \frac{d\vec{e}}{d\sigma} \right|_{(0)} = \frac{d\zeta_1}{d\sigma} \vec{u}_1 + \frac{d\zeta_2}{d\sigma} \vec{u}_2. \quad (6.128)$$

From Eq. (6.126), and applying the dot product to the unit vectors $(\vec{u}_1, \vec{u}_2)$, the following equations are obtained:

$$\frac{d\zeta_1}{d\sigma} = +\vec{a}_f \cdot \vec{u}_2 + b_{py} \zeta_1 \quad (6.129a)$$

$$\frac{d\zeta_2}{d\sigma} = -\vec{a}_f \cdot \vec{u}_1 + b_{py} \zeta_2, \quad (6.129b)$$

which determine the evolution of the coordinates (ζ_1, ζ_2) of the eccentricity vector. Once developed they adopt the form:

$$\frac{d\zeta_1}{d\sigma} = \frac{r^3}{h^2} \left\{ \left[\zeta_1 + \cos(\sigma) \left(1 + \frac{h^2}{r} \right) \right] a_{py} + \frac{h^2}{r} \sin(\sigma) a_{px} \right\} \quad (6.130a)$$

$$\frac{d\zeta_2}{d\sigma} = \frac{r^3}{h^2} \left\{ \left[\zeta_2 + \sin(\sigma) \left(1 + \frac{h^2}{r} \right) \right] a_{py} - \frac{h^2}{r} \cos(\sigma) a_{px} \right\}. \quad (6.130b)$$

Taking into account that $\zeta_3 = 1/h$ and $h^2 = r s$, these equations coincide with two of the equations of the formulation.

6.5 SUMMARY OF EQUATIONS

6.5.1 Using the quaternion linked to $Ox_0y_0z_0$

Using the quaternion $\mathbf{q}_0$ linked to the axes $Ox_0y_0z_0$ to describe the motion of the orbital plane and defining the following auxiliary variables:

$$a_{px} = \vec{a}_p \cdot \vec{i}, \quad a_{py} = \vec{a}_p \cdot \vec{j}, \quad a_{pz} = \vec{a}_p \cdot \vec{k}, \quad (6.131)$$

the governing equations take the form:

$$\frac{d\tau}{d\sigma} = \frac{1}{\zeta_3^3 s^2} \quad (6.132a)$$

$$\frac{d\zeta_1}{d\sigma} = \frac{1}{\zeta_3^4 s^3} \{ s \sin(\sigma) a_{px} + [\zeta_1 + (1+s) \cos(\sigma)] a_{py} \} \quad (6.132b)$$

$$\frac{d\zeta_2}{d\sigma} = \frac{1}{\zeta_3^4 s^3} \{ -s \cos(\sigma) a_{px} + [\zeta_2 + (1+s) \sin(\sigma)] a_{py} \} \quad (6.132c)$$

$$\frac{d\zeta_3}{d\sigma} = -\frac{1}{\zeta_3^3 s^3} a_{py} \quad (6.132d)$$

$$\frac{d\varepsilon_1^0}{d\sigma} = \frac{1}{2} \tilde{\omega}_{01} \{ \varepsilon_4^0 \cos(\sigma - \sigma_0) - \varepsilon_3^0 \sin(\sigma - \sigma_0) \} \quad (6.132e)$$

$$\frac{d\varepsilon_2^0}{d\sigma} = \frac{1}{2} \tilde{\omega}_{01} \{ \varepsilon_3^0 \cos(\sigma - \sigma_0) + \varepsilon_4^0 \sin(\sigma - \sigma_0) \} \quad (6.132f)$$

$$\frac{d\varepsilon_3^0}{d\sigma} = \frac{1}{2} \tilde{\omega}_{01} \{ -\varepsilon_2^0 \cos(\sigma - \sigma_0) + \varepsilon_1^0 \sin(\sigma - \sigma_0) \} \quad (6.132g)$$

$$\frac{d\varepsilon_4^0}{d\sigma} = \frac{1}{2} \tilde{\omega}_{01} \{ -\varepsilon_1^0 \cos(\sigma - \sigma_0) - \varepsilon_2^0 \sin(\sigma - \sigma_0) \} \quad (6.132h)$$

and in them the angular velocity $\tilde{\omega}_{01}$ is:

$$\tilde{\omega}_{01} = \frac{1}{\zeta_3^4 s^3} a_{pz}. \quad (6.133)$$

Besides, the following relations are satisfied:

$$s = 1 + \zeta_1 \cos(\sigma) + \zeta_2 \sin(\sigma) \quad (6.134a)$$

$$\frac{1}{r} = \zeta_3^2 s \quad (6.134b)$$

$$\frac{dr}{d\tau} = \zeta_3 (\zeta_1 \sin(\sigma) - \zeta_2 \cos(\sigma)) \quad (6.134c)$$

$$\varepsilon_1 = \varepsilon_1^0 \cos\left(\frac{\sigma - \sigma_0}{2}\right) + \varepsilon_2^0 \sin\left(\frac{\sigma - \sigma_0}{2}\right) \quad (6.134d)$$

$$\varepsilon_2 = \varepsilon_2^0 \cos\left(\frac{\sigma - \sigma_0}{2}\right) - \varepsilon_1^0 \sin\left(\frac{\sigma - \sigma_0}{2}\right) \quad (6.134e)$$

$$\varepsilon_3 = \varepsilon_3^0 \cos\left(\frac{\sigma - \sigma_0}{2}\right) + \varepsilon_4^0 \sin\left(\frac{\sigma - \sigma_0}{2}\right) \quad (6.134f)$$

$$\varepsilon_4 = \varepsilon_4^0 \cos\left(\frac{\sigma - \sigma_0}{2}\right) - \varepsilon_3^0 \sin\left(\frac{\sigma - \sigma_0}{2}\right). \quad (6.134g)$$

Actually, the perturbing acceleration always appears accompanied by a factor. If the following values are introduced for convenience:

$$\tilde{a}_{px} = \frac{a_{px}}{\zeta_3^4 s^3}, \quad \tilde{a}_{py} = \frac{a_{py}}{\zeta_3^4 s^3}, \quad \tilde{a}_{pz} = \frac{a_{pz}}{\zeta_3^4 s^3}, \quad (6.135)$$

the equations take a more compact form:

$$\frac{d\tau}{d\sigma} = \frac{1}{\zeta_3^3 s^2} \quad (6.136a)$$

$$\frac{d\zeta_1}{d\sigma} = s \sin(\sigma) \tilde{a}_{px} + [\zeta_1 + (1 + s) \cos(\sigma)] \tilde{a}_{py} \quad (6.136b)$$

$$\frac{d\zeta_2}{d\sigma} = -s \cos(\sigma) \tilde{a}_{px} + [\zeta_2 + (1 + s) \sin(\sigma)] \tilde{a}_{py} \quad (6.136c)$$

$$\frac{d\zeta_3}{d\sigma} = -\zeta_3 \tilde{a}_{py} \quad (6.136d)$$

$$\frac{d\varepsilon_1^0}{d\sigma} = \frac{1}{2} \tilde{\omega}_{01} \{ \varepsilon_4^0 \cos(\sigma - \sigma_0) - \varepsilon_3^0 \sin(\sigma - \sigma_0) \} \quad (6.136e)$$

$$\frac{d\varepsilon_2^0}{d\sigma} = \frac{1}{2} \tilde{\omega}_{01} \{ \varepsilon_3^0 \cos(\sigma - \sigma_0) + \varepsilon_4^0 \sin(\sigma - \sigma_0) \} \quad (6.136f)$$

$$\frac{d\varepsilon_3^0}{d\sigma} = \frac{1}{2} \tilde{\omega}_{01} \{ -\varepsilon_2^0 \cos(\sigma - \sigma_0) + \varepsilon_1^0 \sin(\sigma - \sigma_0) \} \quad (6.136g)$$

$$\frac{d\varepsilon_4^0}{d\sigma} = \frac{1}{2} \tilde{\omega}_{01} \{ -\varepsilon_1^0 \cos(\sigma - \sigma_0) - \varepsilon_2^0 \sin(\sigma - \sigma_0) \}, \quad (6.136h)$$

where the angular velocity $\tilde{\omega}_{01}$ is:

$$\tilde{\omega}_{01} = \tilde{a}_{pz}. \quad (6.137)$$

The initial conditions are given by:

$$\sigma = \sigma_0 : \quad \zeta_1(\sigma_0) = e_0, \quad \zeta_2(\sigma_0) = 0, \quad \zeta_3(\sigma_0) = \frac{1}{\sqrt{r_0(1 + e_0 \cos(\sigma_0))}}. \quad (6.138)$$

6.5.2 Using the quaternion linked to $Ox_p y_p z_p$

If instead of the quaternion $\mathbf{q}_0$ linked to the axes $Ox_0 y_0 z_0$ we use the quaternion $\mathbf{n}$ linked to the axes $Ox_p y_p z_p$ to describe the drag motion 0/1, we obtain a similar system of equations but slightly easier because now the value σ_0 no longer

appears. This improves the speed by saving the calculation of a sine and a cosine, and improves the accuracy for long times. In addition, it introduces an important simplification in all those asymptotic schemes that are based on DROMO and also in the schemes that are based on averaging techniques. The system of equations is:

$$\frac{d\tau}{d\sigma} = \frac{1}{\zeta_3^3 s^2} \quad (6.139a)$$

$$\frac{d\zeta_1}{d\sigma} = s \sin(\sigma) \tilde{a}_{px} + [\zeta_1 + (1+s) \cos(\sigma)] \tilde{a}_{py} \quad (6.139b)$$

$$\frac{d\zeta_2}{d\sigma} = -s \cos(\sigma) \tilde{a}_{px} + [\zeta_2 + (1+s) \sin(\sigma)] \tilde{a}_{py} \quad (6.139c)$$

$$\frac{d\zeta_3}{d\sigma} = -\zeta_3 \tilde{a}_{py} \quad (6.139d)$$

$$\frac{d\eta_1}{d\sigma} = \frac{1}{2} \tilde{\omega}_{01} \{ \eta_4 \cos(\sigma) - \eta_3 \sin(\sigma) \} \quad (6.139e)$$

$$\frac{d\eta_2}{d\sigma} = \frac{1}{2} \tilde{\omega}_{01} \{ \eta_3 \cos(\sigma) + \eta_4 \sin(\sigma) \} \quad (6.139f)$$

$$\frac{d\eta_3}{d\sigma} = \frac{1}{2} \tilde{\omega}_{01} \{ -\eta_2 \cos(\sigma) + \eta_1 \sin(\sigma) \} \quad (6.139g)$$

$$\frac{d\eta_4}{d\sigma} = \frac{1}{2} \tilde{\omega}_{01} \{ -\eta_1 \cos(\sigma) - \eta_2 \sin(\sigma) \}, \quad (6.139h)$$

where the angular velocity $\tilde{\omega}_{01}$ is:

$$\tilde{\omega}_{01} = \tilde{a}_{pz}. \quad (6.140)$$

Besides, the following relations hold:

$$s = 1 + \zeta_1 \cos(\sigma) + \zeta_2 \sin(\sigma) \quad (6.141a)$$

$$\frac{1}{r} = \zeta_3^2 s \quad (6.141b)$$

$$\frac{dr}{d\tau} = \zeta_3 (\zeta_1 \sin(\sigma) - \zeta_2 \cos(\sigma)) \quad (6.141c)$$

$$\varepsilon_1 = \eta_1 \cos\left(\frac{\sigma}{2}\right) + \eta_2 \sin\left(\frac{\sigma}{2}\right) \quad (6.141d)$$

$$\varepsilon_2 = \eta_2 \cos\left(\frac{\sigma}{2}\right) - \eta_1 \sin\left(\frac{\sigma}{2}\right) \quad (6.141e)$$

$$\varepsilon_3 = \eta_3 \cos\left(\frac{\sigma}{2}\right) + \eta_4 \sin\left(\frac{\sigma}{2}\right) \quad (6.141f)$$

$$\varepsilon_4 = \eta_4 \cos\left(\frac{\sigma}{2}\right) - \eta_3 \sin\left(\frac{\sigma}{2}\right). \quad (6.141g)$$

The initial conditions are given by:

$$\sigma = \sigma_0 : \quad \zeta_1(\sigma_0) = e_0, \quad \zeta_2(\sigma_0) = 0, \quad \zeta_3(\sigma_0) = \frac{1}{\sqrt{r_0(1 + e_0 \cos(\sigma_0))}}. \quad (6.142)$$

6.6 SINGULARITIES

The presence of singularities is determined by the kind of perturbing force acting upon the satellite. At first sight, there would be a singularity when the denominator ($\zeta_3^4 s^3$) that intervenes in the definitions of ($\tilde{a}_{px}, \tilde{a}_{py}, \tilde{a}_{pz}$) vanishes.

Note that the quantity s can be written as follows:

$$s = 1 + \zeta_1 \cos(\sigma) + \zeta_2 \sin(\sigma) = 1 + e \cos(\sigma - \beta) = 1 + e \cos(\theta), \quad (6.143)$$

being θ the true anomaly. Considering the previous equation, s can vanish if we have

- Parabolic orbit, $e = 1$, when $\theta \rightarrow \pi$. The parabolic orbit has associated a Keplerian energy $E_k = 0$. Consequently, this situation is never reached, because for a parabolic orbit it would be needed an infinite time to reach $\theta = \pi$.
- Rectilinear conics (degenerated conics such that $e = 1$ and $h = 0$) when $\theta \rightarrow \pi$. In a degenerated orbit the Keplerian energy $E_k \neq 0$. This situation is merely theoretical but it does not occur in reality.
- Hyperbolic orbit, $e > 1$, when $\theta \rightarrow \theta_a$, where $\theta_a = \arccos(\frac{-1}{e})$, that is, when we approach to the asymptote of the osculating orbit. This is the only one of the three situations that can appear in reality.

We start with the second case with the intention to know how the quotient

$$\frac{1}{\zeta_3^4 s^3} = \frac{h^4}{s^3} = \frac{a^2 (1 - e^2)^2}{(1 + e \cos(\theta))^3} \quad (6.144)$$

behaves when we approach a rectilinear conic. If we think that $\theta = \pi$, when $e \rightarrow 1$:

$$\frac{1}{\zeta_3^4 s^3} \rightarrow \frac{a^2 (1 + e)^2}{(1 - e)} \rightarrow \infty. \quad (6.145)$$

As a consequence, there is a singularity associated with the rectilinear conics ($s = 0, h = 0, a \neq 0$) that can not be avoided with this formulation. The situation is different for the parabolic orbits and the hyperbolic orbits. In both cases, the singularity $s = 0$ happens when the satellite approaches the infinity point of the conic, that is, when $r \rightarrow \infty$. In this situation, the existence or not of singularity depends on the perturbing forces acting upon the satellite.

6.7 USEFUL RELATIONS

The state vector and eccentricity vector of the orbit are expressed here as a function of the DROMO variables.

6.7.1 Position vector

The position vector of the particle can be expressed in different bases. The values provided are dimensionless, such that to recover the dimensional values only a multiplication by the characteristic length L_c should be applied.

- In the frame $Oxyz$:

$$\vec{x} = r\vec{i} = \frac{1}{\zeta_3^2 s} \vec{i}. \quad (6.146)$$

- In the frame $Ox_0y_0z_0$:

$$\vec{x} = \frac{1}{\zeta_3^2 s} [\cos(\sigma - \sigma_0)\vec{i}_0 + \sin(\sigma - \sigma_0)\vec{j}_0]. \quad (6.147)$$

- In the frame $Ox_py_pz_p$:

$$\vec{x} = \frac{1}{\zeta_3^2 s} [\cos(\sigma)\vec{u}_1 + \sin(\sigma)\vec{u}_2]. \quad (6.148)$$

- In the frame $Ox_1y_1z_1$:

$$\vec{x} = \frac{1}{\zeta_3^2 s} [\vec{i}_1, \vec{j}_1, \vec{k}_1] Q_0 \begin{pmatrix} \cos(\sigma - \sigma_0) \\ \sin(\sigma - \sigma_0) \\ 0 \end{pmatrix}, \quad (6.149)$$

where the rotation matrix Q_0 is the one defined in Eqs. (6.6) and (6.5). Introducing the rectangular matrix N (3×2):

$$N = \begin{bmatrix} 1 - 2((\varepsilon_2^0)^2 + (\varepsilon_3^0)^2) & 2\varepsilon_1^0 \varepsilon_2^0 - 2\varepsilon_4^0 \varepsilon_3^0 \\ 2\varepsilon_1^0 \varepsilon_2^0 + 2\varepsilon_4^0 \varepsilon_3^0 & 1 - 2((\varepsilon_1^0)^2 + (\varepsilon_3^0)^2) \\ 2\varepsilon_1^0 \varepsilon_3^0 - 2\varepsilon_4^0 \varepsilon_2^0 & 2\varepsilon_3^0 \varepsilon_2^0 + 2\varepsilon_4^0 \varepsilon_1^0 \end{bmatrix}, \quad (6.150)$$

it becomes:

$$\vec{x} = \frac{1}{\zeta_3^2 s} [\vec{i}_1, \vec{j}_1, \vec{k}_1] N \begin{pmatrix} \cos(\sigma - \sigma_0) \\ \sin(\sigma - \sigma_0) \\ 0 \end{pmatrix}. \quad (6.151)$$

6.7.2 Velocity vector

The velocity vector of the particle can be expressed in different bases. The values provided are dimensionless, such that to recover the dimensional values only a multiplication by the characteristic value $\omega_c \cdot L_c$ should be applied.

- In the frame $Oxyz$:

$$\vec{v} = \frac{dr}{d\tau} \vec{i} + \frac{h}{r} \vec{j} = \zeta_3 [(\zeta_1 \sin(\sigma) - \zeta_2 \cos(\sigma)) \vec{i} + s \vec{j}]. \quad (6.152)$$

- In the frame $Ox_0y_0z_0$:

$$\begin{aligned} \vec{v} &= \zeta_3 [\vec{i}_0, \vec{j}_0, \vec{k}] R_3(\sigma - \sigma_0) \begin{pmatrix} \zeta_1 \sin(\sigma) - \zeta_2 \cos(\sigma) \\ 1 + \zeta_1 \cos(\sigma) + \zeta_2 \sin(\sigma) \\ 0 \end{pmatrix} \\ &= \zeta_3 [\vec{i}_0, \vec{j}_0, \vec{k}] \begin{pmatrix} -\sin(\sigma - \sigma_0) + \zeta_1 \sin(\sigma_0) - \zeta_2 \cos(\sigma_0) \\ \cos(\sigma - \sigma_0) + \zeta_1 \cos(\sigma_0) + \zeta_2 \sin(\sigma_0) \\ 0 \end{pmatrix}. \end{aligned} \quad (6.153)$$

- In the frame $Ox_py_pz_p$:

$$\vec{v} = \zeta_3 [-(\sin \sigma + \zeta_2) \vec{u}_1 + (\cos(\sigma) + \zeta_1) \vec{u}_2]. \quad (6.154)$$

- In the frame $Ox_1y_1z_1$:

$$\vec{v} = \zeta_3 [\vec{i}_1, \vec{j}_1, \vec{k}_1] Q_0 \begin{pmatrix} -\sin(\sigma - \sigma_0) + \zeta_1 \sin(\sigma_0) - \zeta_2 \cos(\sigma_0) \\ \cos(\sigma - \sigma_0) + \zeta_1 \cos(\sigma_0) + \zeta_2 \sin(\sigma_0) \\ 0 \end{pmatrix}, \quad (6.155)$$

where Q_0 is the matrix defined in Eqs. (6.6) and (6.5). By using the matrix N defined in Eq. (6.150) it turns out

$$\vec{v} = [\vec{i}_1, \vec{j}_1, \vec{k}_1] N \begin{pmatrix} -\sin(\sigma - \sigma_0) + \zeta_1 \sin(\sigma_0) - \zeta_2 \cos(\sigma_0) \\ \cos(\sigma - \sigma_0) + \zeta_1 \cos(\sigma_0) + \zeta_2 \sin(\sigma_0) \\ 0 \end{pmatrix}. \quad (6.156)$$

6.7.3 Eccentricity vector

The eccentricity vector of the particle can be expressed in different bases.

- In the frame $Oxyz$:

$$\vec{e} = [\vec{i}, \vec{j}, \vec{k}] R_3^T(\sigma) \begin{pmatrix} \zeta_1 \\ \zeta_2 \\ 0 \end{pmatrix} = [\vec{i}, \vec{j}, \vec{k}] \begin{pmatrix} \zeta_1 \cos(\sigma) + \zeta_2 \sin(\sigma) \\ -\zeta_1 \sin(\sigma) + \zeta_2 \cos(\sigma) \\ 0 \end{pmatrix}. \quad (6.157)$$

- In the frame $Ox_0y_0z_0$:

$$\vec{e} = [\vec{i}_0, \vec{j}_0, \vec{k}] R_3^T(\sigma_0) \begin{pmatrix} \zeta_1 \\ \zeta_2 \\ 0 \end{pmatrix} = [\vec{i}_0, \vec{j}_0, \vec{k}] \begin{pmatrix} \zeta_1 \cos(\sigma_0) + \zeta_2 \sin(\sigma_0) \\ -\zeta_1 \sin(\sigma_0) + \zeta_2 \cos(\sigma_0) \\ 0 \end{pmatrix}. \quad (6.158)$$

- In the frame $Ox_py_pz_p$:

$$\vec{e} = \zeta_1 \vec{u}_1 + \zeta_2 \vec{u}_2. \quad (6.159)$$

- In the frame $Ox_1y_1z_1$:

$$\vec{e} = \zeta_3 [\vec{i}_1, \vec{j}_1, \vec{k}_1] Q_0 \begin{pmatrix} \zeta_1 \cos(\sigma_0) + \zeta_2 \sin(\sigma_0) \\ -\zeta_1 \sin(\sigma_0) + \zeta_2 \cos(\sigma_0) \\ 0 \end{pmatrix}, \quad (6.160)$$

where Q_0 is the matrix defined in Eqs. (6.6) and (6.5). Making use of the matrix N defined in Eq. (6.150) it becomes

$$\vec{e} = \zeta_3 [\vec{i}_1, \vec{j}_1, \vec{k}_1] N \begin{pmatrix} \zeta_1 \cos(\sigma_0) + \zeta_2 \sin(\sigma_0) \\ -\zeta_1 \sin(\sigma_0) + \zeta_2 \cos(\sigma_0) \end{pmatrix}. \quad (6.161)$$

THE ELLIDROMO REGULARIZATION

The theory of DROMO for elliptic orbits (Baú, Urrutxua, and Peláez [12] and Baú et al. [14]) is presented in the next lines. This special perturbation method is characterized by eight ordinary differential equations.

7.1 BASIC ELEMENTS OF THE THEORY

In the first place we should remember the solution of Kepler's equation by using the eccentric anomaly $\mathfrak{a}$ as a variable. The solution found for the trajectory is:

$$R = \frac{h^2/\mu}{1 + e \cos(\theta)}, \quad (7.1)$$

where θ is the true anomaly. Using dimensionless variables we get:

$$r = \frac{h^2}{1 + e \cos(\theta)}. \quad (7.2)$$

Figure 7.1 shows the basic elements of elliptical Keplerian motion and shows the principal axes $Cx_p y_p$ of the ellipse constituted by the orbit of the particle M. The eccentric anomaly is introduced through the relations:

$$\sin(\theta) = \frac{\sqrt{1 - e^2} \sin(\mathfrak{a})}{1 - e \cos(\mathfrak{a})} \quad (7.3a)$$

$$\cos(\theta) = \frac{\cos(\mathfrak{a}) - e}{1 - e \cos(\mathfrak{a})}, \quad (7.3b)$$

or through the inverse relations:

$$\sin(\mathfrak{a}) = \frac{\sqrt{1 - e^2} \sin(\theta)}{1 + e \cos(\theta)} \quad (7.4a)$$

$$\cos(\mathfrak{a}) = \frac{\cos(\theta) + e}{1 + e \cos(\theta)}. \quad (7.4b)$$

The dimensional velocity is given by:

$$\vec{v} = \omega_c L_C \left(\frac{dr}{d\tau} \vec{i} + \frac{h}{r} \vec{k} \right), \quad (7.5)$$

and the dimensional total energy turns out to be:

$$\tilde{E} = \frac{1}{2} m v^2 - \frac{m\mu}{r} = \frac{1}{2} m \omega_c^2 L_C^2 \left[\left(\frac{dr}{d\tau} \right)^2 + \frac{h^2}{r^2} - \frac{2}{r} \right]. \quad (7.6)$$

In the elliptic case this is

$$\tilde{E} = -\frac{m\mu}{2a}, \quad (7.7)$$

such that, the semi-major axis, in dimensionless form, is:

$$\frac{1}{a} = - \left[\left(\frac{dr}{d\tau} \right)^2 + \frac{h^2}{r^2} - \frac{2}{r} \right]. \quad (7.8)$$

Figure 7.1: Elliptic motion.

The total energy is not independent of the eccentricity; both are linked by means of:

$$e^2 - 1 = \frac{2 \tilde{E} h^2}{m \mu^2}, \quad (7.9)$$

relation that, when it is passed to dimensionless variables, adopts the form:

$$h^2 = a(1 - e^2). \quad (7.10)$$

The classical form to obtain the time equations consists in: 1) expressing the polar coordinates (r, θ) as a function of the eccentric anomaly α ; 2) getting the existing relation between the physical time τ and the eccentric anomaly α . Consequently, the famous Kepler's equation is obtained:

$$\tau - \tau_p = a^{\frac{3}{2}} (\alpha - e \sin(\alpha)), \quad (7.11)$$

where the constant τ_p , the time of perigee passage, has to be estimated from the initial conditions in $\tau = 0$.

7.2 THE ELLIDROMO FORMULATION

We start with the equations of the perturbed problem, rewritten in dimensionless variables, which adopt the following form:

$$\frac{d^2 \mathbf{r}}{d\tau^2} = \frac{h^2}{r^3} - \frac{1}{r^2} + (\vec{a}_p \cdot \vec{i}) \quad (7.12a)$$

$$\frac{dh}{d\tau} = r (\vec{a}_p \cdot \vec{j}). \quad (7.12b)$$

Note that the dimensionless components of the angular velocity can be expressed as a function of the state variables by means of the relations:

$$\omega_1 = \frac{r}{h} (\vec{a}_p \cdot \vec{k}) \quad (7.13a)$$

$$\omega_3 = \frac{h}{r^2}. \quad (7.13b)$$

A change of the independent variable is introduced such that the variable u is taken as independent variable in lieu of τ ; the relation between τ and u is given by the following expression:

$$\frac{d\tau}{du} = r \sqrt{a}, \quad \Rightarrow \quad \tau = k_p + \int_0^u r(u) \sqrt{a(u)} du, \quad (7.14)$$

where the magnitude a is defined by:

$$\frac{1}{a} = - \left[\left(\frac{dr}{d\tau} \right)^2 + \frac{h^2}{r^2} - \frac{2}{r} \right]. \quad (7.15)$$

To rewrite the equations in terms of the new independent variable u the following relations should be taken into account:

$$\frac{dr}{d\tau} = \frac{1}{r \sqrt{a}} \frac{dr}{du} \quad (7.16a)$$

$$\frac{d^2r}{d\tau^2} = \frac{1}{r \sqrt{a}} \frac{d}{du} \left(\frac{1}{r \sqrt{a}} \frac{dr}{du} \right). \quad (7.16b)$$

Once developed, the second derivative turns out to be

$$\frac{d^2r}{d\tau^2} = \frac{1}{r^2 a} \frac{d^2r}{du^2} - \frac{1}{r^3 a} \left(\frac{dr}{du} \right)^2 - \frac{1}{2r^2 a^2} \frac{dr}{du} \frac{da}{du}. \quad (7.17)$$

Equation (7.15) can be written as follows:

$$\left(\frac{dr}{d\tau} \right)^2 = \frac{2}{r} - \frac{1}{a} - \frac{h^2}{r^2}, \quad (7.18)$$

and in terms of the variable u turns out to be:

$$\left(\frac{dr}{du} \right)^2 = 2r a - r^2 - h^2 a. \quad (7.19)$$

Introducing this value into the second derivative, it takes the form:

$$\frac{d^2r}{d\tau^2} = \frac{1}{r^2 a} \frac{d^2r}{du^2} - \frac{2}{r^2} + \frac{1}{r a} + \frac{h^2}{r^3} - \frac{1}{2r^2 a^2} \frac{dr}{du} \frac{da}{du}. \quad (7.20)$$

The equations governing the dynamics take the form:

$$\frac{d^2r}{du^2} + r - a = a r^2 (\vec{a}_p \cdot \vec{i}) + \frac{1}{2a} \frac{dr}{du} \frac{da}{du} \quad (7.21a)$$

$$\frac{dh}{du} = r^2 \sqrt{a} (\vec{a}_p \cdot \vec{j}). \quad (7.21b)$$

Note that the derivative of a is governed by the perturbing accelerations; in fact, its value is

$$\frac{da}{du} = 2a^2 \left[\frac{dr}{du} (\vec{a}_p \cdot \vec{i}) + \sqrt{a} h (\vec{a}_p \cdot \vec{j}) \right]. \quad (7.22)$$

Consequently, the right-hand side of Eq. (7.21a) vanishes when the unperturbed problem is solved, since $\vec{a}_p = \vec{0}$. Obviously, like h , a is constant in the unperturbed problem, but changes over time (or with u) in the perturbed problem.

7.2.1 Solution of the unperturbed problem

Equation (7.21a), when applied to the unperturbed problem, is reduced to

$$\frac{d^2 r}{du^2} + r - a = 0, \quad (7.23)$$

with a constant. The solution to this equation turns out to be:

$$r = a(1 - A_1 \cos(u) - A_2 \sin(u)), \quad (7.24)$$

where a , A_1 y A_2 are integration constants.

7.2.2 Solution of the perturbed problem

In order to solve the perturbed problem, we look for a solution like:

$$r = \lambda_3 (1 - \lambda_1 \cos(u) - \lambda_2 \sin(u)), \quad (7.25)$$

where $(\lambda_1, \lambda_2, \lambda_3)$ are the new dependent variables; in particular λ_3 coincides with the semi-major axis a of the osculating orbit.

We introduce the following notation:

$$\hat{r} = 1 - \lambda_1 \cos(u) - \lambda_2 \sin(u). \quad (7.26)$$

To set out the equations that govern the motion in terms of the new variables, we start by calculating the derivative:

$$\begin{aligned} \frac{dr}{du} &= \frac{d\lambda_3}{du} \hat{r} - \lambda_3 \left(\frac{d\lambda_1}{du} \cos(u) + \frac{d\lambda_2}{du} \sin(u) \right) + \\ &+ \lambda_3 (\lambda_1 \sin(u) - \lambda_2 \cos(u)). \end{aligned} \quad (7.27)$$

The method of variation of constants requires the verification of the condition:

$$\frac{d\lambda_3}{du} \hat{r} - \lambda_3 \left(\frac{d\lambda_1}{du} \cos(u) + \frac{d\lambda_2}{du} \sin(u) \right) = 0, \quad (7.28)$$

so that the osculating orbit is, effectively, osculating of the real trajectory. This way the radial velocity turns out to be:

$$\frac{dr}{du} = \lambda_3 (\lambda_1 \sin(u) - \lambda_2 \cos(u)), \quad (7.29)$$

and from it, the second derivative intervening in Eq. (7.21a) can be estimated. This equation takes the form:

$$\begin{aligned} \frac{d\lambda_1}{du} \sin(u) - \frac{d\lambda_2}{du} \cos(u) &= \\ &= \lambda_3^2 \hat{r}^2 (\vec{a}_p \cdot \vec{i}) - \frac{1}{2} \frac{1}{\lambda_3} \frac{d\lambda_3}{du} [\lambda_1 \sin(u) - \lambda_2 \cos(u)]. \end{aligned} \quad (7.30)$$

Equations (7.28-7.30) constitute a system of two equations with two unknowns, namely the derivatives: $\frac{d\lambda_1}{du}$ and $\frac{d\lambda_2}{du}$. The derivative of λ_3 , which is directly deduced from the relation (7.22), intervenes in the system. This leads to the following system of equations:

$$\frac{d\tau}{du} = \lambda_3^2 \hat{r} \quad (7.31a)$$

$$\frac{d\lambda_1}{du} = \lambda_3^2 \left\{ (\ell^2 \sin u - 2\hat{r}\lambda_2) (\vec{a}_p \cdot \vec{i}) + \ell[(1 + \hat{r}) \cos u - \lambda_1] (\vec{a}_p \cdot \vec{j}) \right\} \quad (7.31b)$$

$$\frac{d\lambda_2}{du} = \lambda_3^2 \left\{ (-\ell^2 \cos u + 2\hat{r}\lambda_1) (\vec{a}_p \cdot \vec{i}) + \ell[(1 + \hat{r}) \sin u - \lambda_2] (\vec{a}_p \cdot \vec{j}) \right\} \quad (7.31c)$$

$$\frac{d\lambda_3}{du} = 2\lambda_3^3 \left\{ (\lambda_1 \sin u - \lambda_2 \cos u) (\vec{a}_p \cdot \vec{i}) + \ell(\vec{a}_p \cdot \vec{j}) \right\}, \quad (7.31d)$$

Figure 7.2: Eccentricity vector.

where ℓ is the quantity:

$$\ell = \sqrt{1 - e^2}. \quad (7.32)$$

7.2.3 The independent variable u

The variable u is not the eccentric anomaly, although both are related. In fact, if you consider the equation of the osculating orbit:

$$r = a(1 - e \cos(\alpha)), \quad (7.33)$$

and since in this formulation

$$r = a(1 - \lambda_1 \cos(u) - \lambda_2 \sin(u)), \quad (7.34)$$

it is immediate to see that the quantities (λ_1, λ_2) should be:

$$\lambda_1 = e \cos(\alpha), \quad \lambda_2 = e \sin(\alpha), \quad \text{with} \quad \alpha = u - \varpi, \quad (7.35)$$

where the r of this formulation corresponds to the r of the osculating orbit.

Consequently, since the eccentricity vector is in the orbital plane, there is a basis $(\vec{a}_1, \vec{a}_2)$ inside that plane that allows to express the eccentricity vector as:

$$\vec{e} = \lambda_1 \vec{a}_1 + \lambda_2 \vec{a}_2, \quad (7.36)$$

i.e., the quantities (λ_1, λ_2) are the coordinates of the eccentricity vector $\vec{e}$ in such basis.

The initial condition of the independent variable u is chosen such as:

$$u_0 = \varpi_0 \quad \text{that is} \quad \alpha_0 = 0, \quad (7.37)$$

where the initial conditions for the quantities (λ_1, λ_2) in $\tau = 0$ are:

$$\lambda_{1_0} = e_0, \quad \lambda_{2_0} = 0. \quad (7.38)$$

The axis Ox , which forms an angle θ (true anomaly) with the vector $\vec{e}$, will form an angle v with the unit vector $\vec{a}_1$ (see Fig. 7.2). Therefore, it is necessary to find the relation between v and the independent variable u . The advantage of the basic formulation of DROMO is that the angle σ between the axis Ox and the unit vector $\vec{u}_1$ coincides with the independent variable. However, the situation here is different.

According to the definition of eccentricity vector:

$$\vec{e} = -\vec{i} - h(\vec{k} \times \vec{v}), \quad (7.39)$$

taking the cross product of the latter relation and the unit vector $\vec{k}$:

$$\vec{k} \times \vec{e} = -\vec{j} + h\vec{v}, \quad (7.40)$$

and taking into account the value of the eccentricity vector, we get:

$$\lambda_1 \vec{a}_2 - \lambda_2 \vec{a}_1 = -\vec{j} + h\vec{v}. \quad (7.41)$$

The procedure consists in differentiating Eq. (??) with respect to the variable u . This yields the expression:

$$\frac{d\lambda_1}{du} \vec{a}_2 - \frac{d\lambda_2}{du} \vec{a}_1 + \lambda_1 \frac{d\vec{a}_2}{du} - \lambda_2 \frac{d\vec{a}_1}{du} = -\frac{d\vec{j}}{du} + \frac{dh}{du} \vec{v} + h \frac{d\vec{v}}{du}. \quad (7.42)$$

The next step is to estimate the derivatives of the unit vectors involved with respect to u .

- Orbital frame $(\vec{i}, \vec{j}, \vec{k})$: because

$$\frac{d}{du}(\cdot) = \frac{d}{d\tau}(\cdot) \lambda_3^{\frac{3}{2}} \hat{r}, \quad (7.43)$$

the derivatives of $(\vec{i}, \vec{j}, \vec{k})$, made in the inertial frame, turn out to be:

$$\frac{d\vec{i}}{du} = \frac{\ell}{\hat{r}} \vec{j} \quad (7.44a)$$

$$\frac{d\vec{j}}{du} = -\frac{\ell}{\hat{r}} \vec{i} + \frac{\lambda_3^2 \hat{r}^2}{\ell} (\vec{a}_p \cdot \vec{k}) \vec{k} \quad (7.44b)$$

$$\frac{d\vec{k}}{du} = -\frac{\lambda_3^2 \hat{r}^2}{\ell} (\vec{a}_p \cdot \vec{k}) \vec{j}. \quad (7.44c)$$

- $Ox_a y_a z_a$ frame $(\vec{a}_1, \vec{a}_2, \vec{k})$: starting from the relation:

$$[\vec{a}_1, \vec{a}_2, \vec{k}] = [\vec{i}, \vec{j}, \vec{k}] M, \quad (7.45)$$

where the matrix M is:

$$M = \begin{pmatrix} \cos(\nu) & \sin(\nu) & 0 \\ -\sin(\nu) & \cos(\nu) & 0 \\ 0 & 0 & 1 \end{pmatrix}. \quad (7.46)$$

Therefore:

$$\left[\frac{d\vec{a}_1}{du}, \frac{d\vec{a}_2}{du}, \frac{d\vec{k}}{du} \right] = \left[\frac{d\vec{i}}{du}, \frac{d\vec{j}}{du}, \frac{d\vec{k}}{du} \right] M + [\vec{i}, \vec{j}, \vec{k}] \frac{dM}{du}. \quad (7.47)$$

Taking into account that

$$\left[\frac{d\vec{i}}{du}, \frac{d\vec{j}}{du}, \frac{d\vec{k}}{du} \right] = [\vec{i}, \vec{j}, \vec{k}] A, \quad (7.48)$$

with

$$A = \begin{pmatrix} 0 & -\hat{\omega}_3 & 0 \\ \hat{\omega}_3 & 0 & -\hat{\omega}_1 \\ 0 & \hat{\omega}_1 & 0 \end{pmatrix}, \quad \begin{cases} \hat{\omega}_1 = \frac{\lambda_3^2 \hat{r}^2}{\ell} (\vec{a}_p \cdot \vec{k}) \\ \hat{\omega}_3 = \frac{\ell}{\hat{r}}, \end{cases} \quad (7.49)$$

and

$$\frac{dM}{du} = \frac{dv}{du} N \quad \text{with} \quad N = \begin{pmatrix} -\sin(\nu) & \cos(\nu) & 0 \\ -\cos(\nu) & -\sin(\nu) & 0 \\ 0 & 0 & 0 \end{pmatrix}, \quad (7.50)$$

we get the relation:

$$\left[\frac{d\vec{a}_1}{du}, \frac{d\vec{a}_2}{du}, \frac{d\vec{k}}{du} \right] = [\vec{a}_1, \vec{a}_2, \vec{k}] \left(M^T A M + \frac{dv}{du} M^T N \right). \quad (7.51)$$

The matrix B:

$$B = M^T A M + \frac{dv}{du} M^T N, \quad (7.52)$$

turns out to be

$$B = \begin{pmatrix} 0 & \frac{dv}{du} - \hat{\omega}_3 & \hat{\omega}_1 \sin(\nu) \\ \hat{\omega}_3 - \frac{dv}{du} & 0 & -\hat{\omega}_1 \cos(\nu) \\ -\hat{\omega}_1 \sin(\nu) & \hat{\omega}_1 \cos(\nu) & 0 \end{pmatrix}, \quad (7.53)$$

where the following relations are deduced:

$$\frac{d\vec{a}_1}{du} = + \left(\hat{\omega}_3 - \frac{dv}{du} \right) \vec{a}_2 - \hat{\omega}_1 \sin(\nu) \vec{k} \quad (7.54a)$$

$$\frac{d\vec{a}_2}{du} = - \left(\hat{\omega}_3 - \frac{dv}{du} \right) \vec{a}_1 + \hat{\omega}_1 \cos(\nu) \vec{k} \quad (7.54b)$$

$$\frac{d\vec{k}}{du} = -\hat{\omega}_1 (-\sin(\nu) \vec{a}_1 + \cos(\nu) \vec{a}_2). \quad (7.54c)$$

Finally, taking the dot product between Eq. (7.42) and unit vectors $(\vec{a}_1, \vec{a}_2, \vec{k})$, keeping in mind the derivatives calculated and the expressions:

$$\vec{a}_1 = \cos(\nu) \vec{i} - \sin(\nu) \vec{j} \quad (7.55a)$$

$$\vec{a}_2 = \sin(\nu) \vec{i} + \cos(\nu) \vec{j} \quad (7.55b)$$

$$\vec{v} = \frac{1}{h} [-(\lambda_2 + \sin(\nu)) \vec{a}_1 + (\lambda_1 + \cos(\nu)) \vec{a}_2] \quad (7.55c)$$

$$\frac{d\vec{v}}{du} = \frac{d\vec{v}}{d\tau} \frac{d\tau}{du} = \left(-\frac{\vec{i}}{r^2} + \vec{a}_p \right) \lambda_3^{\frac{3}{2}} \hat{r}, \quad (7.55d)$$

such that $\vec{v}$ is deduced from Eq. (7.41), we arrive at the following set of equations after simplification:

$$\frac{d\lambda_2}{du} - \left(\frac{dv}{du} - \hat{\omega}_3 \right) \lambda_1 = + \frac{\lambda_3^2 \hat{r}^2}{\ell} \left[(\lambda_2 + \sin(\nu)) a_{py} - \frac{\ell^2}{\hat{r}} (\vec{a}_p \cdot \vec{a}_1) \right] \quad (7.56a)$$

$$\frac{d\lambda_1}{du} + \left(\frac{dv}{du} - \hat{\omega}_3 \right) \lambda_2 = + \frac{\lambda_3^2 \hat{r}^2}{\ell} \left[(\lambda_1 + \cos(\nu)) a_{py} + \frac{\ell^2}{\hat{r}} (\vec{a}_p \cdot \vec{a}_2) \right] \quad (7.56b)$$

$$\frac{\ell^2}{\hat{r}} = 1 + \lambda_1 \cos \nu + \lambda_2 \sin \nu. \quad (7.56c)$$

If we introduce the values of the derivatives of (λ_1, λ_2) given by Eqs. ((7.31b)-(7.31c)) the value of $\hat{\omega}_3$, which is necessary to estimate the time evolution of the orbital plane, can be solved.

7.3 RELATIVE MOTION

In this section we will analyze the orbital motion within the orbital plane. Unlike in previous chapters, where the orbital plane was linked to the axes $Ox_0y_0z_0$ (or to the axes $Ox_py_pz_p$), in this section the orbital plane will be linked to new axes $Ox_ay_az_a$ which are defined by the unit vectors $(\vec{a}_1, \vec{a}_2, \vec{k})$; these axes will be numbered as rigid body (5). The particle M will remain the rigid body (2), the axes $Oxyz$ the rigid body (3) and the inertial axes $Ox_1y_1z_1$ the rigid body (1). The Coriolis theorem will be used for this purpose. If $\vec{A}$ is a variable vector with time, according to this theorem it will be:

$$\left. \frac{d\vec{A}}{du} \right|_{(1)} = \left. \frac{d\vec{A}}{du} \right|_{(5)} + \vec{\Omega}_{51} \times \vec{A}, \quad (7.57)$$

where $\vec{\Omega}_{51}$ is the vector defined by

$$\vec{\Omega}_{51} = \hat{w}_1 \cos(\nu) \vec{a}_1 + \hat{w}_1 \sin(\nu) \vec{a}_2 + \hat{w}_3 \vec{k}, \quad \text{with} \quad \begin{cases} \hat{w}_1 = \lambda_3^2 \frac{\hat{r}^2}{\ell} (\vec{a}_p \cdot \vec{k}) \\ \hat{w}_3 = \frac{\ell}{\hat{r}} - \frac{dv}{du} \end{cases} \quad (7.58)$$

when applying the theory of composition of motions and the relation

$$\frac{d\tau}{du} = \lambda_3^{\frac{3}{2}} \hat{r} \quad (7.59)$$

to the angular velocities $\vec{\omega}_{53}$ and $\vec{\omega}_{31}$. The derivative made in the rigid body (5) assumes that the unit vectors $(\vec{a}_1, \vec{a}_2, \vec{k})$ do not change with time.

In the following we calculate the time derivatives, made in the rigid body (5) for different vectors of interest.

7.3.1 Velocity vector $\vec{v}$

It is known that:

$$\left. \frac{d\vec{v}}{d\tau} \right|_{(1)} = -\frac{\vec{i}}{r^2} + \vec{a}_p, \quad (7.60)$$

therefore

$$\left. \frac{d\vec{v}}{du} \right|_{(1)} = \frac{1}{\lambda_3^{\frac{1}{2}} \hat{r}} (-\vec{i} + \lambda_3^{\frac{3}{2}} \hat{r}^2 \vec{a}_p). \quad (7.61)$$

On the other hand:

$$\vec{\Omega}_{51} \times \vec{v} = -\frac{\ell}{\lambda_3^{\frac{1}{2}} \hat{r}} \hat{w}_3 \vec{i} + \hat{w}_3 \frac{dr}{d\tau} \vec{j} + \frac{\ell}{\lambda_3^{\frac{1}{2}} \hat{r}} \hat{w}_1 \vec{k}. \quad (7.62)$$

From these expressions and the Coriolis theorem:

$$\left. \frac{d\vec{v}}{du} \right|_{(5)} = \left. \frac{d\vec{v}}{du} \right|_{(1)} - \vec{\Omega}_{51} \times \vec{v} \quad (7.63)$$

the following value of the derivative in the rigid body (5) is deduced:

$$\left. \frac{d\vec{v}}{du} \right|_{(5)} = \frac{1}{\lambda_3^{\frac{1}{2}} \hat{r}} (\ell \hat{w}_3 - 1) \vec{i} - \hat{w}_3 \frac{dr}{d\tau} \vec{j} - \frac{\ell}{\lambda_3^{\frac{1}{2}} \hat{r}} \hat{w}_1 \vec{k} + \lambda_3^{\frac{3}{2}} \hat{r} \vec{a}_p. \quad (7.64)$$

Note that, taking into account the value of $\hat{w}_1$, the last two terms of the right-hand side of Eq. (7.64) provides:

$$-\frac{\ell}{\lambda_3^{\frac{1}{2}}\hat{r}}\hat{w}_1\vec{k} + \lambda_3^{\frac{3}{2}}\hat{r}\vec{a}_p = \lambda_3^{\frac{3}{2}}\hat{r}\vec{a}_{\parallel p}, \quad (7.65)$$

and, consequently, it is:

$$\left.\frac{d\vec{v}}{du}\right|_{(5)} = \frac{1}{\lambda_3^{\frac{1}{2}}\hat{r}}(\ell\hat{w}_3 - 1)\vec{i} - \hat{w}_3\frac{dr}{d\tau}\vec{j} + \lambda_3^{\frac{3}{2}}\hat{r}\vec{a}_{\parallel p}, \quad (7.66)$$

because, as expected, the normal components to the orbital plane do not intervene in the motion relative to the rigid body (5).

7.3.2 Angular momentum vector $\vec{h}$

From the angular momentum $\vec{h}$ it is known:

$$\left.\frac{d\vec{h}}{du}\right|_{(1)} = \lambda_3^{\frac{5}{2}}\hat{r}^2\vec{i} \times \vec{a}_p, \quad (7.67)$$

and because:

$$\vec{\Omega}_{51} \times \vec{h} = -\lambda_3^{\frac{1}{2}}\ell\hat{w}_1\vec{j} = -\lambda_3^{\frac{5}{2}}\hat{r}^2\vec{a}_{pz}\vec{j}, \quad (7.68)$$

applying the Coriolis theorem it follows:

$$\left.\frac{d\vec{h}}{du}\right|_{(5)} = \lambda_3^{\frac{5}{2}}\hat{r}^2(\vec{i} \times \vec{a}_{p\parallel}). \quad (7.69)$$

From this equation the following derivative can be deduced:

$$\frac{dh}{du} = \lambda_3^{\frac{5}{2}}\hat{r}^2 a_{py}. \quad (7.70)$$

7.3.3 Eccentricity vector $\vec{e}$

The eccentricity vector is given by:

$$\vec{e} = -\vec{i} - \vec{h} \times \vec{v}, \quad (7.71)$$

therefore, it is:

$$\left.\frac{d\vec{e}}{du}\right|_{(5)} = -\left.\frac{d\vec{i}}{du}\right|_{(5)} - \left.\frac{d\vec{h}}{du}\right|_{(5)} \times \vec{v} - \vec{h} \times \left.\frac{d\vec{v}}{du}\right|_{(5)}. \quad (7.72)$$

The equation of the hodograph:

$$\vec{v} = \frac{1}{h}(\vec{j} + \vec{k} \times \vec{e}) \quad (7.73)$$

allows to estimate the terms of the right-hand side of Eq. (7.72); these turn out to be:

$$\left. \frac{d\vec{i}}{du} \right|_{(5)} = \frac{dv}{du} \vec{j} \quad (7.74a)$$

$$\left. \frac{d\vec{j}}{du} \right|_{(5)} = -\frac{dv}{du} \vec{i} \quad (7.74b)$$

$$\left. \frac{d\vec{h}}{du} \right|_{(5)} \times \vec{v} = -\frac{\lambda_3^2 \hat{r}^2}{\ell} \mathbf{a}_{py} (\vec{i} + \vec{e}) \quad (7.74c)$$

$$\vec{h} \times \left. \frac{d\vec{v}}{du} \right|_{(5)} = \lambda_3^{\frac{1}{2}} \ell \hat{w}_3 \frac{dr}{d\tau} \vec{i} + \frac{\ell}{\hat{r}} (\ell \hat{w}_3 - 1) \vec{j} + \lambda_3^2 \ell \hat{r} \vec{k} \times \vec{a}_{p\parallel}, \quad (7.74d)$$

which finally provide the desired expression:

$$\left. \frac{d\vec{e}}{du} \right|_{(5)} = \frac{\lambda_3^2 \hat{r}^2}{\ell} \left(\mathbf{a}_{py} [\vec{i} + \vec{e}] - \frac{\ell^2}{\hat{r}} \vec{k} \times \vec{a}_{p\parallel} \right) - \hat{w}_3 \vec{k} \times \vec{e}. \quad (7.75)$$

7.3.4 Governing equations

The vector equation (7.75) can be projected into its coordinates in different bases. If the following relations are taken into account:

$$\vec{e} = \lambda_1 \vec{a}_1 + \lambda_2 \vec{a}_2 \quad (7.76a)$$

$$\vec{j} = -\sin(\nu) \vec{a}_1 + \cos(\nu) \vec{a}_2 \quad (7.76b)$$

$$\left. \frac{d\vec{e}}{du} \right|_{(5)} = \frac{d\lambda_1}{du} \vec{a}_1 + \frac{d\lambda_2}{du} \vec{a}_2, \quad (7.76c)$$

from the hodograph equation it is deduced:

$$\vec{v} = \frac{1}{\lambda_3^{1/2} \ell} [-(\lambda_2 + \sin(\nu)) \vec{a}_1 + (\lambda_1 + \cos(\nu)) \vec{a}_2]. \quad (7.77)$$

All these elements allow to develop the evolution equations of (λ_1, λ_2) defined by:

$$\frac{d\lambda_1}{du} = \left. \frac{d\vec{e}}{du} \right|_{(5)} \cdot \vec{a}_1 \quad (7.78a)$$

$$\frac{d\lambda_2}{du} = \left. \frac{d\vec{e}}{du} \right|_{(5)} \cdot \vec{a}_2. \quad (7.78b)$$

However, these equations involve the derivative of ν with respect to u , that is, the component $\hat{w}_3$ of the angular velocity $\vec{\Omega}_{51}$ that is unknown; nevertheless, this derivative can be calculated from Eq. (7.75) which yields:

$$\begin{aligned} \frac{dv}{du} = \frac{\ell}{\hat{r}} + \frac{\lambda_3^2}{1+\ell} \{ [2\hat{r} - \ell(\ell - \hat{r})] \mathbf{a}_{px} + \\ + (\ell - \hat{r})(\lambda_1 \sin(u) - \lambda_2 \cos(u)) \mathbf{a}_{py} \}. \end{aligned} \quad (7.79)$$

7.4 POSITION OF THE AXES Oxyz

As previously mentioned, in this formulation the orbital plane is linked to the rigid body (5), that is, to the axes $Ox_a y_a z_a$ whose unit vectors are $(\vec{a}_1, \vec{a}_2, \vec{k})$. The axes Oxyz are obtained from the axes $Ox_a y_a z_a$ by means of a rotation of an angle ν around Oz. The position of the axes $Ox_a y_a z_a$ is given, with respect to the inertial frame $Ox_1 y_1 z_1$, by an orthogonal matrix $\mathcal{N}$. In this case we have:

$$[\vec{i}, \vec{j}, \vec{k}] = [\vec{a}_1, \vec{a}_2, \vec{k}] R_3(\nu) \quad (7.80a)$$

$$[\vec{a}_1, \vec{a}_2, \vec{k}] = [\vec{i}_1, \vec{j}_1, \vec{k}_1] \mathcal{N}, \quad (7.80b)$$

where the matrix $R_3(\nu)$ is given by:

$$R_3(\nu) = \begin{pmatrix} \cos(\nu) & -\sin(\nu) & 0 \\ \sin(\nu) & \cos(\nu) & 0 \\ 0 & 0 & 1 \end{pmatrix}. \quad (7.81)$$

Consequently, the axes Oxyz are located, with respect to the inertial frame $Ox_1 y_1 z_1$:

$$[\vec{i}, \vec{j}, \vec{k}] = [\vec{i}_1, \vec{j}_1, \vec{k}_1] Q, \quad Q = \mathcal{N} R_3(\nu). \quad (7.82)$$

If we define $\mathbf{n}$ as the quaternion associated to the matrix $\mathcal{N}$ with components:

$$\mathbf{n} = \chi_4 + (\chi_1, \chi_2, \chi_3), \quad (7.83)$$

and $\mathbf{v}_3$ as the quaternion associated to the matrix $R_3(\nu)$:

$$\mathbf{v}_3 = \cos\left(\frac{\nu}{2}\right) + \left(0, 0, \sin\left(\frac{\nu}{2}\right)\right), \quad (7.84)$$

the matrix $Q = \mathcal{N} R_3(\nu)$ is related with the product quaternion $\mathbf{q} = \mathbf{n} \cdot \mathbf{v}_3$.

If:

$$\mathbf{q} = \varepsilon_4 + (\varepsilon_1, \varepsilon_2, \varepsilon_3) \quad (7.85)$$

are the components of this quaternion $\mathbf{q}$, once the product is made, it yields:

$$\varepsilon_1 = \chi_1 \cos\left(\frac{\nu}{2}\right) + \chi_2 \sin\left(\frac{\nu}{2}\right) \quad (7.86a)$$

$$\varepsilon_2 = \chi_2 \cos\left(\frac{\nu}{2}\right) - \chi_1 \sin\left(\frac{\nu}{2}\right) \quad (7.86b)$$

$$\varepsilon_3 = \chi_3 \cos\left(\frac{\nu}{2}\right) + \chi_4 \sin\left(\frac{\nu}{2}\right) \quad (7.86c)$$

$$\varepsilon_4 = \chi_4 \cos\left(\frac{\nu}{2}\right) - \chi_3 \sin\left(\frac{\nu}{2}\right). \quad (7.86d)$$

7.4.1 Determination of ν

The determination of the value of ν is needed for the evolution of the axes Oxyz. This is accomplished in the following way:

1. Starting from the value of u known in the integration process.
2. Estimation of the angle α by using the values (λ_1, λ_2) : $\alpha = \arctan 2(\lambda_2, \lambda_1)$.
3. Calculation of the eccentric anomaly: $\varepsilon = u - \alpha$.

4. Estimation of the eccentricity and true anomaly:

$$e = \sqrt{\lambda_1^2 + \lambda_2^2} \quad (7.87a)$$

$$\sin(\theta) = \frac{\sqrt{1-e^2} \sin(\varpi)}{1 - e \cos(\varpi)} \quad (7.87b)$$

$$\cos(\theta) = \frac{\cos(\varpi) - e}{1 - e \cos(\varpi)} \quad (7.87c)$$

$$\theta = \arctan 2(\sin(\theta), \cos(\theta)). \quad (7.87d)$$

5. Finally, the value of ν is determined as $\nu = \theta + \alpha$.

7.4.2 Initial conditions

The starting initial conditions are the dimensionless initial values of the Cartesian position and velocity vectors in inertial frame coordinates, namely $(\vec{x}_0, \vec{v}_0)$. The initial values, at $t = 0$, of the variables involved in the integration are chosen as follows:

- The unit vector $\vec{a}_1$ is fixed by the initial eccentricity vector $\vec{a}_1 = \frac{1}{e_0} \vec{e}_0$. Later we will see how it should be chosen in the case of circular orbit. Therefore, the angle $\alpha_0 = 0$.
- Because $\nu = \theta + \alpha$, the value of ν_0 coincides with the initial true anomaly: $\nu_0 = \theta_0$.
- Taking into account that $\alpha_0 = 0$, the values of (λ_1, λ_2) are $\lambda_1(0) = e_0$ and $\lambda_2(0) = 0$.
- Because $u = \varpi + \alpha$, the initial value u_0 coincides with the initial eccentric anomaly: $u_0 = \varpi_0$.
- The coordinate λ_3 coincides with the value of the semi-major axis at $t = 0$: $\lambda_3(0) = a_0$.
- To find the initial values of the Euler-Rodrigues parameters $(\chi_1, \chi_2, \chi_3, \chi_4)$ the basis $(\vec{a}_1, \vec{a}_2, \vec{k})$ should be located in the initial instant; this basis coincides with the perifocal frame. From the matrix $\mathcal{N}$, which in the initial instant coincides with the matrix Q , the initial values of the Euler-Rodrigues parameters are obtained.

7.5 SUMMARY OF EQUATIONS

The equations that have to be integrated are:

$$\frac{d\tau}{du} = \lambda_3^{\frac{3}{2}} \hat{r} \quad (7.88a)$$

$$\frac{d\lambda_1}{du} = \lambda_3^2 \{ (\ell^2 \sin(u) - 2\hat{r}\lambda_2) a_{px} + \ell[(1 + \hat{r}) \cos(u) - \lambda_1] a_{py} \} \quad (7.88b)$$

$$\frac{d\lambda_2}{du} = \lambda_3^2 \{ (-\ell^2 \cos(u) + 2\hat{r}\lambda_1) a_{px} + \ell[(1 + \hat{r}) \sin(u) - \lambda_2] a_{py} \} \quad (7.88c)$$

$$\frac{d\lambda_3}{du} = 2\lambda_3^3 [(\lambda_1 \sin(u) - \lambda_2 \cos(u)) a_{px} + \ell a_{py}] \quad (7.88d)$$

$$\frac{d\chi_1}{du} = \frac{1}{2} \{ \chi_4 \hat{w}_1 \cos(v) - \chi_3 \hat{w}_1 \sin(v) + \chi_2 \hat{w}_3 \} \quad (7.88e)$$

$$\frac{d\chi_2}{du} = \frac{1}{2} \{ \chi_3 \hat{w}_1 \cos(v) + \chi_4 \hat{w}_1 \sin(v) - \chi_1 \hat{w}_3 \} \quad (7.88f)$$

$$\frac{d\chi_3}{du} = \frac{1}{2} \{ -\chi_2 \hat{w}_1 \cos(v) + \chi_1 \hat{w}_1 \sin(v) + \chi_4 \hat{w}_3 \} \quad (7.88g)$$

$$\frac{d\chi_4}{du} = \frac{1}{2} \{ -\chi_1 \hat{w}_1 \cos(v) - \chi_2 \hat{w}_1 \sin(v) - \chi_3 \hat{w}_3 \}. \quad (7.88h)$$

In the integration process, the following relations should be taken into account:

$$\ell = \sqrt{1 - e^2} = \sqrt{1 - \lambda_1^2 - \lambda_2^2} \quad (7.89a)$$

$$\hat{r} = 1 - \lambda_1 \cos(u) - \lambda_2 \sin(u) \quad (7.89b)$$

$$r = \lambda_3 \hat{r} \quad (7.89c)$$

$$\frac{dr}{du} = \lambda_3 (\lambda_1 \sin(u) - \lambda_2 \cos(u)) \quad (7.89d)$$

$$\hat{w}_1 = \lambda_3^2 \frac{\hat{r}^2}{\ell} a_{pz} \quad (7.89e)$$

$$\hat{w}_3 = -\frac{\lambda_3^2}{1 + \ell} \{ [2\hat{r} - \ell(\ell - \hat{r})] a_{px} + (\ell - \hat{r})(\lambda_1 \sin(u) - \lambda_2 \cos(u)) a_{py} \} \quad (7.89f)$$

$$\varepsilon_1 = \chi_1 \cos\left(\frac{v}{2}\right) + \chi_2 \sin\left(\frac{v}{2}\right) \quad (7.89g)$$

$$\varepsilon_2 = \chi_2 \cos\left(\frac{v}{2}\right) - \chi_1 \sin\left(\frac{v}{2}\right) \quad (7.89h)$$

$$\varepsilon_3 = \chi_3 \cos\left(\frac{v}{2}\right) + \chi_4 \sin\left(\frac{v}{2}\right) \quad (7.89i)$$

$$\varepsilon_4 = \chi_4 \cos\left(\frac{v}{2}\right) - \chi_3 \sin\left(\frac{v}{2}\right). \quad (7.89j)$$

Note that the value of v should be calculated as indicated in Section 7.4.1. These equations should be integrated from the initial conditions indicated in Section 7.4.2.

7.5.1 Time Element

The equation of time (7.88a), in its right-hand side, has terms that do not cancel when the perturbation tends to zero. To improve the performances of the system, we can introduce a time element that allows to have a homogeneous set of equations, all of them with the same characteristics. A solution for the time τ is sought with the form:

$$\tau = \tau_{np} + \tau_{te}, \quad (7.90)$$

where τ_{np} is based on a solution of the unperturbed problem. This solution is:

$$\tau_{np} = \lambda_3^{3/2} (u - \lambda_1 \sin(u) + \lambda_2 \cos(u)) + k, \quad (7.91)$$

where the constant k is adjusted so that the initial conditions are verified, that is, at the initial instant $\tau_{np} = 0$. Since at the initial instant:

$$\lambda_1(0) = e_0, \quad \lambda_2(0) = 0, \quad \lambda_3(0) = a_0, \quad u_0 = \alpha_0, \quad (7.92)$$

we get the solution:

$$\tau_{np} = \lambda_3^{3/2} (u - \lambda_1 \sin(u) + \lambda_2 \cos(u)) - a_0^{3/2} (\alpha_0 - e_0 \sin(\alpha_0)). \quad (7.93)$$

Equation (7.88a) for the time τ , if it is taken into account the decomposition given by Eq. (7.90), adopts the form:

$$\frac{d\tau_{te}}{du} = \lambda_3^{3/2} \hat{r} - \frac{d\tau_{np}}{du}. \quad (7.94)$$

If the solution given by Eq. (7.93) is introduced in the right-hand side of Eq. (7.94), taking into account that the value $k = -a_0^{3/2} (\alpha_0 - e_0 \sin(\alpha_0))$ is constant and has a null derivative, the following equation is reached:

$$\frac{d\tau_{te}}{du} = \lambda_3^{7/2} \{ -(\ell^2 - q^2 + 3uq - 2\hat{r})(a_{px}) + \ell(2q - 3u)(a_{py}) \}, \quad (7.95)$$

where q is defined as:

$$q = \lambda_1 \sin(u) - \lambda_2 \cos(u). \quad (7.96)$$

Equation (7.95) has the same structure as the rest of the equations of the system, that is, when the perturbation $\vec{a}_p$ tends to zero the right-hand side tends to zero. If the time element is introduced in the system, an additional relation should be considered in the integration process:

$$\tau = \lambda_3^{3/2} (u - q) - a_0^{3/2} (\alpha_0 - e_0 \sin(\alpha_0)) + \tau_{te}. \quad (7.97)$$

7.6 USEFUL RELATIONS

The state vector and eccentricity vector of the orbit are expressed here as a function of the ElliDROMO variables.

7.6.1 Position vector

The position vector of the particle can be expressed in different bases. The values provided are dimensionless, such that to recover dimensional values only a multiplication by the characteristic length L_c should be applied.

- In the frame $Oxyz$:

$$\vec{x} = \vec{r} = \lambda_3 (1 - \lambda_1 \cos(u) - \lambda_2 \sin(u)) \vec{i}. \quad (7.98)$$

- In the frame $Ox_a y_a z_a$:

$$\vec{x} = \lambda_3 (1 - \lambda_1 \cos(u) - \lambda_2 \sin(u)) [\cos(v) \vec{a}_1 + \sin(v) \vec{a}_2]. \quad (7.99)$$

- In the frame $Ox_1y_1z_1$:

$$\vec{x} = \lambda_3 (1 - \lambda_1 \cos(u) - \lambda_2 \sin(u)) [\vec{i}_1, \vec{j}_1, \vec{k}_1] \mathcal{N}_0 \begin{pmatrix} \cos(v) \\ \sin(v) \\ 0 \end{pmatrix}, \quad (7.100)$$

where the rotation matrix $\mathcal{N}_0$ is the matrix defined by:

$$\mathcal{N}_0 = \begin{bmatrix} 1 - 2(\chi_2^2 + \chi_3^2) & 2\chi_1\chi_2 - 2\chi_4\chi_3 & 2\chi_1\chi_3 + 2\chi_4\chi_2 \\ 2\chi_1\chi_2 + 2\chi_4\chi_3 & 1 - 2(\chi_1^2 + \chi_3^2) & 2\chi_2\chi_3 - 2\chi_4\chi_1 \\ 2\chi_1\chi_3 - 2\chi_4\chi_2 & 2\chi_3\chi_2 + 2\chi_4\chi_1 & 1 - 2(\chi_1^2 + \chi_2^2) \end{bmatrix}. \quad (7.101)$$

Introducing the rectangular matrix $\mathcal{N}$ (3×2):

$$\mathcal{N} = \begin{bmatrix} 1 - 2(\chi_2^2 + \chi_3^2) & 2\chi_1\chi_2 - 2\chi_4\chi_3 \\ 2\chi_1\chi_2 + 2\chi_4\chi_3 & 1 - 2(\chi_1^2 + \chi_3^2) \\ 2\chi_1\chi_3 - 2\chi_4\chi_2 & 2\chi_3\chi_2 + 2\chi_4\chi_1 \end{bmatrix}, \quad (7.102)$$

it becomes:

$$\vec{x} = \lambda_3 (1 - \lambda_1 \cos(u) - \lambda_2 \sin(u)) [\vec{i}_1, \vec{j}_1, \vec{k}_1] \mathcal{N} \begin{pmatrix} \cos(v) \\ \sin(v) \end{pmatrix}. \quad (7.103)$$

7.6.2 Velocity vector

The velocity vector of the particle can be expressed in different bases. The values provided are dimensionless, such that to recover dimensional values only a multiplication by the characteristic value $\omega_c \cdot L_c$ should be applied.

- In the frame $Oxyz$:

$$\vec{v} = \lambda_3 [\vec{i}, \vec{j}] \begin{pmatrix} \lambda_1 \sin(v) - \lambda_2 \cos(v) \\ \lambda_1 \cos(v) + \lambda_2 \sin(v) + 1 \end{pmatrix}. \quad (7.104)$$

- In the frame $Ox_a y_a z_a$:

$$\begin{aligned} \vec{v} &= \lambda_3 [\vec{a}_1, \vec{a}_2, \vec{k}] \mathcal{R}_3(v) \begin{pmatrix} \lambda_1 \sin(v) - \lambda_2 \cos(v) \\ \lambda_1 \cos(v) + \lambda_2 \sin(v) + 1 \\ 0 \end{pmatrix} \\ &= \lambda_3 [\vec{a}_1, \vec{a}_2, \vec{k}] \begin{pmatrix} -\lambda_2 - \sin(v) \\ \lambda_1 + \cos(v) \\ 0 \end{pmatrix}. \end{aligned} \quad (7.105)$$

- In the frame $Ox_1y_1z_1$:

$$\begin{aligned} \vec{v} &= \lambda_3 [\vec{i}_1, \vec{j}_1, \vec{k}_1] \mathcal{N}_0 \mathcal{R}_3(v) \begin{pmatrix} \lambda_1 \sin(v) - \lambda_2 \cos(v) \\ \lambda_1 \cos(v) + \lambda_2 \sin(v) + 1 \\ 0 \end{pmatrix} \\ &= \lambda_3 [\vec{i}_1, \vec{j}_1, \vec{k}_1] \mathcal{N}_0 \begin{pmatrix} -\lambda_2 - \sin(v) \\ \lambda_1 + \cos(v) \\ 0 \end{pmatrix}, \end{aligned} \quad (7.106)$$

where $\mathcal{N}_0$ is the matrix defined in Eq. (7.101).

7.6.3 Eccentricity vector

The eccentricity vector of the particle can be expressed in different bases.

- In the frame $Oxyz$:

$$\vec{e} = [\vec{i}, \vec{j}, \vec{k}] R_3^T(\nu) \begin{pmatrix} \lambda_1 \\ \lambda_2 \\ 0 \end{pmatrix} = [\vec{i}, \vec{j}, \vec{k}] \begin{pmatrix} \lambda_1 \cos(\nu) + \lambda_2 \sin(\nu) \\ -\lambda_1 \sin(\nu) + \lambda_2 \cos(\nu) \\ 0 \end{pmatrix}. \quad (7.107)$$

- In the frame $Ox_a y_a z_a$:

$$\vec{e} = \lambda_1 \vec{a}_1 + \lambda_2 \vec{a}_2. \quad (7.108)$$

- In the frame $Ox_1 y_1 z_1$:

$$\vec{e} = [\vec{i}_1, \vec{j}_1, \vec{k}_1] N \begin{pmatrix} \lambda_1 \\ \lambda_2 \end{pmatrix}, \quad (7.109)$$

where N is the matrix defined in Eq. (7.102).

DROMO propagator was conceived as a high-performance propagator to integrate slightly perturbed environments. However, when the perturbation is significant, as for example in deep flybys, the performance of DROMO deteriorates due to singularity problems in the evolution equations (Roa, Sanjurjo-Rivo, and Peláez [73]).

The following formulation, intended to extend the DROMO performance for Strongly Perturbed Environments (SPE), is based on the definition of new axes to describe the motion of the particle in the orbital plane. It is determined not only for DROMO propagator, but for ElliDROMO as well.

8.1 THE DROMO-SPE FORMULATION

The relative motion of the particle M was described in Chapter 6 with respect to the axes $Ox_p y_p z_p$ through the variables $(\zeta_1, \zeta_2, \zeta_3)$. This parameterization can be reformulated by considering the variables (ξ_1, ξ_2, ξ_3) , that is, the variables related to the orbital frame $Oxyz$ (Fig. 8.1). The drag motion is again described with the quaternion $\mathbf{n}$ related to the axes $Ox_p y_p z_p$.

8.1.1 Summary of equations

The motion of the particle M in the orbital frame is described by the variables:

$$\xi_1 = \zeta_1 \cos(\sigma) + \zeta_2 \sin(\sigma) \quad (8.1a)$$

$$\xi_2 = -\zeta_1 \sin(\sigma) + \zeta_2 \cos(\sigma) \quad (8.1b)$$

$$\xi_3 = \zeta_3, \quad (8.1c)$$

where σ is the angle defined by the axes Ox_p and Ox (Fig. 8.1). Therefore, there is a bijective correspondence between $(\zeta_1, \zeta_2, \zeta_3)$ and (ξ_1, ξ_2, ξ_3) . If the latter are known, the first can be obtained by the change of variable:

$$\zeta_1 = \xi_1 \cos(\sigma) - \xi_2 \sin(\sigma) \quad (8.2a)$$

$$\zeta_2 = \xi_1 \sin(\sigma) + \xi_2 \cos(\sigma) \quad (8.2b)$$

$$\zeta_3 = \xi_3. \quad (8.2c)$$

The initial conditions are given by:

$$\sigma = \sigma_0 \Rightarrow \begin{cases} \xi_1(\sigma_0) = e_0 \cos(\sigma_0) \\ \xi_2(\sigma_0) = -e_0 \sin(\sigma_0) \\ \xi_3(\sigma_0) = \frac{1}{\sqrt{r_0(1 + e_0 \cos(\sigma_0))}}. \end{cases} \quad (8.6)$$

The structure of the equations is simpler, but an important characteristic is lost; indeed, the coordinates (ζ_1, ζ_2) are integrals of the unperturbed problem and, therefore, maintain a constant value in that problem. However, the variables (ξ_1, ξ_2) are not integrals of the unperturbed problem, and unlike the coordinates (ζ_1, ζ_2) , they do not maintain constant values in the problem. Consequently, the goodness of the equations should be checked numerically to know if there is any problem with the accuracy of the solution.

This change of variables provides a simpler system of equations to describe the relative motion where two sines and two cosines do not explicitly appear in the equations. This improves the precision of the calculations by reducing the truncation error with less degradation in the propagation of the error. In addition, when the magnitude of the perturbing acceleration is close to unity, the terms on the right-hand side of Eqs. (8.3b), (8.3c) and (8.3d), related with the partial derivatives of (ξ_1, ξ_2, ξ_3) , become of the same order of magnitude. As it will be shown later, it improves the performance of the propagator giving more stability.

8.1.2 Useful relations

The state vector and eccentricity vector of the orbit are expressed here as a function of the DROMO-SPE variables in different bases and with dimensionless values. The dimensional values can be easily obtained by following the same procedure as in the classical DROMO.

8.1.2.1 Position vector

- In the frame Oxyz:

$$\vec{x} = r\vec{i} = \frac{1}{\xi_3^2(1 + \xi_1)} \vec{i}. \quad (8.7)$$

- In the frame O $x_p y_p z_p$:

$$\vec{x} = \frac{1}{\xi_3^2(1 + \xi_1)} [\cos(\sigma)\vec{u}_1 + \sin(\sigma)\vec{u}_2]. \quad (8.8)$$

- In the frame O $x_1 y_1 z_1$:

$$\vec{x} = \frac{1}{\xi_3^2(1 + \xi_1)} [\vec{i}_1, \vec{j}_1, \vec{k}_1] P_0 \begin{pmatrix} \cos(\sigma) \\ \sin(\sigma) \\ 0 \end{pmatrix}, \quad (8.9)$$

where the rotation matrix P_0 is the defined in Eqs. (6.12) and (6.11). Introducing the rectangular matrix N (3×2):

$$N = \begin{bmatrix} 1 - 2(\eta_2^2 + \eta_3^2) & 2\eta_1\eta_2 - 2\eta_4\eta_3 \\ 2\eta_1\eta_2 + 2\eta_4\eta_3 & 1 - 2(\eta_1^2 + \eta_3^2) \\ 2\eta_1\eta_3 - 2\eta_4\eta_2 & 2\eta_3\eta_2 + 2\eta_4\eta_1 \end{bmatrix}, \quad (8.10)$$

it becomes:

$$\vec{x} = \frac{1}{\xi_3^2(1 + \xi_1)} [\vec{i}_1, \vec{j}_1, \vec{k}_1] N \begin{pmatrix} \cos(\sigma) \\ \sin(\sigma) \end{pmatrix}. \quad (8.11)$$

8.1.2.2 *Velocity vector*

- In the frame $Oxyz$:

$$\vec{v} = \frac{dr}{d\tau} \vec{i} + \frac{h}{r} \vec{j} = \xi_3 [-\xi_2 \vec{i} + (1 + \xi_1) \vec{j}]. \quad (8.12)$$

- In the frame $Ox_p y_p z_p$:

$$\vec{v} = \xi_3 [(-\sin(\sigma) - \xi_1 \sin(\sigma) - \xi_2 \cos(\sigma)) \vec{u}_1 + (\cos(\sigma) + \xi_1 \cos(\sigma) - \xi_2 \sin(\sigma)) \vec{u}_2]. \quad (8.13)$$

- In the frame $Ox_1 y_1 z_1$:

$$\vec{v} = \xi_3 [\vec{i}_1, \vec{j}_1, \vec{k}_1] P_0 \begin{pmatrix} -\sin(\sigma) - \xi_1 \sin(\sigma) - \xi_2 \cos(\sigma) \\ \cos(\sigma) + \xi_1 \cos(\sigma) - \xi_2 \sin(\sigma) \\ 0 \end{pmatrix}, \quad (8.14)$$

where P_0 is the matrix defined in Eqs. (6.12) and (6.11). By using the matrix N defined in Eq. (8.10) it turns out

$$\vec{v} = \xi_3 [\vec{i}_1, \vec{j}_1, \vec{k}_1] N \begin{pmatrix} -\sin(\sigma) - \xi_1 \sin(\sigma) - \xi_2 \cos(\sigma) \\ \cos(\sigma) + \xi_1 \cos(\sigma) - \xi_2 \sin(\sigma) \end{pmatrix}. \quad (8.15)$$

8.1.2.3 *Eccentricity vector*

- In the frame $Oxyz$:

$$\vec{e} = \xi_1 \vec{i} + \xi_2 \vec{j}. \quad (8.16)$$

- In the frame $Ox_p y_p z_p$:

$$\vec{e} = [\vec{u}_1, \vec{u}_2, \vec{k}] R_3(\sigma) \begin{pmatrix} \xi_1 \\ \xi_2 \\ 0 \end{pmatrix} = [\vec{u}_1, \vec{u}_2, \vec{k}] \begin{pmatrix} \xi_1 \cos(\sigma) - \xi_2 \sin(\sigma) \\ \xi_1 \sin(\sigma) + \xi_2 \cos(\sigma) \\ 0 \end{pmatrix}. \quad (8.17)$$

- In the frame $Ox_1 y_1 z_1$:

$$\vec{e} = [\vec{i}_1, \vec{j}_1, \vec{k}_1] P_0 \begin{pmatrix} \xi_1 \cos(\sigma) - \xi_2 \sin(\sigma) \\ \xi_1 \sin(\sigma) + \xi_2 \cos(\sigma) \\ 0 \end{pmatrix}, \quad (8.18)$$

where P_0 is the matrix defined in Eqs. (6.12) and (6.11). Making use of the matrix N defined in Eq. (8.10) it becomes

$$\vec{e} = [\vec{i}_1, \vec{j}_1, \vec{k}_1] N \begin{pmatrix} \xi_1 \cos(\sigma) - \xi_2 \sin(\sigma) \\ \xi_1 \sin(\sigma) + \xi_2 \cos(\sigma) \end{pmatrix}. \quad (8.19)$$

8.2 THE ELLIDROMO-SPE FORMULATION

The relative motion of the particle M particle was described in Chapter 7 with respect to the axes $Ox_a y_a z_a$ through the variables $(\lambda_1, \lambda_2, \lambda_3)$. This parameterization can be reformulated by considering the variables $(\varphi_1, \varphi_2, \varphi_3)$, that is, the variables related to the frame $Ox_4 y_4 z_4$ (see Fig. 8.2). These axes are chosen so that they do not depend on the angle ν , but on the independent variable u , defining the axis Ox_4 the eccentric anomaly. The drag motion is again described with the quaternion $\mathbf{n}$ associated with the axes $Ox_a y_a z_a$.

Figure 8.2: Kinematic elements in ElliDROMO-SPE.

8.2.1 Summary of equations

The motion of the particle M in the frame $Ox_4y_4z_4$ is described by the variables:

$$\varphi_1 = \lambda_1 \cos(u) + \lambda_2 \sin(u) \quad (8.20a)$$

$$\varphi_2 = -\lambda_1 \sin(u) + \lambda_2 \cos(u) \quad (8.20b)$$

$$\varphi_3 = \lambda_3, \quad (8.20c)$$

where u is the angle defined by the axes Ox_a and Ox_4 (Fig. 8.2). There is a bijective correspondence between $(\lambda_1, \lambda_2, \lambda_3)$ and $(\varphi_1, \varphi_2, \varphi_3)$. If the latter are known, the first can be obtained by the change of variable:

$$\lambda_1 = \varphi_1 \cos(u) - \varphi_2 \sin(u) \quad (8.21a)$$

$$\lambda_2 = \varphi_1 \sin(u) + \varphi_2 \cos(u) \quad (8.21b)$$

$$\lambda_3 = \varphi_3. \quad (8.21c)$$

8.2.1.1 Using the quaternion linked to $Ox_a y_a z_a$

In terms of the new variables, the governing equations take the form:

$$\frac{d\tau}{du} = \varphi_3^{\frac{3}{2}} \hat{r} \quad (8.22a)$$

$$\frac{d\varphi_1}{du} = \varphi_2 + \varphi_3^2 [-2\hat{r}\varphi_2 a_{px} + \ell(1 + \hat{r} - \varphi_1) a_{py}] \quad (8.22b)$$

$$\frac{d\varphi_2}{du} = -\varphi_1 + \varphi_3^2 [(-\ell^2 + 2\hat{r}\varphi_1) a_{px} - \ell\varphi_2 a_{py}] \quad (8.22c)$$

$$\frac{d\varphi_3}{du} = 2\varphi_3^3 [-\varphi_2 a_{px} + \ell a_{py}] \quad (8.22d)$$

$$\frac{d\chi_1}{du} = \frac{1}{2} \{ \chi_4 \hat{w}_1 \cos(v) - \chi_3 \hat{w}_1 \sin(v) + \chi_2 \hat{w}_3 \} \quad (8.22e)$$

$$\frac{d\chi_2}{du} = \frac{1}{2} \{ \chi_3 \hat{w}_1 \cos(v) + \chi_4 \hat{w}_1 \sin(v) - \chi_1 \hat{w}_3 \} \quad (8.22f)$$

$$\frac{d\chi_3}{du} = \frac{1}{2} \{ -\chi_2 \hat{w}_1 \cos(v) + \chi_1 \hat{w}_1 \sin(v) + \chi_4 \hat{w}_3 \} \quad (8.22g)$$

$$\frac{d\chi_4}{du} = \frac{1}{2} \{ -\chi_1 \hat{w}_1 \cos(v) - \chi_2 \hat{w}_1 \sin(v) - \chi_3 \hat{w}_3 \}. \quad (8.22h)$$

The following relations should be taken into account in the integration process:

$$\ell = \sqrt{1 - e^2} = \sqrt{1 - \varphi_1^2 - \varphi_2^2} \quad (8.23a)$$

$$\hat{r} = 1 - \varphi_1 \quad (8.23b)$$

$$r = \varphi_3 \hat{r} \quad (8.23c)$$

$$\frac{dr}{du} = -\varphi_2 \varphi_3 \quad (8.23d)$$

$$\hat{w}_1 = \varphi_3^2 \frac{\hat{r}^2}{\ell} a_{pz} \quad (8.23e)$$

$$\hat{w}_3 = -\frac{\varphi_3^2}{1 + \ell} \{ [2\hat{r} - \ell(\ell - \hat{r})] a_{px} - (\ell - \hat{r}) \varphi_2 a_{py} \} \quad (8.23f)$$

$$\varepsilon_1 = \chi_1 \cos\left(\frac{\nu}{2}\right) + \chi_2 \sin\left(\frac{\nu}{2}\right) \quad (8.23g)$$

$$\varepsilon_2 = \chi_2 \cos\left(\frac{\nu}{2}\right) - \chi_1 \sin\left(\frac{\nu}{2}\right) \quad (8.23h)$$

$$\varepsilon_3 = \chi_3 \cos\left(\frac{\nu}{2}\right) + \chi_4 \sin\left(\frac{\nu}{2}\right) \quad (8.23i)$$

$$\varepsilon_4 = \chi_4 \cos\left(\frac{\nu}{2}\right) - \chi_3 \sin\left(\frac{\nu}{2}\right). \quad (8.23j)$$

The initial conditions are given by:

$$u = u_0 \Rightarrow \begin{cases} \varphi_1(u_0) = e_0 \cos(\alpha_0) \\ \varphi_2(u_0) = -e_0 \sin(\alpha_0) \\ \varphi_3(u_0) = a_0. \end{cases} \quad (8.24)$$

The new variables (φ_1, φ_2) , as in the case of DROMO-SPE, are not integrals of the unperturbed problem. Nevertheless, the change of variables avoids the appearance of three sines and three cosines in the equations of the relative motion resulting in a similar behavior than DROMO-SPE with respect to the classical version.

If the time element is incorporated, the differential equation (8.22a) is replaced by

$$\frac{d\tau_{te}}{du} = \varphi_3^{\frac{7}{2}} [-(\ell^2 - \varphi_2^2 - 3u\varphi_2 - 2\hat{r}) a_{px} + \ell(-2\varphi_2 - 3u) a_{py}], \quad (8.25)$$

where the following relation is considered in the integration process:

$$\tau = \varphi_3^{\frac{3}{2}} (u + \varphi_2) - a_0^{\frac{3}{2}} (\alpha_0 - e_0 \sin(\alpha_0)) + \tau_{te}. \quad (8.26)$$

8.2.2 Useful relations

The state vector and eccentricity vector of the orbit are expressed here as a function of the ElliDROMO-SPE variables in different bases. The determination of the values is analogous to the classical ElliDROMO, that is, providing dimensionless values.

8.2.2.1 Position vector

- In the frame Oxyz:

$$\vec{x} = r\vec{i} = \varphi_3 (1 - \varphi_1) \vec{i}. \quad (8.27)$$

- In the frame $Ox_a y_a z_a$:

$$\vec{x} = \varphi_3 (1 - \varphi_1) [\cos(\nu) \vec{a}_1 + \sin(\nu) \vec{a}_2]. \quad (8.28)$$

- In the frame $Ox_4y_4z_4$:

$$\vec{x} = \varphi_3 (1 - \varphi_1) [\cos(\nu - u) \vec{i}_4 + \sin(\nu - u) \vec{j}_4]. \quad (8.29)$$

- In the frame $Ox_1y_1z_1$:

$$\vec{x} = \varphi_3 (1 - \varphi_1) [\vec{i}_1, \vec{j}_1, \vec{k}_1] \mathcal{N}_0 \begin{pmatrix} \cos(\nu) \\ \sin(\nu) \\ 0 \end{pmatrix}, \quad (8.30)$$

where $\mathcal{N}_0$ is the rotation matrix defined in Eq. (7.101). Introducing the rectangular matrix $\mathcal{N}$ given by Eq. (7.102) it becomes

$$\vec{x} = \varphi_3 (1 - \varphi_1) [\vec{i}_1, \vec{j}_1, \vec{k}_1] \mathcal{N} \begin{pmatrix} \cos(\nu) \\ \sin(\nu) \end{pmatrix}. \quad (8.31)$$

8.2.2.2 Velocity vector

- In the frame $Oxyz$:

$$\vec{v} = \varphi_3 [(\sin(\nu - u) \varphi_1 - \cos(\nu - u) \varphi_2) \vec{i} + (1 + \cos(\nu - u) \varphi_1 + \sin(\nu - u) \varphi_2) \vec{j}]. \quad (8.32)$$

- In the frame $Ox_a y_a z_a$:

$$\vec{v} = \varphi_3 [\vec{a}_1, \vec{a}_2, \vec{k}] R_3(\nu) \begin{pmatrix} \sin(\nu - u) \varphi_1 - \cos(\nu - u) \varphi_2 \\ 1 + \cos(\nu - u) \varphi_1 + \sin(\nu - u) \varphi_2 \\ 0 \end{pmatrix}. \quad (8.33)$$

- In the frame $Ox_4y_4z_4$:

$$\begin{aligned} \vec{v} &= \varphi_3 [\vec{i}_4, \vec{j}_4, \vec{k}] R_3(\nu - u) \begin{pmatrix} \sin(\nu - u) \varphi_1 - \cos(\nu - u) \varphi_2 \\ 1 + \cos(\nu - u) \varphi_1 + \sin(\nu - u) \varphi_2 \\ 0 \end{pmatrix} \\ &= \varphi_3 [\vec{i}_4, \vec{j}_4, \vec{k}] \begin{pmatrix} -\varphi_2 - \sin(\nu - u) \\ \varphi_1 + \cos(\nu - u) \\ 0 \end{pmatrix}. \end{aligned} \quad (8.34)$$

- In the frame $Ox_1y_1z_1$:

$$\vec{v} = \varphi_3 [\vec{i}_1, \vec{j}_1, \vec{k}_1] \mathcal{N}_0 R_3(\nu) \begin{pmatrix} \sin(\nu - u) \varphi_1 - \cos(\nu - u) \varphi_2 \\ 1 + \cos(\nu - u) \varphi_1 + \sin(\nu - u) \varphi_2 \\ 0 \end{pmatrix}, \quad (8.35)$$

where $\mathcal{N}_0$ is the matrix defined in Eq. (7.101).

8.2.2.3 Eccentricity vector

- In the frame $Oxyz$:

$$\begin{aligned} \vec{e} &= (\cos(\nu - u) \varphi_1 + \sin(\nu - u) \varphi_2) \vec{i} + \\ &\quad (-\sin(\nu - u) \varphi_1 + \cos(\nu - u) \varphi_2) \vec{j}. \end{aligned} \quad (8.36)$$

- In the frame $Ox_a y_a z_a$:

$$\vec{e} = [\vec{a}_1, \vec{a}_2, \vec{k}] R_3(\nu) \begin{pmatrix} \cos(\nu - u) \varphi_1 + \sin(\nu - u) \varphi_2 \\ -\sin(\nu - u) \varphi_1 + \cos(\nu - u) \varphi_2 \\ 0 \end{pmatrix}. \quad (8.37)$$

- In the frame $Ox_4 y_4 z_4$:

$$\vec{e} = \varphi_1 \vec{i}_4 + \varphi_2 \vec{j}_4. \quad (8.38)$$

- In the frame $Ox_1 y_1 z_1$:

$$\vec{e} = [\vec{i}_1, \vec{j}_1, \vec{k}_1] \mathcal{N}_0 R_3(\nu) \begin{pmatrix} \cos(\nu - u) \varphi_1 + \sin(\nu - u) \varphi_2 \\ -\sin(\nu - u) \varphi_1 + \cos(\nu - u) \varphi_2 \\ 0 \end{pmatrix}, \quad (8.39)$$

where $\mathcal{N}_0$ is the matrix defined in Eq. (7.101).

8.3 TSIENT'S PROBLEM

Tsien's problem (Tsien [89]) is a planar motion under the application of a constant radial thrust acceleration a_r upon a spacecraft initially on a circular orbit. Let R_0 be the radius of the departure circular orbit, ω_0 its angular velocity and $R_0 \omega_0$ the circular velocity. Expressing the equations of motion in non-dimensional form, the spacecraft is acted upon by a constant dimensionless thrust given by

$$\epsilon = \frac{8 a_r}{R_0 \omega_0^2}. \quad (8.40)$$

Depending on the magnitude of ϵ , three different kinds of motion are distinguished:

- $\epsilon < 1$: the trajectory is bounded by two concentric circles.
- $\epsilon = 1$: the trajectory tends asymptotically to a circular orbit.
- $\epsilon > 1$: the trajectory escapes.

8.3.1 Analysis

The asymptotic case $\epsilon = 1$ corresponds to a highly perturbed orbital scenario, where the perturbation is as strong as the Keplerian term. This scenario is therefore an appropriate playground for assessing the performance of the new DROMO propagators. In such a case, the trajectory of the spacecraft tends asymptotically to a circular orbit of radius $r = 2 R_0$. For ideal conditions, the spacecraft remains in that orbit. However, the numerical errors committed during the integration process prevent the permanence in the asymptotic circular orbit and the spacecraft finally enters in the bounded region or escapes. As a consequence, the case $\epsilon = 1$ of the Tsien's problem is an excellent tool to evaluate the stability of numerical propagators (Baù et al. [13]) by counting the number of revolutions that the spacecraft completes on the asymptotic circular orbit before escaping. The relations between this number and the number of function calls and the runtime employed provide valuable information about the effectiveness of the method.

The calculations have been carried out in a computer with an Intel(R) Xeon(R) E7-4820 v4 @ 2.00 GHz microprocessor under the Windows 10 operative system 64 bits

Figure 8.3: Number of revolutions with respect to the number of function calls (upper) and the runtime (lower) in double precision.

with the Intel Visual Fortran compiler. The trajectory is integrated with an 8th order Runge-Kutta-Fehlberg algorithm (RKF8(7)) with varying integration tolerances in double (from 10^{-9} to 10^{-19}) and quadruple (from 10^{-9} to 10^{-24}) precision. It is started with the initial state vector $\vec{x}_0 = (0, 1)$, $\vec{v}_0 = (-1, 0)$ and propagated until the asymptotic orbit is entered. The integration finishes when the asymptotic orbit is left.

8.3.2 Comparison of results

Tsien's problem for $\epsilon = 1$ is solved with the method proposed in this study in their different versions: DROMO-SPE, ElliDROMO-SPE and ElliDROMO-SPE(TE) (time element incorporated). The new extension has been compared with the classic formulations: DROMO, ElliDROMO and ElliDROMO(TE) to asses the improvement in performance. Besides, the same analysis has been performed with typical special perturbations methods, for the sake of comparison, namely:

- Cowell's method.
- The Kustaanheimo-Stiefel (KS) propagator.
- The Sperling-Bürdet (SB) propagator.
- The Palacios propagator.

Figures 8.3 and 8.4 provide the results in double precision. Figure 8.3 compares the DROMO-SPE extension with respect to the different propagators. Figure 8.4 compares all the DROMO and ElliDROMO formulations, including the new extensions. As it is shown in Figure 8.3, the maximum number of revolutions ranges

Figure 8.4: Number of revolutions with respect to the number of function calls (upper) and the runtime (lower) in double precision.

between three and four before escaping. Palacios and DROMO-SPE methods have a similar evolution and need less function calls to reach any number of revolutions as compared to the other propagators. The same happens with the computational cost (see Fig. 8.3). DROMO and KS methods are the second best performance, followed by SB and Cowell. Note that the formulations based on orbital elements provides mostly better results than the ones based on the state vector.

Moreover, all the new extensions improve their original counterpart being more stable and requiring less computational time (see Fig. 8.4). The comparison between ElliDROMO(TE)-SPE and ElliDROMO-SPE reveals the role of the time element in the new formulation. Unlike ElliDROMO, the ElliDROMO-SPE coordinates are not integrals of the unperturbed problem (see Eqs. (8.22b) and (8.22c)). This means that the time element does not bring any further improvement in the ElliDROMO-SPE formulation because it does not homogenize the set of equations. Consequently, Fig. 8.4 shows that ElliDROMO-SPE requires less function calls and runtime to reach the same revolution on asymptotic orbit than ElliDROMO(TE)-SPE.

Figures 8.5 and 8.6 provide the same results but in quadruple precision. The conclusions are similar to the ones obtained in double precision but now with a greater accuracy, thus preventing numerical noise that usually appears in numerical simulations. Palacios and DROMO-SPE practically define the same curve (see Fig. 8.5), followed by DROMO. SB needs fewer function calls during the integration than Cowell, however, it is the slowest (see Fig. 8.5). The results of Fig. 8.6 are analogous to those computed in double precision. DROMO improves with respect to ElliDROMO(TE)-SPE while the classic versions of ElliDROMO take longer and require more function calls.

Figure 8.5: Number of revolutions with respect to the number of function calls (upper) and the runtime (lower) in quadruple precision.

Figure 8.6: Number of revolutions with respect to the number of function calls (upper) and the runtime (lower) in quadruple precision.

8.4 ARTIFICIAL SATELLITE ORBITING PLUTO

Consider the N-body problem of an artificial satellite orbiting in the Pluto system. The perturbations acting on the satellite, besides the central gravitational term of Pluto, are the gravitational forces of the third body relative to the Sun, the planets (Mercury, Venus, Earth, Mars, Jupiter, Saturn, Uranus and Neptune), the natural satellites of Pluto (Charon, Nix, Hydra, Kerberos and Styx) and the main asteroids (Ceres, Pallas and Vesta).

8.4.1 Analysis

In order to assess the performance of several propagators, the trajectory of the satellite is integrated from two different initial conditions with respect to Pluto at the given epoch. The description of both scenarios is as follows:

Scenario 1:

- Starting state vector:

$$\vec{x}_0 = x_0 [0, 0, -1], \quad \vec{v}_0 = v_0 [\cos(i_0), \sin(i_0), 0], \quad (8.41)$$

such that $x_0 = 10 R_{\text{Pluto}}$ and $v_0 = 1.001 \sqrt{\frac{GM_{\text{Pluto}}}{x_0}}$.

- Initial eccentricity: $e_0 = 0.002001$.
- Initial inclination: $i_0 = 45^\circ$.
- Timespan: 2019/01/01, 12 : 00 : 00 TDB - 2020/01/01, 12 : 00 : 00 TDB.

Scenario 2:

- Starting state vector:

$$\vec{x}_0 = x_0 [1, 0, 0], \quad \vec{v}_0 = v_0 [0, \cos(i_0), \sin(i_0)], \quad (8.42)$$

such that $x_0 = 15 R_{\text{Pluto}}$ and $v_0 = 1.001 \sqrt{\frac{GM_{\text{Pluto}}}{x_0}}$.

- Initial eccentricity: $e_0 = 0.002001$.
- Initial inclination: $i_0 = 45^\circ$.
- Timespan: 2019/01/01, 12 : 00 : 00 TDB - 2021/01/01, 12 : 00 : 00 TDB.

The values selected for the eccentricity and inclination are chosen with the purpose of obtaining an orbit confined around Pluto and highly non linear at the same time. Figures 8.7 and 8.8 show the evolution of these parameters for Scenarios 1 and 2.

The positions, radii and masses of the celestial bodies involved in the study are those offered by the ephemerides DE432 and PLU055 through the information system named SPICE¹.

8.4.1.1 Scenario 1

The calculations have been carried out in a MacBook Pro with an Intel(R) Core(TM) i7 CPU @ 3.00 GHz and a macOS operative system. All the trajectories are integrated with the ode113 integrator of Matlab and varying integration tolerances, from 10^{-10} to 10^{-13} , using double precision arithmetics.

In order to select the reference solution for the comparison, the orbit of the artificial satellite is propagated with the most stringent tolerance provided by Matlab

¹ <https://naif.jpl.nasa.gov/naif/>

Figure 8.7: Eccentricity (upper left), inclination (upper right), semi-major axis (lower left) and right ascension of the ascending node (lower right) of the reference solution in Scenario 1.

Figure 8.8: Eccentricity (upper left), inclination (upper right), semi-major axis (lower left) and right ascension of the ascending node (lower right) of the reference solution in Scenario 2.

$(2.22 \cdot 10^{-14})$ for each of the propagators considered in the study. After that, the respective errors are compared by pairs to determine the time limit where the difference always remains below one meter. The result of this comparison provides as time threshold a period of one year. Within this timespan any of the solutions can be taken as reference solution admitting a precision of one meter, as long as the measured errors are much greater than one meter. The solution calculated with the KS method and a tolerance of $2.22 \cdot 10^{-14}$ is selected as reference solution. The solutions in double precision are determined at the very same instants of time as the solution in quadruple precision by applying a spline interpolation, so that solutions can be compared pointwise throughout the integration timespan.

Figure 8.9 shows the orbits described by the satellites of Pluto as well as the trajectory of the artificial satellite given by the reference solution. All the orbits are practically contained in a plane except for the orbit of the artificial satellite (blue color) which is oscillating. In spite of being strongly perturbed, the trajectory keeps confined between Pluto and Charon.

8.4.1.2 Scenario 2

The calculations have been carried out in a computer with an Intel(R) Core(TM) i5-7500 CPU @ 3.40 GHz microprocessor under the Debian 8.3.0-2 operative system 64 bits with the gfortran compiler. The trajectory is integrated with a 5th order Runge-Kutta-Fehlberg algorithm (RKF5(4)) and varying integration tolerances from 10^{-9} to 10^{-18} using double precision arithmetics.

The reference solution is calculated by using Cowell's method in quadruple precision and a tolerance equal to 10^{-16} , where this value is chosen in the calibration stage as a compromise between the accuracy of the solution and the computational time required. By using dense output propagator the propagator in quadruple precision evaluates the solution provided by the double precision propagator at the very same instants of time, so that a point-by-point direct comparison between both solutions can be made throughout the integration timespan. Indeed, for each solution in double precision there will be another in quadruple precision.

Figure 8.10 shows the trajectories of the satellites of Pluto as well as the trajectory of the artificial satellite given by the reference solution. The orbits of the natural satellites are practically contained in a plane except for the orbit of the artificial satellite (blue color) which is perpendicular to them. The orbit, whose initial conditions give place to an initial position very close to Pluto, ends up moving away from the planet with significant oscillations. The resulting trajectory is not confined to the neighborhood of Pluto, but goes beyond the orbit of the remaining satellites.

8.4.2 Comparison of results

The trajectory of the satellite is estimated with the DROMO-SPE method and compared with different propagators:

- The DROMO propagator.
- Cowell's method.
- The Kustaanheimo-Stiefel (KS) propagator.
- The Sperling-Bürdet (SB) propagator.
- The Palacios propagator.

Figure 8.9: Trajectory of the satellites orbiting Pluto for Scenario 1.

Figure 8.10: Trajectory of the satellites orbiting Pluto for Scenario 2.

8.4.2.1 Scenario 1

Figure 8.11 describes the performance of the different special perturbation methods using a reference solution to estimate the accuracy of the solution. As it is shown, the best results are provided by the propagators based on the Hansen ideal frame concept, that is, Palacios, DROMO and DROMO-SPE. In particular, DROMO and DROMO-SPE seem to be slightly better than Palacios when the accuracy increases, that is, when decreasing the value of the integration tolerance. In this case the DROMO-SPE propagator stand out from the rest. Cowell is the method that requires the higher number of function calls to reach the same average error, while KS and SB methods evolve in a similar way.

Figure 8.11: Performance curves relative to the reference solution in Scenario 1.

8.4.2.2 Scenario 2

Figure 8.12 describes the performance of the propagators previously introduced. It is observed that the best result, in terms of minimum average error, is given by KS followed by SB. The first propagator relies in coordinates as dependent integration variables, while the second includes a combination of coordinates and orbital elements. This fact is not trivial and its effect in the results will be explained in detail in Section 8.5. Palacios, DROMO and DROMO-SPE follow similar curves, specially when the number of function calls is less than one million. However, due to the fact that these formulations are based only on orbital elements, they provide a degraded accuracy solution when compared to KS and SB.

8.5 APPLICABILITY DOMAIN OF THE SPE FORMULATION

The new formulation has been assessed in two different environments, namely in Tsien's problem ($\epsilon = 1$) and in an N-body problem. While the performance significantly improves in the first problem, the results of the second problem reveal a lack of efficiency compared to other methods, specially in the second Scenario. In what follows we provide an explanation to this situation.

Figure 8.13 shows the magnitude of the perturbing acceleration broken down in the different sources that contribute to the total perturbing acceleration involved in

Figure 8.12: Performance curves relative to the reference solution in Scenario 2.

Scenario 2 as a function of time for the first 100 days of integration. In particular, we are only going to consider Charon as third-body. This can be accepted because the dynamical role of the remaining satellites in terms of perturbation is negligible compared with Charon and they only add stiffness to the equations, but without noticeably affecting the underlying dynamical structure of the problem. This way, the accelerations shown in Fig. 8.13 are decomposed in the following constituents:

- The direct component of Charon's third-body perturbation:

$$G M_{\text{Charon}} \frac{\vec{x}_{\text{Charon}} - \vec{x}}{|\vec{x}_{\text{Charon}} - \vec{x}|^3}. \quad (8.43)$$

- The indirect component of Charon's indirect perturbation:

$$- G M_{\text{Charon}} \frac{\vec{x}_{\text{Charon}}}{|\vec{x}_{\text{Charon}}|^3}. \quad (8.44)$$

- The total component (direct+indirect) of Charon's perturbation:

$$G M_{\text{Charon}} \left(\frac{\vec{x}_{\text{Charon}} - \vec{x}}{|\vec{x}_{\text{Charon}} - \vec{x}|^3} - \frac{\vec{x}_{\text{Charon}}}{|\vec{x}_{\text{Charon}}|^3} \right). \quad (8.45)$$

- The Keplerian term of Pluto's perturbation:

$$- G M_{\text{Pluto}} \frac{\vec{x}}{|\vec{x}|^3}. \quad (8.46)$$

The mass of Charon is approximately 1/8 of the mass of Pluto and consequently the gravitational attraction of Charon is always smaller than the gravitational attraction of Pluto, which decreases as the inverse of the square of the distance. However, as it is shown in Fig. 8.13, for some timespans the third-body perturbation (red color) becomes greater than the Keplerian term (dark blue color). What is happening is that when calculating the third-body perturbation due to Charon, the indirect term (or drag term) is often greater than the direct term (green color) and in some

cases larger than the Keplerian term itself, thus yielding a strongly perturbed dynamical environment where the perturbing term can become orders of magnitude larger than the Keplerian term. This is a direct consequence of trying to model as a perturbed two-body problem what is clearly a three-body problem.

Indeed, since Charon's orbit is almost circular, the distance to Pluto is almost constant and the indirect term of the third-body perturbation remains practically constant. When the artificial satellite is far from Pluto (see Fig. 8.10), the indirect term dominates over the other terms, while the gravitational terms are larger when the satellite is close to Pluto. As it is expected, the Keplerian term is always larger than the direct term of Charon's third-body perturbation.

The formulations based on orbital elements (Palacios, DROMO and DROMO-SPE) are consequently not appropriated to describe this particular kind of strongly perturbed environments where the trajectory is far from Keplerian. The orbital elements can not be properly described because they are subject to very fast variations (see Fig. 8.8) so their key advantage of being slowly-varying quantities is no longer fulfilled, and thus they exhibit no advantage (performance-wise) compared to formulations using Cartesian coordinates as state vector. As opposed to this particular scenario, the new formulation does prove very efficient when, despite the existence of a strong perturbing acceleration, the trajectory is smoothly varying such that orbital elements still represent an appropriate geometrical representation for the trajectory stemming from the strongly perturbed dynamical environment. This is the case of Tsien's problem which tends asymptotically to a circular orbit when $\epsilon = 1$, and the Scenario 1 of the second problem which keeps a similar shape along the integration.

Figure 8.13: Magnitude of perturbing accelerations (upper) and accelerations (lower) involved in Scenario 2.

TRAJECTORY OF ARTIFICIAL METEORS

The Japanese company Astro Live Experience (ALE Co., Ltd.) is a Tokyo-based space company founded by Lena Okajima. The aim of ALE is to realize the world's first artificial meteor shower with its own satellites, which has been described by ALE as the *Shooting Star Challenge*. On January 18, 2019 the first satellite was launched to space on Epsilon Rocket 4 from JAXA's Uchinoura Space Observatory in Kagoshima (Japan) as part of JAXA's *Innovate Satellite Technology Demonstration Program*. The Epsilon Rocket for this program includes seven satellites, one being ALE's first satellite whose main duties are:

- Investigate the feasibility of man-made meteors and its marketability.
- Obtain data about the Earth's upper atmosphere such as density, wind direction or composition by observing the artificial meteors.
- Understand how the reentry can change and affect the meteors' trajectory.

The second ALE's satellite, which is in the final stage of assembly, is expected to be launched in fall 2019.

In order to minimize potential effects on other Earth orbiting satellites, the satellite has a triple redundant attitude determination system such that the meteor release will occur only if three independent CPUs agree on the satellite attitude. After reaching space, the satellite, which is detached from the Epsilon Rocket at an altitude of 500 km, reaches its orbit using the De-Orbit Mechanism (DOM). The release of the meteor sources occurs when the satellite is below an altitude of 400 km. To achieve this the satellite activates a DOM and opens up a membrane in outer space making the membrane increases atmospheric drag on the satellite and help the satellite slowly descend down to the altitude of 400 km. The satellite is in a sun-synchronous orbit, which allows the satellite to pass over a specific location at the same time every day. The microsatellite (Fig. 9.1) is about 60 cm in size and capable of storing and delivering between 300 and 400 pellets. Each pellet will burn for about five to ten seconds and will be visible for up to 200 km. These 1 cm particles are composed of non-toxic materials such as lithium, potassium and copper such that varying their composition results in sparkles of different colors. The particles experience a significant aeroheating during reentry and convert their kinetic energy into brightness. The material interaction with the air generates a wide range of bright colors and safely disintegrates before reaching 60 km.

ALE's meteor project is being developed by several institutions. The main satellite systems were developed by Dr. Toshinori Kuwahara from Tohoku University and the release mechanism by Dr. Takeo Watanabe. Dr. Hironori Sahara, from Tokyo Metropolitan University, developed the meteor source together with Dr. Shinsuke Abe from Nihon University, who are responsible for the satellite design and manufacturing as well as the orbital simulation.

As part of my doctoral research, I have had the opportunity to work under the supervision of Dr. Hironori Sahara in the Space Systems Laboratory at Tokyo Metropolitan University. The topic is related to orbital simulation inside ALE's project (Kimura [48]). In particular, I have been focused on the estimation of the

Figure 9.1: Microsatellite launched by ALE.

meteors's trajectory and the sensitivity analysis of the parameters involved in the simulation. The aim is to increase the understanding of the relations between input and output variables in the system and their consequences from a physical viewpoint in order to establish appropriate initial conditions for the release. Besides, it allows the opportunity to test the DROMO's formulations and extend its application for meteors. This work has been developed from April 1, 2019 until July 1, 2019.

9.1 INTRODUCTION

Most asteroids in the Solar System are located in the main asteroid belt between Mars and Jupiter. However, some asteroids orbit the Sun passing close to the Earth. Sometimes asteroids can smash into another causing small pieces of the asteroid to break off. These small rocky bodies located in space are named meteoroids. If the meteoroid passes close enough to the Earth so as to enter its atmosphere, it can disintegrate and produce a light phenomenon. In this case we refer to them as meteors, more commonly known as shooting stars. However, if the rocks do not disintegrate completely in the atmosphere and they reach the surface of the Earth, they are referred to as meteorites.

Meteors are formed at altitudes between 130 km and 70 km. Typical meteors are associated with meteoroids with diameters between 0.01 mm and 20 cm, which correspond to a mass ranging from 10^{-9} to 10^4 g. The final masses for the meteorite, after the passage through the atmosphere, ranges from 10^{-2} to 10^8 g. The ability of entering in the atmosphere strongly depends on the velocity of the meteoroid. The mass loss due to ablation causes an upper limit on the velocity of about 30 km/s for the occurrence of a meteorite fall (Abe [1]).

The reentry of meteoroids (see Fig. 9.2) is divided in three regimes:

- Preheating: at altitudes ranging from 400 km to 120 km, when the meteoroid approaches the Earth, the surface temperature of the meteoroid increases but due to low atmospheric density the ablation is not significant above 150 km.

Figure 9.2: Meteoroid phases (Abe [1]).

- Ablation: thermal ablation begins at altitudes between 130 km and 120 km, being this altitude independent of the meteoroid mass. It happens because of the momentum transfer in collisions between atmospheric gases and the meteoroid. The disintegration starts when the temperature of the meteoroid increases to about 2000 K, which depends on the melting point of its constituent material. In most of the cases the kinetic energy of the meteoroids is consumed during this regime and the deceleration increases when entering deeper into the atmosphere. Typical meteor ablation occurs at altitudes of approximately 130 km to 70 km. More details about the ablation process can be found in Davuluri, Zhang, and Martin [25], Park [61], and Vinnikov et al. [92].
- Dark-flight: the dark flight starts when the meteoroid decelerates below the velocity at which the ablation takes place. This regime starts without emitting light. The terminal velocity observed for many fireballs is typically between 3 km/s and 4 km/s. If there is no wind, the remnant of the meteoroid follows a free fall. The impact velocities on the Earth's surface are in the order of several tens of meters per second for smaller meteorites (~ 10 g), and several hundreds of meters per second for larger meteorites (~ 10 kg).

Natural meteors can be observed with visual, radar or photographic methods. However, they can not be observed continuously because its availability is limited in space and time. Artificial meteors can avoid this problem because it can be decided when and where they are observed without interruption. Moreover, continuous observations and the knowledge of the precise meteor's trajectory allows to study the Earth's atmosphere by analyzing the evolution of the meteor along its trajectory. In particular, it allows to study the mesosphere, i.e., the atmosphere's layer where the meteorite is disintegrated. Several methods have been developed to describe and determine the meteor trajectory, some of them based on observations as photographs (Borovicka [16]) or multiple camera views (Gural [39]), while others have been based on theoretical studies to provide accurate analytic models (Lyne, M., and Fought [52]). In what follows the equations and the parameters involved in the orbital simulation are presented and described.

9.2 PHYSICAL MODELING

Let M be an artificial meteor with initial mass m_0 which is released from a satellite and injected behind it into an elliptic orbit at its apogee. The meteor moves with respect to the inertial frame $Ox_1y_1z_1$ where O is the center of mass of the Earth. We assume that the meteor is acted upon by: 1) the main term of the gravitational attraction of the Earth; and 2) the drag force.

The governing equations are given by the system:

$$m \frac{d^2 \vec{x}}{dt^2} = -\frac{m \mu}{|\vec{x}|^3} \vec{x} - \frac{1}{2} C_d A \rho \vec{v}^2 \frac{\vec{v}}{|\vec{v}|} \quad (9.1a)$$

$$L^* \frac{dm}{dt} = -\frac{1}{2} C_h A \rho |\vec{v}|^3. \quad (9.1b)$$

Equation (9.1a) describes the linear momentum equation where $\vec{x}$ is the position vector, $\vec{v}$ the velocity vector, m the mass of the meteor source, C_d the drag coefficient, A the cross-section area of the meteor and ρ the atmospheric density. Equation (9.1b) corresponds to the mass loss equation due to ablation such that $L^* = 10^6$ J/kg is the ablation specific heat and C_h the heat transfer coefficient.

Note that the linear momentum equation has been simplified by assuming a uniform mass transfer around the surface of the meteor (Davies and Park [24]). Indeed, the rate of change of momentum turns out to be

$$\frac{d(m \vec{v})}{dt} = m \frac{d\vec{v}}{dt} - \vec{v}_l \frac{dm}{dt}, \quad (9.2)$$

where $\vec{v}_l$ is the velocity of the material leaving the meteor with respect to the meteor. Considering an uniform mass transfer this velocity becomes equal to zero.

9.2.1 Cross-section area

The aerodynamic area changes as the mass decreases. Their relation is expressed by the shape change coefficient ν (Gritsevich and Koschny [38]) as

$$\frac{A}{A_0} = \left(\frac{m}{m_0} \right)^\nu, \quad (9.3)$$

where A_0 and m_0 are the initial values before ablation of the cross-section area and mass respectively.

The shape change coefficient is a parameter that characterizes the rotation of the meteor along the flight. If $\nu = 0$, it means that the meteor is stable without rotation, while $\nu = 0.66$ refers to a meteor remaining spherical.

9.2.2 Atmospheric parameters

A body moving close to the Earth is affected by the atmospheric drag. Satellites in Low Earth Orbit (LEO) encounter atmospheric drag from gases in the thermosphere (~ 80 km - 500 km), while objects like meteors are especially affected during reentry (~ 120 km). This perturbing acceleration depends on the atmospheric density ρ_{air} and atmospheric temperature T_{air} which need to be modeled with high accuracy in order to properly describe the constant change of the environment. For this study the NRLMSISE-00 empirical model of the atmosphere (Picone, Hedin, and Drob [63]) has been selected. It allows to use a single model by covering the atmosphere from the surface to lower exosphere (~ 0 km - 1000 km). The meteor, which is released in the thermosphere, can benefit from this advantage. The NRLMSISE-00 database includes:

- Total mass density from satellite accelerometers and from orbit determination (including the Jacchia and Barlier data sets).
- Temperature from incoherent scatter radar covering 1981-1997.
- Molecular oxygen number density from solar ultraviolet occultation aboard the Solar Maximum Mission. The anomalous oxygen, which contributes to the drag above 500 km, is also considered.

The inputs of this model include parameters describing the solar and geomagnetic activity:

- Daily and 81-day averaged values of the $F_{10.7}$ index, i.e., the solar radio flux of the 10.7 cm wavelength (2.8 GHz).
- A_p index, i.e., the geomagnetic activity index.

The greater the solar activity and geomagnetic activity are, the larger the atmospheric density and temperature of the air are. $F_{10.7} = 150$ and $A_p = 4$ correspond to a medium activity.

9.2.3 Surface temperature

The temperature of the surface of the meteor is a parameter that is taken into account in the atmospheric resistance, in particular in the drag coefficient. The meteors are acted upon not only by the aerodynamic heating, but by complex processes as melting and spalling. Consequently, the surface temperature becomes difficult to estimate. One option is to consider it equal to the atmospheric air at the corresponding altitude. Following the work by Kimura [48], four distinct expressions are considered for the sensitivity analysis:

$$T_s = \left(\frac{q}{\sigma \epsilon} \right)^{0.25} \quad \epsilon = 0.25 \quad (9.4a)$$

$$T_s = \left(\frac{q}{\sigma \epsilon} \right)^{0.25} \quad \epsilon = 0.75 \quad (9.4b)$$

$$T_s = \left(T_{\text{air}}^4 + \frac{q}{\sigma \epsilon} \right)^{0.25} \quad \epsilon = 0.25 \quad (9.4c)$$

$$T_s = \left(T_{\text{air}}^4 + \frac{q}{\sigma \epsilon} \right)^{0.25} \quad \epsilon = 0.75, \quad (9.4d)$$

where $\sigma = 5.670373 \cdot 10^{-8} \text{ W/m}^2/\text{K}^4$ is the Stefan-Boltzmann constant, ϵ is the emissivity value and q is the heating rate given by the Detra, Kemp, and Ridell [29] formula:

$$q = 110.35 \left(\frac{|\vec{v}|}{|\vec{v}_{\text{ref}}|} \right)^{3.15} \left(\frac{|\vec{x}|}{|\vec{x}_{\text{ref}}|} \right)^{-0.5} \left(\frac{\rho}{\rho_{\text{ref}}} \right)^{0.5}, \quad (9.5)$$

where $|\vec{v}_{\text{ref}}| = 7925 \text{ m/s}$, $|\vec{x}_{\text{ref}}| = 1 \text{ m}$ and $\rho_{\text{ref}} = 1.225 \text{ kg/m}^3$ are the reference values for the velocity, position and density respectively.

9.2.4 Drag coefficient

The drag coefficient is the main parameter in the atmospheric drag, which in the case of reentry objects is not constant, but significantly fluctuates along its trajectory. The expression considered for this parameter is (Henderson [41]):

$$C_d = \frac{0.9 + \frac{0.34}{\text{Ma}^2} + 1.86 \left(\frac{\text{Ma}}{\text{Re}} \right)^{\frac{1}{2}} \left[2 + \frac{2}{(\text{Ma}\sqrt{\zeta/2})^2} + \frac{1.058}{\text{Ma}\sqrt{\zeta/2}} \left(\frac{T_s}{T_{\text{air}}} \right)^{\frac{1}{2}} - \frac{1}{(\text{Ma}\sqrt{\zeta/2})^4} \right]}{1 + 1.86 \left(\frac{\text{Ma}}{\text{Re}} \right)^{\frac{1}{2}}}, \quad (9.6)$$

Figure 9.3: Trajectory of the meteor.

such that:

$$Ma = \frac{|\vec{v}|}{\sqrt{\frac{\zeta R_{gas} T_{air}}{W_{air}}}} \quad (9.7)$$

is the Mach number with $R_{gas} = 8.3144598 \text{ J/mol/K}$ the molar gas constant, $W_{air} = 0.0289644 \text{ kg/mol}$ the molar mass of dry air and $\zeta = 1.4$ the heat capacity ratio of air.

$$Re = \frac{2 \rho_{air} |\vec{v}| R_m}{\mu_{air}} \quad (9.8)$$

is the Reynolds number with R_m the radius of the meteor and μ_{air} the atmospheric viscosity (Sutherland and Bass [88]) given by

$$\mu_{air} = \mu_{ref} \left(\frac{T_0 + T_{ref}}{T_{air} + T_{ref}} \right) \left(\frac{T_{air}}{T_0} \right)^{\frac{3}{2}}, \quad (9.9)$$

with $T_0 = 273.11 \text{ K}$ and $\mu_{ref} = 6.7894 \cdot 10^{-5} \text{ kg/m/s}$ and $T_{ref} = 110.56 \text{ K}$ reference values for viscosity and temperature respectively.

9.2.5 Heat transfer coefficient

The heating coefficient is the main parameter in the mass loss equation and it is related with the mass reduction of the meteor. The formula used in this study is taken from the work developed by Prabhu and Saunders [65] and Prevèreaud [66]:

$$C_h = \frac{2 q}{\rho_{air} |\vec{v}|^3} \frac{\int_0^{\frac{\pi}{2}} \left[\sin^2(t) + \frac{\cos^2(t)}{1 + \zeta Ma^2} \right] \left(\frac{\pi}{2} - t \right) \cos(t) \sin(t) dt}{\sqrt{\int_0^{\frac{\pi}{2}} \left[\sin^2(t) + \frac{\cos^2(t)}{1 + \zeta Ma^2} \right] \left(\frac{\pi}{2} - t \right) \cos^2(t) dt}}. \quad (9.10)$$

9.3 PROPAGATION OF THE ORBIT

The orbit of the meteor is propagated by considering the previous models in the formulation (Fig. 9.3). For that, a priori initial conditions for the satellite and the

meteor need to be introduced. The present study is an illustrative example of meteor's trajectory. Consequently, we choose representative values for the problem, for the sake of simplicity. Instead of considering a sun-synchronous orbit, we consider a satellite with a polar circular orbit and altitude $H_{\text{sat}} = 375$ km. The meteor is shot just behind the satellite with relative velocity $v_{\text{rel}} = 350$ m/s. The release point of the meteor is defined by the following parameters:

- Inclination: $i = 90^\circ$.
- Eccentricity: $e = 0$.
- Semi-major axis $a = R_{\text{eq}} + H_{\text{sat}}$.
- True longitude $l = \Omega + \omega + \theta = 257^\circ$.

where R_{eq} is the terrestrial equatorial radius according to the reference ellipsoid WGS84, $\Omega + \omega = 257^\circ$ and $\theta = 0^\circ$. The start date is 2020/01/01, 00 : 00 : 00 TDB such that the orbit is propagated until the mass reaches a value equal or smaller than 10^{-7} kg or the meteor reaches the surface of the Earth.

9.3.1 Initial position

The initial position of the meteor in the perifocal frame is given by

$$x_p = \frac{p}{1 + e \cos(\theta)} \cos(\theta) \quad (9.11a)$$

$$y_p = \frac{p}{1 + e \cos(\theta)} \sin(\theta) \quad (9.11b)$$

$$z_p = 0, \quad (9.11c)$$

where $p = a(1 - e^2)$ is the *semilatus rectum*. By using the matrix P_0 defined in Chapter 6, the initial coordinates of the meteor can be obtained:

$$\begin{pmatrix} x_0 \\ y_0 \\ z_0 \end{pmatrix} = P_0 \begin{pmatrix} x_p \\ y_p \\ z_p \end{pmatrix}. \quad (9.12)$$

9.3.2 Initial velocity

The velocity of the satellite in the perifocal frame is defined by

$$v_{x_{\text{sat}}} = -\frac{\mu}{h} \sin(\theta) \quad (9.13a)$$

$$v_{y_{\text{sat}}} = \frac{\mu}{h} (e + \cos(\theta)) \quad (9.13b)$$

$$v_{z_{\text{sat}}} = 0, \quad (9.13c)$$

where μ is the Earth gravitational parameter and $h = \sqrt{\mu p}$ the specific angular momentum. Therefore, the velocity and period of the satellite are:

$$v_{\text{sat}} = \sqrt{\frac{\mu}{R_{\text{eq}} + H_{\text{sat}}}} = 7.683 \text{ km/s} \quad (9.14)$$

$$T_{\text{sat}} = 2\pi \frac{R_{\text{eq}} + H_{\text{sat}}}{v_{\text{sat}}} = 92.049 \text{ min.} \quad (9.15)$$

If the meteor is released behind the satellite with a relative velocity $v_{\text{rel}} = 350$ m/s, that is, in the same direction of motion but with opposite sense, then the velocity of the meteor in the perifocal frame is

$$v_{x_p} = v_{x_{\text{sat}}} - v_{\text{rel}} \cdot \frac{v_{x_{\text{sat}}}}{|\vec{v}_{\text{sat}}|} \quad (9.16a)$$

$$v_{y_p} = v_{y_{\text{sat}}} - v_{\text{rel}} \cdot \frac{v_{y_{\text{sat}}}}{|\vec{v}_{\text{sat}}|} \quad (9.16b)$$

$$v_{z_p} = 0, \quad (9.16c)$$

with $\vec{v}_{\text{sat}} = (v_{x_{\text{sat}}}, v_{y_{\text{sat}}}, v_{z_{\text{sat}}})$. By using the matrix P_0 defined in Chapter 6, the velocity of the meteor at the initial instant in the inertial frame results:

$$\begin{pmatrix} v_{x_0} \\ v_{y_0} \\ v_{z_0} \end{pmatrix} = P_0 \begin{pmatrix} v_{x_p} \\ v_{y_p} \\ v_{z_p} \end{pmatrix}. \quad (9.17)$$

9.3.3 Initial mass

The initial mass of the meteor depends on two parameters, the initial radius R_{m0} and the initial density ρ_{m0} through the equation

$$m_0 = \frac{4}{3} \pi R_{m0}^3 \rho_{m0}. \quad (9.18)$$

A spherical shape is assumed with a uniform density distribution.

9.3.4 Mathematical formulations

The orbit propagation has been carried out with DROMO and DROMO-SPE propagators as well as with Cowell's method. In what follows, the system of equations for each of these formulations is described for the meteor problem.

9.3.4.1 DROMO's governing equations

$$\frac{d\tau}{d\sigma} = \frac{1}{\zeta_3^3 s^2} \quad (9.19a)$$

$$\frac{d\zeta_1}{d\sigma} = +s \sin(\sigma) \tilde{a}_{px} + [\zeta_1 + (1+s) \cos(\sigma)] \tilde{a}_{py} \quad (9.19b)$$

$$\frac{d\zeta_2}{d\sigma} = -s \cos(\sigma) \tilde{a}_{px} + [\zeta_2 + (1+s) \sin(\sigma)] \tilde{a}_{py} \quad (9.19c)$$

$$\frac{d\zeta_3}{d\sigma} = -\zeta_3 \tilde{a}_{py} \quad (9.19d)$$

$$\frac{d\eta_1}{d\sigma} = \frac{1}{2} \tilde{\omega}_{01} \{ \eta_4 \cos(\sigma) - \eta_3 \sin(\sigma) \} \quad (9.19e)$$

$$\frac{d\eta_2}{d\sigma} = \frac{1}{2} \tilde{\omega}_{01} \{ \eta_3 \cos(\sigma) + \eta_4 \sin(\sigma) \} \quad (9.19f)$$

$$\frac{d\eta_3}{d\sigma} = \frac{1}{2} \tilde{\omega}_{01} \{ -\eta_2 \cos(\sigma) + \eta_1 \sin(\sigma) \} \quad (9.19g)$$

$$\frac{d\eta_4}{d\sigma} = \frac{1}{2} \tilde{\omega}_{01} \{ -\eta_1 \cos(\sigma) - \eta_2 \sin(\sigma) \} \quad (9.19h)$$

$$\frac{dm}{d\sigma} = -\frac{1}{2L^*} \frac{1}{\omega_c \zeta_3^3 s^2} C_h A \rho |\vec{v}|^3. \quad (9.19i)$$

The mass loss equation is expressed in kg and with respect to the independent variable σ by the chain rule:

$$\frac{dm}{d\sigma} = \frac{dm}{d\tau} \frac{d\tau}{d\sigma}, \quad (9.20)$$

with

$$\frac{dm}{d\tau} = \frac{dm}{dt} \frac{1}{\omega_c}. \quad (9.21)$$

The perturbing acceleration corresponds to the atmospheric drag introduced in Eq. (9.1a) but in dimensionless form:

$$\vec{a}_p = -\frac{1}{2} \frac{1}{m} \frac{1}{\omega_c^2 L_c} C_d A \rho \vec{v}^2 \frac{\vec{v}}{|\vec{v}|}, \quad (9.22)$$

such that

$$\tilde{\vec{a}}_p = (\tilde{a}_{px}, \tilde{a}_{py}, \tilde{a}_{pz}) = \vec{a}_p \frac{1}{\zeta_3^4 s^3}. \quad (9.23)$$

9.3.4.2 DROMO-SPE's governing equations

$$\frac{d\tau}{d\sigma} = \frac{1}{\xi_3^3 (1 + \xi_1)^2} \quad (9.24a)$$

$$\frac{d\xi_1}{d\sigma} = \xi_2 + 2(1 + \xi_1) \tilde{a}_{py} \quad (9.24b)$$

$$\frac{d\xi_2}{d\sigma} = -\xi_1 - (1 + \xi_1) \tilde{a}_{px} + \xi_2 \tilde{a}_{py} \quad (9.24c)$$

$$\frac{d\xi_3}{d\sigma} = -\xi_3 \tilde{a}_{py} \quad (9.24d)$$

$$\frac{d\eta_1}{d\sigma} = \frac{1}{2} \tilde{\omega}_{01} \{ \eta_4 \cos(\sigma) - \eta_3 \sin(\sigma) \} \quad (9.24e)$$

$$\frac{d\eta_2}{d\sigma} = \frac{1}{2} \tilde{\omega}_{01} \{ \eta_3 \cos(\sigma) + \eta_4 \sin(\sigma) \} \quad (9.24f)$$

$$\frac{d\eta_3}{d\sigma} = \frac{1}{2} \tilde{\omega}_{01} \{ -\eta_2 \cos(\sigma) + \eta_1 \sin(\sigma) \} \quad (9.24g)$$

$$\frac{d\eta_4}{d\sigma} = \frac{1}{2} \tilde{\omega}_{01} \{ -\eta_1 \cos(\sigma) - \eta_2 \sin(\sigma) \} \quad (9.24h)$$

$$\frac{dm}{d\sigma} = -\frac{1}{2L^*} \frac{1}{\omega_c \xi_3^3 (1 + \xi_1)^2} C_h A \rho |\vec{v}|^3, \quad (9.24i)$$

where the relations are analogous to DROMO's formulations but with respect to the DROMO-SPE variables. In this case the perturbing acceleration is given by:

$$\tilde{\vec{a}}_p = (\tilde{a}_{px}, \tilde{a}_{py}, \tilde{a}_{pz}) = \vec{a}_p \frac{1}{\xi_3^4 (1 + \xi_1)^3}. \quad (9.25)$$

9.3.4.3 Cowell's governing equations

$$\frac{dx}{dt} = v_x \quad (9.26a)$$

$$\frac{dy}{dt} = v_y \quad (9.26b)$$

$$\frac{dz}{dt} = v_z \quad (9.26c)$$

$$\frac{dv_x}{dt} = -\frac{\mu}{|\vec{x}|^3}x + a_{px} \quad (9.26d)$$

$$\frac{dv_y}{dt} = -\frac{\mu}{|\vec{x}|^3}y + a_{py} \quad (9.26e)$$

$$\frac{dv_z}{dt} = -\frac{\mu}{|\vec{x}|^3}z + a_{pz} \quad (9.26f)$$

$$\frac{dm}{dt} = -\frac{1}{2L^*} \frac{1}{\omega_c} C_h A \rho |\vec{v}|^3, \quad (9.26g)$$

where $\vec{a}_p$ is the perturbing acceleration due to the atmospheric drag defined in DROMO'S formulation and $(\vec{x}, \vec{v})$ the state vector in the inertial frame. The variables here are defined in dimensionless form following the same procedure as in DROMO and DROMO-SPE.

9.4 SCENARIOS

The calculations have been carried out in a computer with an Intel(R) Core(TM) i5-3470 CPU @ 3.20 GHz microprocessor under the Windows 10 operative system 64 bits. The solutions provided by DROMO, DROMO-SPE and Cowell are integrated with the ode113 integrator of Matlab by using a tolerance equal to 10^{-12} . This value is fixed for both relative and absolute tolerance because all the equations in DROMO, DROMO-SPE and Cowell are dimensionless, and therefore it makes sense to fix the same value to both tolerances. The reference solution selected for the comparison is the one provided by Cowell's method by using the most stringent tolerance of Matlab ($2.22 \cdot 10^{-14}$). Besides, in order to assess the influence of several parameters in the propagation, 18 different scenarios are analyzed with respect to the reference Scenario X (see Table 9). The Scenario X contains the default values for the parameters involved in the analysis as explained in the preceding section.

9.4.1 Sensitivity analysis

The different scenarios are evaluated by using the reference solution. Table 10 shows the results obtained at the end of the meteor's flight taking into account that the initial altitude of the meteor for all the scenarios is 375 km:

- Scenario 1: the meteor is disintegrated at altitudes of 74.150 km and 70.677 km for 1a and 1b respectively. The increase of the drag coefficient (Scenario 1b) reduces the altitude of disintegration such that when the meteor reaches and altitude below 120 km, the mass starts to decrease very fast (see Fig. 9.4). From this altitude the eccentricity increases and tends to a parabolic orbit. The exception is the Scenario 1a, when C_d is small, whose eccentricity remains below 0.2 until the end of the flight. The timespan is 15.292 min and 15.181 min for 1a and 1b respectively.
- Scenario 2: the flight finishes due to mass depletion with an altitude on the order of 71 km. Scenarios 2a and 2c, as well as 2b and 2d, provide the same re-

Table 9: Scenarios considered in the study.

Scenario	C_d	T_s [K]	C_h	ν	$F_{10.7}$	A_p	v_{rel} [m/s]	R_{m0} [m]	ρ_{m0} [kg/m ³]
X	(9.6)	T_{air}	(2.12)	0.66	150	4	350	0.005	5000
1a	0.1	T_{air}	(9.10)	0.66	150	4	350	0.005	5000
1b	2	T_{air}	(9.10)	0.66	150	4	350	0.005	5000
2a	(9.6)	(9.4a)	(9.10)	0.66	150	4	350	0.005	5000
2b	(9.6)	(9.4b)	(9.10)	0.66	150	4	350	0.005	5000
2c	(9.6)	(9.4c)	(9.10)	0.66	150	4	350	0.005	5000
2d	(9.6)	(9.4d)	(9.10)	0.66	150	4	350	0.005	5000
3a	(9.6)	T_{air}	0.1	0.66	150	4	350	0.005	5000
3b	(9.6)	T_{air}	1	0.66	150	4	350	0.005	5000
4a	(9.6)	T_{air}	(9.10)	0.1	150	4	350	0.005	5000
4b	(9.6)	T_{air}	(9.10)	0.8	150	4	350	0.005	5000
5a	(9.6)	T_{air}	(9.10)	0.66	240	40	350	0.005	5000
5b	(9.6)	T_{air}	(9.10)	0.66	60	1	350	0.005	5000
6a	(9.6)	T_{air}	(9.10)	0.66	150	4	250	0.005	5000
6b	(9.6)	T_{air}	(9.10)	0.66	150	4	450	0.005	5000
7a	(9.6)	T_{air}	(9.10)	0.66	150	4	350	0.001	5000
7b	(9.6)	T_{air}	(9.10)	0.66	150	4	350	0.01	5000
8a	(9.6)	T_{air}	(9.10)	0.66	150	4	350	0.005	1000
8b	(9.6)	T_{air}	(9.10)	0.66	150	4	350	0.005	10000

sults. It means, the temperature of the air and the emissivity parameter do not seem to have a noticeable influence in the surface temperature of the meteor to provide a remarkable change. As in Scenario 1, the duration of the flight is approximately 15 min. The mass of the meteor starts to decrease faster when entering in the atmosphere (~ 120 km), with an eccentricity of the orbit larger than 0.8 just before full disintegration (see Fig. 9.5).

- Scenario 3: when the heat transfer coefficient is significantly small (Scenario 3a), the effect of the ablation is reduced, allowing the meteorite to reach the surface in 22.911 minutes with a final mass of 0.255 g. When the value of C_h is larger, like in Scenario 3b, the meteor depletes all its mass when reaches an altitude of 73.155 km in 15.232 min. Figure 9.6 reflects this situation and the parabolic behavior of the orbit in Scenario 3a when the altitude is below 70 km. The final eccentricity and mass decrease with the increase of C_h . Figure 9.12 shows how the trajectory reaches lower latitudes when decreasing C_h .
- Scenario 4: the shape change coefficient determines the stability of the meteor. The mass and eccentricity evolution remains the same until the entrance in the atmosphere (see Fig. 9.7). The decreases of ν makes the orbit more eccentric at the end of the flight, while the disintegration altitude increases. When $\nu = 0.1$ the final mass depletion happens at 80.897 km altitude with a timespan of 15.009 min. In the case $\nu = 0.8$ it requires 15.418 min and occurs at 67.636 km altitude.

Table 10: State of the meteor at the end of the flight (MD = Mass Depletion; SR = Surface Reached).

Scenario	Initial mass [g]	Final mass [g]	Final altitude [km]	Timespan [min]	Event
X	2.618	10^{-4}	71.578	15.292	MD
1a	2.618	10^{-4}	74.150	15.181	MD
1b	2.618	10^{-4}	70.677	15.341	MD
2a	2.618	10^{-4}	71.046	15.321	MD
2b	2.618	10^{-4}	71.149	15.315	MD
2c	2.618	10^{-4}	71.046	15.321	MD
2d	2.618	10^{-4}	71.149	15.315	MD
3a	2.618	0.255	0	22.911	SR
3b	2.618	10^{-4}	73.155	15.232	MD
4a	2.618	10^{-4}	80.897	15.009	MD
4b	2.618	10^{-4}	67.636	15.418	MD
5a	2.618	10^{-4}	71.566	15.291	MD
5b	2.618	10^{-4}	71.468	15.297	MD
6a	2.618	10^{-4}	75.979	18.247	MD
6b	2.618	10^{-4}	27.734	16.969	MD
7a	0.021	10^{-4}	111.070	14.148	MD
7b	20.944	0.598	0	22.973	SR
8a	0.524	10^{-4}	94.993	14.606	MD
8b	5.236	0.012	0	22.339	SR

- Scenario 5: the effect of solar activity is mainly noticeable at the beginning of the propagation, before entering in the atmosphere (see Fig. massecc5). The eccentricity does not vary between scenarios although all of them become very eccentric in the final phase of the meteor's atmospheric dive. In Scenario 5a, or active periods of solar activity, the disintegration takes place at 71.566 km altitude and needs 15.291 min. On the contrary, for quiet periods of solar activity (Scenario 5b) it occurs at 71.468 km and requires 15.297 min.
- Scenario 6: the relative velocity of the meteor determines not only the altitude of disintegration, but the latitude where it occurs. When increasing the relative velocity with respect to the satellite, the final eccentricity of the orbit increases too (see Fig. 9.9). In Scenario 6b the orbit turns out to be parabolic at the end of the flight and the mass variation becomes slower in this phase. Besides, the complete mass depletion occurs at lower altitudes and higher latitudes (see Fig. 9.12). Thus, in Scenario 6a the meteor requires 18.247 min to disintegrate at 75.979 km altitude, while Scenario 6b needs 16.969 min and an altitude of 27.734 km. The mass variation, although is small between them, it starts from high altitudes being larger for Scenario 6b (Fig. 9.9).
- Scenario 7: the initial radius of the meteor defines its evolution. For larger values the initial mass is larger and the ablation process can not fully deplete all the mass along the trajectory. This is the case of Scenario 7b with an initial mass of 20.944 g which decreases to 0.598 g until the meteorite reaches the

surface in a timespan of 22.973 min. In Scenario 7a the meteor starts with 0.021 g and disintegrates at 111.070 km altitude in 14.148 min. As in Scenario 6b, the mass variation in Scenario 7b decreases more slowly than in the last phase of the trajectory when the eccentricity increases to one (see Fig. 9.10). Likewise, Fig. 9.13 reveals that Scenario 7b reaches lower latitudes compared with Scenario 7a.

- Scenario 8: the influence of the initial density of the meteor is similar to that of the radius (see Fig. 9.11). In Scenario 8a the meteor has an initial mass of 0.524 g and depletes all its mass when the altitude is 94.993 km and the time elapsed in 14.606 min. In Scenario 8b, with an initial mass of 5.236 g, the meteorite reaches the ground after 22.339 min with a final mass of 0.012 kg. Compared with X and Scenario 8a, the trajectory of the meteorite in Scenario 8b passes through lower latitudes (9.13).

Taking into account the initial requirements for the artificial meteor release, the most suitable scenarios would be X, 1a, 1b, 2a, 2b, 2c, 2d, 3b, 4a, 4b, 5a, 5b and 6a. In all of them the material is disintegrated before reaching 60 km in time periods ranging from 15 min to 19 min. The parameters providing the most significant changes in the outcome of the simulation are the heat transfer coefficient, the initial radius and the initial density of the meteor. All of them are involved in the ablation process. The heating coefficient describes the ablation process while the remaining parameters characterized the material, with a specific melting point, affected by ablation.

9.4.2 Comparison of formulations

Table 11 displays the final position error of the meteor for the different scenarios with respect to the reference solution. Three formulations have been considered in the analysis, that is, DROMO, DROMO-SPE and Cowell. This value is defined as follows:

$$\Delta x = |\vec{x}(t_f) - \vec{x}_{ref}(t_f)|, \quad (9.27)$$

where t_f is the instant of surface impact or complete mass depletion, $\vec{x}_{ref}(t_f)$ is the final position provided by the reference solution and $\vec{x}(t_f)$ is the final position provided by DROMO, DROMO-SPE or Cowell with a tolerance equal to 10^{-12} . The results provided in Tab. 11 do not include, however, the Scenarios 3a, 6b, 7b and 8b. In the latter scenarios the flight of the meteor finished because it reached the surface of the Earth or it was disintegrated very close to the surface (see Table 10). This fact makes the angular momentum of the particle tend to zero and the eccentricity tend to one (see Figs. 9.6, 9.9, 9.10 and 9.11). This situation corresponds to one of the singularities described in Chapter 6 when approaching a rectilinear motion. Indeed, taking into account that

$$h = \frac{1}{\zeta_3}, \quad \zeta_3 = \xi_3, \quad (9.28)$$

we can consider, without loss of generality, the DROMO system of equations (9.19) to analyze the singularity occurring both in DROMO and DROMO-SPE formulations. If the angular momentum tends to zero, the quantity ζ_3 , which is its inverse, tends to infinity. Besides, the auxiliary variable s defined by

$$s = 1 + \zeta_1 \cos(\sigma) + \zeta_2 \sin(\sigma) \quad (9.29)$$

also tends to zero. Note that the trajectory tends to a rectilinear motion passing through the origin which makes the eccentricity tends to one.

Table 11: Final position error with respect to the reference solution and number of function calls (D = DROMO; DS = DROMO-SPE; C = Cowell).

Scenario	Δx_D [mm]	FCalls _D	Δx_{DS} [mm]	FCalls _{DS}	Δx_C [mm]	FCalls _C
X	0.110	17244	1.257	17826	2.374	13502
1a	2.814	4618	1.914	5184	0.467	4102
1b	4.992	18922	0.056	19763	2.327	14924
2a	0.945	18236	4.874	18886	1.091	14496
2b	1.553	17740	3.033	19021	4.518	14016
2c	0.302	18129	3.602	19451	2.443	14134
2d	0.159	17886	3.596	18829	2.889	14161
3b	0.349	14353	0.221	15390	0.930	11136
4a	5.793	14947	0.696	15796	2.535	11746
4b	0.809	16952	0.593	17509	1.684	13523
5a	5.337	17482	0.735	18461	0.466	14244
5b	6.929	17025	4.412	17774	9.600	13847
6a	0.586	21359	2.312	21462	1.752	16826
7a	42.841	3780	26.195	4136	81.171	2936
8a	4.575	9935	0.803	10582	0.592	7925

Equation (9.19a), which describes the variation of τ , i.e. the derivative of τ with respect to σ , depends on both variables ζ_3 and s . Thus, to see which one freezes, it is necessary to evaluate the rate of convergence of $1/\zeta_3$ and s to zero. Figure 9.14 displays the decimal logarithm of $1/\zeta_3$ and s versus σ showing that s decreases faster. Consequently, Eq. (9.19a) tends to infinity and the time τ increases while σ freezes, tending to a stationary value (see Fig. 9.14). This fact is due to the trajectory tends to a rectilinear conic passing through the origin.

In this scenario the integration does not advance and the propagation never reaches the stop condition. As it is shown in Figs. 9.6, 9.9, 9.10 and 9.11, it happens at altitudes below 60 km. The Scenario X, which does not present this problem, is included as a reference to compare the magnitude of these quantities. The specific angular momentum at the beginning of the integration is 49518.994 km²/s for Scenarios X, 3a, 7b and 8b. In the case of Scenario 6b, where the relative velocity is larger, it takes the value 48843.680 km²/s. The final value in Scenario X is 24778.885 km²/s, while in Scenarios 3a, 6b, 7b and 8b is $2.220 \cdot 10^{-12}$ km²/s, 0.195 km²/s, $1.508 \cdot 10^{-12}$ km²/s and $3.924 \cdot 10^{-9}$ km²/s respectively.

The differences shown in Table 11 are, in most of the cases, smaller than 5 mm. In 6 out of 15 scenarios, DROMO and DROMO-SPE provide the smallest errors, while Cowell does it in 3 out of 15. The largest differences are reached when the decrease of the meteor radius (Scenario 7a), being of the order of centimeters. In this case Cowell's error is approximately three times the error of DROMO-SPE. In Scenarios 2b, 3b, 4b, 5b and 7a the position error for DROMO and DROMO-SPE becomes smaller than Cowell. However, Cowell requires a smaller number of function calls than DROMO and DROMO-SPE (see Table 11). This fact is arguably due to the aforementioned singularity of DROMO. Indeed, the propagator requires more

function calls when the dependent variables are subject to very fast variations along the integration, which explains the increased computational cost.

If we consider a point-by-point direct comparison between the DROMO solution and the reference solution, it is possible to assess the relative position error throughout the integration timespan. To do that, the propagator providing the reference solution uses dense output to determine the solution at the very same instants of time as the solution provided by DROMO. The results show that in all the scenarios the number of function calls tends to increase with time. Figure 9.15 displays this trend for Scenarios 1b, 2c, 3b, 4a, 5a, 5b. In some cases, like in Scenarios 1b, 4a, 5a and 5b, the relative position error describes an asymptotic behavior which clearly jeopardizes the accuracy of the solution as it can be seen in Table 11. This fact is mainly due to when σ freezes because of the singularity. Nevertheless, in most of the scenarios the error is uniform, i.e., the integration is not long enough to show an exponential error growth. Consequently, the property of the regularized method of mitigating the error growth can not take advantage.

Figure 9.4: Mass (upper), eccentricity (middle) and angular momentum (lower) in Scenario 1.

Figure 9.5: Mass (upper), eccentricity (middle) and angular momentum (lower) in Scenario 2.

Figure 9.6: Mass (upper), eccentricity (middle) and angular momentum (lower) in Scenario 3.

Figure 9.7: Mass (upper), eccentricity (middle) and angular momentum (lower) in Scenario 4.

Figure 9.8: Mass (upper), eccentricity (middle) and angular momentum (lower) in Scenario 5.

Figure 9.9: Mass (upper), eccentricity (middle) and angular momentum (lower) in Scenario 6.

Figure 9.10: Mass (upper), eccentricity (middle) and angular momentum (lower) in Scenario 7.

Figure 9.11: Mass (upper), eccentricity (middle) and angular momentum (lower) in Scenario 8.

Figure 9.12: Trajectories of the meteor in Scenarios 3 (upper) and 6 (lower).

Figure 9.13: Trajectories of the meteor in Scenarios 7 (upper) and 8 (lower).

Figure 9.14: Evolution of $1/\zeta_3$ and s in Scenarios 3a, 6b (upper) and 7b, 8b (middle) and evolution of τ in Scenarios X, 3a, 6b, 7b and 8b (lower) with DROMO propagator.

Figure 9.15: DROMO's position error with respect to the reference solution; from left to right: 1b, 2c (upper), 3b, 4a (middle) and 5a, 5b (lower).

CONCLUSIONS AND FUTURE WORK

The aim of this thesis has not only been to produce efficient numerical tools, but also to give answers to orbital mechanics problems by a thorough study and physical interpretation of the results.

In the field of orbit determination we should underline the stability and reliability of the novel SDG scheme for solving Kepler's equation. The method, combined with the Newton-Raphson algorithm in its different versions always guarantees the convergence by optimizing the initial seed. The classical Newton-Raphson provides the solution to the elliptic Kepler's equation (EKE) in 98.599 % of the cases with only one single iteration, and the solution to the hyperbolic Kepler's equation (HKE) in 99.866 % of the cases with only one or two iterations. This is remarkable result that is not found in other algorithms used in the Astrodynamics community. Improving the algorithm by using the Conway or the modified Newton-Raphson permits to obtain the right EKE solution with only one iteration in 98.631 % and 98.737 % of the cases respectively and the HKE solution with only one or two iterations in 99.868 % and 99.874 % of the cases respectively. The low number of iterations permits to use quadruple precision if it is required, because the speed of the calculations is not meaningfully jeopardized. Therefore, we obtain the benefit of a greater accuracy with a minimal cost. The method has a global application domain since it provides successfully the solution Kepler's equation in the singular corner. The asymptotic expansions used to generate the initial seed ensures the reliability and convergence of the Newton-Raphson iterative algorithm. This is another remarkable results since the convergence in this special region is not always guaranteed by some of the algorithms usually considered in the literature. Consequently, the SDG method is a robust, reliable and efficient tool for determining the initial orbit of the non-perturbed problem.

Inspired by the results obtained with the SDG code, the second contribution related to preliminary orbit determination approaches the two-body Lambert's problem, where Kepler's problem needs to be solved. A new formulation of the problem has been presented by choosing as a free parameter the true anomaly of the bisector defined by the directions of the two position vectors. This parameter, χ , describes without ambiguity and with high simplicity the different conics involved in the analysis by using the properties of the fundamental and minimum-energy ellipses. Moreover, the determination of χ in the prograde transfer allows to directly obtain the corresponding value for the conjugate orbit and its counterpart in the retrograde transfer. As part of future work, the next step of this study will be the assessment of the performance of such formulation by considering as initial seed of the problem the one computed with the SDG code. After that, a comparison and numerical analysis of the new approach with respect to different methods currently in use, like Battin and Vaughan [9], Izzo [46], Avanzini [4] and Arora and Russell [3], will be applied to draw conclusions on the performance and efficiency of the method. This solution can then be used as a initial starter for the perturbed Lambert's problem.

In the field of orbit propagation an extension to the DROMO regularization has been derived for highly perturbed problems. As it has been confirmed by the analysis carried out in different scenarios, the DROMO-SPE formulation be-

comes specially convenient for environments with a strong perturbing acceleration that nonetheless yields smooth variations of the orbital geometry, i.e. acceleration that despite being of order unity, yield trajectories that are still conveniently representable with orbital elements based on geometric features of the trajectory. In this case the DROMO-SPE propagator improves the performance of DROMO, keeping its effectiveness when the magnitude of the perturbation is not necessarily high. This suggests that the SPE formulation has various other applications and orbital scenarios, as for instance the study of deep flybys, where the perturbation exhibits a varying range of magnitudes. This exploration of potential avenues for applications is left as future work.

The last part of the dissertation deals with the study of meteor trajectories by considering several special perturbation methods. The application of DROMO and DROMO-SPE reveals the reliability of these propagators when the mass depletion of the meteor happens in the mesosphere. However, if the meteor reaches the surface, the performance of DROMO and DROMO-SPE is affected by the intrinsic singularity that results from approaching a rectilinear motion passing through the origin. This makes the computational time to increase by rising the number of function calls because of the rapid variations of some orbital elements as the ideal anomaly approaches a limiting value and freezes. This fact prevents the propagation from reaching the prescribed stop condition, and thus the integration process gets stuck and the computational cost explodes. In this study the physical model has only considered the mass ablation. However, as it is referenced in Briani et al. [17], the energy and mass balance are affected by several physical processes which condition their evolution during the atmospheric interaction. Some of these mechanisms are the melting and evaporation of the material. A potential follow up would be to explore the effects of taking into account these parameters in order to obtain more realistic physical and dynamical equations and analyze their contributions. The results obtained could then be compared with the expected observations from ALE's satellite to study the atmospheric density distribution at different altitudes.

Part III

APPENDIX

THE FIFTH DEGREE POLYNOMIAL $p_1(x, e)$

The polynomial $p_1(x, e)$ is given by the expression:

$$p_1(x, e) = a_0^{(1)}(e) + a_1^{(1)}(e)x + a_2^{(1)}(e)x^2 + a_3^{(1)}(e)x^3 + a_4^{(1)}(e)x^4 + a_5^{(1)}(e)x^5$$

where the coefficients are defined as a function of the eccentricity by

$$a_5^{(1)}(e) = \frac{432.0 e (0.5393817 e^3 - 0.8743466 e^2 + 0.2650564 e + 0.0453931)}{d_5^{(1)}(e)}$$

$$a_4^{(1)}(e) = \frac{72.0 e (0.4657683 e^3 - 0.402960 e^2 - 0.125744 e - 0.002438)}{d_4^{(1)}(e)}$$

$$a_3^{(1)}(e) = \frac{12.0 (2.1335977 e^3 - 7.174617 e^2 + 8.438043 e - 3.4460552) e}{d_3^{(1)}(e)}$$

$$a_2^{(1)}(e) = 0$$

$$a_1^{(1)}(e) = \frac{-1.0}{-1.0 + e}$$

$$a_0^{(1)}(e) = 0$$

with

$$d_5^{(1)}(e) = (3739.002236 e^3 - 2549.554211 e^4 + 926.7532457 e^5 - 140.2961154 e^6 - 3082.919872 e^2 + 1355.064876 e - 248.0502135) (-1.0 + e) (9.0 e^2 - 18.84955592 e + 9.869604404)$$

$$d_4^{(1)}(e) = (3.141592654 - 3.0 e) (231.3573259 e - 323.4979098 e^2 + 200.9177486 e^3 - 46.76537182 e^4 - 62.01255338) (-1.0 + e) (6.928203232 e - 3.0 e^2 - 4.0)$$

$$d_3^{(1)}(e) = (-27.0 e^3 + 84.82300166 e^2 - 88.82643964 e + 31.00627669) (5.196152424 e^3 - 18.0 e^2 + 20.78460970 e - 8.0) (-1.0 + e)$$

THE EKE-SDG CODE

The next functions provide the solution in double precision of the EKE-SDG code by using the Modified Newton-Rhapson method (authors: V. Raposo Pulido and J. Pelaez):

- <ellipk.h>: header file where the different modules and functions of the code are declared.
- <polinomios24.c>: polynomials for the seed (only the first polynomial is presented due to the extension of the expressions).
- <keplere.c>: set of functions to estimate the polynomial and asymptotic approximations for the seed as well as the numerical method to estimate the final value of the eccentric anomaly.
- <main.c>: module containing a driver to check the code. Is an example about how the function must be run. In this function the eccentricity and mean anomaly are defined.

B.O.1 <ellipk.h>

```

////////////////////////////////////
The KEPLERE program solves the Kepler equation for elliptic Keplerian orbits. The
code is collected in the following modules: keplere.c, polinomios24.c and ellipk.h.
The module main.c is a driver to check good operation of the program
Copyright (C) 2018 by the Universidad Politecnica de Madrid (UPM)
Authors: Virginia Raposo-Pulido and Jesus Pelaez
This program is free software: you can redistribute it and/or modify it under the
terms of the GNU General Public License as published by the Free Software Found-
ation, either version 3 of the License, or (at your option) any later version.
This program is distributed in the hope that it will be useful, but WITHOUT ANY
WARRANTY; without even the implied warranty of MERCHANTABILITY or FIT-
NESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for
more details.
You should have received a copy of the GNU General Public License along with this
program. If not, see <https://www.gnu.org/licenses/>.
////////////////////////////////////

////////////////////////////////////
FILE keplere.c
////////////////////////////////////
double ke_24(double Mo, double eo, double eps);
double etao_as(double xi, double epsilon);
double interno(double M, double e);
double nu_as(double sigmao, double epsilon);
double intermedio(double M, double e);
double gtotal(double M, double e);
double *FKE(double M, double e, double u, double *caja);
int ecuacion_de_kepler_sdg(double Mo, double eo, double *pu, double *resto, dou-
ble eps);

```

```

int kepler_eq(double M1, double e1, double *ph1, double *resto1, double eps1);
////////////////////////////////////

////////////////////////////////////
MODULE polinomios24.c
////////////////////////////////////
double PP0_ke(double e, double x);
double PP0_ke25(double e, double x);
double PP1_ke(double e, double x);
double PP2_ke(double e, double x);
double PP3_ke(double e, double x);
double PP4_ke(double e, double x);
double PP5_ke(double e, double x);
double PP6_ke(double e, double x);
double PP7_ke(double e, double x);
double PP8_ke(double e, double x);
double PP9_ke(double e, double x);
double PP10_ke(double e, double x);
double PP11_ke(double e, double x);
double PP12_ke(double e, double x);
double PP13_ke(double e, double x);
double PP14_ke(double e, double x);
double PP15_ke(double e, double x);
double PP16_ke(double e, double x);
double PP17_ke(double e, double x);
double PP18_ke(double e, double x);
double PP19_ke(double e, double x);
double PP20_ke(double e, double x);
double PP21_ke(double e, double x);
double PP22_ke(double e, double x);
double PP23_ke(double e, double x);

```

B.0.2 <polinomios24.c>

```

////////////////////////////////////
The KEPLERE program solves the Kepler equation for elliptic Keplerian orbits. The
code is collected in the following modules: keplere.c, polinomios24.c and ellipk.h.
The module main.c is a driver to check good operation of the program
Copyright (C) 2018 by the Universidad Politecnica de Madrid (UPM)
Authors: Virginia Raposo-Pulido and Jesus Pelaez
This program is free software: you can redistribute it and/or modify it under the
terms of the GNU General Public License as published by the Free Software Founda-
tion, either version 3 of the License, or (at your option) any later version.
This program is distributed in the hope that it will be useful, but WITHOUT ANY
WARRANTY; without even the implied warranty of MERCHANTABILITY or FIT-
NESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for
more details.
You should have received a copy of the GNU General Public License along with this
program. If not, see <https://www.gnu.org/licenses/>.
////////////////////////////////////

```



```
#include "ellipk.h"
```

```
////////////////////////////////////
```

This file contains the 24 polynomials $P_{Pi\_ke}(e, M)$ that provide the seed (E) of the modified Newton-Rapson method that solves the elliptic Kepler equation. The polynomials in this module are grade 5.

```
////////////////////////////////////
```

```
double PPO_ke (double e, double x)
```

```
{
double t1, t11, t17, t19, t2, t23, t26, t6, t8, t9;
```

```
t1 = x * x;
t2 = t1 * t1;
t6 = 0.1e1 / (-0.1e1 + e);
t8 = e * e;
t9 = t8 * t8;
t11 = t8 * e;
t17 = t9 * t9;
t19 = t9 * t11;
t23 = t9 * e;
t26 = t9 * t8;
```

```
return(0.3456e4*t2*x*e*t6*(0.1169099091706114e-1 + 0.671492819444721191e-1 * e -
0.11087037484241450e0 * t9 + 0.3131071359544849e0 * t11 - 0.2810909188261130e0 *
t8) / (-0.48963149565645032525e4 + 0.43120783089460391712e5 * e - 0.37026103758017141520e5
* t17 + 0.15137138580134853217e6 * t19 + 0.4025153789434402896e4 * t17 * e + 0.55340321949460409412e6
* t23 - 0.56558223538618654188e6 * t9 - 0.36098496542315134366e6 * t26 + 0.38534644417281762299e6
* t11 - 0.16877736682374551478e6 * t8) - 0.576e3 * t2 * e * t6 * (-0.768461018223e-4
- 0.397103535353572809e-2 * e - 0.1604309194516221e-1 * t9 + 0.291952768559462e-1 *
t11 - 0.91228165902839e-2 * t8) / (-0.15585454565440389959e4 + 0.12184968692226749445e5
* e - 0.1296000000000000003e4 * t17 + 0.10610566308309836480e5 * t19 + 0.77785503962289416959e5
* t23 - 0.99500937539998574645e5 * t9 - 0.38005094959292002980e5 * t26 + 0.81456789267902728562e5
* t11 - 0.41677250274894150139e5 * t8) + 0.24e2 * t1 * x * e * t6 * (-0.34451994696369384e1
+ 0.13190372883781691919e2 * e - 0.290999476081273939e1 * t9 + 0.121221300468499857e2
* t11 - 0.189573364698870189e2 * t8) / (-0.49610042688479712283e3 + 0.33881432129428691286e4
* e + 0.41727995695687749988e3 * t19 + 0.92079665847908949561e4 * t23 - 0.15731011270073840002e5
* t9 - 0.29942550071872819232e4 * t26 + 0.16124685389283909272e5 * t11 - 0.99167084398296184588e4
* t8) - 0.1e1 * x * t6);
}
```

```
B.O.3 <keplere.c>
```

```
////////////////////////////////////
```

The KEPLERE program solves the Kepler equation for elliptic Keplerian orbits. The code is collected in the following modules: keplere.c, polinomios24.c and ellipk.h. The module main.c is a driver to check good operation of the program

Copyright (C) 2018 by the Universidad Politecnica de Madrid (UPM)

Authors: Virginia Raposo-Pulido and Jesus Pelaez

This program is free software: you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation, either version 3 of the License, or (at your option) any later version.

This program is distributed in the hope that it will be useful, but WITHOUT ANY

WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details.

You should have received a copy of the GNU General Public License along with this program. If not, see <<https://www.gnu.org/licenses/>>.

////////////////////////////////////

```
#include "ellipk.h"
```

```
#include <math.h>
```

```
#include <stdio.h>
```

```
#define abs(x) ((x)>0.oL ? (x) : -(x))
```

////////////////////////////////////

The function:

```
double ke_24(double Mo, double eo, double eps)
```

provides the starting seed to use with any Newton-Raphson algorithm

The arguments are:

Mo → mean anomaly (known data)

eo → eccentricity of the elliptic orbit (0< eo <1)

eps → tolerance

The function returns the value of E (unknown of the Kepler equation) in radians

////////////////////////////////////

```
double ke_24(double Mo, double eo, double eps)
```

```
{
```

```
double XV[25], z, M, e, DBL_EPSILON, theeps, PI, DOSPI;
```

```
int i=0, j;
```

```
PI = 3.141592653589793238462643383279502884L;
```

```
DOSPI = 6.283185307179586476925286766559005768L;
```

```
DBL_EPSILON = 2.22*pow(10, -16);
```

```
if (DBL_EPSILON < eps) theeps = DBL_EPSILON;
```

```
else theeps = eps;
```

```
if (abs(Mo - 0.oL) <= theeps) return(0.oL);
```

```
if (abs(Mo - PI) <= theeps) return(PI);
```

```
if (abs(Mo - DOSPI) <= theeps) return(DOSPI);
```

```
if(Mo > PI) M = DOSPI - Mo;
```

```
else M = Mo;
```

```
e = eo;
```

```
if (abs(1.oL - e)<=0.25L && abs(M) < 0.15L)
```

```
{
```

```
z = gtotal(M, e);
```

```
if(Mo > PI) z = DOSPI - z;
```

```
return(z);
```

```
}
```

//ends of the 24 intervals associated with the 24 polynomials P in which it is divided [0, pi]

```

XV[0] = (0.0L);
XV[1] = (0.130899693899574718269276807636645953L -
0.130526192220051591548406227895489010L*e);
XV[2] = (0.261799387799149436538553615273291907L -
0.258819045102520762348898837624048328L*e);
XV[3] = (0.392699081698724154807830422909937860L -
0.382683432365089771728459984030398866L*e);
XV[4] = (0.523598775598298873077107230546583814L -
0.500000000000000000000000000000000000L*e);
XV[5] = (0.654498469497873591346384038183229766L -
0.608761429008720639416097542898164003L*e);
XV[6] = (0.785398163397448309615660845819875720L -
0.707106781186547524400844362104849039L*e);
XV[7] = (0.916297857297023027884937653456521674L -
0.793353340291235164579776961501299276L*e);
XV[8] = (1.04719755119659774615421446109316763L -
0.866025403784438646763723170752936184L*e);
XV[9] = (1.17809724509617246442349126872981358L -
0.923879532511286756128183189396788286L*e);
XV[10] = (1.30899693899574718269276807636645953L -
0.965925826289068286749743199728897366L*e);
XV[11] = (1.43989663289532190096204488400310549L -
0.991444861373810411144557526928562871L*e);
XV[12] = (1.57079632679489661923132169163975144L -
1.000000000000000000000000000000000000L*e);
XV[13] = (1.70169602069447133750059849927639739L -
0.991444861373810411144557526928562872L*e);
XV[14] = (1.83259571459404605576987530691304335L -
0.965925826289068286749743199728897367L*e);
XV[15] = (1.96349540849362077403915211454968930L -
0.923879532511286756128183189396788288L*e);
XV[16] = (2.09439510239319549230842892218633525L -
0.866025403784438646763723170752936187L*e);
XV[17] = (2.22529479629277021057770572982298121L -
0.793353340291235164579776961501299276L*e);
XV[18] = (2.35619449019234492884698253745962716L -
0.707106781186547524400844362104849042L*e);
XV[19] = (2.48709418409191964711625934509627311L -
0.608761429008720639416097542898164010L*e);
XV[20] = (2.61799387799149436538553615273291907L -
0.500000000000000000000000000000000000L*e);
XV[21] = (2.74889357189106908365481296036956502L -
0.382683432365089771728459984030398870L*e);
XV[22] = (2.87979326579064380192408976800621097L -
0.258819045102520762348898837624048335L*e);
XV[23] = (3.01069295969021852019336657564285693L -
0.130526192220051591548406227895489011L*e);
XV[24] = (PI);

while( M - XV[i] >= 0.0L) i++;
j = i -1;

```

```

switch(j)
{
    case 0:
        if(e < 0.90L) z = PP0_ke25(e,M);
        else z = PP0_ke(e,M);
        break;
    case 1:
        if(e < 0.90L) z = PP1_ke(e,M);
        else z = PP0_ke(e,M);
        break;
    case 2: z = PP2_ke(e,M); break;
    case 3: z = PP3_ke(e,M); break;
    case 4: z = PP4_ke(e,M); break;
    case 5: z = PP5_ke(e,M); break;
    case 6: z = PP6_ke(e,M); break;
    case 7: z = PP7_ke(e,M); break;
    case 8: z = PP8_ke(e,M); break;
    case 9: z = PP9_ke(e,M); break;
    case 10: z = PP10_ke(e,M); break;
    case 11: z = PP11_ke(e,M); break;
    case 12: z = PP12_ke(e,M); break;
    case 13: z = PP13_ke(e,M); break;
    case 14: z = PP14_ke(e,M); break;
    case 15: z = PP15_ke(e,M); break;
    case 16: z = PP16_ke(e,M); break;
    case 17: z = PP17_ke(e,M); break;
    case 18: z = PP18_ke(e,M); break;
    case 19: z = PP19_ke(e,M); break;
    case 20: z = PP20_ke(e,M); break;
    case 21: z = PP21_ke(e,M); break;
    case 22: z = PP22_ke(e,M); break;
    case 23: z = PP23_ke(e,M); break;
    default: z = 0.0L; break;
}

```

```

if(Mo > PI) z = DOSPI - z;
return(z);
}

```

```

////////////////////////////////////

```

The functions

```

    double interno(double M, double e)
    double etao_as (double xi, double epsilon)
control the development of interno in the singular corner
////////////////////////////////////

```

```

double etao_as (double xi, double epsilon)
{
    double t1, t2;

    t1 = xi * xi;
    t2 = t1 * epsilon;
    return(xi*(t2*(epsilon*(t1/0.12e2 + 0.1e1/0.6e1 + t2*((t1*(0.275e3*t1 + 0.1008e4)/0.6480e4

```

```

+ 0.1e1/0.12e2 - t2*(t1*(0.455e3*t1 + 0.2035e4) + 0.2016e4)/0.12960e5)* epsilon - t1/0.18e2
- 0.19e2/0.120e3)) - 0.1e1/0.6e1 + 0.1e1));
}

```

```

double interno(double M, double e)
{
double xi, epsilon;

epsilon = 0.1e1 - e;
xi = M/epsilon/epsilon;
return(epsilon*etao_as(xi,epsilon));
}

```

```

////////////////////////////////////

```

The functions

```

double intermedio(double M, double e)
double nu_as (double sigmao, double epsilon)
control the development of intermedio in the singular corner
////////////////////////////////////

```

```

double nu_as (double sigmao, double epsilon)
{
double t1;
double t2;
double t3;
double t4;

```

```

t1 = sigmao * sigmao;
t2 = t1 + 0.2e1;
t2 = 0.1e1 / t2;
t3 = t2*t2 * epsilon * t1;
t4 = t3 * (t1 * (t1 * (t1 * (t1 * (t1 * (t1 * (t1 * (t1 * (t1 * (0.757195e6 * t1
+ 0.44808682e8) + 0.1224969238e10) + 0.20654633560e11) + 0.243708581800e12) +
0.2211750858328e13) + 0.18349996321600e14) + 0.128222589683200e15) + 0.623846188000000e15)
+ 0.1882207378800000e16) + 0.3157146518400000e16) / 0.63567504000000e14 + 0.462e3
/ 0.13e2 + t3 * (t1 * (t1 * (t1 * (t1 * (t1 * (t1 * (t1 * (t1 * (t1 * (t1 * (t1 * (t1 *
(0.39895597e8 * t1 + 0.2708147672e10) + 0.85624740744e11) + 0.1679468000096e13) +
0.23019850794160e14) + 0.237013796890368e15) + 0.1962922508203200e16) + 0.15128222076687360e17)
+ 0.108665136123240960e18) + 0.614804420307200000e18) +
0.2399006473260160000e19) + 0.5951303807554560000e19) +
0.8445756809297920000e19) + 0.5235928169185280000e19) /
0.4576860288000000e17);
return(sigmao * (t2 * epsilon * t1 * (t1 / 0.60e2 + 0.1e1 / 0.3e1 + t3 * (t1 * (t1 * (t1
+ 0.25e2) + 0.340e3) / 0.1400e4 + 0.3e1 / 0.5e1 + t3 * (t1 * (t1 * (t1 * (0.5e1 *
t1 + 0.166e3) + 0.2505e4) + 0.28240e5) + 0.124100e6) / 0.126000e6 + 0.10e2 / 0.7e1
+ t3 * (t1 * (t1 * (t1 * (t1 * (t1 * (0.387e3 * t1 + 0.16172e5) + 0.306228e6) +
0.3619848e7) + 0.35945312e8) + 0.205356480e9) + 0.568176000e9) / 0.155232000e9 +
0.35e2 / 0.9e1 + t3 * (t1 * (t1 * (t1 * (t1 * (t1 * (t1 * (t1 * (t1 * (0.8491e4 * t1 +
0.428551e6) + 0.9895902e7) + 0.140539840e9) + 0.1425806620e10) + 0.12837145200e11)
+ 0.83879753200e11) + 0.327717760000e12) +
0.678930000000e12) / 0.50450400000e11 + 0.126e3 / 0.11e2 + t4)))))) +
0.1e1));

```

```

}

double intermedio(double M, double e)
{
double epsilon, chi, sigmao,s, L;

epsilon = 0.1e1 - e;
chi = M / sqrt(epsilon*epsilon*epsilon);
L = sqrt(8.0L + 9.0L*chi*chi);
s = pow(L + 3.0L*chi, 1.0L/3.0L);
sigmao = 6.0L*chi/( 2.0L + s*s + 4.0L/s/s);
return(sqrt(epsilon)*nu_as(sigmao,epsilon));
}

////////////////////////////////////
The function
    double gtotal(double M, double e);
compute the seed provided to the modified Newton-Raphson in the singular corner
The arguments are:
    M → mean anomaly
    e → eccentricity
Return the value of the eccentric anomaly which is given as seed to the N-R
////////////////////////////////////

double gtotal(double M, double e)
{
double epsilon, s, y;

epsilon = 0.1e1 - e;
s = M/pow(epsilon,1.5L);
if(s < 0.001L) y = interno(M,e);
else y = intermedio(M,e);
return(y);
}

////////////////////////////////////
The function
    double *FKE(double M, double e, double u, double *caja)
calculate the function of the kepler equation  $f(u) = u - e \sin(u) - M$  and its first two
derivatives. The arguments are:
    M → mean anomaly (known data)
    e → eccentricity of the orbit ( $0 < e < 1$ ) (known data)
    u → eccentric anomaly = E (here is a known data)
    caja → memory to store the calculated values. It is the responsibility of the
user to make the appropriate reservation of such memory
Returns a pointer to caja, which is where the calculated values are stored. The func-
tion uses the external variable nfun which collects the number of times the function
is called
////////////////////////////////////

long int nfun;
double *FKE(double M, double e, double u, double *caja)
{

```

```

extern long int nfun;

nfun++;
caja[2] = e*sin(u); // second derivative
caja[1] = 1.0L - e*cos(u); // first derivative
caja[0] = (u-caja[2]) -M; // function
return(caja);
}

////////////////////////////////////
The function
    int ecuacion_de_kepler_sdg(double M, double e, double *pu, double *resto,
double eps)
solve the kepler equation, in the elliptic case with a given precision. The arguments
are:
    M → mean anomaly (known data)
    e → eccentricity of the orbit (0<e<1) (known data)
    pu → pointer to a variable where the eccentric anomaly (in
        radians) is collected = E (unknown)
    resto → pointer to a variable where the residual of Kepler's
        equation is collected (u-e*sin(u)-M)
    eps → tolerance
The function uses the Modified Newton Raphson algorithm (MNR). The function
returns an integer that matches the number of iterations made with the MNR
////////////////////////////////////

int ecuacion_de_kepler_sdg(double Mo, double eo, double *pu, double *resto, dou-
ble eps)
{
double M, e, eps;
double p, q, r, u,s, delta, caja[3], *pd;
int i=0;

M = Mo;
e = eo;
eps = eps;
u = ke_24(M, e, eps); // seed
pd = FKE(M, e, u, caja);
p = caja[0];
q = caja[1];
r = caja[2];
if(abs(p)< eps)
    {*pu = u; *resto = p; return(i);}

for(i=1; i<50; i++)
{
    if( q > 0.0L) delta = 2.0L*p/(q + sqrt(abs(q*q-2.0L*p*r ))); // MNR
    else delta = 2.0L*p/(q - sqrt(abs(q*q-2.0L*p*r )));
    s = 1.0L-q; // improvement with third derivative s = 1 - q
    p = 2.0L*q - delta*(2.0L*r - delta*s);
    if( abs(p)> eps)
    {
        p = - delta*delta*s/p/3.0L;
    }
}

```

```

        delta *= (1.0L + p);
    }
    if( i >=6) delta *= pow(0.990L,(double)i);
    u -= delta;

    pd = FKE(M, e, u, caja);
    p = caja[0];
    q = caja[1];
    r = caja[2];

    if((abs(delta) < eps) || (abs(p) < eps))
    {
        break;
    }
}
*pu = (double)u;
*resto = (double)p;
return(i);
}

int ekepler_eq(double M1, double e1, double *ph1, double *resto1, double eps1)
{
    int i1;
    double M11, DOSPI, aux;

    DOSPI = 6.283185307179586476925286766559005768L;

    if (abs(M1) > DOSPI)
    {
        if (M1 < 0)
        {
            M11 = -(abs(M1) - floor(abs(M1) / DOSPI)*DOSPI);
            aux = floor(abs(M1) / DOSPI)*DOSPI;
            i1 = ecuacion_de_kepler_sdg(abs(M11), e1, ph1, resto1, eps1);
            *ph1 = -*ph1-aux;
        }
        else
        {
            M11 = M1 - floor(M1 / DOSPI)*DOSPI;
            aux = floor(M1 / DOSPI)*DOSPI;
            i1 = ecuacion_de_kepler_sdg(M11, e1, ph1, resto1, eps1);
            *ph1 = *ph1 + aux;
        }
    }
    else
    {
        if (M1 < 0)
        {
            i1 = ecuacion_de_kepler_sdg(abs(M1), e1, ph1, resto1, eps1);
            *ph1 = -*ph1;
        }
        else
        {

```



```

        i1 = ecuacion_de_kepler_sdg(M1, e1, ph1, resto1, eps1);
    }
}
return(i1);
}

```

B.O.4 *main.c*

```

////////////////////////////////////
The KEPLERE program solves the Kepler equation for elliptic Keplerian orbits. The
code is collected in the following modules: keplere.c, polinomios24.c and ellipk.h.
The module main.c is a driver to check good operation of the program
Copyright (C) 2018 by the Universidad Politecnica de Madrid (UPM)
Authors: Virginia Raposo-Pulido and Jesus Pelaez
This program is free software: you can redistribute it and/or modify it under the
terms of the GNU General Public License as published by the Free Software Foun-
dation, either version 3 of the License, or (at your option) any later version.
This program is distributed in the hope that it will be useful, but WITHOUT ANY
WARRANTY; without even the implied warranty of MERCHANTABILITY or FIT-
NESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for
more details.
You should have received a copy of the GNU General Public License along with this
program. If not, see <https://www.gnu.org/licenses/>.
////////////////////////////////////

#include "ellipk.h"
#include <math.h>
#include <stdio.h>

void main(void)
{
    //////////////////////////////////
    Input
        e: eccentricity of the elliptic orbit (known data)  $0 < e < 1$ 
        M: mean anomaly (known data)
    Output
        ph1: eccentric anomaly
    Note: the solution is given in double precision for the EKE-SDG code by using the
    Modified Newton-Rhapson (MNR)
    Code developed by V. Raposo-Pulido and J. Peláez
    //////////////////////////////////

    double e, M, ph1, resto1, eps;
    int i;

    eps = 2.22*pow(10, -16);
    e = 0.45;
    M = 2.5;
    i = ekepler_eq(M, e, &ph1, &resto1, eps);
    printf("\n Eccentricity: %+020.18E \n", e);
    printf("\n Mean anomaly: %+020.18E \n", M);
    printf("\n Eccentric anomaly: %+020.18E \n", ph1);

```

```
printf("\n Residual: %+020.18E \n", resto1);  
printf("\n Number of iterations: %d \n", i);  
}
```

REGISTRO TERRITORIAL
DE LA PROPIEDAD
INTELECTUAL
COMUNIDAD DE MADRID

MINISTERIO
DE CULTURA
Y DEPORTE

REGISTRO GENERAL DE LA PROPIEDAD INTELECTUAL

Según lo dispuesto en la Ley de Propiedad Intelectual (Real Decreto Legislativo 1/1996, de 12 de abril), quedan inscritos en este Registro los derechos de propiedad intelectual en la forma que se determina seguidamente:

NÚMERO DE ASIENTO REGISTRAL 16 / 2019 / 872

Título: The EKE-SDG code to solve the elliptic Kepler equation

Objeto de propiedad intelectual: Programa de ordenador

Clase de obra: Programa de ordenador

PRIMERA INSCRIPCIÓN

Autor/es y titular/es originarios de derechos

- **Apellidos y nombre:** PELÁEZ ÁLVAREZ, Jesús

Nacionalidad: España

D.N.I./N.I.F./Pasaporte: 01483058H

% sobre derechos de explotación: 50

- **Apellidos y nombre:** RAPOSO PULIDO, Virginia

Nacionalidad: España

D.N.I./N.I.F./Pasaporte: 05290453Q

% sobre derechos de explotación: 50

Transmisión de derechos

- **Titular cesionario:** UNIVERSIDAD POLITÉCNICA DE MADRID

Nacionalidad: España

D.N.I./N.I.F./Pasaporte: Q2818015F

Titular/es cesionario/s por transmisión intervivos de los derechos de explotación en la naturaleza y condiciones que se expresan a continuación:

Derechos cedidos: Reproducción, distribución, comunicación pública y transformación

Modalidades cedidas: Cualquiera de las existentes y conocidas al día de la fecha

Ámbito territorial: Mundial

Ámbito temporal: Hasta su paso a dominio público

Carácter de la cesión: Exclusiva y onerosa

- **Cedente:** PELÁEZ ÁLVAREZ, Jesús

Nacionalidad: España

D.N.I./N.I.F./Pasaporte: 01483058H

- **Cedente:** RAPOSO PULIDO, Virginia

Nacionalidad: España

D.N.I./N.I.F./Pasaporte: 05290453Q

(Continúa la inscripción)

La autenticidad de este documento se puede comprobar en www.madrid.org/csv mediante el siguiente código seguro de verificación: 0908878539389334793764

REGISTRO TERRITORIAL
DE LA PROPIEDAD
INTELECTUAL
COMUNIDAD DE MADRID

MINISTERIO
DE CULTURA
Y DEPORTE

Extracto: Mediante documento privado de fecha 13 de septiembre de 2018, D^a. Virginia Raposo Pulido y D. Jesús Peláez Álvarez, en su propio nombre y derecho y en calidad de autores, y D^a. Asunción Gómez Pérez, como Vicerrectora de Investigación, Innovación y Doctorado de la Universidad Politécnica de Madrid, en su nombre y representación, según delegación de competencias otorgada el 27 de abril de 2016 por el Sr. Rector Magnífico de la citada Universidad, acuerdan, que los primeros ceden con carácter oneroso y de forma exclusiva a la Universidad Politécnica de Madrid, que acepta, los derechos de reproducción, distribución, comunicación pública y transformación sobre la presente obra, en las condiciones señaladas arriba de este extracto.

En consecuencia, quedan inscritos los citados derechos a favor de la Universidad Politécnica de Madrid.

Datos de la solicitud

Núm. solicitud: M-6673-18

Fecha de presentación y efectos: 03/10/2018

Hora: 13:53

En MADRID, a diecinueve de febrero de dos mil diecinueve

El/La titular del Registro

Firmado digitalmente por ALEJANDRO PUERTO MENDOZA
Emitido por FNMT
Fecha: 2019.02.19 10:27:34 CET
Huella dig.: 33961a4a0f3d140800d273ca92117638536b8347

Firmado: Alejandro Puerto Mendoza

La autenticidad de este documento se puede comprobar en www.madrid.org/csv
mediante el siguiente código seguro de verificación: **0908878539389334793764**

Doc.: 09/015252.6/19

Exp.: 09-RTPI-07544.2/2018

Ref. 16/2019/872
UNIVERSIDAD POLITÉCNICA DE MADRID
A/A Asunción de María Gómez Pérez

Para su conocimiento y efectos oportunos, se le notifica que el Registro Territorial de la Propiedad Intelectual de la Comunidad de Madrid ha adoptado, con fecha 18 de febrero de 2019, la siguiente Resolución:

*"Vista la solicitud de inscripción de derechos sobre la obra titulada '**The EKE-SDG code to solve the elliptic Kepler equation**' presentada en este Registro como Programa de ordenador, el **03 de octubre de 2018**, a la que correspondió el número **M-006673/2018** y examinada la documentación aportada junto a ella, ha obtenido calificación jurídica favorable.*

Por ello, de conformidad con lo dispuesto por el artículo 145.2 del Texto Refundido de la Ley de Propiedad Intelectual, aprobado por Real Decreto Legislativo 1/1996, de 12 de abril (TRLPI), y de acuerdo con las funciones atribuidas a los Registros Territoriales por el artículo 3 del Reglamento del Registro General de la Propiedad Intelectual, aprobado por Real Decreto 281/2003, de 7 de marzo (RRGPI),

RESUELVO

*Practicar, de conformidad con la matriz que se incorpora como anexo a la presente resolución, la inscripción de derechos instada por D/Dª **UNIVERSIDAD POLITÉCNICA DE MADRID**, sobre la obra titulada '**The EKE-SDG code to solve the elliptic Kepler equation**', presentada en el Registro de la Propiedad Intelectual, el **03 de octubre de 2018**, a la que correspondió el número **M-006673/2018**."*

En cumplimiento de esta resolución, se remite copia de la matriz de inscripción con número de asiento registral **16/2019/872**, de fecha **19 de febrero de 2019**.

De conformidad con lo dispuesto en el artículo 25 del RRGPI la presente Resolución podrá ser impugnada:

- En la vía jurisdiccional civil, ante los Juzgados de lo Mercantil de Madrid, en el caso de acuerdos relativos a la inscripción y fundados en la validez o invalidez de los títulos, en la capacidad de las partes o en la existencia o inexistencia de los derechos inscribibles, así como en cualquier otra cuestión de naturaleza jurídico-privada.
- En vía administrativa, en el caso de resoluciones que tengan su fundamento en la aplicación de normas de procedimiento administrativo. En este caso, se significa que esta Resolución no pone fin a la vía administrativa, pudiendo interponer recurso de alzada ante el Viceconsejero de Cultura, Turismo y Deportes, o ante el Registrador Territorial de la Propiedad Intelectual, en el plazo de un mes contado a partir del día siguiente al de la recepción de la presente notificación.

En caso de concurrencia simultánea de las causas previstas en los apartados a) y b), la vía de impugnación procedente será la civil.

Todo ello, sin perjuicio de que pueda ejercitar cualquier otro recurso que estime procedente.

Madrid, 19 de febrero de 2019
EL REGISTRADOR DE LA PROPIEDAD INTELECTUAL
(P.D.F. Resolución de 7.03.17)
EL JEFE DEL SERVICIO DE PUBLICIDAD REGISTRAL
PEDRO JIMÉNEZ SANTIAGO

Firmado digitalmente por PEDRO JIMENEZ SANTIAGO
Emitido por FNMT
Fecha: 2019.02.19 14:12:01 CET
Huella dig.: 827f923479e45005f76f8d6790b4744bf9198abc

THE HKE-SDG CODE

The next functions provide the solution in double precision of the HKE-SDG code by using the Modified Newton-Rhapson method (authors: V. Raposo Pulido and J. Pelaez):

- <hyperk.h>: header file where the different modules and functions of the code are declared.
- <polinomios25.c> and <polinomios25Q.c>: polynomials for the seed (only the first polynomial for each function is presented due to the extension of the expressions).
- <keplerh.c>: set of functions to estimate the polynomial and asymptotic approximations for the seed as well as the numerical method to estimate the final value of the hyperbolic anomaly.
- <main.c>: module containing a driver to check the code. Is an example about how the function must be run. In this function the eccentricity and mean anomaly are defined.

C.0.1 <hyperk.h>

```
////////////////////////////////////
The KEPLERH program solves the Kepler equation for hyperbolic Keplerian or-
bits. The code is collected in the following modules: keplerh.c, polinomios25.c poli-
nomios25Q.c and hyperk.h. The module main.c is a driver to check good operation
of the program
```

Copyright (C) 2018 by the Universidad Politecnica de Madrid (UPM)

Authors: Virginia Raposo-Pulido and Jesus Pelaez

This program is free software: you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation, either version 3 of the License, or (at your option) any later version.

This program is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details.

You should have received a copy of the GNU General Public License along with this program. If not, see <<https://www.gnu.org/licenses/>>.

```
////////////////////////////////////
```

```
////////////////////////////////////
```

```
FILE keplerh.c
```

```
////////////////////////////////////
```

```
double keh_25(double Mo, double eo, double eps);
double nuh_as(double sigmao, double epsilon);
double intermedih(double M, double e);
double gtotalh(double M, double e);
double *FKEH(double M, double e, double u, double *caja);
int ECUACION_DE_KEPLERH_sdgh(double Mo, double eo, double *pu, double
*resto, double eps);
```

```

double FCT(double S, double x, double e);
double gtotalhDOS(double M, double e);
double gtotalhTRES(double M, double e);
int hkepler_eq(double M1, double e1, double *ph1, double *resto1, double eps1);
////////////////////////////////////

```

```

////////////////////////////////////

```

```

MODULE polinomios25.c

```

```

////////////////////////////////////

```

```

double PPHo_ke(double e, double x);
double PPH1_ke(double e, double x);
double PPH2_ke(double e, double x);
double PPH3_ke(double e, double x);
double PPH4_ke(double e, double x);
double PPH5_ke(double e, double x);
double PPH6_ke(double e, double x);
double PPH7_ke(double e, double x);
double PPH8_ke(double e, double x);
double PPH9_ke(double e, double x);
double PPH10_ke(double e, double x);
double PPH11_ke(double e, double x);
double PPH12_ke(double e, double x);
double PPH13_ke(double e, double x);
double PPH14_ke(double e, double x);
double PPH15_ke(double e, double x);
double PPH16_ke(double e, double x);
double PPH17_ke(double e, double x);
double PPH18_ke(double e, double x);
double PPH19_ke(double e, double x);
double PPH20_ke(double e, double x);
double PPH21_ke(double e, double x);
double PPH22_ke(double e, double x);
double PPH23_ke(double e, double x);
double PPH24_ke(double e, double x);

```

```

////////////////////////////////////

```

```

MODULE polinomios25Q.c

```

```

////////////////////////////////////

```

```

double QQHo_ke(double e, double x);
double QQH1_ke(double e, double x);
double QQH2_ke(double e, double x);
double QQH3_ke(double e, double x);
double QQH4_ke(double e, double x);
double QQH5_ke(double e, double x);
double QQH6_ke(double e, double x);
double QQH7_ke(double e, double x);
double QQH8_ke(double e, double x);
double QQH9_ke(double e, double x);
double QQH10_ke(double e, double x);
double QQH11_ke(double e, double x);
double QQH12_ke(double e, double x);
double QQH13_ke(double e, double x);
double QQH14_ke(double e, double x);

```



```
double QQH15_ke(double e, double x);
double QQH16_ke(double e, double x);
double QQH17_ke(double e, double x);
double QQH18_ke(double e, double x);
double QQH19_ke(double e, double x);
double QQH20_ke(double e, double x);
double QQH21_ke(double e, double x);
double QQH22_ke(double e, double x);
double QQH23_ke(double e, double x);
```

c.o.2 <polinomios25.c>

```
////////////////////////////////////
```

The KEPLERH program solves the Kepler equation for hyperbolic Keplerian orbits. The code is collected in the following modules: keplerh.c, polinomios25.c polinomios25Q.c and hyperk.h. The module main.c is a driver to check good operation of the program

Copyright (C) 2018 by the Universidad Politecnica de Madrid (UPM)

Authors: Virginia Raposo-Pulido and Jesus Pelaez

This program is free software: you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation, either version 3 of the License, or (at your option) any later version.

This program is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details.

You should have received a copy of the GNU General Public License along with this program. If not, see <<https://www.gnu.org/licenses/>>.

```
////////////////////////////////////
```

```
#include "hyperk.h"
```

```
////////////////////////////////////
```

This file contains the 25 polynomials PPHi_ke(e, M) that provide the seed ($S = \sinh(H)$) of the modified Newton-Rapson method that solves the hyperbolic Kepler equation. The polynomials in this module are grade 5.

```
////////////////////////////////////
```

```
double PPHo_ke (double e, double x)
```

```
{
double t8o;
```

```
t8o = x * x;
```

```
return((0.1e1 / (e - 0.1e1) +
((0.125619031568129558537380271410332022e1 +
(-0.386896297445922329942791253526031698e1 +
(0.397458463649162035319156450269314720e1 -
0.136187300221481648906471060024501584e1 * e) * e) *
e) / (-0.753709874513349572842363902433619830e1 +
(0.533644703178604011903435655253666572e2 +
(-0.161926955103827310067286221499564231e3 +
```

```

(0.272966483185125737342108674783906395e3 +
(-0.276087334554369700721547069525986956e3 +
(0.167544674481392071366040635772371738e3 +
(-0.564856332122875304429233576403064543e2 +
0.816139363123982706168741160854905003e1 * e) * e) * e) *
e) * e) * e) * e) + ((-0.958688217924823935501782842000181705e-4 +
(-0.510121027425997692040825072704324965e-2 +
(-0.16474166644863896333205206206751994e-1 +
0.220780757484086884956517410243543940e-1 * e) * e) *
e) / (0.753709874513349572842363902433619829e1 +
(-0.609519169783737701972277800942650357e2 +
(0.215647900761431692125272092573189541e3 +
(-0.435975112406404674084898863736474596e3 +
(0.550877237315339524214194941686440662e3 +
(-0.445476275938404291166795839088250074e3 +
(0.225149508247949125762430962753541289e3 +
(-0.650243515910619529228042795155324044e2 +
0.821591184439085054140512639701441960e1 * e) * e) * e) *
e) * e) * e) * e) * e) + (0.252674257989487261929593560802687026e-1 +
(0.181156144842423602994829135698156169e0 +
(-0.421277152858462836797440502550471456e0 +
0.214548459711471257973372417227719640e0 * e) * e) *
e) / (-0.301483949805339829136945560973447932e1 +
(0.27415745455548556816447978652653657e2 +
(-0.110802790927857499178963701893328363e3 +
(0.261225417553932260103325655073133759e3 +
(-0.395905867604755066389384777273332807e3 +
(0.400013352079265784684681827805794974e3 +
(-0.269440628547842955871503875019778378e3 +
(0.116671144565895074989564934132360149e3 +
(-0.294698507720984166784906571885258136e2 +
0.330831769595936095049525210814559361e1 * e) * e) * e) *
e) * e) * e) * e) * e) * e) * x) * x) * t80) * x);
}

```

c.o.3 <polinomios25Q.c>

////////////////////////////////////

The KEPLERH program solves the Kepler equation for hyperbolic Keplerian orbits. The code is collected in the following modules: keplerh.c, polinomios25.c polinomios25Q.c and hyperk.h. The module main.c is a driver to check good operation of the program

Copyright (C) 2018 by the Universidad Politecnica de Madrid (UPM)

Authors: Virginia Raposo-Pulido and Jesus Pelaez

This program is free software: you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation, either version 3 of the License, or (at your option) any later version.

This program is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details.

You should have received a copy of the GNU General Public License along with this program. If not, see <<https://www.gnu.org/licenses/>>.

[illegible]

[illegible]

[illegible]

C.O.4 <keplerh.c>

////////////////////////////////////

The KEPLERH program solves the Kepler equation for hyperbolic Keplerian orbits. The code is collected in the following modules: `keplerh.c`, `polinomios25.c`, `polinomios25Q.c` and `hyperk.h`. The module `main.c` is a driver to check good operation of the program

Copyright (C) 2018 by the Universidad Politecnica de Madrid (UPM)

Authors: Virginia Raposo-Pulido and Jesus Pelaez

This program is free software: you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation, either version 3 of the License, or (at your option) any later version.

This program is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details.

You should have received a copy of the GNU General Public License along with this program. If not, see <<https://www.gnu.org/licenses/>>.

```

////////////////////////////////////

#include "hyperk.h"
#include <math.h>
#include <stdio.h>

#define abs(x) ((x)>0.oL ? (x) : (-(x)) )

////////////////////////////////////
The function:
    double keh_25(double Mo, double eo, double eps)
provides the starting seed to use with any Newton-Raphson algorithm
The arguments are:
    Mo → mean anomaly (known data)
    eo → eccentricity of the hyperbolic orbit (eo >1)
    eps → tolerance
The function returns the value of S=sinh(H) (unknown of the Kepler equation)
////////////////////////////////////

double keh_25(double Mo, double eo, double eps)
{
    double XV[51], z, M, e, phi, tt;
    double xi;

    if (abs(e - 1.oL)<=0.5L && abs(M) < 0.25L)
    {
        z = gtotalh(M, e);
        return(sinh(z));
    }

    //ends of the 25 intervals associated with the 25 polynomials P

    XV[0] = (0.oL);
    XV[1] = (- 0.2L + 0.201336002541093987625568243010317373L*e);
    XV[2] = (- 0.4L + 0.410752325802815508540210013844698104L*e);
    XV[3] = (- 0.6L + 0.636653582148241271123454375465148319L*e);
    XV[4] = (- 0.8L + 0.888105982187623006574717573189756981L*e);
    XV[5] = (- 1.0L + 1.17520119364380145688238185059560082L*e);
    XV[6] = (- 1.2L + 1.50946135541217269644289491125921093L*e);
    XV[7] = (- 1.4L + 1.90430150145153405514212382769742631L*e);
    XV[8] = (- 1.6L + 2.37556795320022969758455354439030725L*e);
    XV[9] = (- 1.8L + 2.94217428809567977271710961629775564L*e);
    XV[10] = (- 2.0L + 3.62686040784701876766821398280126170L*e);
    XV[11] = (- 2.2L + 4.45710517053589352156881637051936233L*e);
    XV[12] = (- 2.4L + 5.46622921367609457443138377479405527L*e);
    XV[13] = (- 2.6L + 6.69473222839367825866130738120707085L*e);
    XV[14] = (- 2.8L + 8.19191835423591595325119731137053734L*e);
    XV[15] = (- 3.0L + 10.0178749274099018989745936194658281L*e);
    XV[16] = (- 3.2L + 12.2458839965654912141970922329098858L*e);
    XV[17] = (- 3.4L + 14.9653633887183436343401764511296270L*e);
    XV[18] = (- 3.6L + 18.2854553606153475958966014183740521L*e);
    XV[19] = (- 3.8L + 22.3394068607223287208896227032622527L*e);
    XV[20] = (- 4.0L + 27.2899171971277524489082715907938186L*e);

```

```

XV[21] = (- 4.2L + 33.3356677320523319694048751468816484L*e);
XV[22] = (- 4.4L + 40.7192956625325245016109361699722550L*e);
XV[23] = (- 4.6L + 49.7371319030945875769059601966175554L*e);
XV[24] = (- 4.8L + 60.7510938858429303641033771355589455L*e);
XV[25] = (- 5.0L + 74.2032105777887589770094719960645656L*e);

```

```
//ends of the 24 intervals associated with the 24 polynomials Q
```

```

XV[26] = (0.100166750019844025823729383521905024L*e - 0.1L);
XV[27] = (0.304520293447142618958435267005095229L*e - 0.3L);
XV[28] = (0.521095305493747361622425626411491559L*e - 0.5L);
XV[29] = (0.758583701839533503459874647592768154L*e - 0.7L);
XV[30] = (1.02651672570817527595833616197842235L*e - 0.9L);
XV[31] = (1.33564747012417677938478052357867844L*e - 1.1L);
XV[32] = (1.69838243729261580866757837422406835L*e - 1.3L);
XV[33] = (2.12927945509481749683438749467763165L*e - 1.5L);
XV[34] = (2.64563193383723255528348091262507841L*e - 1.7L);
XV[35] = (3.26816291152831718171575932929456328L*e - 1.9L);
XV[36] = (4.02185674215733408161554001720300250L*e - 2.1L);
XV[37] = (4.93696180554595850311383723155543542L*e - 2.3L);
XV[38] = (6.05020448103978732145032363835040319L*e - 2.5L);
XV[39] = (7.40626310606654217337122065930621505L*e - 2.7L);
XV[40] = (9.05956107469332685682315502164694588L*e - 2.9L);
XV[41] = (11.0764510395240377993798196445394570L*e - 3.1L);
XV[42] = (13.5378778766283237106863842027448534L*e - 3.3L);
XV[43] = (16.5426272876349976249567315290124982L*e - 3.5L);
XV[44] = (20.2112904167985255688457159280278653L*e - 3.7L);
XV[45] = (24.6911035970421847456447294987383451L*e - 3.9L);
XV[46] = (30.1618574609801041249726180050204907L*e - 4.1L);
XV[47] = (36.8431125702917979900023602980633864L*e - 4.3L);
XV[48] = (45.0030111519917856218096568056437152L*e - 4.5L);
XV[49] = (54.9690385875109015313183231905375222L*e - 4.7L);
XV[50] = (67.1411665509322802497701958639658740L*e - 4.9L);

```

```

while( M - XV[i] >= 0.0L && i <26) i++;
j = i -1;

```

```

if (j==25) k=25;
else if (j==0 && M<XV[26]) k=0;
else if (j==24 && M>XV[50]) k=24;
else if (M<XV[j+26])
{
    if ((M-XV[j])<(XV[j+26]-M)) k=j;
    else k=j+25;
}
else if (M>XV[j+26])
{
    if ((M-XV[j+26])<(XV[j+1]-M)) k=j+26;
    else k=j;
}

switch(k)
{

```

```

case 0: z = PPHo_ke(e,M); break;
case 1: z = PPH1_ke(e,M); break;
case 2: z = PPH2_ke(e,M); break;
case 3: z = PPH3_ke(e,M); break;
case 4: z = PPH4_ke(e,M); break;
case 5: z = PPH5_ke(e,M); break;
case 6: z = PPH6_ke(e,M); break;
case 7: z = PPH7_ke(e,M); break;
case 8: z = PPH8_ke(e,M); break;
case 9: z = PPH9_ke(e,M); break;
case 10: z = PPH10_ke(e,M); break;
case 11: z = PPH11_ke(e,M); break;
case 12: z = PPH12_ke(e,M); break;
case 13: z = PPH13_ke(e,M); break;
case 14: z = PPH14_ke(e,M); break;
case 15: z = PPH15_ke(e,M); break;
case 16: z = PPH16_ke(e,M); break;
case 17: z = PPH17_ke(e,M); break;
case 18: z = PPH18_ke(e,M); break;
case 19: z = PPH19_ke(e,M); break;
case 20: z = PPH20_ke(e,M); break;
case 21: z = PPH21_ke(e,M); break;
case 22: z = PPH22_ke(e,M); break;
case 23: z = PPH23_ke(e,M); break;
case 24: z = PPH24_ke(e,M); break;
case 25:
{
double le, l2, lm;
le = log(e);
l2 = log(2.0L);
lm = log(M);
tt = sqrt(e*e + M*M);
phi = (tt*(log(tt+M)-le))/((tt-1.0)*M);
lm = log(e/2.0L/M);
xi = -lm*lm/2/M/M/M;
z = (M/e)*(1.0+phi+xi);
{
break;
case 26: z = QQHo_ke(e,M); break;
case 27: z = QQH1_ke(e,M); break;
case 28: z = QQH2_ke(e,M); break;
case 29: z = QQH3_ke(e,M); break;
case 30: z = QQH4_ke(e,M); break;
case 31: z = QQH5_ke(e,M); break;
case 32: z = QQH6_ke(e,M); break;
case 33: z = QQH7_ke(e,M); break;
case 34: z = QQH8_ke(e,M); break;
case 35: z = QQH9_ke(e,M); break;
case 36: z = QQH10_ke(e,M); break;
case 37: z = QQH11_ke(e,M); break;
case 38: z = QQH12_ke(e,M); break;
case 39: z = QQH13_ke(e,M); break;
case 40: z = QQH14_ke(e,M); break;

```


[illegible]

```

}

double intermedih(double M, double e)
{
double epsilon, chi, sigmao, sdos, s, L;

epsilon = e - 0.1e1
chi = M / sqrt(epsilon*epsilon*epsilon);
L = sqrt(8.0L + 9.0L*chi*chi);
s = pow(L + 3.0L*chi, 1.0L/3.0L);
sdos = s*s;
sigmao = 6.0L*chi/( 2.0L + sdos + 4.0L/sdos);
return(sqrt(epsilon)*nuh_as(sigmao,epsilon));
}

////////////////////////////////////
The function
    double gtotalh(double M, double e);
compute the seed provided to the modified Newton-Raphson in the singular corner
The arguments are:
    M → mean anomaly
    e → eccentricity
Return the value S=sinh(H) which is given as seed to the N-R
////////////////////////////////////

double gtotalh(double M, double e)
{
double y;

y = intermedih(M,e);
return(y);
}

double FCT(double S, double x, double e)
{
double z;

z = (x + log(S + sqrt(S*S + 1.0L)))/e;
return(z);
}

double gtotalhDOS(double M, double e)
{
int n,i;
double S, u;

n = 10;

for (i = 0; i < n; i++)
{
    if (i == 0) u = (M / e);
    else u = FCT(S, M, e);
    S = u;
}

```

```

}
return(S);
}

double gtotalhTRES(double M, double e)
{
int n,i;
double S, u, z;
double le, l2, lm, tt, phi, xi;

le = log(e);
l2 = log(2.0L);
lm = log(M);
tt = sqrt(e*e + M*M);
phi = (tt*(log(tt + M) - le)) / ((tt - 1.0)*M);
xi = (-1.0L / (2.0L*pow(M, 3.0L)))*(pow(le - l2, 2.0L) + lm*(lm + 2.0L*(l2 - le)));
z = (M / e)*(1.0 + phi + xi);

n = 10;

printf("\n M= %020.18lf e= %020.18lf", (double)M, (double)e);
for (i = 0; i < n; i++)
{
    if (i == 0) u = z;
    else u = FCT(S, M, e);
    S = u;
printf("\n S= %020.18lf", (double)S);
}
printf("\n S= %020.18lf\n\n", (double)S);
return(S);
}

////////////////////////////////////
The function
    double *FKEH(double M, double e, double u, double *caja)
calculate the function of the kepler equation  $f(u) = u - \log(u + \sqrt{1 + \text{pow}(u, 2)}) / e - M/e$  and its first two derivatives. The arguments are:
    M → mean anomaly (known data)
    e → eccentricity of the orbit ( $e > 1$ ) (known data)
    u → hyperbolic sine of the hyperbolic anomaly =  $\sinh(H)$  (here is a known
data)
    caja → memory to store the calculated values. It is the responsibility of the
user to make the appropriate reservation of such memory
Returns a pointer to caja, which is where the calculated values are stored. The func-
tion uses the external variable nfun which collects the number of times the function
is called
////////////////////////////////////

long int nfun;
double *FKEH(double M, double e, double u, double *caja)
{
extern long int nfun;

```

```

double a,ad, at, ac;

nfun++;

ad = 1.oL + u*u;
a = sqrt(ad);
at = ad*a;
ac = at*ad;

c aja[3] = (3.oL - 2.oL*ad)/ac/e; // third derivative
c aja[2] = u/at/e; // second derivative
c aja[1] = 1.oL - 1.oL/a/e; // first derivative
c aja[0] = u - log(u+a)/e -M/e; // function
return(c aja);
}

////////////////////////////////////
The function
    int ECUACION_DE_KEPLERH_sdgh(double M, double e, double *pu,
double *resto, double eps)
solve the kepler equation, in the hyperbolic case with a given precision. The arguments are:
    M → mean anomaly (known data)
    e → eccentricity of the orbit (e>1) (known data)
    pu → pointer to a variable where the hyperbolic sine of the hyperbolic anomaly is
        collected = sinh(H) (unknown)
    resto → pointer to a variable where the residual of Kepler's equation is collected (e*u-log(u+sqrt(1+pow(u,2)))-M)
    eps → tolerance
The function uses the Modified Newton Raphson algorithm (MNR). The function returns an integer that matches the number of iterations made with the MNR
////////////////////////////////////

int ECUACION_DE_KEPLERH_sdgh(double Mo, double eo, double *pu, double *resto, double eps)
{
double M, e, eps, u, tt, delta, c[4], *pd;
double DBL_EPSILON;
int i=0;

DBL_EPSILON = 2.22*pow(10, -16);
M = Mo;
e = eo;
eps = eps0;
u = keh_25(M, e, eps); // seed
pd = FKEH(M, e, u, c);
if(abs(c[0])< eps)
    {*pu = u; *resto = c[0]; return(i);}

for(i=1; i<50; i++)
{
    if( c[1] > 0.oL) delta = 2.oL*c[0]/(c[1] + sqrt(abs(c[1]*c[1]

```

```

-2.0L*c[o]*c[2] )); // MNR
else delta = 2.0L*c[o]/(c[1] - sqrt(abs(c[1]*c[1]-2.0L*c[o]*c[2] )));
tt = -c[3]*delta*delta*delta/(3.0L*c[3]*delta*delta -6.0L*c[2]*delta+6.0L*c[1]);
delta += tt;

u -= delta;
pd = FKEH(M, e, u, c);

if (abs(u)<= 1.0L)
{
    if((abs(c[o]) < eps) || (abs(delta) < DBL_EPSILON)) break;
}
else
{
    if ((abs(c[o]) < eps) || (abs(delta)<abs(u*DBL_EPSILON))) break;
}
}
*pu = u;
*resto = c[o];
return(i);
}

////////////////////////////////////
The function
    int hkepler_eq(double M1, double e1, double *ph1, double *resto1, double eps1)
consider the sign of the mean anomaly to solve the kepler equation, in the hyperbolic case with a given precision. The arguments are:
    M1 → mean anomaly (known data)
    e1 → eccentricity of the orbit (e>1) (known data)
    ph1 → pointer to a variable where the hyperbolic anomaly is collected
        = H (unknown)
    resto1 → pointer to a variable where the residual of Kepler's
        equation is collected (e1*pu1-log(pu1+sqrt(1+pow(pu1,2)))-M1)
    eps → tolerance
The function returns an integer that matches the number of iterations made with the MNR
////////////////////////////////////

int hkepler_eq(double M1, double e1, double *ph1, double *resto1, double eps1)
{
    int i1;
    double s;

    if (M1 >= 0)
    {
        i1 = ECUACION_DE_KEPLERH_sdgh(M1, e1, &s, resto1, eps1);
        *ph1 = log(s + sqrt(1.0 + s*s));
    }
    else
    {
        M1 = -M1;
        i1 = ECUACION_DE_KEPLERH_sdgh(M1, e1, &s, resto1, eps1);
    }
}

```

```

    *ph1 = -log(s + sqrt(1.0 + s*s))
}
return(i1);
}

```

C.0.5 *main.c*

```

////////////////////////////////////
The KEPLERH program solves the Kepler equation for hyperbolic Keplerian or-
bits. The code is collected in the following modules: keplerh.c, polinomios25.c poli-
nomios25Q.c and hyperk.h. The module main.c is a driver to check good operation
of the program

```

Copyright (C) 2018 by the Universidad Politecnica de Madrid (UPM)

Authors: Virginia Raposo-Pulido and Jesus Pelaez

This program is free software: you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation, either version 3 of the License, or (at your option) any later version.

This program is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details.

You should have received a copy of the GNU General Public License along with this program. If not, see <<https://www.gnu.org/licenses/>>.

```

////////////////////////////////////

```

```

#include "hyperk.h"
#include <math.h>
#include <stdio.h>

```

```

void main(void)
{

```

```

////////////////////////////////////

```

Input

e: eccentricity of the hyperbolic orbit (known data) $e > 1$

M: mean anomaly (known data)

Output

ph1: hyperbolic anomaly

Note: the solution is given in double precision for the HKE-SDG code by using the Modified Newton-Rhapson (MNR)

Code developed by V. Raposo-Pulido and J. Peláez

```

////////////////////////////////////

```

```

double e, M, ph1, resto1, eps;

```

```

int i;

```

```

eps = 2.22*pow(10, -16);

```

```

e = 2.25;

```

```

M = 40.2;

```

```

i = hkepler_eq(M, e, &ph1, &resto1, eps);

```

```

printf("\n Eccentricity: %+020.18E \n", e);

```

```

printf("\n Mean anomaly: %+020.18E \n", M);

```

```

printf("\n Hyperbolic anomaly: %+020.18E \n", ph1);

```

```
printf("\n Residual: %+020.18E \n", resto1);  
printf("\n Number of iterations: %d \n", i);  
}
```

REGISTRO TERRITORIAL
DE LA PROPIEDAD
INTELECTUAL
COMUNIDAD DE MADRID

MINISTERIO
DE CULTURA
Y DEPORTE

REGISTRO GENERAL DE LA PROPIEDAD INTELECTUAL

Según lo dispuesto en la Ley de Propiedad Intelectual (Real Decreto Legislativo 1/1996, de 12 de abril), quedan inscritos en este Registro los derechos de propiedad intelectual en la forma que se determina seguidamente:

NÚMERO DE ASIENTO REGISTRAL 16 / 2019 / 743

Título: The HKE-SDG code to solve the hyperbolic Kepler equation

Objeto de propiedad intelectual: Programa de ordenador

Clase de obra: Programa de ordenador

PRIMERA INSCRIPCIÓN

Autor/es y titular/es originarios de derechos

- **Apellidos y nombre:** PELÁEZ ÁLVAREZ, Jesús

Nacionalidad: España

D.N.I./N.I.F./Pasaporte: 01483058H

% sobre derechos de explotación: 50

- **Apellidos y nombre:** RAPOSO PULIDO, Virginia

Nacionalidad: España

D.N.I./N.I.F./Pasaporte: 05290453Q

% sobre derechos de explotación: 50

Transmisión de derechos

- **Titular cesionario:** UNIVERSIDAD POLITÉCNICA DE MADRID

Nacionalidad: España

D.N.I./N.I.F./Pasaporte: Q2818015F

Titular/es cesionario/s por transmisión intervivos de los derechos de explotación en la naturaleza y condiciones que se expresan a continuación:

Derechos cedidos: Reproducción, distribución, comunicación pública y transformación

Modalidades cedidas: Según el Art. 43.2 de la L.P.I.

Ámbito territorial: Mundial

Ámbito temporal: Hasta su paso a dominio público

Carácter de la cesión: Exclusiva y onerosa

- **Cedente:** PELÁEZ ÁLVAREZ, Jesús

Nacionalidad: España

D.N.I./N.I.F./Pasaporte: 01483058H

- **Cedente:** RAPOSO PULIDO, Virginia

Nacionalidad: España

D.N.I./N.I.F./Pasaporte: 05290453Q

(Continúa la inscripción)

La autenticidad de este documento se puede comprobar en www.madrid.org/csv mediante el siguiente código seguro de verificación: 124085070416548556195

REGISTRO TERRITORIAL
DE LA PROPIEDAD
INTELECTUAL
COMUNIDAD DE MADRID

MINISTERIO
DE CULTURA
Y DEPORTE

Extracto: Mediante documento privado de fecha 30 de agosto de 2018, D^a. Virginia Raposo Pulido y D. Jesús Peláez Álvarez, en su propio nombre y derecho y en calidad de autores, y D^a, Asunción Gómez Pérez, como Vicerrectora de Investigación, Innovación y Doctorado de la Universidad Politécnica de Madrid, en su nombre y representación, según delegación de competencias otorgada el 27 de abril de 2016 por el Sr. Rector Magnífico de la citada Universidad, acuerdan, que los primeros ceden con carácter oneroso y de forma exclusiva a la Universidad Politécnica de Madrid, que acepta, los derechos de reproducción, distribución, comunicación pública y transformación sobre la presente obra, en las condiciones señaladas arriba de este extracto.

En consecuencia, quedan inscritos los citados derechos a favor de la Universidad Politécnica de Madrid.

Datos de la solicitud

Núm. solicitud: M-6230-18

Fecha de presentación y efectos: 18/09/2018

Hora: 13:37

En MADRID, a quince de febrero de dos mil diecinueve

El/La titular del Registro

Firmado digitalmente por ALEJANDRO PUERTO MENDOZA
Emitido por FNMT
Fecha: 2019.02.15 11:22:04 CET
Huella dig.: e3ca44339dcec502bcd423c5ec7e6f55484d4b52

Firmado: Alejandro Puerto Mendoza

La autenticidad de este documento se puede comprobar en www.madrid.org/csv
mediante el siguiente código seguro de verificación: **1240850704165486556195**

Doc.: 09/014126.6/19

Exp.: 09-RTPI-07042.3/2018

Ref. 16/2019/743
UNIVERSIDAD POLITÉCNICA DE MADRID
A/A Asunción de María Gómez Pérez

Para su conocimiento y efectos oportunos, se le notifica que el Registro Territorial de la Propiedad Intelectual de la Comunidad de Madrid ha adoptado, con fecha 14 de febrero de 2019, la siguiente Resolución:

"Vista la solicitud de inscripción de derechos sobre la obra titulada 'The HKE-SDG code to solve the hyperbolic Kepler equation' presentada en este Registro como Programa de ordenador, el 18 de septiembre de 2018, a la que correspondió el número M-006230/2018 y examinada la documentación aportada junto a ella, ha obtenido calificación jurídica favorable.

Por ello, de conformidad con lo dispuesto por el artículo 145.2 del Texto Refundido de la Ley de Propiedad Intelectual, aprobado por Real Decreto Legislativo 1/1996, de 12 de abril (TRLPI), y de acuerdo con las funciones atribuidas a los Registros Territoriales por el artículo 3 del Reglamento del Registro General de la Propiedad Intelectual, aprobado por Real Decreto 281/2003, de 7 de marzo (RRGPI),

RESUELVO

Practicar, de conformidad con la matriz que se incorpora como anexo a la presente resolución, la inscripción de derechos instada por la UNIVERSIDAD POLITÉCNICA DE MADRID, sobre la obra titulada 'The HKE-SDG code to solve the hyperbolic Kepler equation', presentada en el Registro de la Propiedad Intelectual, el 18 de septiembre de 2018, a la que correspondió el número M-006230/2018."

En cumplimiento de esta resolución, se remite copia de la matriz de inscripción con número de asiento registral 16/2019/743, de fecha 15 de febrero de 2019.

De conformidad con lo dispuesto en el artículo 25 del RRGPI la presente Resolución podrá ser impugnada:

- En la vía jurisdiccional civil, ante los Juzgados de lo Mercantil de Madrid, en el caso de acuerdos relativos a la inscripción y fundados en la validez o invalidez de los títulos, en la capacidad de las partes o en la existencia o inexistencia de los derechos inscribibles, así como en cualquier otra cuestión de naturaleza jurídico-privada.
- En vía administrativa, en el caso de resoluciones que tengan su fundamento en la aplicación de normas de procedimiento administrativo. En este caso, se significa que esta Resolución no pone fin a la vía administrativa, pudiendo interponer recurso de alzada ante el Viceconsejero de Cultura, Turismo y Deportes, o ante el Registrador Territorial de la Propiedad Intelectual, en el plazo de un mes contado a partir del día siguiente al de la recepción de la presente notificación.

En caso de concurrencia simultánea de las causas previstas en los apartados a) y b), la vía de impugnación procedente será la civil.

Todo ello, sin perjuicio de que pueda ejercitar cualquier otro recurso que estime procedente.

Madrid, 15 de febrero de 2019
EL REGISTRADOR DE LA PROPIEDAD INTELECTUAL
(P.D.F. Resolución de 7.03.17)
EL JEFE DEL SERVICIO DE PUBLICIDAD REGISTRAL
PEDRO JIMÉNEZ SANTIAGO

Firmado digitalmente por PEDRO JIMENEZ SANTIAGO
Emitido por FNMT
Fecha: 2019.02.15 13:45:51 CET
Huella dig.: a688e3caab735ebd63c9d85dc60461d94e8cd304

BIBLIOGRAPHY

- [1] S. Abe. *Meteoroids and meteors - Observations and connection to parent bodies*. Chapter in *Lecture Notes in Physics*. 2009.
- [2] Fatheah Ahmed Alhindi and M.A. Sharaf. "Homotopy Continuation Method of Arbitrary Order of Convergence for Solving Differenced Hyperbolic Kepler's Equation." In: *Journal of Advances in Mathematics* 10.5 (2014), pp. 3457–3462.
- [3] Arora and Russell. "A fast and robust multiple revolution Lambert algorithm using a cosine transformation." In: *AAS/AIAA Astrodynamics Specialist Conference, Astrodynamics - Hilton Head Island, SC, Vol. 150, 2013*. Vol. 150. 2013, pp. 411–430.
- [4] G. Avanzini. "A simple Lambert algorithm." In: *Journal of Guidance, Control, and Dynamics* 31.6 (2008), pp. 1587–1594.
- [5] M. Avendano, V. Martín-Molina, and J. Ortigas-Galindo. "A new method for obtaining approximate solutions of the hyperbolic Kepler's equation." In: *arXiv preprint arXiv:1503.01641* (2015).
- [6] Martín Avendano, Verónica Martín-Molina, and Jorge Ortigas-Galindo. "Approximate solutions of the hyperbolic Kepler equation." In: *Celestial Mechanics and Dynamical Astronomy* 123.4 (2015), pp. 435–451.
- [7] G.S. Balaraman and D. Vranceanu. "Numerical solution of modified-Kepler problem using a splitting method." In: *Physics Letters A* 369.3 (2007), pp. 188–195.
- [8] R.R. Bate, D.D. Mueller, and J.E. White. *Fundamentals of astrodynamics*. Dover Publications, New York, 1971.
- [9] Battin and Vaughan. "An elegant Lambert algorithm." In: *Journal of Guidance, Control, and Dynamics* 7.6 (1984), pp. 662–670.
- [10] R.H. Battin. *Astronautical guidance*. McGraw Hill, New York, 1964.
- [11] R.H. Battin. *An introduction to the mathematics and methods of astrodynamics*. Revised edition AIAA Reston, VA, 1999.
- [12] G. Baú, H. Urrutxua, and J. Peláez. "EDROMO: An Accurate Propagator for Elliptical Orbits in the Perturbed Two-Body Problem." In: *Adv Astronaut Sci* (2014).
- [13] Giulio Baù, Alexander Huhn, Hodei Urrutxua, Claudio Bombardelli, and Jesús Peláez. "DROMO: A New Regularized Orbit Propagator." In: *International Symposium on Orbit Propagation and Determination*. 2011.
- [14] Giulio Baù, Claudio Bombardelli, Jesús Peláez, and Enrico Lorenzini. "Non-singular orbital elements for special perturbations in the two-body problem." In: *Monthly Notices of the Royal Astronomical Society* 454 (2015), pp. 2890–2908.
- [15] H. Beust, M. Bonnefoy, A-L. Maire, D. Ehrenreich, A-M. Lagrange, and G. Chauvin. "Orbital fitting of imaged planetary companions with high eccentricities and unbound orbits-Their application to Fomalhaut b and PZ Teocopii B." In: *Astronomy & Astrophysics* 587 (2016), A89. DOI: [10.1051/0004-6361/201527388](https://doi.org/10.1051/0004-6361/201527388).
- [16] J. Borovicka. "The comparison of two methods of determining meteor trajectories from photographs." In: *House of the Czechoslovak Academy of Sciences* 41.6 (1990), pp. 391–396.

- [17] G. Briani, Pace. E., S.N. Shore, G. Pupillo, A. Passaro, and S. Aiello. "Simulations of micrometeoroid interactions with the Earth atmosphere." In: *Astronomy & Astrophysics* (2014).
- [18] C.A. Bürdet. "Theory of Kepler motion: the general perturbed two body problem." In: *Zeitschrift für angewandte Mathematik und Physik* 19.2 (1968), pp. 345–368.
- [19] Peter Colwell. *Solving Kepler's equation over three centuries*. ISBN 0-943396-40-9. Willmann-Bell, Richmond, Virginia, 1993.
- [20] Bruce A. Conway. "An improved algorithm due to Laguerre for the solution of Kepler's equation." In: *Celestial mechanics* 39.2 (1986), pp. 199–211.
- [21] J. Danby. *Fundamentals of celestial mechanics*. 2nd ed., ISBN = 0-943396-20-4. Willman-Bell Inc., Richmond, Virginia 23235, 1992.
- [22] M. Davidson. "Determination of the true anomaly and the time of perihelion passage in hyperbolic orbits." In: *Monthly Notices of the Royal Astronomical Society* 93 (1932), p. 33.
- [23] M. Davidson. "Note on Davidson's method of determining the true anomaly and perihelion-time in hyperbolic orbits, Note on Dr. Smiley's paper." In: *Monthly Notices of the Royal Astronomical Society* 95 (1934), p. 66.
- [24] C.B. Davies and C. Park. "Trajectories of solid particles spalled from a carbonaceous heat shield." In: *AIAA 20th Aerospace Sciences Meeting, Orlando, Florida*. 1982.
- [25] R.S.C. Davuluri, H. Zhang, and A. Martin. "Numerical study of spallation phenomenon in arc-jet environment." In: *Journal of Thermophysics and Heat Transfer* 30.1 (2016).
- [26] W. Dehnen and D.M. Hernandez. "Symplectic fourth-order maps for the collisional N-body problem." In: *Monthly Notices of the Royal Astronomical Society* 465 (2016), pp. 1201–1217.
- [27] A. Deprit. "Ideal elements for perturbed Keplerian motions." In: *Journal of Research of the National Bureau of Standards. Sec. B: Math. Sci.* 79B, No. 1-2 (1975). Paper 79B1&2-416, pp. 1–15.
- [28] A. Deprit. "Ideal frames for perturbed Keplerian motions." In: *Celestial Mechanics and Dynamical Astronomy* 13.2 (1976), pp. 253–263. URL: [D : /GDTWS / LIBRERIAS/DROM0.V2/DOC/PAPERS/deprit.19875.idel.frames.pdf](http://dx.doi.org/10.1007/BF01205131).
- [29] R.W. Detra, N.H. Kemp, and F.R. Ridell. "Addendum to Heat transfer to satellite vehicles re-entering the atmosphere." In: *Jet Propulsion* 27.12 (1957), pp. 1256–1257.
- [30] Johann Franz Encke and L. Schäfli. *Über die allgemeinen Störungen der Planeten*. Berliner Astronomisches Jahrbuch für 1857, (Königlich Akademie der Wissenschaften, Berlin), 1854.
- [31] M.B. Faintich. "Application of the restricted hyperbolic three-body problem to a star-sun-comet system." In: *Celestial mechanics* 6.1 (1972), pp. 22–26.
- [32] Toshio Fukushima. "A method solving Kepler's equation without transcendental function evaluations." In: *Celestial Mechanics and Dynamical Astronomy* 66.3 (1996), pp. 309–319.
- [33] Toshio Fukushima. "A method solving Kepler's equation for hyperbolic case." In: *Celestial Mechanics and Dynamical Astronomy* 68.2 (1997), pp. 121–137.
- [34] R.H. Gooding. *Universal procedures for conversion of orbital elements to and from position and velocity (unperturbed orbits)*. Tech. rep. Royal Aircraft Establishment, 1987.

- [35] R.H. Gooding. "A procedure for the solution of Lambert's orbital boundary-value problem." In: *Celestial Mechanics and Dynamical Astronomy* 48.2 (1990), pp. 145–165.
- [36] R.H. Gooding and A.W. Odell. "The hyperbolic Kepler equation (and the elliptic equation revisited)." In: *Celestial mechanics* 44.3 (1988), pp. 267–282.
- [37] William H. Goodyear. "Completely general closed-form solution for coordinates and partial derivative of the two-body problem." In: *The Astronomical Journal* 70 (1965), p. 189.
- [38] M. Gritsevich and D. Koschny. "Constraining the luminous efficiency of meteors." In: *International Journal of Solar System Studies* (2011).
- [39] Peter S. Gural. "A new method of meteor trajectory determination applied to multiple unsynchronized video cameras." In: *Meteoritics & Planetary Science* 9 (2012), pp. 1405–1418.
- [40] P. A. Hansen. "Auseinandersetzung einer Zweckmässigen Methode zur Berechnung der absoluten Störungen der Kleinen Planeten." In: *Abh. der Math.-Phys. Cl. der Kon. Sachs. Ges. der Wissensch.* 5 (1857).
- [41] C.B. Henderson. "Drag coefficients of spheres in continuum and rarefield flows." In: *AIAA Journal* 14.6 (1976).
- [42] D. M. Hernandez and E. Bertschinger. "Symplectic integration for the collisional gravitational N-body problem." In: *Monthly Notices of the Royal Astronomical Society* 452 (2015), pp. 1934–1944.
- [43] S. Herrick. *Astro dynamics: Orbit correction, perturbation theory, integration*. Van Nostrand Reinhold, New York, 1971.
- [44] Gerald R. Hintz. "Survey of orbit element sets." In: *Journal of guidance, control, and dynamics* 31.3 (2008), pp. 785–790.
- [45] A. Hurwitz. "Über die Reduktion der binären quadratischen Formath." In: *Werke* 2 (1933).
- [46] Dario Izzo. "Revisiting Lambert's problem." In: *arXiv:1403.2705* (2014).
- [47] R. Kalman. "A New Approach to Linear Filtering and Prediction Problems." In: *ASME Journal of Basic Engineering* (1960).
- [48] Natsumi Kimura. "Sensitivity analysis on aerodynamic and thermal parameters for trajectory and brightness of artificial meteors." MA thesis. Tokyo Metropolitan University, 2018.
- [49] Rudolf Kurth. "A global particular solution to the initial-value problems of stellar dynamics." In: *Quarterly of Applied Mathematics* 36.3 (1978), pp. 325–329.
- [50] P.E. Kustaanheimo, A. Schinzel, H. Davenport, and E.L. Stiefel. "Perturbation theory of Kepler motion based on spinor regularization." In: *Crelles Journal* (1965).
- [51] C.T. Leondes. *Theory and Applications of Kalman Filters*. 1970.
- [52] J.E. Lyne, Tauber M., and R. Fought. "An analytical model of the atmospheric entry of large meteors and its application to the Tunguska Event." In: *Journal of Geophysical Research* 101 (1996).
- [53] F.L. Markley. "Kepler equation solver." In: *Celestial Mechanics and Dynamical Astronomy* 63.1 (1995), pp. 101–111.
- [54] A. Milani and A.M. Nobili. "Integration error over very long time spans." In: *Celestial mechanics* 43.1-4 (1987), pp. 1–34.

- [55] A. Milani, A.M. Nobili, K. Fox, and M. Carpino. "Long-term changes in the semimajor axes of the outer planets." In: *Nature* 319 (1986), pp. 386–388.
- [56] O. Montenbruck and E. Gill. *Satellite Orbits. Models, methods and applications*. Springer-Verlag Berlin Heidelberg New York, 2000.
- [57] Daniele Mortari and Antonio Elife. "Solving Kepler's equation using implicit functions." In: *Celestial Mechanics and Dynamical Astronomy* 118.1 (2014), pp. 1–11.
- [58] Albert Nijenhuis. "Solving Kepler's equation with high efficiency and accuracy." In: *Celestial Mechanics and Dynamical Astronomy* 51.4 (1991), pp. 319–330.
- [59] A. Pál. "An analytical solution for Kepler's problem." In: *Monthly Notices of the Royal Astronomical Society* 396.3 (2009), pp. 1737–1742.
- [60] M. Palacios and M.C. Calvo. "Ideal frames and regularization in numerical orbit computation." In: *The Journal of the Astronautical Sciences* 44.1 (1996), pp. 63–77.
- [61] C. Park. "Interaction of spalled particles with shock layer flow." In: *Journal of Thermophysics and Heat Transfer* 13.4 (1999).
- [62] J. Peláez, J.M. Hedo, and P. de Andres. "A special perturbation method in orbital dynamics." In: *Celestial Mechanics and Dynamical Astronomy* 97.2 (2007), pp. 131–150.
- [63] J.M. Picone, A.E. Hedin, and D.P. Drob. "NRLMSISE-00 empirical model of the atmosphere: Statistical comparisons and scientific issues." In: *Journal of Geophysical Research* 107 (2002).
- [64] E.T. Pitkin. "A regularized approach to universal orbit variables." In: *AIAA Journal* 3.8 (1965), pp. 1508–1511.
- [65] D. Prabhu and D. Saunders. "Thermophysics issues relevant to high-speed Earth entry of large asteroids." In: *AIAA SciTech*. 2016.
- [66] Y. Prevèreaud. "Contribution a la modelisation de la rentree atmospherique des debris spatiaux." PhD thesis. Institute Supèrieur de l'Aeronautique et de l'Espace, l'Universite de Toulouse, 2014.
- [67] A.A. Rambaut. "Mechanical solution of Kepler's equation." In: *Monthly Notices of the Royal Astronomical Society* 66 (1906), p. 520.
- [68] V. Raposo-Pulido and J. Peláez. "An efficient code to solve the Kepler equation. Elliptic case." In: *Monthly Notices of the Royal Astronomical Society* 467.2 (2017), pp. 1702–1713.
- [69] V. Raposo-Pulido and J. Peláez. "An efficient code to solve the Kepler equation. Hyperbolic case." In: *Astronomy and Astrophysics* (2018).
- [70] A. Rauh and J. Parisi. "Quantum mechanics of hyperbolic orbits in the Kepler problem." In: *Physical Review A* 83.4 (2011), p. 042101.
- [71] H. Rein and D. Tamayo. "WHFAST: a fast and unbiased implementation of a symplectic Wisdom-Holman integrator for long-term gravitational simulations." In: *Monthly Notices of the Royal Astronomical Society* 452 (2015), pp. 376–388.
- [72] J. Roa. *Regularization in Orbital Mechanics. Theory and Practice*. De Gruyter, 2017.
- [73] Javier Roa, Manuel Sanjurjo-Rivo, and Jesus Peláez. "Singularities in Dromo formulation. Analysis of deep flybys." In: *Advances in Space Research* 56 (2015), pp. 569–581.

- [74] Javier Roa, Hodei Urrutxua, and Jesús Peláez. "Satbility and chaos in Kustaanheimo-Stiefel space induced by the Hopf fibration." In: *Monthly Notices of the Royal Astronomical Society* (2016).
- [75] A.E. Roy. *Orbital motion (3rd revised and enlarged edition)*. Adam Hilger, Bristol, U.K., 1988.
- [76] A.E. Roy and P.E. Moran. "Studies in the application of recurrence relations to special perturbation methods." In: *Celestial mechanics* 7.2 (1973), pp. 236–255.
- [77] A.J. Sarnecki. "Minimal orbital dynamics." In: *Acta Astronautica* 17.8 (1988), pp. 881–891.
- [78] R.A. Serafin. "Bounds on the solution to Kepler's equation." In: *Celestial mechanics* 38.2 (1986), pp. 111–121.
- [79] M.A. Sharaf and A.A. Sharaf. "Homotopy continuation method of arbitrary order of convergence for the two-body universal initial value problem." In: *Celestial Mechanics and Dynamical Astronomy* 86.4 (2003), pp. 351–362.
- [80] C.E. Siewert and E.E. Burniston. "An exact analytical solution of Kepler's equation." In: *Celestial mechanics* 6.3 (1972), pp. 294–304.
- [81] G. Sitarski. "Digital Computer Solution of the Equation of Position of a Comet on a Keplerian Orbit." In: *Acta Astronomica* 18 (1968), p. 197.
- [82] C.H. Smiley. "Note on Davidson's method of determining the true anomaly and perihelion-time in hyperbolic orbits." In: *Monthly Notices of the Royal Astronomical Society* 95 (1934), p. 63.
- [83] H. Sperling. "Computation of Keplerian conic sections." In: *ARSJ* 31.5 (1961), pp. 660–661.
- [84] E.L. Stiefel and G. Scheifele. *Linear and regular celestial mechanics: Perturbed two-body motion, numerical methods, canonical theory*. Springer-Verlag (Berlin and New York), 1971.
- [85] Laura Stumpf. "Chaotic behaviour in the newton iterative function associated with Kepler's equation." In: *Celestial Mechanics and Dynamical Astronomy* 74.2 (1999), pp. 95–109.
- [86] K. Stumpff. *On the application of spinors to the pproblem of celestial mechanics*. Tech. rep. National Aeronautics and Space Administration, 1968.
- [87] K.F. Sundman. "Mémoire sur le problème des trois corps." In: (1912).
- [88] Louis C. Sutherland and Henry E. Bass. "Atmospheric absorption in the atmosphere up to 160 km." In: *The Journal of the Acoustical Society of America* 115 (2004).
- [89] H.S. Tsien. "Take-Off from Satellite Orbit." In: *Journal of the American Rocket Society* 23.4 (1953), pp. 233–236.
- [90] S.E. Urban and P. Kenneth Seidelmann. *Explanatory supplement to the astronomical almanac*. University Science Books, 2013.
- [91] David A. Vallado. *Fundamentals of Astrodynamics and Applications*. Ed. by Space Technology Library. Microcosm Press and Springer, 2007.
- [92] V. Vinnikov, M. Gritsevich, D. Kuznetsova, and V. Lukashenko. "Drag, ablation and fragmentation of a meteor body in the atmosphere." In: *Proceedings of the XLI Summer School-Conference "Advanced problems in mechanics"*. 2013, pp. 619–624.
- [93] M. Wahde. "Determination of orbital parameters of interacting galaxies using a genetic algorithm." In: *Astronomy & Astrophysics Supplement Series* (1998), pp. 417–429.

- [94] Jack Wisdom and David M. Hernandez. "A fast and accurate universal Kepler solver without Stumpff series." In: *Monthly Notices of the Royal Astronomical Society* 453.3 (2015), pp. 3015–3023.
- [95] Jack Wisdom and Matthew Holman. "Symplectic maps for the n-body problem." In: *The Astronomical Journal* 102 (1991), pp. 1528–1538.
- [96] Shin Yabushita. "The possibility of capture in the restricted problem of three bodies and formation of bridges between galaxies." In: *Monthly Notices of the Royal Astronomical Society* 153.1 (1971), pp. 97–109.