UNIVERSITAT POLITÈCNICA
DE CATALUNYA
BARCELONATECH

Spacecraft state propagation and orbit determination using jet transport

Jianlin Chen

Spacecraft State Propagation and Orbit Determination Using Jet Transport

Author
Jianlin Chen

Thesis Advisor
Josep J. Masdemont, Gerard Gómez, JianPing Yuan

PhD dissertation

submitted to the Universitat Politècnica de Catalunya (UPC) in
partial fulfillment of the requirements for the degree of
DOCTOR OF PHILOSOPHY

Ph.D. program on Aerospace Science and Technology
Barcelona, January 2021

Thesis written by Jianlin Chen

Spacecraft State Propagation and Orbit Determination Using Jet Transport

Ph.D. program on Aerospace Science and Technology

This work has been supported by the China Scholarship Council (Grant 201606290066), National Natural Science Foundation of China (Grant 11572248), Ministerio de Economia, Industria y comptetividad (MINECO)-Fondo Europeo de Desarrollo Regional (FEDER) (Grant MTM2015-65715-P, MTM2016-80117-P, PGC2018-100928-B), and Catalan government (Grant 2017SGR-1049, 2017SGR-1374).

Acknowledgments

This dissertation is a full summary of my PhD research work of the past three and a half years. During this period, I have received a lot of selfless help and support from my advisors, parents and friends. There is no doubt that they also made a very important contribution to the development of this research work. I would like to express my sincere gratitude to all of them herein.

First and foremost, I would like to thank the members of my advisory committee, Prof. Josep J. Masdemont from Universitat Politècnica de Catalunya (UPC), Prof. Gerard Gómez from Universitat de Barcelona (UB), and Prof. Jianping Yuan from Northwestern Polytechnical University (NPU), for all their guidance, support, and enthusiasm for my research. Thanks to them, I was able to attend some exciting academic conferences and share my work with other researchers. In Barcelona, Prof. Josep J. Masdemont and Prof. Gerard Gómez not only provides me many great tips to improve my research, but also introduces me a lot of local cultures and festivals, making me enjoy the life in Barcelona. Gerard's humor made me feel relax in the weekly meeting. Josep always encouraged me to be more confident and not be afraid of the unknown. Those factors make me more optimistic and active in both life and research now. I also would like to thank Josep's family members for their kindness and open-mindedness, making me feel at home. Being with them saves me from the pain of homesickness. In Xi'an, Prof. Jianping Yuan always taught me to set high goals in research and in life. His continuous encouragement and unconditional trust gave me the opportunity to study in Barcelona. I also want to thank the staff of the Department of Applied Mathematics I, UPC, and National Key Lab of Aerospace Flight Dynamics, NPU, for their warm-heartedness and concern.

I am very grateful to all my friends, from UPC and NPU. In particular, I want to thank the following persons. Duan Xun and Pablo Machuca, thanks for the exciting life and experiences we enjoyed together in and outside the office; Qiu Zihua and Fang Jian, thanks for the joys and sorrows we shared during my PhD period and the continuous supports to me.

Finally, I would like to express my deep appreciation and love to my parents for their selfless and endless support and trust. I feel very grateful they raised me to the person that I am now. I will never forget their great and unconditional love, support and advice.

Abstract

The motion of space objects orbiting the Earth is not only affected by the Keplerian central gravity field of the Earth, but it is also subjected to some complex perturbations, such as solar radiation pressure, luni-solar gravitational attraction, Earth's non-spherical gravity, atmospheric drag, as the main ones, and also others of smaller significance. Since the equations of motion are strongly nonlinear and non-integrable, it is difficult to obtain closed-form analytical solutions for predicting the satellite motion and only long and particular expansions can be available. Therefore, numerical integration techniques are the main tools to accurately propagate spacecraft trajectories. However, due to the rapid time-varying characteristics of satellite dynamics, small integration time steps have to be often employed, which results in a heavy computational time, specially for many simulations to obtain statistical distributions. To improve the computational efficiency, some alternatives, such as covariance analysis and unscented transformations, have been developed at the cost of accuracy loss. Consequently, using these state transition methods, it is hard to achieve high propagation accuracy and efficiency.

On the other hand, these state transition methods are indispensable in the design of sequential Bayesian filters, so the developed orbit estimators inherit their weaknesses. This is, the problem is either a bad prediction accuracy or a low computational efficiency. In order to alleviate this dilemma about the computational accuracy and efficiency in both orbit propagation and orbit determination applications, a set of polynomial-based high order nonlinear propagators and estimators have been developed. In particular, this dissertation proposes a specific polynomial algebraic software based in an arbitrary number of variables, called Jet Transport (JT), which enables to perform the precise and efficient Taylor or Chebyshev polynomial algebra and its implementation in nonlinear state propagation and estimation algorithms.

1) ***An efficient polynomial operation tool has been proposed.*** This tool defines a series of polynomial algebraic manipulation in a modern computer, such as polynomial storage, addition, subtraction, multiplication, division, differentiation, integration and composition with an arbitrary number of variables. The tool includes as well useful algorithms, for instance, polynomial-based Runge-Kutta integrators and efficient polynomial evaluation. In overall, the methodology provides an efficient and accurate way to calculate polynomial expansions (such as Taylor and Chebyshev series), up to an arbitrary order, for practical problems defined by a set of nonlinear functions or ordinary differential equations (ODEs).

2) ***Efficient and accurate propagations of the spacecraft trajectories and its associated statistics, like advanced Monte Carlo (MC) methods, are put forward.*** This newly proposed MC method is constructed via the combination of high order polynomial propagators with polynomial evaluation techniques at real vectors. Both algorithms are implemented in the JT framework. To be specific, the high order polynomial propagators are based on either Taylor series expansion or Chebyshev interpolation technique and employed to approximate the flow of the dynamics with the polynomial results. Then, the polynomial evaluation technique is used to transport the state vectors in a neighborhood of an initial state to determine statistical distributions.

3) ***An augmented high order extended Kalman filter is proposed in the JT framework (JT-AHEKF) to jointly estimate the spacecraft trajectories and their parameters.*** In the design of the JT-AHEKF filter, the aforementioned high order Taylor series expansion method is employed to achieve the a priori prediction of the state and measurement (tracking) vectors. Apparently, the JT-AHEKF filter enables to extract more nonlinear information from the dynamical and measurement models, such that both spacecraft trajectories and associated parameters are estimated accurately, even when very adverse conditions, such as large initial state deviations and low measurement frequency, are considered. Furthermore, a standard high order extended Kalman filter can be also deduced in the JT framework (JT-HEKF) by degrading the augmented JT-AHEKF filter without consideration of parameter estimation.

4) ***To avoid the pollution of false measurements on the performance of the JT-HEKF filter, three fault-tolerant strategies have been put forward and analyzed.*** These practical strategies are proposed based on either the direct abandon of the identified false measurements or the adaptive adjustment of the measurement noise covariance matrix with the usage of a single scale factor and an adaptive scale matrix.

5) ***An adaptive order-switching strategy tailored for the JT-HEKF filter (JT-OSHEKF) has been proposed to further mitigate the dilemma about the computational accuracy and efficiency in orbit determination applications.*** Although, to a great extent, the JT-HEKF filter reaches an efficient balance between the computational accuracy and efficiency, its continuous usage in a whole estimation process can be uneconomic since a high order JT-HEKF filter might be not necessary at the filter steady stage. A specific algorithm, based on a chi-square test, for detecting the filter consistency has been designed to dynamically and automatically switch the filter order within one single run, making the filtering process even more efficient.

To conclude, this dissertation proposes a set of nonlinear state propagators and estimators using polynomial expansion techniques, whose computational efficiency are improved by the newly developed Jet Transport software. The feasibility and reliability of the algorithms have been tested in propagation and estimation problems involving geosyn-

chronous trajectories. The results show that the proposed propagators and estimators are highly efficient and accurate. Note that these propagators and filters can be not only applied in practical space missions, but also regarded as a useful tool to achieve state prediction and estimation in general engineering problems.

Resum

El moviment de cossos orbitant la Terra no només està influenciat pel camp gravitatori Keplerià central d'aquesta, sinó que a més, està subjecte a perturbacions complexes: la pressió de radiació solar, l'atracció luni-solar, les perturbacions del camp gravitatori terrestre i el frenat atmosfèric són les més importants, però també n'hi ha d'altres de menor significància. Degut a que les equacions del moviment són fortament no lineals i no integrables, és difícil d'obtenir expressions analítiques de la solució per a fer prediccions del moviment de satèl·lits i només es disposa d'expansions llarges i particulars. Aleshores, les tècniques d'integració numèriques són les eines més importants per a la propagació de trajectòries de manera precisa. Passa però, que degut a variacions temporals ràpides de la dinàmica, cal usar passos d'integració petits que es tradueix en un cost computacional elevat quan calen moltes simulacions per a trobar distribucions estadístiques. A fi de millorar aquesta carrega computacional, s'han desenvolupat algunes alternatives, com l'anàlisi de covariança i transformacions no biaxades (unscented), però amb el cost de pèrdua de precisió. Consequentment, usant aquests mètodes de transició d'estat és difícil d'aconseguir nivells de precisió i eficiència alts.

D'altra banda, els mètodes de transició d'estat són indispensables en el disseny de filtres Bayesians seqüencials, i per tant, els estimadors d'òrbita que se'n derivin herederan les seves febleses. Així que, els límits de la balança del problema els tindrem entre prediccions pobres i baixa eficiencia computacional. Per a alleugerir aquest dilema computacional de precisió respecte d'eficiència, ja sigui en la propagació d'òrbites o en el problema de determinació d'òrbita, s'han desenvolupat un conjunt d'integradors i estimadors d'ordre alt basats en àlgebra polinomial amb un nombre arbitrari de variables. En particular, aquesta memòria implementa un software específic, anomenat Jet Transport (JT), que permet, de manera precisa i eficient, obtenir expansions polinomials de Taylor o de Chebyshev, que s'implementen en algoritmes per a problemes de propagació i estimació no lineals.

1) ***Es proposa una eina operacional polinòmica eficient.*** Aquesta eina defineix manipulacions algebràiques polinomials adaptades a computadores modernes, entre elles, enmagatzematge de polinomis, adició, multiplicació, divisió, diferenciació, integració i composició amb un nombre arbitrari de variables. L'eina inclou també altres algoritmes útils com per exemple, integradors Runge-Kutta basats en àlgebra polinomial i metodologies eficients per a l'avaluació dels polinomis. En resum, la metodologia proveeix una manera eficient i precisa de calcular expansions polinomials (ja siguin de Taylor o de Chebyshev) d'ordre arbitrari, per a problemes definits

per un conjunt de funcions no lineals o bé per equacions diferencials ordinàries (ODEs).

2) ***Es proveeix una propagació precisa i eficient per a trajectòries de satèl·lits i suport per l'estadística associada via mètodes Montecarlo avançats.*** Aquesta nova metodologia de Montecarlo es construeix via la combinació d'integradors numèrics d'ordre alt basats en àlgebra polinomial i tècniques per a la seva avaluació vectorial. Ambdues metodologies s'implementen dins l'entorn JT. Dit de manera una mica més precisa, els propagadors polinomials d'ordre elevat estan basats en expansions de Taylor o d'interpolacions de Chebyshev i s'empren per a aproximar el flux de la dinàmica obtenint-ne expressions polinomials. Aleshores amb això, s'usa la tècnica d'avaluació polinomial per a transportar els vectors d'estat a l'entorn d'una condició inicial per a determinar distribucions estadístiques.

3) ***Es proposa un filtre de Kalman augmentat d'ordre alt dins l'entorn JT (JT-AHEKF) per a fer estimacions conjuntes de trajectòries de satèl·lits i de paràmetres associats.*** En el disseny del filtre JT-AHEKF, s'usen les expansions de Taylor abans esmentades per a aconseguir la predicció a priori d'estat i els vectors de mesura (de seguiment). S'evidencia que el filtre JT-AHEKF permet extreure més informació no lineal dels models dinàmics i de mesura. De manera que la trajectòria del satèl·lit i els paràmetres associats s'estimen de manera precisa, inclús en condicions molt adverses, com podrien ser grans desviacions de la condició inicial o baixa freqüència de mesura. A més, es desenvolupa un filtre de Kalman estàndard d'ordre alt dins l'entorn JT, anomenat JT-HEKF, per degradació del JT-AHEKF, sense considerar la part d'estimació de paràmetres.

4) ***A fi d'evitar la pol·lució deguda a mesures errònies en el rendiment del filtre JT-HEKF, es proposen i analitzen tres estratègies tolerants a fallades.*** Aquestes estratègies es proposen de manera pràctica, basades o bé en el descartament directe de les mesures identificades com a errònies o bé en l'ajust adaptatiu de la matriu de covariança que quantifica el soroll de mesura, cosa que s'aconsegueix usant un sol factor d'escala i un escalat adaptatiu de la matriu.

5) ***Es proposa una estratègia de variació d'ordre pel filtre JT-HEKF (JT-OSHEKF) per a mitigar, encara més, el dilema entre el cost i la precisió computacional en problemes de determinació d'òrbita.*** Malgrat que en gran mesura el filtre JT-HEKF ja aconsegueix un balanç molt eficient entre precisió i cost computacional, el seu ús continu en totes les parts del procés d'estimació d'òrbita pot ser poc econòmic, ja que un filtre JT-HEKF d'ordre alt pot no ser necessari en estats estacionaris. Per a això, s'ha dissenyat un algoritme específic, basat en el test de khi-quadrat, per a detectar la consistència del filtre i, de manera dinàmica i automàtica, adaptar l'ordre del filtre dins del mateix cicle d'aplicació, fent així que el filtre sigui encara més eficient.

En conclusió, aquesta memòria proposa un conjunt de propagadors i estimadors per a dinàmiques no lineals usant tècniques d'expansions polinomials, l'eficiència computacional

de les quals es veu millorada per un software de transport de jet. La viabilitat i fiabilitat dels algoritmes ha estat testejada en problemes de propagació i estimació d'òrbita associats a trajectòries geosíncrones. Els resultats mostren que els propagadors i estimadors desenvolupats són molt eficients i precisos. Notem finalment que aquests integradors i filtres no són només d'aplicació en problemes de missions espacials, sinó que es poden considerar una eina útil per a problemes d'enginyeria en general.

Contents

1 Introduction **1**
- 1.1 Advanced space activities . . . 1
  - 1.1.1 Geosynchronous earth orbit synthetic aperture radars . . . 2
  - 1.1.2 Space situational awareness mission . . . 3
- 1.2 Orbit uncertainty propagation methods . . . 5
  - 1.2.1 Monte Carlo simulations . . . 7
  - 1.2.2 Local linearized methods . . . 8
  - 1.2.3 Nonlinear uncertainty propagation methods . . . 9
- 1.3 Sequential filtering techniques . . . 12
  - 1.3.1 Linear Kalman filters . . . 12
  - 1.3.2 Nonlinear sequential filters . . . 13
  - 1.3.3 Particle filter . . . 14
- 1.4 Research overview and contributions . . . 15
- 1.5 Publications and academic activities . . . 17

2 Jet transport technique **19**
- 2.1 Polynomial storage . . . 20
- 2.2 Polynomial algebra . . . 23
  - 2.2.1 Basic polynomial operations . . . 23
  - 2.2.2 Differential polynomial algebra . . . 27
  - 2.2.3 Polynomial approximation to univariate functions . . . 29
  - 2.2.4 Polynomial approximation to a multi-variable function . . . 33
  - 2.2.5 Numerical simulations . . . 34
- 2.3 Polynomial-based flow propagation . . . 37
  - 2.3.1 Parameterization method . . . 38
  - 2.3.2 Design of polynomial-based integrators . . . 40
  - 2.3.3 Semi-analytical polynomial propagation . . . 43
  - 2.3.4 Polynomial evaluation . . . 44
  - 2.3.5 Numerical simulations . . . 45
- 2.4 Remarks and conclusions . . . 50

3 Analysis of high order orbit uncertainty propagation **53**
- 3.1 Dynamical models . . . 54
  - 3.1.1 Cartesian dynamic model . . . 54
  - 3.1.2 Cylindrical coordinates . . . 55
  - 3.1.3 GEO dynamical model . . . 55

3.1.4 Coordinate transformations . . . 56
3.2 Analysis of dominant perturbation accelerations . . . 58
3.2.1 Perturbation modelling . . . 58
3.2.2 Perturbation analysis . . . 59
3.3 Geostationary orbit propagation . . . 61
3.3.1 Selection of fixed step integrator . . . 61
3.3.2 Polynomial-based orbit propagation . . . 61
3.3.3 Comparison between different coordinate representations . . . 68
3.4 Remarks . . . 69

4 Jet transport-based joint orbit and parameter estimation 71
4.1 Augmented high order extended Kalman filter . . . 71
4.2 Joint orbit and parameter estimation . . . 75
4.2.1 Equations of motion . . . 75
4.2.2 Measurement model . . . 76
4.2.3 Case A: spacecraft state and physical parameter estimation . . . 77
4.2.4 Case B: spacecraft state and tracking station position estimation . 77
4.2.5 Evaluation metrics . . . 78
4.3 Numerical simulations . . . 79
4.3.1 Spacecraft physical parameter estimation . . . 79
4.3.2 Tracking station position estimation . . . 88
4.4 Remarks and conclusions . . . 89

5 Autonomous orbit determination and fault-tolerant designs 91
5.1 Standard high order extended Kalman filter . . . 92
5.2 Fault-tolerant variants of the JT-HEKF-n filter . . . 94
5.2.1 Fault measurement detection . . . 94
5.2.2 False measurement-discarding based JT-HEKF-n filter . . . 95
5.2.3 Single and multiple scale factor based JT-HEKF-n filter . . . 95
5.3 Model description . . . 98
5.3.1 Equations of motion . . . 98
5.3.2 Measurement model . . . 99
5.4 Numerical simulations . . . 99
5.4.1 Autonomous nonlinear orbit determination . . . 100
5.4.2 Autonomous fault-tolerant orbit determination . . . 108
5.5 Remarks and conclusions . . . 114

6 An efficient nonlinear statistical adaptive order-switching filter 117
6.1 Order-switching based JT-HEKF-n filter . . . 118
6.1.1 Design of the adaptive order-switching strategy . . . 118
6.1.2 Detailed implementation of a JT-OSHEKF-n filter . . . 120
6.2 Model description . . . 121
6.2.1 Equations of motion . . . 121
6.2.2 Measurement model . . . 122
6.3 Numerical simulations . . . 122
6.3.1 Case study A: adverse simulation scenario . . . 123
6.3.2 Case study B: mild simulation scenario . . . 125
6.4 Remarks and conclusions . . . 130

7 Conclusions **131**
7.1 Main Conclusions 131
7.2 Future work 134

Appendix **135**

Bibliography **139**

List of Figures

1.1 Diagram of Iridium 33 and Kosmos 2251 Collision. 4
1.2 Three segments of ESA SSA programme. 4
1.3 Basic architecture of a SSA system. 5

2.1 One-to-one correspondence among the coefficient index k, multi-index vector $\boldsymbol{\gamma} = (\gamma_1, \gamma_2)$, and basis element $\boldsymbol{\beta}_{\boldsymbol{\gamma}}(\boldsymbol{x})$ in JTTNM-3 and JTCNM-3 implementations. 23
2.2 Absolute errors of both JTTNM-10 and JTCNM-10 for approximating basic functions. 35
2.3 Maximum absolute errors of both JTTNM-n and JTCNM-n vs. the expansion order n for approximating basic functions. 35
2.4 Maximum absolute errors of both JTTNM-n and JTCNM-n vs. the expansion order n for approximating a complicated univariate function $\frac{e^{\sin x}}{4+\cos x}$. 36
2.5 Absolute errors of both JTTNM-10 and JTCNM-10 for approximating a complicated univariate function $\frac{e^{\sin x}}{4+\cos x}$. 37
2.6 Results of JTTNM-1, JTTNM-2, JTTNM-8, and classical MC methods for propagating an initial state uncertainty circle with radius $\delta\check{r} = 0.005$. . . . 46
2.7 Results of JTCNM-1, JTCNM-2, JTCNM-8, and classical MC methods for propagating an initial state uncertainty circle with radius $\delta\check{r} = 0.005$. . . . 47
2.8 RMSEs in dimensionless position and velocity computed by the JTTNM-n method versus the expansion order n. 47
2.9 RMSEs in dimensionless position and velocity computed by the JTCNM-n method versus the expansion order n. 48
2.10 Ratio of CPU time versus the number of samples, indicating the computational efficiency of the JTTNM-n relative to a classical MC simulation. . . 49
2.11 Ratio of CPU time versus the number of samples, indicating the computational efficiency of the JTCNM-n relative to a classical MC simulation. . . 50

3.1 Optimal expansion orders for approximating four dominant GEO perturbations vs. the size of the position uncertainty domain. 61
3.2 MAEs in position and velocity vs. the expansion order for propagating an initial set $\Omega_r : u_r(0) = (u_x, u_y, u_z) = (50, 50, 50)$ km using both JTTNM-n and JTCNM-n for 2 days. 63
3.3 Position error distribution of both JTTNM-3 and JTCNM-3 for propagating an initial set $\Omega_r : u_r(0) = (u_x, u_y, u_z) = (50, 50, 50)$ km for 2 days. . . 64

3.4 Velocity error distribution of both JTTNM-3 and JTCNM-3 for propagating an initial set $\Omega_r : u_r(0) = (u_x, u_y, u_z) = (50, 50, 50)$ km for 2 days. . . 65
3.5 The CPU time vs. the expansion order for both JTTNM-n and JTCNM-n. 66
3.6 Position error distribution of the ADS-based JTTNM-3 and normal JTCNM-3 for propagating an initial set $\Omega_r : u_r(0) = (u_x, u_y, u_z) = (50, 50, 50)$ km for 2 days. 66
3.7 Velocity error distribution of the ADS-based JTTNM-3 and normal JTCNM-3 for propagating an initial set $\Omega_r : u_r(0) = (u_x, u_y, u_z) = (50, 50, 50)$ km for 2 days. 67
3.8 MAEs in position and velocity vs. the expansion order for propagating an initial set $\Omega_r : u_r(0) = (u_x, u_y, u_z) = (50, 50, 50)$ km, $u_v(0) = (u_{v_x}, u_{v_y}, u_{v_z}) = (5, 5, 5)$ m/s using three different coordinate representations. 68
3.9 The CPU time vs. the expansion order for the JTTNM-n implementation of three different models. 69

4.1 Case A: Profiles of estimation errors in position, velocity and area-to-mass ratio with a measurement frequency 7 times/night. Implemented in the GEO representation. 80
4.2 Case A: Profiles of estimation errors in position, velocity and area-to-mass ratio with a measurement frequency 7 times/night. Implemented in the Cartesian representation. 81
4.3 Case A: Sensitivity analysis relative to the observational geometry with a measurement frequency of 7 times/night. Implemented in the GEO representation. 83
4.4 Case A: Sensitivity analysis relative to initial state deviations with a measurement frequency of 7 times/night. Implemented in the GEO representation. 85
4.5 Case A: Sensitivity analysis relative to initial state deviations with a measurement frequency of 7 times/night. Implemented in the GEO representation and incorporating the relative distance information. 86
4.6 Case A: Profiles of estimation errors in position, velocity and area-to-mass ratio with a measurement frequency 14 times/night. Implemented in the GEO representation. 87
4.7 Case A: Profiles of estimation errors in position, velocity and area-to-mass ratio with a measurement frequency 21 times/night. Implemented in GEO representation. 87
4.8 Case B: Profiles of estimation errors in position, velocity and height of the ground tracking station with a measurement frequency 7 times/night. Implemented in the GEO representation. 89

5.1 Schematic representation of the GEO-GEO observation configuration. . . 100
5.2 Visible tracklets of the GEO-GEO observation configuration. 101
5.3 Profiles of estimation errors in position and velocity with measurement acquisition period 1 hour and measurement noise 0.64 arc-sec in the GEO-GEO observation configuration. 102

5.4 Accuracy profiles of 25 MC runs with the worst estimation errors obtained by JT-HEKF-1 and JT-HEKF-2 implementations. Measurement acquisition period 1 hour and measurement noise 0.64 arc-sec. 103
5.5 Innovation and innovation standard deviation bounds of the JT-HEKF-n implemented at orders $n = 1$ (top), 2 (middle), and 3 (bottom). 103
5.6 Schematic representation of the MEO-GEO observation configuration. . . 106
5.7 Visible tracklets of the MEO-GEO observation configuration. 107
5.8 Profiles of estimation errors in position and velocity with measurement acquisition period of 1 hour and measurement noise 0.2 arc-sec in the MEO-GEO observing configuration. 107
5.9 Profiles of estimation errors in position and velocity computed by the JT-HEKF-2, JT-DHEKF-2, JT-SHEKF-2 and JT-MHEKF-2 in the case of a constant measurement bias inside the time interval $t \in [70.8, 76.8]$ h. . . . 109
5.10 10 MC runs with the worst estimation accuracy for the JT-HEKF-2, JT-DHEKF-2, JT-SHEKF-2 and JT-MHEKF-2 in the case of a constant measurement bias inside the time interval $t \in [70.8, 76.8]$ h. 110
5.11 Position and velocity errors of the JT-HEKF-2, JT-DHEKF-2, JT-SHEKF-2 and JT-MHEKF-2 in the case of a random measurement bias inside the time interval $t \in [70.8, 76.8]$ h. 111
5.12 10 MC runs with the worst estimation accuracy for the JT-HEKF-2, JT-DHEKF-2, JT-SHEKF-2 and JT-MHEKF-2 in the case of a random measurement bias inside the time interval $t \in [70.8, 76.8]$ h. 112
5.13 Position and velocity errors for the JT-HEKF-2, JT-DHEKF-2, JT-SHEKF-2 and JT-MHEKF-2 in the case of a zero sensor output at $t = 70.8$ h. . . 113
5.14 10 MC runs with the worst estimation accuracy for the JT-HEKF-2, JT-DHEKF-2, JT-SHEKF-2 and JT-MHEKF-2 in the case of a zero sensor output at $t = 70.8$ h. 113

6.1 Comparison of position and velocity errors for case study A. 124
6.2 Comparison of standard deviations of position and velocity errors for case study A. 124
6.3 Comparison of position and velocity errors for case study A with and without an angular measurement noise at $t = 53$ h. 126
6.4 Order-switching procedures for the JT-OSHEKF algorithm with and without an angular measurement noise at $t = 53$h. 126
6.5 Sensitivity analysis relative to initial state deviations for case study A. . . 127
6.6 Comparison of position and velocity errors for case study B. 128
6.7 Comparison of the position and velocity error estimators for case study B. 128
6.8 Sensitivity analysis relative to initial state deviations for case study B. . . 129

List of Tables

2.1 RMSEs in dimensionless position and velocity computed by the JTTNM-n at propagation time $t = 0.75T$, implemented at expansion orders from $n = 1$ to 8. 48

2.2 RMSEs in dimensionless position and velocity computed by the JTCNM-n at propagation time $t = 0.75T$, implemented at expansion orders from $n = 1$ to 8. 49

3.1 Nominal initial values of spacecraft states and parameters 62

3.2 Selection of integration step-sizes. 62

3.3 Performance comparison between both JTTNM-3 and JTCNM-3. 64

3.4 Influence of the size of the initial uncertainty domain on the performances of JTTNM-3 and JTCNM-3. 65

3.5 Performance comparison between ADS-based JTTNM-3 and JTCNM-3. . 67

4.1 True initial values of spacecraft state, parameters, and the ground tracking station position 80

4.2 Computation time (s) 82

4.3 Case A: Sensitivity analysis relative to the observational geometry 83

4.4 Case A: Sensitivity analysis relative to initial state deviations 84

4.5 Case A: Sensitivity analysis relative to initial state deviations incorporating the relative distance information 85

5.1 State and area-to-mass ratio at initial epoch July 4th 2019, 03:20:00.000 UTC. 101

5.2 RMSEs in position and velocity of the JT-HEKF-n at the steady stage $t \in [96, 144]$ h and the associated CPU time. 102

5.3 Sensitivity analysis of the JT-HEKF-n with respect to initial state deviations. ($\hat{\sigma}_r$ km, $\hat{\sigma}_v$ m/s) indicates the standard deviation of the Gaussian distribution describing the initial spacecraft state vector. The unit of ${}_n\bar{\varepsilon}_\Upsilon$ and ${}_n\sigma_\Upsilon$ are kilometers and meters per second. The - symbol indicates filtering divergence. 104

5.4 Position and velocity RMSEs in the sensitivity analysis of the JT-HEKF-n with respect to the measurement noise and acquisition period. The units of ${}_n\bar{\varepsilon}_r$ and ${}_n\bar{\varepsilon}_v$ are kilometers and meters per second. 105

5.5 SDs of the position and velocity RMSEs in the sensitivity analysis of the JT-HEKF-n with respect to the measurement noise and acquisition period. The units of ${}_n\sigma_r$ and ${}_n\sigma_v$ are kilometers and meters per second. 106
5.6 State and area-to-mass ratio at initial epoch July 3rd 2019, 07:20:00.000 UTC. 107
5.7 RMSEs in position and velocity of the JT-HEKF-2, JT-DHEKF-2, JT-SHEKF-2, and JT-MHEKF-2 at the steady stage $t \in [96, 144]$ h in the case of a constant measurement bias and the associated CPU time. 109
5.8 RMSEs in position and velocity of the JT-HEKF-2, JT-DHEKF-2, JT-SHEKF-2, and JT-MHEKF-2 at the steady stage $t \in [96, 144]$ h in the case of a random measurement bias and the associated CPU time. 111
5.9 RMSEs in position and velocity of the JT-HEKF-2, JT-DHEKF-2, JT-SHEKF-2, and JT-MHEKF-2 at the steady stage $t \in [96, 144]$ h in the case of an instantaneous zero sensor output and the associated CPU time. 114

6.1 Initial states and area-to-mass ratio of GeoSat-A and GeoSat-B. 123
6.2 Case study A: RMSEs in position and velocity over $t \in [48, 144]$ h and the required CPU time. 123
6.3 Case study B: RMSEs in position and velocity over $t \in [48, 144]$ h and the required CPU time. 127

Acronyms

ADS Automatic Domain Splitting technique

CAC Cartesian Coordinates

CKF Cubature Kalman Filter

CYC Cylindrical Coordinates

DA Differential Algebra

ECEF Earth-Centered Earth-Fixed Reference Frame

ECI Earth-Centered Inertial Reference Frame

EKF Extended Kalman Filter

FPE Fokker–Planck Equation

FPE-BF FPE-based Bayesian Filter

GEO Geosynchronous Earth Orbit

GES GEO Element Set

GMM Gaussian Mixture Model

GSF Gaussian Sum Filter

HEKF High-order Extended Kalman Filter

JT Jet Transport

JTCNM Jet Transport based Chebyshev polynomial Nonlinear Mapping

JTRK JT-based Runge-Kutta method

JTTNM Jet Transport based Taylor polynomial Nonlinear Mapping

JT-AHEKF JT-based Augmented High-order Extended Kalman Filter

JT-HEKF JT-based High-order Extended Kalman Filter

JT-DHEKF false measurement-Discarding based JT-HEKF

JT-MHEKF Multiple scale factors based JT-HEKF

JT-OSHEKF Order-Switching based JT-HEKF

JT-SHEKF Single scale factor based JT-HEKF

KF Kalman Filter

LEO Low Earth Orbit

LinCov Linear Covariance

LVLH Local Vertical Local Horizontal reference frame

MAE Maximum Absolute Error

MC Monte Carlo

NEO Near-Earth Object

NIS Normalized Innovations Squared

OD Orbit Determination

ODE Ordinary Differential Eequation

OP Orbit Prediction

PC Polynomial Chaos

PCBF Polynomial Chaos-based Bayesian Filter

PDF Probability Density Function

PF Particle Filter

RA Right Ascension

RAA Radial, Azimuthal, and Axial directions

RK Runge-Kutta method

RMSE Root Mean Square Error

RSO Resident Space Object

SAR Synthetic Aperture Radar

SBO Space-Based Optical sensor

SD Standard Deviation

SRP Solar Radiation Pressure

SSA Space Situation Awareness

SSN Space Surveillance Network

SST Space Surveillance and Tracking

STM State Transition Matrix

STT State Transition Tensor

SWE Space Weather

TLE Two-Line Elements

UKF Unscented Kalman Filter

UT Unscented Transformation

1

Chapter 1
Introduction

1.1 Advanced space activities

The successful launch of the first artificial Earth satellite Sputnik-1 on 4th October 1957, triggered a completely new era of space exploration. It firstly created the needs for accurate and efficient orbit prediction, tracking and determination techniques. As of the rapid development of space technology, highly efficient and accurate orbit prediction and estimation methods for space objects, including active spacecraft and space debris, have become the most critical factor concerning whether the space mission could be successfully performed or not. Based on those techniques, over the past six decades, a large quantity of high-value but costly satellites with various practical functions have been launched and deployed into terrestrial space. As a reward, those space missions provide a wide variety of appealing space services, including communications [1,2], weather forecasts [3], remote sensing [4], global navigation and so on, which remarkably change the way we live, and the economical and scientific activities we take.

However, despite that, the human being still has imperious demands to achieve more progressive space missions and applications, such as the state-of-the-art synthetic aperture radar (SAR) onboard a geosynchronous Earth orbit (GEO) satellite [5]. The development of an unprecedented orbit prediction and determination accuracy (centimeter level in the radial direction) is requisite for GEO SAR missions, in the interest of significantly reducing the influence of orbit errors on the SAR imaging and interferometric processing [6,7]. On the other hand, the security of high-value space assets, including the aforementioned GEO SAR satellites, is being threatened by many resident space objects (RSOs), which

are inevitably produced due to the increasing human activities in space. To reduce or even eliminate the risks of space collisions, the space situational awareness (SSA) project arises at the historic moment. The main constituent part of an impeccable SSA system is the timely updating of a working catalog of the RSOs resident in the near-Earth orbits through precise observation, prediction and estimation of their physical locations. The ultimate objective of the SSA programs is to predict the collision probability, plan collision avoidance paths, protect expensive spacecraft, and ensure space sustainability [8]. In essence, a SSA surveillance activity can be cast as orbit prediction and estimation problems of some known or unknown objects of interest (i.e., active satellites or inactive debris), where only some partial information on the dynamical behavior and a series of observations produced by sensors are available. Therefore, the development of orbit prediction and estimation techniques with a high precision is of vital importance.

1.1.1 Geosynchronous earth orbit synthetic aperture radars

Due to the unique feature of the geosynchronous orbit regime for maintaining a spacecraft in a high altitude and almost fixed position relative to the Earth-centered Earth-fixed reference frame (ECEF), SAR devices mounted on the GEO spacecraft possess obvious advantages, such as enormous instantaneous coverage and short revisit time [9]. These excellent characteristics make a permanent radar monitoring of local regions of interest possible, thus hold huge potential application value on a variety of urgent missions requiring continuous imaging and fast assessment, such as the timely natural hazard monitoring and disaster alarming of volcanic activity, earthquakes and forest fire [10,11]. In addition, the observation peculiarity of the GEO SAR is also benefiting to achieve a series of appealing atmospheric and terrestrial applications [12,13], which makes up for the deficiency existing in the traditional observation mode of low Earth orbit SAR (LEO SAR) [14].

The concept of the GEO SAR was firstly introduced by Kiyo Tomiyasu [5, 15]. In 2003, Jet Propulsion Laboratory put forward a great plan in which SAR devices installed on an inclination GEO satellite were studied for enabling earthquake prediction, that in turn obtained many valuable experiment results in orbit design [16]. From then on, a lot of studies focusing on GEO SAR missions have been extensively carried out and devoted to enriching the relevant theory [17–20]. Furthermore, the quasi-geostationary SAR concept was also proposed and proved feasible for achieving GEO SAR missions [21]. A comprehensive comparison of practicable orbits, imaging performance and intended applications between the classical inclination geosynchronous SAR and quasi-geostationary SAR were given in [22].

Up to now, a variety of indispensable techniques for achieving GEO SAR missions have been studied, such as signal modelling, resolution calculation, and imaging algorithms [23]. Nevertheless, there is little literature about the related research of the orbit design, prediction and determination, as well as the influence analysis of orbital errors on the GEO SAR imaging. The lack of these work vastly limits the advance of GEO SAR missions. Current research about GEO SAR missions mainly focuses on theoretical analysis and simulation stage. In [24], the influence of orbit errors on the Doppler parameters and final

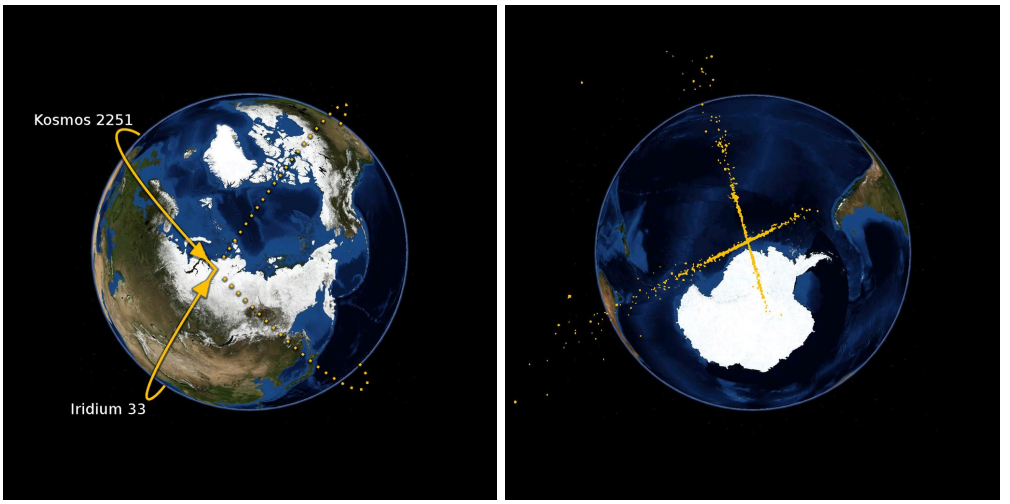
SAR imaging performance was simply analyzed. Analogously, the significant impact of the satellite orbit error on the radar interferometric phase of differential interferometric SAR was investigated and verified [25]. Subsequent studies were devoted to analyzing the effect of orbit determination errors on the GEO SAR imaging from quantitative and qualitative perspective [6, 7, 9]. It concluded that the accuracy of orbit determination approaches should be at centimeter level in the radial direction in order to satisfy the demands for future GEO SAR missions.

1.1.2 Space situational awareness mission

The space collision is extremely dangerous, not only because it always severely damages the artificial spacecraft, but also because it would create tens of thousands of debris to further deteriorates the space environment. For instance, on 10th February 2009, an operational United States' communications satellite, called Iridium 33, was struck and destroyed by a long-defunct Russian communications satellite, called Kosmos 2251, in a low Earth orbit [26], as shown in Fig. 1.1. It has been observed that this severe collision produced over 2500 pieces of debris (larger than 10 centimeters in diameter) [27], which often experience very slow natural reentry and likely remain in the low Earth orbits for tens of years. The serious trouble is that these debris, in turn, would significantly threaten the safety of satellites operating in the nearby orbit region for a long time duration. Moreover, the rapid accumulation of debris very easily results in the soon reach of a critical number threshold of the RSOs, where an incidental collision would cause a runaway exponential increase of space collisions. This accidental space collision occurred completely due to the space congestion and lack of accurate orbit estimation and prediction information of both satellites. As a fundamental foundation, the development of accurate orbit prediction and determination techniques are extremely urgent and useful.

As a solution to understand and maintain awareness of the RSOs' population in the Earth orbits and further identify the collision risks to existing missions, the concept of the SSA comes into being and obtains an increasing attention from a great number of experienced space agencies. Based on the efficient tracking and identification of the RSOs, the SSA project is expected to obtain a comprehensive knowledge of the near-Earth space environment with the purpose of achieving various functions, including anomaly detection, tracking and data association, conjunction analysis, probability of collision calculation, and sensor resource management [28]. The main tasks in SSA missions contain: 1) efficient tracking and identification of the RSOs; 2) precise orbit prediction and estimation; 3) timely maintaining and updating a catalog of the RSOs; 4) predict space events and maintain a collision-free space environment. Note that the accurate and complete position and velocity information of the RSOs, provided by a SSA system, is the fundamental to compute the probability of collision events, such that the collision-free space environment can be maintained via performing effective evasive maneuvers if necessary.

At present, some SSA programs have been put forward and acknowledged. The well-known one was proposed in 2009 by European Space Agency, which consists of three



(a) Point of collision **(b)** Debris fields after 50 minutes

Figure 1.1: Diagram of Iridium 33 and Kosmos 2251 Collision.

segments [29], including Space Weather (SWE) system, Near-Earth Objects (NEO) system, and Space Surveillance and Tracking (SST) system, as shown in Fig. 1.2. Another distinguished SSA system is Space Surveillance Network (SSN) operated by United States Strategic Command Joint Space Operations Center, which is the first operational SSA system and currently tracks around 21,000 objects with diameters greater than 10 centimeters [30].

(a) SWE **(b)** NEO **(c)** SST

Figure 1.2: Three segments of ESA SSA programme.

Figure 1.3 illustrates a classical architecture of a SSA system. The orbit determination (OD) module, typically running in a Bayesian framework, contains both orbit prediction (OP) and update steps. The OP module is used to predict the orbits and associated uncertainty through the dynamics of the RSOs, while the observation module is employed to sequentially observe the RSOs of interest and provides useful range or angular information to the filters for updating the predicted orbits. Therefore, the accuracy of the OD algorithm essentially depends on prediction errors involved in OP methods and the observation errors induced by measurement sensors. It is worth mentioning that the accurate OD module is a major component for updating and maintaining a catalog of the

RSOs. Besides, the timely update of the catalog of the RSOs in turn enables to provide a priori state information as inputs to both orbit prediction and observation modules. The space weather data monitoring module is requisite for measuring the space physical parameters, such as the atmospheric density, solar photon intensity and so on, which always remarkably affects the dynamics of the RSOs.

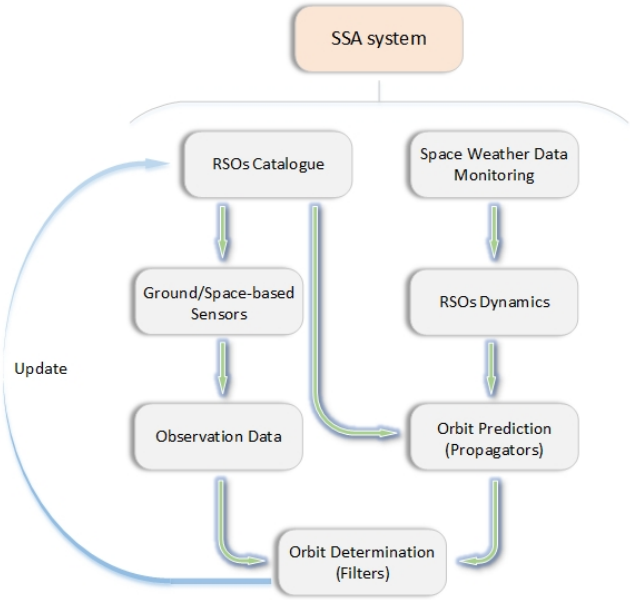


Figure 1.3: Basic architecture of a SSA system.

Finally, it is worth to mention that a key issue of SSA missions is to settle the measurement data-starved problem. As the space launch activities increase, the number of the RSOs are likely to be much larger than that of the ground or space observation stations for actively tracking them, such that it is almost impossible to provide high frequency observations for any single RSO. That is, OD algorithms inevitably have to employ increasingly sparse observations. It further requires that the adopted OP method possesses high prediction accuracy in between measurements over long time intervals. As a conclusion, one topic of recent interest in SSA programs is to design both OP and OD modules in a highly efficient and accurate way to predict and estimate the orbits, even when encounter high nonlinearity, sparse observations or large initial state deviations. Another topic of interest lies in how to achieve the trade-off between the accuracy and computational cost for designate tasks.

1.2 Orbit uncertainty propagation methods

The orbit uncertainty propagation concept refers to the prediction of the state probability density function (PDF) or state moments of the RSOs (generally mean and covariance ma-

trix). The commonly used propagation methods often employ the physics-based analytical high-fidelity models, described by a set of ordinary differential equations (ODEs) with consideration of several dominant orbital perturbations, including Earth non-spherical, luni-solar gravitational forces, solar radiation pressure (SRP), and atmospheric drag [31]. In general, uncertainties existing in the initial state vectors and model parameters stem from the lack of accurate orbital observation data, space environmental models or actual physical characteristics of the RSOs. Except these factors, uncertainties can also arise from the necessary simplifications of the high-fidelity models and truncation errors in the numerical simulations [32].

The main purpose of orbit uncertainty propagation methods is to accurately transport the uncertainties associated with the initial states and parameters of a RSO to future possible states. Therefore, it clearly plays an important role in the ongoing space projects such as the aforementioned SSA and GEO SAR missions. For instance, in SSA missions, a variety of orbit prediction methods have been extensively studied and applied to perform a series of vital tasks, including but not limited to space tracking and data association, collision detection and sensor resource management [28]. On the other hand, since the state prediction step of Bayesian filters relates to the state propagation technique, therefore, the filtering accuracy heavily rides on the error of the adopted state propagation method. Consequently, for space OD problems, highly precise state propagation approaches are badly requisite to transport the state PDF in between measurement updates. Homoplastically, in GEO SAR missions, nonlinear Bayesian filters are expected to accurately determine the spacecraft orbit, calculate the interferometer baseline, and further distinctly decrease the influence of orbit errors on the GEO SAR imaging quality.

Up to now, a variety of orbit uncertainty propagation methods proposed in the literature usually can be divided into three categories: analytical, numerical and semi-analytical [33, 34]. Ideally, using analytical integration methods, a closed-form analytical solution of an ODE system can be represented as an explicit function of time, initial state and model parameters [35–37]. Clearly, it is preferred since the complicated state uncertainty propagation along the stochastic, nonlinear ODEs is reduced to the simple evaluation of explicitly known mapping functions, which makes the orbit uncertainty propagation very efficient. In particular, the analytical solution is usually valid for all or at least large ranges of initial conditions. However, except in some special cases, it is hard to analytically solve an ODE system describing a real physical procedure due to its complicated nonlinearity. As an exception, under the Kepler assumption, the motion of the RSOs described by a set of ODE equations can be analytically solved, but it significantly loses the accuracy resulting from the omission of all orbital perturbations [31].

To consider the influence of orbital perturbations, semi-analytical method is another elegant technique to propagate the orbit uncertainties. The effects of dominant orbital perturbations separate the constant, short periodic and long-periodic terms [38]. With the direct usage of the averaging technique on the variational equations or Hamiltonian formulation of the dynamics [39–41], the short periodic effects, which normally have small amplitudes, are filtered analytically. Therefore, the obtained averaged equations only include long periodic perturbation effects and can be integrated numerically with very large step sizes, making semi-analytical methods more computational efficient. The

main shortcoming is that the derivation procedure of the averaged equations is often complicated and time-consuming with the consideration of specific orbital perturbations. Another type of semi-analytical methods employ first order polynomials to approximate the vector fields governing the motion of the RSOs, then carry out analytical integration of the simplified models. Clearly, this semi-analytical solution is only an approximation of the actual system, thus again encounters the "accuracy loss" issue.

As a useful alternative to analytical and semi-analytical approaches, numerical methods completely consider the nonlinear perturbation vector fields, which strongly benefits the accuracy of the state propagation problem. Consequently, it enables to numerically solve ODE systems within an acceptable precision. However, accurate calculation of the intricate nonlinear vector fields requires small time-steps adopted in the numerical integration procedures, which significantly reduces the state propagation speed. In particular, it is a huge disaster to numerical methods when a great amount of the repetitive computation over the same time interval are carried out to investigate state statistical characteristics, such as Monte Carlo (MC) simulations.

For the sake of comparison, the most well-known MC simulations, linearized models, and nonlinear propagation methods are reviewed in what follows. Note that two key indicators, i.e., computational accuracy and efficiency, are employed to describe the performance of these orbit uncertainty propagation methods. The computational accuracy is often represented by the degree of closeness of the propagated state to its true value, while the computational efficiency is frequently addressed by the amount of computational burden.

1.2.1 Monte Carlo simulations

As an intuitive and comprehensible uncertainty propagation method, a MC simulation enables to completely describe the statistical properties of an uncertainty propagation process through the implementation of particle-type studies. Therefore, it naturally possesses two features: 1) it enables to propagate nonlinear and non-Gaussian distributions; 2) when the number of sampling points approaches to infinity, its result perfectly approximates to the true probability distribution. With the consideration of these two merits, for nonlinear orbit uncertainty propagation problems, the MC simulation is frequently employed to predict future trajectory statistics. Besides, due to the high propagation accuracy, the MC result is usually cast as the reference value to validate the accuracy of other linear or nonlinear methods.

In the existing literature, the MC method has been extensively adopted to propagate the orbit uncertainties in many applications. In [42, 43], the MC method was used to propagate the non-Gaussian characteristics of orbit uncertainties in different coordinate reference frames. Besides, a similar method was further explored to accurately transport the trajectory distributions under thrust errors for non-impulsive orbital transfer missions [44]. Ghrist et al. also adopted MC simulations to investigate the effect of non-Gaussian error volumes on collision probability computation [45]. However, if the

dynamics in question is a high dimensional system, the MC method has to adopt a great number of sampled trajectories for realizing good convergent statistics, which clearly results in very heavy computational burden [46]. In other words, the MC method is possible to encounter the so-called "curse of dimensionality" issue when it is adopted to handle high dimensional propagation problems. One key topic of interest about the variant of MC methods is to improve the computational efficiency.

An alternative approach is to create a minimal particle representation of the transient state uncertainty distributions in order to achieve the identical performance to that of traditional MC simulations [47]. An elegant solution is to dynamically and adaptively adjust the number of particles with a single run of the MC simulation in order to balance the computational accuracy and burden [48]. To speed up the implementation of MC simulations, two advanced techniques based on line sampling and subset simulation were put forward [49, 50]. Another alternative to decrease the computational burden of MC simulations is to employ the parallel manner of modern computer, such as the usage of graphics processing units [51, 52].

1.2.2 Local linearized methods

To obtain accurate non-Gaussian statistical characteristics of orbit uncertainty propagation, MC simulations often consume a heavy computational burden with the increasing dimensions of considered problems, or even encounter the "curse of dimensionality" issue. However, not all space applications require such high trajectory propagation accuracy. On the contrary, in some specific situations, the computational efficiency is also as vital important as the required accuracy requirements.

Instead of the well-known MC simulations, the linearized mapping technique based on state transition matrix (STM) is employed to simplify the dynamical models and accelerate the propagation of state uncertainties [31, 53]. In general, linearized models are capable of yielding analytical solutions under the linearization assumption, thus they usually possess high computational efficiency. However, they are only adopted when the following two assumptions hold: 1) the local motion relative to a nominal trajectory is sufficiently approximated by a first order expansion model; 2) the uncertainty distribution can be assumed as Gaussian. Otherwise, it possibly fails to describe the characteristics of state uncertainties, especially for highly nonlinear dynamical systems or long-duration propagation intervals, since the omission of higher order terms always results in the significant approximation errors. Therefore, linearized models can be only applied in the linear or moderately nonlinear systems.

Based on the STM, Battin put forward a local linearization method, named linear covariance (LinCov) analysis, to quantitatively study the impacts of navigation estimation errors on the trajectory dispersion [54]. Afterwards, Maybeck presented another LinCov formulation describing the state error covariance in a general manner [46]. Based on Battin's and Maybeck's work, Geller adopted the LinCov technique to predict the trajec-

tory dispersion under navigation errors and impulsive maneuver variations, and further proposed a new trajectory control and navigation analysis method [55]. Due to the high computational efficiency, the LinCov technique was extensively adopted in practical space applications, such as the orbital rendezvous problem [56], lunar descent landing problem [57,58], attitude estimation and control problem [59] and spacecraft atmospheric entry problem [60]. In addition, the LinCov technique can be also employed to study the influences of dynamical modelling errors [61], actuator errors [62] and sensor errors [63] on the trajectory dispersion and estimation accuracy.

1.2.3 Nonlinear uncertainty propagation methods

Generally, the nonlinearity degree of a physical system is not an attribute intrinsic to the system itself and often can be alleviated through a reasonable mathematical description, such as the choice of coordinates, regularization and averaging methods [64]. Even so, the dynamics of the Earth-orbiting spacecraft has been verified to be highly nonlinear (in both Kepler case and perturbed case) [65]. Therefore, the propagation behavior of an orbit family over a long time interval, resulting from a large initial state uncertainty, cannot be accurately approximated by a linearized model. In other words, the accuracy of the orbit uncertainty propagation often drops off due to the inconsistency between the Gaussian distribution assumption and actual statistics [66]. On the other hand, the commonly used MC approach is high-precision and easy to implement orbit uncertainty propagation, but it is very computationally expensive to obtain the convergent statistics [52]. Consequently, neither of them enables to reach a superior trade-off between the computational accuracy and efficiency. This sharp contradiction inspires a lot of researchers to study computationally efficient approaches for accurately capturing the nonlinearity of the dynamics. In addition, the stability and chaotic issues also should be carefully considered in the orbital propagation over a long duration, which can be effectively discussed through calculating the future behaviors of many neighboring orbits or computing indicators of chaos [67, 68].

In [42, 64], Junkins et al. first discussed the nonlinear transformation method to capture non-Gaussian characteristics of the orbit uncertainty propagation. Scheeres et al. deduced a new formulation of the constraint on covariance matrices that arises from topological considerations through a series of observation on fundamental constraints existing on the propagation of orbit uncertainties [69, 70]. From then on, the study of the nonlinear uncertainty propagation has received an increasing attention. A variety of analytical and semi-analytical approaches were presented, which contain three types: 1) sample-based unscented transformation (UT) and polynomial chaos (PC) expansions, 2) dynamics-based state transition tensors (STTs) and differential algebra (DA), 3) PDF-based Gaussian mixture model (GMM) and Fokker–Planck equations' solution (FPE). These methods effectually improve the weakness of the MC method and linearized model.

Both UT and PC methods are based on a straightforward and basic idea, that is, for complicated nonlinear dynamical systems, it may be easier and faster to approximate the state probability distribution than to approximate the nonlinear mapping process.

The probability distribution at a future epoch can be approximated by full numerical integration of a few samples chosen from the initial state distribution. To address the imperfection of linearization, the UT method, proposed by Julier et al. [71–73], utilizes $2n+1$ deterministic sigma-points to capture the system nonlinearity and delivers a second order approximation of the first two moment of the propagated statistical distributions. That is, only means and covariance matrices of state uncertainty distributions can be accurately mapped, which results in the fact that the UT method is usually appropriate for uncertainty propagation of moderately nonlinear systems. For strong nonlinear systems, the UT method is often inaccurate, or even ineffectual since the higher order terms could not be mapped.

Instead of using deterministic sampling points, the non-intrusive uncertainty propagation method based on PC expansions utilizes a few random samples to accurately approximate the solution to a stochastic ODE system [74]. It is worth highlighting that no simplification of sophisticated nonlinear dynamical models is taken in the implementation of the PC expansion procedure. Therefore, the PC method enables to elegantly approximate any finite-variance and possibly non-Gaussian solution. Compared with the UT method, it has been verified that the PC expansion technique is capable of extracting more nonlinear information from the perturbed orbital dynamics and delivering high order moments of the mapped state statistical distributions [75]. Compared with the MC method, the PC expansion technique possesses exponential, mean-squares convergence rate relative to the expansion order of the polynomial basis even in the mapping of non-Gaussian uncertainty distributions. However, although the PC method alleviates the computational cost relative to the MC simulation, it is still likely to encounter the so-called "curse of dimensionality" phenomenon [76]. To reduce the overall computational burden and tackle the issue of dimensionality, several elegant variants of the PC method were put forward [76–78].

Apart from the aforementioned sample-based nonlinear uncertainty propagation methods, two novel dynamics-based propagators were discussed. Park and Scheeres proposed a semi-analytic method, i.e., STTs, for addressing the localized nonlinear motion relative to the nominal trajectory and analytically mapping the state uncertainty [79]. It has been proved that the result computed by STTs affords good agreement with that calculated by the MC method. Besides, the STTs method enables to accurately and efficiently propagate both Gaussian and non-Gaussian statistical distributions. Fujimoto et al. further formulated a precise and consistent polynomial representation to transport orbit uncertainties in both conservative and non-conservative dynamical systems [80, 81]. Nevertheless, manual computation of high order derivatives of the governing dynamics is very tedious, which limits its application in many cases, especially for high-dimensional or high-fidelity dynamical systems. To address this disadvantage, several techniques such as differential algebra (DA) [82], jet transport (JT) [83], automatic differentiation [84, 85] and modified Picard integrator [86] were put forward in succession to achieve fast and automatic computations of the STTs within a computerized environment.

The first DA formalization was done by Berz and Makino for the study of particle beam accelerators, and implemented in their COSY Infinity package [82]. Subsequently, the DA technique was extensively studied and applied into the transport of orbit uncertain-

ties [33, 87–89]. In 2009, another well-known polynomial algebra implementation named JT technique was put forward to automatically compute the STTs of the flow of an ODE system [83]. Since then, it has been widely employed to propagate the statistics of initial orbit uncertainties to a future epoch [90]. It is worth mentioning that the accuracy of the STTs method or its DA and JT implementations often dramatically decreases as the magnitude of orbit uncertainties increases because of the natural properties of Taylor series expansions. To accurately deliver large sets of orbit uncertainties, the automatic domain splitting (ADS) technique was applied to both DA and JT implementations [91, 92]. Furthermore, with the usage of polynomial-based integration, semi-analytical DA and JT solutions of an ODE system can be further adopted to develop an advanced MC method. Compared with the conventional MC method, it significantly accelerates the state propagation of sampling particles and reduce the computational load [93].

The evident disadvantage of the STTs or its DA and JT implementations is that these methods require the propagated dynamical models to be continuous and differentiable. In general, they are not suitable for problems with the consideration of solar radiation pressure and atmospheric drag perturbations due to the entry and exit of the Earth shadow or non-differentiable model of the atmospheric density. Instead of Taylor series expansions, the non-intrusive Chebyshev polynomial interpolation technique has been investigated to address this issue in the implementations of DA and JT procedures [94,95].

In addition to the aforementioned nonlinear uncertainty propagators, the exact evolution of non-Gaussian distributions can be described by the associated PDF through the solution of FPE equations. Kumar et al. proposed a series of useful methods to solve the FPEs [96–99]. Nevertheless, these methods are inadaptable to solve a high dimensional FPE due to the huge computational complexity. For instance, the FPE describing an orbital problem with six variables is almost impossible to be solved due to the unbearable computational load, even using super computing platforms. Until 2016, the direct solution of the transient FPE for the perturbed two-body problem was just for the first time obtained in a personal computer with the usage of the tensor decomposition [100]. However, it is still a huge challenge to solve the FPE for higher dimensional dynamical systems. An alternative method for approximating the time evolution of the PDF, associated with a high dimensional system, was to adopt the GMM model [101, 102]. It is worth to mention that the accuracy of the GMM method fully depends on the number of Gaussian mixtures. To some extent, the GMM method enables to alleviate the computational efficiency issue. However, if the number of the adopted Gaussian mixtures is too many, the GMM method would encounter the "curse of dimensionality" issue; otherwise, its accuracy maybe deteriorates. To improve the computational efficiency, the adaptive GMM method was further studied [103–105], however, the recomputation of the weights of Gaussian mixture components is also computational intensive.

Except the aforementioned methods, the combination of linear and nonlinear uncertainty propagators were occasionally studied [106, 107]. The original inspiration of the hybrid method arises from the absorption of all individual advantages. For instance, a new propagator was proposed to express the closed-form PDF by combining both GMM and STTs methods [106], which inherits the superior accuracy of the GMM method and the high computational efficiency of the STTs method if it is implemented by the specialized

technique, such as JT or DA. Furthermore, with the development of artificial intelligence technology, several machine learning approaches were discussed to improve the accuracy of orbit propagation problems [108, 109]. The historical orbit data of a RSO can be employed to train the adopted artificial neural network, such that the trained artificial neural network can be in turn used to propagate orbit uncertainties in nonlinear space environments [110, 111].

1.3 Sequential filtering techniques

This section introduces the state-of-the-art sequential filtering techniques developed in the Bayesian framework for accurately estimating the state vector of a RSO. In general, the Bayesian filtering technique always consists of both state prediction and updating procedures, where the former step is used to transport the state vector and its associated PDF in between measurements, while the latter one is employed to correct the a priori prediction of the state vector by incorporating actual measurements and further obtains the a posteriori estimated state vector. Therefore, the accuracy of a filter strongly rides on the magnitudes of error statistical properties output from both prediction and measurement processes.

It is worth underlying that state uncertainty propagators discussed in section 1.2 relate to the prediction step of a Bayesian filter, such that they can be directly employed to construct different filters. Based on the types of adopted state uncertainty propagation approaches, a variety of relevant filtering techniques have been proposed. We review mostly used Bayesian filters in what follows, as well as their underlying pros and cons. Note that, for the solution of space-related OD problems, an elegant sequential filtering algorithm should be accurate, robust and low computational burden.

1.3.1 Linear Kalman filters

The well-known linear Kalman filter (KF) was first proposed in 1960 by Rudolf E. Kalman [112, 113], who opened up a completely new field in the estimation theory. Theoretically the KF is an optimal mean square error filter for linear systems, in which both process and measurement noises are assumed to be zero-mean Gaussian distributions with known covariance matrices [112–114]. In actual estimation problems, standard deviations of both process and measurement noises are not easy to precisely determine, such that the accuracy of the KF is likely to reduce. In such a case, a robust alternative is to minimize the worst-case estimation error rather than its variance. Based on this idea, an improved linear filter, called H-∞ filter, was proposed to enhance the robustness of the KF for time-varying systems or the cases with inaccurate or unknown process and measurement noises [115, 116]. For more detailed information about linear filters, an outstanding description was given in [117, 118]. It is worth emphasizing that these linear filters were broadly adopted in many early space applications due to the simplicity and effectiveness.

1.3.2 Nonlinear sequential filters

The motion of Earth-orbiting RSOs subjected to the central gravity and perturbations is extremely nonlinear and time-varying. Meanwhile, the measurement models, describing the information of range, range rate, angle, angular rate and so on, are always nonlinear functions of position and velocity vectors of the RSOs. With the consideration of nonlinear behaviors, linear KFs are no longer adequate to precisely describe the state statistics (i.e., mean and covariance matrix). Many research attempts to improve the estimation accuracy of common KF filters within limiting the computational burden, including extended Kalman filter (EKF) [112, 119], unscented Kalman filter (UKF) [73], cubature Kalman filter (CKF) [120, 121], PC-based Bayesian filter (PCBF) [122], high order extended Kalman filter (HEKF) [123–125], Gaussian sum filter (GSF) [126], FPE-based Bayesian filter (FPE-BF) [127] and so on.

For moderately nonlinear systems, the dynamical and measurement equations most often can be approximated by linearized models. One of the most renowned nonlinear variants of the KF, called EKF, was firstly put forward based on the linearization assumption and widely employed to solve nonlinear state estimation problems [31, 112, 119, 128]. The straightforward rationale of the EKF is to achieve the a priori state prediction using first order Taylor expansions of nonlinear systems, then, produce the a posterior estimate of the state vector through the incorporation with the new measurements at each time sample. The prediction error between the adopted linearized approximation model and the actual nonlinear dynamics dramatically depends on the local slope of the dynamics. To guarantee that the linearized model is sufficiently accurate to the actual dynamical system, high measurement frequency and small initial state uncertainties are requisite such that the partial derivative of the dynamics relative to both the time and state remains almost fixed [129]. In such condition, the approximation error generally can be controlled within a relatively small scope. Otherwise, if the EKF encounters poor conditions, for instance in systems with low frequency measurements or when only bad initial state guesses are available, the linearization-based approximation may be inefficient, thus results in the loss of estimation accuracy, or even divergence [130]. In other words, large initial state deviations, sparse measurements available, as well as their interactions remarkably amplify the effect of system nonlinearity, making the EKF inappropriate for the nonlinear state estimation. Note that if nonlinear systems are not differentiable, the EKF is very likely to suffer from a large loss of accuracy in the estimation of mean and covariance matrix due to the derivative-based structure. On the other hand, the EKF is difficult to implement and tune since the cumbersome derivation of the Jacobian matrices always depends on the considered dynamical models.

In order to improve the shortcomings of the EKF, two alternatives, including UKF and CKF, were proposed based on the direct representation of probability distributions approximated by the full numerical integration of specially selected sampling points. The UKF adopts a UT transformation to transport state Gaussian distributions with the usage of $2n+1$ deterministically chosen sigma points [72, 131–133]. Analogously, based on a third-degree spherical-radical cubature rule, the CKF adopts $2n$ so-called cubature points to numerically compute Gaussian-weighted multi-dimensional integrals [134]. Compared

with the CKF, the UKF contains one more central sampling point of the state Gaussian distribution, which often possesses more weight. However, for systems of dimension more than 3, the weight of the central sampling point is commonly negative, which always results in the failure of the UKF due to the non-positive semi-definite covariance matrix. In other words, the CKF has the better numerical stability property than the UKF when being employed to solve high-dimensional estimation problems [135]. Although these two filters are designed to solve the nonlinear estimation problem, they still assume the propagated state distribution being Gaussian. Another sample-based filter, called PCBF, was proposed to provide non-Gaussian solutions by abolishing the Gaussian error distribution assumption [122]. It is worth mentioning that, compared with the EKF, the aforementioned three sample-based filters, i.e., UKF, CKF and PCBF, do not require the estimated model being differentiable. On the other hand, except the PCBF, both UKF and CKF enable to provide a higher accuracy at the cost of a slight increment on the computational complexity (although generally, on the same order of magnitude) [136, 137], while the computational cost of the PCBF increases exponentially with the state dimensionality.

From another point of view, along with the development of the STTs concept, the common EKF has been generalized into the so-called HEKF [123–125]. These filters enable to extract more nonlinear information from dynamical and measurement models, thus provide a much better performance, specially in the case of low measurement frequencies and large initial state deviations. However, the manual derivation of high order STTs not only requires nonlinear systems to be continuous and differentiable, but also causes the computational complexity in high-fidelity and/or high-dimensional systems. In other words, the bottleneck of the HEKF lies in the accurate and efficient computation of the required high order STTs. With the development of computing hardware performance and new automatic differentiation and integration techniques, such as DA [138, 139] and JT [83], the computational efficiency of the STTs has been remarkably improved, making the HEKF possible into the practical application nowadays. However, the antagonism between the estimation accuracy and the computational burden still persists in the HEKF implementation and, in general, with the increase of the expansion order, the estimation error decreases, but the computational cost increases significantly.

Besides, some specialized filters, such as GSF [126] and FPE-BF [127], that cater to various classes of nonlinear estimation problems of interest, have been also put forward along with the development of other nonlinear uncertainty propagators, i.e., GMM, FPE and tensor decomposition. However, these filters always inherit the computational inefficiency from the adopted state propagation approaches, such that they are rarely used in actual space-related missions.

1.3.3 Particle filter

Particle filter (PF) is another well-known nonlinear recursive Bayesian filter, which was firstly proposed by Gordon et al. in 1993 with the name of bootstrap filter [140]. The basic rationale underlying the PF is to represent the state PDF in both prediction and updating procedures as a weighted set of random samples [141]. Clearly, it provides an

alternative approach to accurately estimate the nonlinear states and parameters occupying non-Gaussian uncertainty distributions [142, 143]. The PF uses a MC simulation to approximate the state PDF after each measurement updating, which is called "particle evolution". Clearly, the PF has no restriction on the differentiability of the systems. Note that, to guarantee the approximation accuracy, the required number of particles has to increase exponentially with the system dimensionality. Hence, the PF always encounters "curse of dimensionality" phenomenon, which makes it hard to apply in real-time estimation missions [144]. To improve the computational efficiency, a JT-based MC propagator was adopted to develop a more excellent filter called Jet Transport particle filter (JTPF) [145]. Instead of the repetitive numerical integration adopted by the classical PF, the JTPF applies the JT technology to achieve the particle evolution process, which contains two steps: 1) Flow expansion: the vector fields of the ODEs are expanded around a center particle into high order Taylor series and further integrated to obtain the approximated polynomial of the associated flow; 2) Particle evolution: substitute the initial deviation of each particle into polynomial results to yield the approximate value of each particle's evolution. Clearly, the polynomial integration in the "Flow expansion" step consumes more time than the numerical integration of one particle, but it is executed only once in between measurements. Meanwhile once the high order Taylor series expansion of the flow is obtained, the tedious repetitive numerical integration procedure for evaluating all particles can be replaced with the elegant "Particle evolution" step, i.e., faster numerical algebra operation, which enables to significantly save the computational cost. Consequently, the JTPF costs much less processing time in total than the conventional PF method, especially when the filter needs a huge quantity of particles. However, even so, compared with other concise filters, the JTPF still consumes a large amount of the computational burden. Summarizing, the PF and its variant JTPF overcome almost all intrinsic limitations of the aforementioned nonlinear filters except the heavy computational burden.

1.4 Research overview and contributions

In the discussions above, precise and efficient orbit propagation and determination techniques are requisite in current or future advanced space missions, such as the SSA and GEO SAR, especially when only bad initial state guess or/and sparse measurements are available. However, the existing propagators and estimators almost can not satisfy both accuracy and efficiency requirements simultaneously. On the other hand, it is also of importance to identify sensor malfunctions and maintain strong robustness in the orbit determination problem. Therefore, this work is mostly devoted to achieving the following intended contributions:

- Develop a polynomial algebra-based JT framework to automatically implement polynomial approximation operations of nonlinear systems, including nonlinear functions and general ODE systems.

- Based on the JT technique, nonlinear propagators using high order STTs are ex-

plored to perform the efficient and accurate orbit propagation of the RSOs in the high-fidelity space environment.

- Propose a novel JT-based high order extended Kalman filter (JT-HEKF) to accurately and efficiently solve the joint state and parameter estimation problem in space-related applications.

- Design three superior fault-tolerance strategies to improve the robustness and reliability of the JT-HEKF against sensor malfunctions.

- Put forward a more efficient statistical adaptive order-switching methodology for further improving the computational efficiency of the JT-HEKF algorithm.

Following is a brief outline of the organization of this dissertation.

- In chapter 2, a JT framework is constructed in the modern computer with the usage of C plus plus. A series of polynomial operations are defined in the JT framework, including polynomial storage, addition, subtraction, multiplication, division, differentiation, integration and composition. The tool includes as well useful algorithms, for instance, the polynomial-based Runge-Kutta integrator and polynomial evaluation. Furthermore, the JT technique provides an efficient and accurate way to calculate polynomial expansions (such as Taylor and Chebyshev series), up to an arbitrary order, for practical problems described by a set of nonlinear functions or ordinary differential equations.

- Based on automatic computation of the STTs in the proposed JT framework, Chapter 3 develops an accurate and efficient nonlinear method for propagating trajectory uncertainties in the high-fidelity space environment. Without loss of generality, an illustrative example for the orbit uncertainty propagation in the GEO regime is performed in two forms, Taylor and Chebyshev expansions, and in different coordinate representations. Except the Earth's central gravity, the considered vector field contains other four dominant perturbations: SRP, Earth's potential and luni-solar gravitational attractions. Taking into account the size of uncertainty neighborhoods as well as the expansion order of approximation polynomials and the time step of integration schemes, abundant combinations of the above factors are simulated and a series of look-up tables with recommendations on the best options to address the propagations are given.

- Along with the development of JT-based nonlinear prediction methods, Chapter 4 proposes an augmented JT-HEKF (JT-AHEKF) for simultaneously estimating the spacecraft state vector and uncertain parameters, either physically related with the spacecraft or with the measurement procedure. Two different coordinate representations, including Cartesian and hybrid geostationary orbital elements, are exploited for dealing with the OD problem in the GEO regime.

- In Chapter 5, three fault-tolerant algorithms are further put forward to suppress

measurement faults. In the light of the insensitiveness of the proposed JT-HEKFs to the measurement frequency, the first strategy directly discards the identified false measurements, being equivalent someway to the decrease of the measurement frequency. The second and third strategies employ a single scale factor and an adaptive scale matrix, respectively, as a multiplier to the measurement noise covariance matrix in order to alleviate the pollution of false measurements.

- The JT-HEKF generally encounters a dilemma between the estimation accuracy delivered and the computational burden associated. Targeting to improve this shortcoming, achieving estimations with high accuracy and computational efficiency, Chapter 6 investigates a new adaptive order-switching variant of the JT-HEKF via automatic adjustment of the order within one single filter run. At each filter step, an innovation-based function, accounting for the filter consistency, is put forward to judge the necessity of an order-switching operation.

- In Chapter 7, some useful conclusions are highlighted and possible future extension of this dissertation is introduced.

1.5 Publications and academic activities

The published articles and academic activities carried out during the thesis are summarized below:

Articles

- J. Chen, J. J. Masdemont, G. Gómez, and J. Yuan, Jet transport-based nonlinear state and parameter estimation for geostationary spacecraft, *Acta Astronautica*, 164(1), 2019.

- J. Chen, J. J. Masdemont, G. Gómez, and J. Yuan, Analysis of jet transport-based geostationary trajectory uncertainty propagation, *Journal of Guidance, Control, and Dynamics*, 43(6), 2020.

- J. Chen, J. J. Masdemont, G. Gómez, J. Yuan, and Z. Zhu, Rotation–translation coupling analysis on perturbed spacecraft relative translational motion, *Nonlinear Dynamics*, 2020.

- J. Chen, J. J. Masdemont, G. Gómez, and J. Yuan, An efficient statistical adaptive order-switching methodology for kalman filters, *Communications in Nonlinear Science and Numerical Simulation*, 93(1), 2020.

- J. Chen, J. J. Masdemont, G. Gómez, and J. Yuan, A polynomial-based autonomous orbit determination technique and its three fault-tolerant designs, *In preparation.*

- C. Ma, Z. Zheng, J. Chen, and J. Yuan, Jet transport particle filter for attitude estimation of tumbling space objects, *Aerospace Science and Technology*, 107(1), 2020.

Conferences (speaker & first author)

- J. Chen, J. J. Masdemont, G. Gómez, and J. Yuan, High accuracy state and parameter estimation of geo-stationary satellites using jet transport based nonlinear filtering algorithm, in *IAC Papers Archive*, 2018, pp. 1–12.

2

Chapter 2
Jet transport technique

This chapter is devoted to introducing a powerful tool, i.e., JT scheme. It is an expert software framework constructed on the basis of the modern computer technology. In essence, JT technique developed in this thesis is composed of polynomial expansion techniques (in Taylor or Chebyshev basis) and a series of associated algebra defined in the polynomial space. The polynomial approximation technique employs Taylor or Chebyshev polynomials to fit basic nonlinear functions up to an arbitrary order, such as exponentials, powers, logarithms, and trigonometric functions. The specific algebra trivially defines not only polynomial addition, subtraction, multiplication and division operations, but also some useful algorithms, for instance, complicated polynomial composition, polynomial-based integrators and advanced MC simulators. In what follows, all these polynomial algebraic operations are introduced in order to build a tailored JT framework for carrying out the polynomial-based semi-analytical method in an automatic, accurate and efficient way.

Note that the core concept of JT technique focuses on how to achieve the automatic Taylor and Chebyshev polynomial approximation representation of a given nonlinear procedure (i.e., a function mapping or an ODE system) by the modern computer technology, thus it is also known as *differential algebra*, *automatic differentiation* or *polynomial algebra*. For brevity, the JT-based n-th order Taylor and Chebyshev approximations of a nonlinear procedure are respectively referred to as JTTNM-n and JTCNM-n methods in the following discussions.

2.1 Polynomial storage

Theoretically, a polynomial $P_{n,d}(\boldsymbol{\beta}(\boldsymbol{x}))$, in d variables of an infinite order n, enables to adequately address the full information of a given nonlinear procedure $f(\boldsymbol{x})$ over a bounded domain $\Omega = [\boldsymbol{x}_l, \boldsymbol{x}_u] \in \mathbb{R}^d$, where $\boldsymbol{x}_l \in \mathbb{R}^d$ and $\boldsymbol{x}_u \in \mathbb{R}^d$ jointly confine the range of interest, and the polynomial basis of choice $\boldsymbol{\beta}(\boldsymbol{x})$ is a function of the state vector $\boldsymbol{x} \in \mathbb{R}^d$. However, in practical JT implementations, it is impossible and superfluous to store all terms of designated polynomials in the modern computer due to an enormous demand of computer's internal storage and CPU time. As a consequence, a truncated polynomial $P_{n,d}(\boldsymbol{\beta}(\boldsymbol{x}))$ in a finite expansion order n is preferred to approximate a nonlinear function $f(\boldsymbol{x})$ within an acceptable error tolerance $R_n(\boldsymbol{x})$ in the JT framework, that is

$$f(\boldsymbol{x}) = P_{n,d}(\boldsymbol{\beta}(\boldsymbol{x})) + R_n(\boldsymbol{x})\,, \tag{2.1}$$

it is apparent that both the approximation error $R_n(\boldsymbol{x})$ and the needed amount of memory used for the polynomial storage depend on the selected expansion order n. For instance, a natural univariate exponential function e^x can be expanded as $e^x = 1+x+\frac{x^2}{2!}+\frac{x^3}{3!}+o(x^4)$ around the expansion point $x = 0$ by means of the JTTNM-n method. Its second order polynomial result $P_{2,1}(x) = 1 + x + \frac{x^2}{2!}$ possesses second order accuracy and needs to store three coefficients, while its third order polynomial result $P_{3,1}(x) = 1 + x + \frac{x^2}{2!} + \frac{x^3}{3!}$ has better accuracy (third order) but needs to store one more coefficient than the second order expansion. In other words, the higher expansion order is adopted, the better approximation accuracy can be obtained, meanwhile the more CPU time and the larger amount of memory must be consumed. Therefore, there is a trade-off between the approximation accuracy and CPU time in the practical choice of the expansion order n.

On the other hand, if the JTCNM-n method is employed in the approximation procedure of a given nonlinear law $f(\boldsymbol{x})$, the Chebyshev polynomial interpolation, with an assigned number of Chebyshev nodes (for instance 100), is often carried out in the Chebyshev basis. Generally, in advance of the interpolation procedure, it requires to convert the range of state vector $\Omega = [\boldsymbol{x}_l, \boldsymbol{x}_u] \in \mathbb{R}^d$ into a standard interval $\bar{\Omega} = [\boldsymbol{-1}, \boldsymbol{1}] \in \mathbb{R}^d$. Without loss of generality, using the following linear mapping on each component,

$$x_i' = \frac{-2x_i}{x_{l,i} - x_{u,i}} + \frac{x_{l,i} + x_{u,i}}{x_{l,i} - x_{u,i}}, \quad i = 1, \cdots, d \tag{2.2}$$

the polynomial defined on $\boldsymbol{x} \in \Omega$ can be converted and reconstructed on a unified hyper-rectangle domain $\boldsymbol{x}' \in \bar{\Omega}$. Note that this tedious transformation is unnecessary in the JTTNM-n implementation. Besides, the numbers of coefficients and polynomial terms constituting a full polynomial $P_{n,d}(\boldsymbol{\beta}(\boldsymbol{x}))$ are identical in the JTTNM-n and JTCNM-n implementations, that is

$$\mathcal{N}_{n,d} = \frac{(n+d)!}{n!\,d!}\,. \tag{2.3}$$

The most basic but vital point of both JTTNM-n and JTCNM-n implementations is how to store and retrieve Taylor or Chebyshev polynomials. Without question, the polynomial

storage and retrieval speed is one of the key factors dramatically affecting the execution efficiency of polynomial algebra [68]. In general, a polynomial of order n in d-dimensional variables $P_{n,d}(\boldsymbol{\beta}(\boldsymbol{x}))$ can be expressed as

$$P_{n,d}(\boldsymbol{\beta}(\boldsymbol{x})) = \sum_{|\boldsymbol{\gamma}|\leq n} \boldsymbol{c}_{\boldsymbol{\gamma}} \boldsymbol{\beta}_{\boldsymbol{\gamma}}(\boldsymbol{x}) \,, \tag{2.4}$$

where the multi-index vector $\boldsymbol{\gamma} = (\gamma_1, \cdots, \gamma_d)$ is a d-dimensional vector with each component $\gamma_i \in [0, n] \in \mathbb{N}$ and its Manhattan norm $|\boldsymbol{\gamma}| = \sum_{i=1}^{d} \gamma_i$, $\boldsymbol{\beta}_{\boldsymbol{\gamma}}(\boldsymbol{x}) = \beta_{\gamma_1}(x_1) \cdots \beta_{\gamma_d}(x_d)$ represents an arbitrary element of the selected orthogonal basis $\boldsymbol{\beta}(\boldsymbol{x})$, corresponding to a multi-index vector $\boldsymbol{\gamma}$, $\boldsymbol{c}_{\boldsymbol{\gamma}} \in \mathbb{R}^{\mathcal{N}_{n,d}}$ indicates the set of all coefficients defining the polynomial (2.4). In accordance to the form of polynomial (2.4), its storage can be achieved by archiving both the selected orthogonal basis $\boldsymbol{\beta}(\boldsymbol{x})$ and the associated coefficients $\boldsymbol{c}_{\boldsymbol{\gamma}}$. An efficient approach is to store all elements of the adopted orthogonal basis $\boldsymbol{\beta}(\boldsymbol{x})$ a priori by means of a default rule with the usage of the multi-index vector $\boldsymbol{\gamma}$. Subsequently, the storage of polynomials can be simplified into the placement of coefficients $\boldsymbol{c}_{\boldsymbol{\gamma}}$ in the $\mathcal{N}_{n,d}$-dimensional vector C, again in terms of the multi-index vector $\boldsymbol{\gamma}$. Clearly, the multi-index vector $\boldsymbol{\gamma} = (\gamma_1, \cdots, \gamma_d)$ used in (2.4) is a substitutive way to represent an arbitrary element $\boldsymbol{\beta}_{\boldsymbol{\gamma}}(\boldsymbol{x}) = \beta_{\gamma_1}(x_1) \cdots \beta_{\gamma_d}(x_d)$ since $\boldsymbol{\gamma}$ uniquely determines the placement of the element $\boldsymbol{\beta}_{\boldsymbol{\gamma}}(\boldsymbol{x})$ in Taylor or Chebyshev polynomial $P_{n,d}(\boldsymbol{\beta}(\boldsymbol{x}))$.

The purpose of this section is to precisely define the storage rule of the selected orthogonal basis $\boldsymbol{\beta}(\boldsymbol{x})$, and one-to-one correspondence between the element $\boldsymbol{\beta}_{\boldsymbol{\gamma}}(\boldsymbol{x})$ and its coefficient $C[k]$, that is, the element $\boldsymbol{\beta}_{\boldsymbol{\gamma}}(\boldsymbol{x})$, coefficient index k, and coefficient $C[k]$ must be uniquely determined and bundled by means of the multi-index vector $\boldsymbol{\gamma}$. The default storage rule of the selected orthogonal basis is defined as follows: firstly, all elements are segmented and allocated with serial numbers in accordance to the Manhattan norm of its milti-index $|\boldsymbol{\gamma}|$. Generally, for a certain element $\boldsymbol{\beta}_{\boldsymbol{\gamma}}(\boldsymbol{x})$, the lower $|\boldsymbol{\gamma}|$ the element possesses, the smaller serial number will be allocated; secondly, the total number of the elements whose order less than $|\boldsymbol{\gamma}|$ is calculated by means of (2.3) and denoted as $\mathcal{N}_{|\boldsymbol{\gamma}|-1,d}$; thirdly, the relative position k_r of a designated element $\boldsymbol{\beta}_{\boldsymbol{\gamma}}(\boldsymbol{x})$ inside the segment occupying the same degree $|\boldsymbol{\gamma}|$ is calculated by the reserve lexicographic order with vector access to the multi-index vector $\boldsymbol{\gamma}$ [146]; finally, the serial number $k = k_r + \mathcal{N}_{|\boldsymbol{\gamma}|-1,d}$ is allocated to the designated element $\boldsymbol{\beta}_{\boldsymbol{\gamma}}(\boldsymbol{x})$. Follow the rule above, all elements consisting of the selected orthogonal basis $\boldsymbol{\beta}(\boldsymbol{x})$ are sequentially distributed and known a priori. As a consequence, each element is allocated with its unique serial number k, which is also adopted to determine the position of the associated coefficient inside the $\mathcal{N}_{n,d}$-dimensional vector C. On the other hand, if the coefficient index k is given, the coefficient $C[k]$ can be retrieved from the coefficient vector C and its associated multi-index vector $\boldsymbol{\gamma} = (\gamma_1, \cdots, \gamma_d)$ can be also uniquely calculated. Note that Algorithm 2.2 and 2.3 explain these detailed procedures.

As an illustrative example, Fig. 2.1 illustrates in detail the storage procedure of a polynomial of order 3 in 2-dimensional variables, expressed in both Taylor and Chebyshev bases. Clearly, each term contains one selected orthogonal basis element $\boldsymbol{\beta}_{\boldsymbol{\gamma}}(\boldsymbol{x})$ and its relevant coefficient $C[k]$. Their one-to-one correspondence relationship can be constructed straightforward. For instance, in JTTNM-3 and JTCNM-3 implementations, if the element $x_1^2 x_2$ or $T_2(x_1)T_1(x_2)$ is considered, it is easy to obtain its multi-index vector

Algorithm 2.1 Form an auxiliary matrix N_M with dimension $(d+1) \times |\boldsymbol{\gamma}|$

1: All terms in the first row of N_M are zero;
2: All terms in the second row of N_M orderly equal $1, 2, \cdots, |\boldsymbol{\gamma}|$;
3: The terms in other rows $N_M[i][j]$ can be calculated as
- if k equal zero, $N_M[i][0] = 1$;
- else $N_M[i][j] = N_M[i-1][j] + N_M[i][j-1]$.

Algorithm 2.2 Calculate the coefficient index k in terms of the multi-index vector $\boldsymbol{\gamma}$

1: Determine the state dimension d and calculate the Manhattan norm of the multi-index vector $|\boldsymbol{\gamma}|$;
2: Using Algorithm 2.1, form an auxiliary matrix N_M with dimension $(d+1) \times |\boldsymbol{\gamma}|$;
3: Based on the lexicographic order with vector access to the multi-index vector $\boldsymbol{\gamma}$, the coefficient index k is calculated as follows
- if $|\boldsymbol{\gamma}|$ equals zero, then $k = 0$; else retrieve the multi-index vector $\boldsymbol{\gamma} = (\gamma_1, \cdots, \gamma_d)$ from left to right in order to obtain the index i of the last non-zero term;
- set $k = 0$, $i = i - 1$, $j = l - i$ and $l = -1$;
- Loop, while $(i \geq 0)$ $\{l = l + \gamma_{i+1}; i- = 1; k+ = N_M[j][l]; j+ = 1\}$ end.

$\boldsymbol{\gamma} = (2, 1)$ and further calculate the coefficient index $k = 7$ by means of Algorithm 2.2; homoplastically, if the coefficient index $k = 7$ is given, one can compute the multi-index vector $\boldsymbol{\gamma} = (2, 1)$ through the usage of Algorithm 2.3 and determine the element $x_1^2 x_2$ or $T_2(x_1)T_1(x_2)$.

Algorithm 2.3 Calculate the multi-index vector $\boldsymbol{\gamma}$ using the coefficient index k

1: Determine the state dimension d and use the trial-and-error method to obtain the maximum integer γ_{max} satisfying $\frac{(\gamma_{max}+d-1)!}{(\gamma_{max}-1)!d!} < k + 1$;
2: Using Algorithm 2.1, form an auxiliary matrix N_M with dimension $(d+1) \times |\boldsymbol{\gamma}|$;
3: According to the reserve lexicographic order and the coefficient index k, calculate the multi-index vector $\boldsymbol{\gamma} = (\gamma_1, \cdots, \gamma_d)$ as follows
Loop:
S1: set l equals the state dimension d
S2: for $(i = 0; i < l; i = i + 1)\{$
S3: if k equals zero, set all components $\gamma_{i+1}, \cdots, \gamma_d$ as zero and return the multi-index vector $\boldsymbol{\gamma}$; end;
S4: if i equals zero, $\{$ $j = \gamma_{max}$ $\}$; else $\{$ for $(j = 0;$ $k \geq N_M[l][j];$ $j = j + 1, \gamma_i = \gamma_i - j);$ $\}$ end;
S5: $\gamma_{i+1} = j$; $k = k - N_M[l][j-1]$; $l = l - 1$;
S6: $\}$ end; return the multi-index vector $\boldsymbol{\gamma}$.

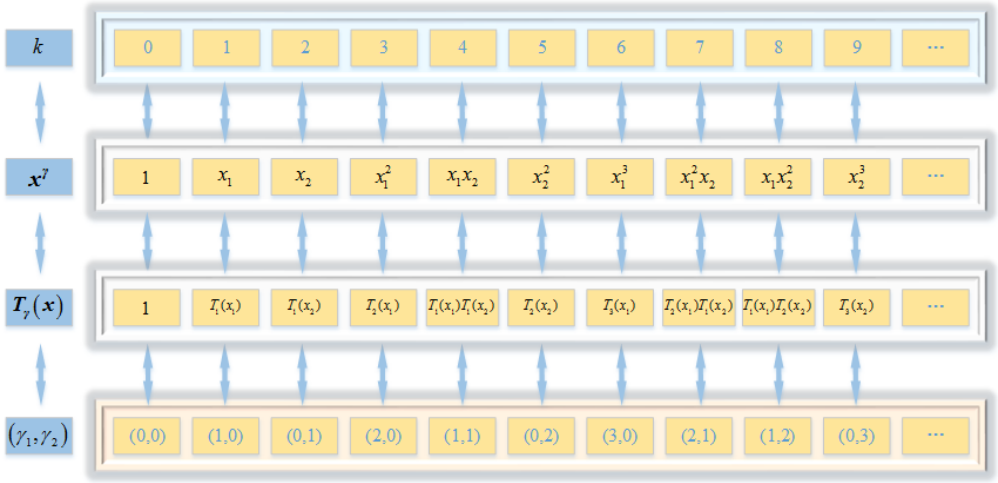


Figure 2.1: One-to-one correspondence among the coefficient index k, multi-index vector $\boldsymbol{\gamma} = (\gamma_1, \gamma_2)$, and basis element $\boldsymbol{\beta}_{\boldsymbol{\gamma}}(\boldsymbol{x})$ in JTTNM-3 and JTCNM-3 implementations.

2.2 Polynomial algebra

This section introduces the definition of polynomial algebra for the JT framework. To be specific, we firstly describe the formulae for basic polynomial operations and differential polynomial algebra; then, the polynomial approximations to basic nonlinear univariate functions are investigated via the Taylor expansion approach or Chebyshev interpolation technique; furthermore, the polynomial composition operation is proposed to approximate multi-variable functions. In order to validate the performance of the proposed approximation methods, several numerical simulations are provided.

2.2.1 Basic polynomial operations

As the basis of JTTNM-n and JTCNM-n implementations, a set of elementary arithmetic operations $\odot$, including polynomial addition, subtraction, multiplication and composition, are trivially defined, generating in this way an algebra in the polynomial space $[\mathcal{P}_{n,d}(\boldsymbol{\beta}_{\boldsymbol{\gamma}}(\boldsymbol{x})), \odot]$. In order to illustrate them in a concise and comprehensible way, we firstly define two polynomials ${}^{a}P_{n,d}(\boldsymbol{\beta}_{\boldsymbol{\gamma}_a})$ and ${}^{b}P_{n,d}(\boldsymbol{\beta}_{\boldsymbol{\gamma}_b})$ in d variables of degree n, in Taylor or Chebyshev basis

$$ {}^{a}P_{n,d}(\boldsymbol{\beta}_{\boldsymbol{\gamma}_a}) = \sum_{|\boldsymbol{\gamma}_a| \le n} {}^{a}\boldsymbol{c}_{\boldsymbol{\gamma}_a} \boldsymbol{\beta}_{\boldsymbol{\gamma}_a}(\boldsymbol{x}) \,, \quad {}^{b}P_{n,d}(\boldsymbol{\beta}_{\boldsymbol{\gamma}_b}) = \sum_{|\boldsymbol{\gamma}_b| \le n} {}^{b}\boldsymbol{c}_{\boldsymbol{\gamma}_b} \boldsymbol{\beta}_{\boldsymbol{\gamma}_b}(\boldsymbol{x}) \,, \tag{2.5} $$

where both ${}^{a}P_{n,d}(\boldsymbol{\beta}_{\boldsymbol{\gamma}_a})$ and ${}^{b}P_{n,d}(\boldsymbol{\beta}_{\boldsymbol{\gamma}_b})$, identified by two sets of coefficients ${}^{a}\boldsymbol{c}_{\boldsymbol{\gamma}_a}, {}^{b}\boldsymbol{c}_{\boldsymbol{\gamma}_b} \in \mathbb{R}^{\mathcal{N}_n^d}$, are defined in the polynomial space $[\mathcal{P}_{n,d}(\boldsymbol{\beta}_{\boldsymbol{\gamma}}(\boldsymbol{x})), \odot]$. Note that if the Taylor basis

is adopted,

$$\boldsymbol{\beta}_{\boldsymbol{\gamma}}(\boldsymbol{x}) = \boldsymbol{x}^{\boldsymbol{\gamma}} = \prod_{i=1}^{d} x_i^{\gamma_i}\,, \quad \boldsymbol{\gamma} = (\gamma_1, \cdots, \gamma_d)\,, \tag{2.6}$$

otherwise, if the Chebyshev basis is adopted

$$\boldsymbol{\beta}_{\boldsymbol{\gamma}}(\boldsymbol{x}) = \boldsymbol{T}_{\boldsymbol{\gamma}}(\boldsymbol{x}) = \prod_{i=1}^{d} T_{\gamma_i}(x_i)\,, \quad \boldsymbol{\gamma} = (\gamma_1, \cdots, \gamma_d). \tag{2.7}$$

On the other hand, as discussed above, two vectors ${}^aC$ and ${}^bC$ with dimension $\mathcal{N}_{n,d}$ are used to store the coefficients ${}^a\boldsymbol{c}_{\boldsymbol{\gamma}_a}$ and ${}^b\boldsymbol{c}_{\boldsymbol{\gamma}_b}$, thus also employed to identify the polynomials ${}^aP_{n,d}(\boldsymbol{\beta}_{\boldsymbol{\gamma}_a})$ and ${}^bP_{n,d}(\boldsymbol{\beta}_{\boldsymbol{\gamma}_b})$. Besides, unless otherwise stated, polynomial results of elementary arithmetic operations $\odot$ are expressed as

$${}^rP_{n,d}(\boldsymbol{\beta}_{\boldsymbol{\gamma}_r}) = \sum_{|\boldsymbol{\gamma}_r| \leq n} {}^r\boldsymbol{c}_{\boldsymbol{\gamma}_r} \boldsymbol{\beta}_{\boldsymbol{\gamma}_r}(\boldsymbol{x})\,. \tag{2.8}$$

2.2.1.1 Scalar multiplication of polynomials

Assume a given scalar $\lambda \in \mathbb{R}$ and a polynomial ${}^aP_{n,d}(\boldsymbol{\beta}_{\boldsymbol{\gamma}_a})$, the scalar multiplication can be expressed as ${}^rP_{n,d}(\boldsymbol{\beta}_{\boldsymbol{\gamma}_r}) = \lambda \cdot {}^aP_{n,d}(\boldsymbol{\beta}_{\boldsymbol{\gamma}_a})$, thus polynomial coefficients are

$${}^rC[k] = \lambda \cdot {}^aC[k]\,, \quad k = 0, \cdots, \mathcal{N}_{n,d} - 1\,, \tag{2.9}$$

where ${}^rP_{n,d}(\boldsymbol{\beta}_{\boldsymbol{\gamma}_r})$ indicates the polynomial result of the scalar multiplication, identified by its coefficient vector ${}^rC$, as well as the set of coefficients ${}^r\boldsymbol{c}_{\boldsymbol{\gamma}_r}$.

2.2.1.2 Polynomial addition and subtraction

Let two polynomials ${}^aP_{n,d}(\boldsymbol{\beta}_{\boldsymbol{\gamma}_a})$ and ${}^bP_{n,d}(\boldsymbol{\beta}_{\boldsymbol{\gamma}_b})$ being two elements in the polynomial space $[\mathcal{P}_{n,d}(\boldsymbol{\beta}_{\boldsymbol{\gamma}}(\boldsymbol{x})), \odot]$, identified by two corresponding coefficient vectors ${}^aC$ and ${}^bC$. The addition and subtraction operations ${}^rP_{n,d}(\boldsymbol{\beta}_{\boldsymbol{\gamma}_r}) = {}^aP_{n,d}(\boldsymbol{\beta}_{\boldsymbol{\gamma}_a}) \pm {}^bP_{n,d}(\boldsymbol{\beta}_{\boldsymbol{\gamma}_b})$ can be again achieved coefficient by coefficient, i.e.,

$${}^rC[k] = {}^aC[k] \pm {}^bC[k]\,, \quad k = 0, \cdots, \mathcal{N}_{n,d} - 1\,. \tag{2.10}$$

2.2.1.3 Polynomial multiplication

The multiplication of two polynomials is defined as ${}^rP_{n,d}(\boldsymbol{\beta}_{\boldsymbol{\gamma}_r}) = {}^aP_{n,d}(\boldsymbol{\beta}_{\boldsymbol{\gamma}_a}) \cdot {}^bP_{n,d}(\boldsymbol{\beta}_{\boldsymbol{\gamma}_b})$, that is

$${}^rP_{n,d}(\boldsymbol{\beta}_{\boldsymbol{\gamma}_r}) = \sum_{|\boldsymbol{\gamma}_r| \leq n} {}^r\boldsymbol{c}_{\boldsymbol{\gamma}_r} \boldsymbol{\beta}_{\boldsymbol{\gamma}_r}(\boldsymbol{x}) = \sum_{|\boldsymbol{\gamma}_a| \leq n} {}^a\boldsymbol{c}_{\boldsymbol{\gamma}_a} \boldsymbol{\beta}_{\boldsymbol{\gamma}_a}(\boldsymbol{x}) \cdot \sum_{|\boldsymbol{\gamma}_b| \leq n} {}^2\boldsymbol{c}_{\boldsymbol{\gamma}_b} \boldsymbol{\beta}_{\boldsymbol{\gamma}_b}(\boldsymbol{x})\,. \tag{2.11}$$

In contrast to the scalar multiplication, addition and subtraction discussed above, it is impossible to address the polynomial multiplication in a unified form for Taylor and Chebyshev cases. Therefore, a separate discussion will be carried out. Analogously, all coefficients ${}^{r}\boldsymbol{c}_{\boldsymbol{\gamma}_r}$, ${}^{a}\boldsymbol{c}_{\boldsymbol{\gamma}_a}$ and ${}^{b}\boldsymbol{c}_{\boldsymbol{\gamma}_b}$ are stored into $\mathcal{N}_{n,d}$-dimensional vectors ${}^{r}C$, ${}^{a}C$ and ${}^{b}C$.

Taylor polynomial case

In the Taylor polynomial case, (2.11) can be rewritten in the monomial basis as

$$ {}^{r}P_{n,d}(\boldsymbol{x}^{\boldsymbol{\gamma}_r}) = \sum_{|\boldsymbol{\gamma}_r|\leq n} {}^{r}\boldsymbol{c}_{\boldsymbol{\gamma}_r}\boldsymbol{x}^{\boldsymbol{\gamma}_r} = \sum_{|\boldsymbol{\gamma}_a|\leq n} {}^{a}\boldsymbol{c}_{\boldsymbol{\gamma}_a}\boldsymbol{x}^{\boldsymbol{\gamma}_a} \cdot \sum_{|\boldsymbol{\gamma}_b|\leq n} {}^{b}\boldsymbol{c}_{\boldsymbol{\gamma}_b}\boldsymbol{x}^{\boldsymbol{\gamma}_b}\,, \tag{2.12} $$

where

$$ \boldsymbol{x}^{\boldsymbol{\gamma}_a}\cdot\boldsymbol{x}^{\boldsymbol{\gamma}_b} = \begin{cases} \boldsymbol{x}^{\boldsymbol{\gamma}_a+\boldsymbol{\gamma}_b}, & |\boldsymbol{\gamma}_a+\boldsymbol{\gamma}_b|\leq n \\ 0, & n<|\boldsymbol{\gamma}_a+\boldsymbol{\gamma}_b|\leq 2n\,. \end{cases} $$

It is clear that each coefficient ${}^{r}\boldsymbol{c}_{\boldsymbol{\gamma}_r}$ will contain the product of coefficients ${}^{a}\boldsymbol{c}_{\boldsymbol{\gamma}_a}$ and ${}^{b}\boldsymbol{c}_{\boldsymbol{\gamma}_b}$. Therefore, the general formula to calculate ${}^{r}\boldsymbol{c}_{\boldsymbol{\gamma}_r}$ can be obtained through matching those terms in both sides of (2.12) with the same multi-index vector,

$$ {}^{r}\boldsymbol{c}_{\boldsymbol{\gamma}_r} = \sum_{\boldsymbol{\gamma}_a\leq\boldsymbol{\gamma}_r} {}^{a}\boldsymbol{c}_{\boldsymbol{\gamma}_a}\cdot{}^{b}\boldsymbol{c}_{\boldsymbol{\gamma}_r-\boldsymbol{\gamma}_a}\,, \quad |\boldsymbol{\gamma}_r| = 0,\cdots,n\,, \tag{2.13} $$

where the multi-index vectors $\boldsymbol{\gamma}_r, \boldsymbol{\gamma}_a \in \mathbb{N}^d$. The detailed procedure to calculate the coefficient and its associated index k corresponding to $\boldsymbol{\gamma}_r$ includes: 1) assign a multi-index vector $\boldsymbol{\gamma}_r$ and seek all multi-index vectors $\boldsymbol{\gamma}_a$ satisfying the condition $\boldsymbol{\gamma}_a \leq \boldsymbol{\gamma}_r$ (i.e., $\forall i \in [1,d] \subset \mathbb{N}$, $[\gamma_a]_i \leq [\gamma_r]_i$); 2) retrieve the coefficient of the term identified by the multi-index vector $\boldsymbol{\gamma}_a$ or $\boldsymbol{\gamma}_r - \boldsymbol{\gamma}_a$ from ${}^{a}C$ and ${}^{b}C$ using Algorithm 2.2; 3) compute the coefficient associated with the multi-index vector $\boldsymbol{\gamma}_r$ by means of (2.13), 4) store it inside the coefficient vector ${}^{r}C$ through Algorithm 2.3. Note that the full implementation of polynomial multiplication should go through all possible multi-index vector $\boldsymbol{\gamma}_r$ cohering with $|\boldsymbol{\gamma}_r| \leq n$.

In order to clearly explain the polynomial multiplication operation, ${}^{r}P_{2,2}(\boldsymbol{x}) = {}^{a}P_{2,2}(\boldsymbol{x}) \cdot {}^{b}P_{2,2}(\boldsymbol{x})$ is taken as an illustrative example. The detailed polynomials can be expressed as

$$ \begin{cases} {}^{a}P_{2,2}(\boldsymbol{x}) &= a_{0,0} + a_{1,0}x_1 + a_{0,1}x_2 + a_{2,0}x_1^2 + a_{1,1}x_1x_2 + a_{0,2}x_2^2\,, \\ {}^{b}P_{2,2}(\boldsymbol{x}) &= b_{0,0} + b_{1,0}x_1 + b_{0,1}x_2 + b_{2,0}x_1^2 + b_{1,1}x_1x_2 + b_{0,2}x_2^2\,, \\ {}^{r}P_{2,2}(\boldsymbol{x}) &= r_{0,0} + r_{1,0}x_1 + r_{0,1}x_2 + r_{2,0}x_1^2 + r_{1,1}x_1x_2 + r_{0,2}x_2^2\,, \end{cases} \tag{2.14} $$

then coefficients of the terms whose multi-index vector satisfy $|\gamma_r| \leq 2$ are calculated. All feasible multi-index vectors associated with $|\gamma_r| = 2$ contain $\gamma_r = (2,0)$, $(1,1)$ and $(0,2)$. One can obtain three resulting coefficients by means of (2.13) as follows

$$\begin{cases} r_{2,0} &= a_{2,0}b_{0,0} + a_{1,0}b_{1,0} + a_{0,0}b_{2,0}\,, \\ r_{1,1} &= a_{1,1}b_{0,0} + a_{1,0}b_{0,1} + a_{0,1}b_{1,0} + a_{0,0}b_{1,1}\,, \\ r_{0,2} &= a_{0,2}b_{0,0} + a_{0,1}b_{0,1} + a_{0,0}b_{0,2}\,. \end{cases} \tag{2.15}$$

Finally, all coefficients of ${}^{r}P_{2,2}(\boldsymbol{x})$ can be obtained through the traversal of all possibilities satisfying the condition $|\gamma_r| \leq 2$. Based on Algorithm 2.3, the coefficient index k associated with a certain multi-index vector $\boldsymbol{\gamma}_r$ can be calculated, such that the coefficient can be stored into the vector ${}^{r}C$.

Chebyshev polynomial case

In the Chebyshev polynomial case, (2.11) is expressed in the Chebyshev basis

$${}^{r}P_{n,d}(\boldsymbol{T}_{\boldsymbol{\gamma}_r}(\boldsymbol{x})) = \sum_{|\boldsymbol{\gamma}_r|\leq n} {}^{r}\boldsymbol{c}_{\boldsymbol{\gamma}_r}\boldsymbol{T}_{\boldsymbol{\gamma}_r}(\boldsymbol{x}) = \sum_{|\boldsymbol{\gamma}_a|\leq n} {}^{a}\boldsymbol{c}_{\boldsymbol{\gamma}_a}\boldsymbol{T}_{\boldsymbol{\gamma}_a}(\boldsymbol{x}) \sum_{|\boldsymbol{\gamma}_b|\leq n} {}^{b}\boldsymbol{c}_{\boldsymbol{\gamma}_b}\boldsymbol{T}_{\boldsymbol{\gamma}_b}(\boldsymbol{x})\,, \tag{2.16}$$

where $\boldsymbol{T}_{\Upsilon}(\boldsymbol{x}), \Upsilon = \boldsymbol{\gamma}_r, \boldsymbol{\gamma}_a, \boldsymbol{\gamma}_b$ indicates the Chebyshev basis. However, the first attempt to develop a direct product of Chebyshev polynomials was discarded due to the calculation complexity of Chebyshev polynomial multiplications [147], limiting, in principle, the applicability of the nonlinear mapping method based on Chebyshev series expansions [94]. A new approach converting polynomials from the Chebyshev basis to the monomial basis was proposed in order to accelerate the multiplication of Chebyshev polynomials [95,148,149]. The detailed procedure can be given as follows: 1) convert polynomials from the Chebyshev basis into the monomial basis, ${}^{r}P_{n,d}(\boldsymbol{T}_{\boldsymbol{\gamma}_r}(\boldsymbol{x})) \rightarrow {}^{r}P_{n,d}(\boldsymbol{x}^{\boldsymbol{\gamma}_r})$, thus (2.16) can be rewritten as

$$\begin{aligned} {}^{r}P_{n,d}(\boldsymbol{T}_{\boldsymbol{\gamma}_r}(\boldsymbol{x})) &= \sum_{|\boldsymbol{\gamma}_a|\leq n} {}^{a}\boldsymbol{c}_{\boldsymbol{\gamma}_a}\boldsymbol{T}_{\boldsymbol{\gamma}_a}(\boldsymbol{x}) \sum_{|\boldsymbol{\gamma}_b|\leq n} {}^{b}\boldsymbol{c}_{\boldsymbol{\gamma}_b}\boldsymbol{T}_{\boldsymbol{\gamma}_b}(\boldsymbol{x}) \\ &= \sum_{|\boldsymbol{\gamma}_a|\leq n} {}^{a}\bar{\boldsymbol{c}}_{\boldsymbol{\gamma}_a}\boldsymbol{x}^{\boldsymbol{\gamma}_a} \sum_{|\boldsymbol{\gamma}_b|\leq n} {}^{b}\bar{\boldsymbol{c}}_{\boldsymbol{\gamma}_b}\boldsymbol{x}^{\boldsymbol{\gamma}_b}\,, \end{aligned} \tag{2.17}$$

2) implement the polynomial multiplication using the formula (2.13) which was defined in the polynomial space in the monomial basis $(\mathcal{P}_{n,d}(\boldsymbol{x}^{\boldsymbol{i}}), \odot)$, and keep polynomial order up to $\lfloor 1.5n+1 \rfloor$, that is,

$$\boldsymbol{x}^{\boldsymbol{\gamma}_a} \cdot \boldsymbol{x}^{\boldsymbol{\gamma}_b} = \begin{cases} \boldsymbol{x}^{\boldsymbol{\gamma}_a+\boldsymbol{\gamma}_b}, & \boldsymbol{\gamma}_a + \boldsymbol{\gamma}_b \leq \lfloor 1.5n+1 \rfloor, \\ 0, & \boldsymbol{\gamma}_a + \boldsymbol{\gamma}_b > \lfloor 1.5n+1 \rfloor, \end{cases} \tag{2.18}$$

where $\lfloor 1.5n+1 \rfloor$ indicates the smallest integer bigger than $1.5n+1$; 3) reverse the basis transformation and truncation at order n, this is ${}^{r}P_{\lfloor 1.5n+1 \rfloor,d}(\boldsymbol{x}^{\boldsymbol{\gamma}_r}) \rightarrow {}^{r}P_{n,d}(\boldsymbol{T}_{\boldsymbol{\gamma}_r}(\boldsymbol{x}))$. It is worth mentioning that this improved algorithm for performing the Chebyshev polynomial multiplication frequently employs the basis transformation, where high order terms in the Chebyshev basis contribute to low order terms in the monomial basis, such that it is possible to produce the loss of accuracy. However, it has been proved that this handling method for truncating the polynomial result at order $\lfloor 1.5n+1 \rfloor$ enables to minimize the loss of accuracy in the basis transformation procedure [95].

2.2.1.4 Polynomial division

The division of two polynomials is defined as ${}^{r}P_{n,d}(\boldsymbol{\beta}_{\boldsymbol{\gamma}_r}) = {}^{a}P_{n,d}(\boldsymbol{\beta}_{\boldsymbol{\gamma}_a})/{}^{b}P_{n,d}(\boldsymbol{\beta}_{\boldsymbol{\gamma}_b})$, which can be rewritten as

$$ {}^{r}P_{n,d}(\boldsymbol{\beta}_{\boldsymbol{\gamma}_r}) {}^{b}P_{n,d}(\boldsymbol{\beta}_{\boldsymbol{\gamma}_b}) = {}^{a}P_{n,d}(\boldsymbol{\beta}_{\boldsymbol{\gamma}_a}) \,. \tag{2.19} $$

It is clear that (2.19) converts the polynomial division into the polynomial multiplication. Note that if the Chebyshev polynomial division operation is implemented, the transformation from the Chebyshev basis to the monomial basis is again employed in order to accelerate the evaluation of the division operation. For space brevity, only the division operation implemented in the Taylor polynomial case is discussed herein, while that implemented in the Chebyshev polynomial case is similar except the indispensable basis transformation.

In the Taylor polynomial case, the following equation can be obtained in terms of (2.13),

$$ {}^{a}\boldsymbol{c}_{\boldsymbol{\gamma}_a} = \sum_{\boldsymbol{\gamma}_r \leq \boldsymbol{\gamma}_a} {}^{r}\boldsymbol{c}_{\boldsymbol{\gamma}_r} \cdot {}^{b}\boldsymbol{c}_{\boldsymbol{\gamma}_a - \boldsymbol{\gamma}_r} = \sum_{\boldsymbol{\gamma}_r < \boldsymbol{\gamma}_a} {}^{r}\boldsymbol{c}_{\boldsymbol{\gamma}_r} \cdot {}^{b}\boldsymbol{c}_{\boldsymbol{\gamma}_a - \boldsymbol{\gamma}_r} + {}^{r}\boldsymbol{c}_{\boldsymbol{\gamma}_a} \cdot {}^{b}\boldsymbol{c}_{\mathbf{0}} \,, \quad |\boldsymbol{\gamma}_a| = 0, \cdots, n \,, \tag{2.20} $$

thus, it is easy to calculate the result of the division between two Taylor polynomials, and the coefficients of the polynomial result are,

$$ {}^{r}\boldsymbol{c}_{\boldsymbol{\gamma}_a} = \frac{1}{{}^{b}\boldsymbol{c}_{\mathbf{0}}} \left({}^{a}\boldsymbol{c}_{\boldsymbol{\gamma}_a} - \sum_{\boldsymbol{\gamma}_r < \boldsymbol{\gamma}_a} {}^{r}\boldsymbol{c}_{\boldsymbol{\gamma}_r} \cdot {}^{b}\boldsymbol{c}_{\boldsymbol{\gamma}_a - \boldsymbol{\gamma}_r} \right) \,, \quad |\boldsymbol{\gamma}_a| = 0, \cdots, n \,. \tag{2.21} $$

Note in particular that all terms in the right hand side of (2.21) are obtained either from the previous calculations (i.e., ${}^{r}\boldsymbol{c}_{\boldsymbol{\gamma}_r}$ with $\boldsymbol{\gamma}_r < \boldsymbol{\gamma}_a$) or from the known coefficients ${}^{a}\boldsymbol{c}_{\boldsymbol{\gamma}_a}$ and ${}^{b}\boldsymbol{c}_{\boldsymbol{\gamma}_b}$. Again, to explain the polynomial division procedure, ${}^{r}P_{2,2}(\boldsymbol{x}) = {}^{a}P_{2,2}(\boldsymbol{x})/{}^{b}P_{2,2}(\boldsymbol{x})$ is taken as an example with the expressions in (2.14). All feasible multi-index vectors associated with $|\gamma_r| \leq 2$ are $\gamma_r = (0,0)$, $(1,0)$, $(0,1)$, $(2,0)$, $(1,1)$ and $(0,2)$. Using (2.21), we can obtain the corresponding coefficients as follows

$$ \begin{aligned} r_{0,0} &= \frac{a_{0,0}}{b_{0,0}} \,, & r_{1,0} &= \frac{a_{1,0} - r_{0,0}b_{1,0}}{b_{0,0}} \,, \\ r_{0,1} &= \frac{a_{0,1} - r_{0,0}b_{0,1}}{b_{0,0}} \,, & r_{2,0} &= \frac{a_{2,0} - r_{1,0}b_{1,0} - r_{0,0}b_{2,0}}{b_{0,0}} \,, \\ r_{1,1} &= \frac{a_{1,1} - r_{1,0}b_{0,1} - r_{0,1}b_{1,0} - r_{0,0}b_{1,1}}{b_{0,0}} \,, & r_{0,2} &= \frac{a_{0,2} - r_{0,1}b_{0,1} - r_{0,0}b_{0,2}}{b_{0,0}} \,. \end{aligned} $$

2.2.2 Differential polynomial algebra

Apart from the basic operations, the calculation procedures for the derivatives and integrals of polynomials are also two issues of interest. It is of vital importance to differentiate

and integrate a given polynomial with respect to one of its variables. Without loss of generality, only the differentiation and integration of Taylor polynomials are explained in detail in what follows. Note in particular that, if the differentiation and integration of Chebyshev polynomials are employed, the differential operations suitable for Taylor polynomials can be easily extended to those suitable for Chebyshev polynomials via a similar basis transformation procedure adopted in the polynomial multiplication operation.

2.2.2.1 Polynomial differentiation

Given a Taylor polynomial of order n in d-dimensional variables

$$P_{n,d}(\boldsymbol{x}^{\boldsymbol{\gamma}}) = \sum_{|\boldsymbol{\gamma}|\leq n} \boldsymbol{c}_{\boldsymbol{\gamma}} \boldsymbol{x}^{\boldsymbol{\gamma}}\,, \tag{2.22}$$

its partial derivative with respect to the ith variable x_i is computed by the usual differentiation law for polynomials,

$${}^{r}P_{n-1,d}(\boldsymbol{x}^{\bar{\boldsymbol{\gamma}}}) = \frac{\partial P_{n,d}(\boldsymbol{x}^{\boldsymbol{\gamma}})}{\partial x_i} = \sum_{|\boldsymbol{\gamma}|\leq n} \gamma_i \boldsymbol{c}_{\boldsymbol{\gamma}} \boldsymbol{x}^{\boldsymbol{\gamma}-\tilde{\boldsymbol{\gamma}}} = \sum_{|\bar{\boldsymbol{\gamma}}|\leq n-1} {}^{r}\boldsymbol{c}_{\bar{\boldsymbol{\gamma}}} \boldsymbol{x}^{\bar{\boldsymbol{\gamma}}}\,, \tag{2.23}$$

where ${}^{r}\boldsymbol{c}_{\bar{\boldsymbol{\gamma}}} = \gamma_i \boldsymbol{c}_{\boldsymbol{\gamma}}$ and $\bar{\boldsymbol{\gamma}} = \boldsymbol{\gamma} - \tilde{\boldsymbol{\gamma}}$. Note in particular that the multi-index vector $\tilde{\boldsymbol{\gamma}}$ is employed to identify the position of ith variable x_i, its elements are defined as

$$\tilde{\gamma}_j = \begin{cases} 0\,, & j \neq i\,, \\ 1\,, & j = i\,, \end{cases} \quad j = 1, \cdots, d\,. \tag{2.24}$$

Therefore, the detailed procedure for calculating the result of polynomial partial derivative is: 1) determine the variable x_i and associated multi-index vector $\tilde{\boldsymbol{\gamma}}$ based on (2.24); 2) use the multi-index vector $\boldsymbol{\gamma}$ to calculate the coefficient index $k_{\boldsymbol{\gamma}}$ through Algorithm 2.2 and retrieve the corresponding coefficient $C[k_{\boldsymbol{\gamma}}]$; 3) compute the new multi-index vector $\bar{\boldsymbol{\gamma}} = \boldsymbol{\gamma} - \tilde{\boldsymbol{\gamma}}$ and further obtain the corresponding coefficient index $k_{\bar{\boldsymbol{\gamma}}}$ based on Algorithm 2.3; 4) calculate and store the value of $\gamma_i \cdot C[k_{\boldsymbol{\gamma}}]$ into ${}^{r}C[k_{\bar{\boldsymbol{\gamma}}}]$; 5) traverse all possible multi-index vectors $\boldsymbol{\gamma}$ cohering with $|\boldsymbol{\gamma}| \leq n$ one by one and achieve the polynomial differentiation. It is clear that if a polynomial is differentiated one time relative to one of its variable, its order is reduced by one.

2.2.2.2 Polynomial integration

A polynomial integration process is another type of differential operations. Similar to the polynomial differentiation, only the integration operation of Taylor polynomials is exhibited, while the integration operation of Chebyshev polynomials can be achieved by combining the basis transformation and integration operation of Taylor polynomials.

Given a multivariate Taylor polynomial of order $n-1$, i.e.,

$$P_{n-1,d}(\boldsymbol{x}^{\boldsymbol{\gamma}}) = \sum_{|\boldsymbol{\gamma}|\leq n-1} \boldsymbol{c}_{\boldsymbol{\gamma}} \boldsymbol{x}^{\boldsymbol{\gamma}} \,. \tag{2.25}$$

Its indefinite integral with respect to the variable x_i can be calculated by the common integration law for polynomials,

$${}^{r}P_{n,d}(\boldsymbol{x}^{\bar{\boldsymbol{\gamma}}}) = \int P_{n-1,d}(\boldsymbol{x}^{\boldsymbol{\gamma}}) dx_i = \sum_{|\boldsymbol{\gamma}|\leq n-1} \frac{1}{1+\gamma_i} \boldsymbol{c}_{\boldsymbol{\gamma}} \boldsymbol{x}^{\boldsymbol{\gamma}+\tilde{\boldsymbol{\gamma}}} = \sum_{|\bar{\boldsymbol{\gamma}}|\leq n} {}^{r}\boldsymbol{c}_{\bar{\boldsymbol{\gamma}}} \boldsymbol{x}^{\bar{\boldsymbol{\gamma}}} \,, \tag{2.26}$$

with

$${}^{r}\boldsymbol{c}_{\bar{\boldsymbol{\gamma}}} = \frac{1}{1+\gamma_i} \boldsymbol{c}_{\boldsymbol{\gamma}} \,, \quad \bar{\boldsymbol{\gamma}} = \boldsymbol{\gamma} + \tilde{\boldsymbol{\gamma}} \,, \quad \tilde{\gamma}_j = \begin{cases} 0 \,, & j \neq i \,, \\ 1 \,, & j = i \,. \end{cases}$$

As it happens with the polynomial differentiation, the detailed procedure for implementing the polynomial integration is described as: 1) determine the variable x_i and associated multi-index vector $\tilde{\boldsymbol{\gamma}}$; 2) use the multi-index vector $\boldsymbol{\gamma}$ to calculate the coefficient index $k_{\boldsymbol{\gamma}}$ through Algorithm 2.2 and retrieve the coefficient $C[k_{\boldsymbol{\gamma}}]$; 3) compute the new multi-index vector $\bar{\boldsymbol{\gamma}}$ and the associated coefficient index $k_{\bar{\boldsymbol{\gamma}}}$ based on Algorithm 2.3; 4) calculate and store the value of $\frac{1}{1+\gamma_i} \cdot C[k_{\boldsymbol{\gamma}}]$ into ${}^{r}C[k_{\bar{\boldsymbol{\gamma}}}]$; 5) traverse all possible multi-index vector $\boldsymbol{\gamma}$ cohering with $|\boldsymbol{\gamma}| \leq n-1$ in order to achieve the full polynomial integration. Observe that, after one time polynomial integration, the order is increased by 1, which results in the omission of the terms with original order n since the integration operation is performed in the polynomial space $[\mathcal{P}_{n,d}(\boldsymbol{\beta}_{\boldsymbol{\gamma}}(\boldsymbol{x})), \odot]$. This fact justifies the abandon of the terms with order n in the polynomial integration formula (2.26) from the perspective of saving the computational cost.

2.2.3 Polynomial approximation to univariate functions

As a bridge, the polynomial approximation technique of univariate functions, in either Taylor or Chebyshev basis, is significantly important, enabling to convert the operations implemented in the function space into those carried out in the polynomial space. Clearly, all complicated univariate functions consist of several basic functions, such as exponentials, powers, logarithms, trigonometric functions and so on. Therefore, the main work herein focuses on efficient polynomial representations of basic univariate functions.

In what follows, we will give the detailed procedures for both JTTNM-n and JTCNM-n implementations to approximate a nonlinear univariate function. In principle, the JTTNM-n method using Taylor series expansion technique can be considered as a representative of a local derivative-based function approximation method, while the JTCNM-n method employing Chebyshev polynomial interpolation technique can be understood as a hyper-interpolation-based function approximation approach.

2.2.3.1 Taylor series expansion technique

Define $y = f(x) : \Omega_x \subset \mathbb{R} \to \mathbb{R}$ as a univariate function being $(n+1)$ times continuously partially differentiable on a set containing the domain $\Omega_x = [x_l, x_u]$. Clearly, $\bar{x} = \frac{x_l + x_u}{2}$ is the central point of the domain Ω_x, thus the nth-order Taylor series expansion of the function $y = f(x)$ around the central point is

$$y = f(\bar{x}) + f'(\bar{x})\delta x + \frac{f''(\bar{x})}{2!}\delta x^2 + \cdots + \frac{f^{(n)}(\bar{x})}{n!}\delta x^n + R_n(\delta x)\,, \tag{2.27}$$

where $\delta x = x - \bar{x}$ indicates the distance between the mapping point x and the expansion point $\bar{x}$. We further denote its nth-order Taylor truncation polynomial as

$$y \approx P_{n,1}(\bar{x}, \delta x) = a_0 + a_1\delta x + a_2\delta x^2 + \cdots + a_n\delta x^n, \quad \text{and} \quad a_n = \frac{f^{(n)}(\bar{x})}{n!}\,. \tag{2.28}$$

The truncation error is defined as the difference between $f(x)$ and $P_{n,1}(\bar{x}, \delta x)$, which can be obtained in terms of a repeated usage of Rolle's theorem [150], that is, $\forall x \in [x_l, x_u]$, $\exists\, \xi(x) \in (x_l, x_u)$, **s.t.**

$$R_n(\delta x) = \frac{f^{(n+1)}(\xi)}{(n+1)!}\delta x^{n+1}\,, \tag{2.29}$$

let the supremum norm of $f^{(n+1)}(\xi)$ over the domain $\Omega_x = [x_l, x_u]$ being M_{n+1}, thus the supremum of the truncation error is

$$|R_n(\delta x)| \leqslant \frac{M_{n+1}}{(n+1)!}\delta x^{n+1} \leqslant \frac{M_{n+1}}{(n+1)!}\frac{(x_u - x_l)^{n+1}}{2^{n+1}}\,. \tag{2.30}$$

As a consequence, a nth-order Taylor polynomial of an arbitrary univariate function can be easily obtained by means of (2.28), meanwhile (2.29) clearly illustrates that the polynomial truncation error not only rides on the $(n+1)$th-order derivative $f^{(n+1)}(\xi)$ and but also on the displacement of the mapping point relative to the expansion point δx.

2.2.3.2 Chebyshev polynomial interpolation technique

As one well-known rigorous computation tool for performing interval arithmetic, the Taylor series expansion technique is widely adopted to propagate not only the information about function values, but also convey much other properties about the function itself [149, 151]. However, it is very likely to yield fairly poor approximation results, especially in the margins of a relatively large expansion domain. Besides, if a nonlinear function is not $n+1$ times continuously differentiable over the expansion domain, the Taylor series expansion method is also not applicable.

In order to overcome these shortcomings, a natural alternative is to employ a better polynomial approximation instead of the Taylor series expansion method [149]. As a

hyper-interpolation-based function approximation approach, the Chebyshev polynomial interpolation technique is therefore investigated in the JT framework. The accuracy and efficiency of the Chebyshev polynomial interpolation method strongly depend on two factors, including the number of the interpolating nodes and the type of the selected orthogonal basis. It is worth to highlight that the selections of these two factors should satisfy not only the convenient implementation of the basic polynomial algebra, such as polynomial addition, multiplication, composition and so on, but also the compact calculation of truncation remainders. In what follows, the Chebyshev basis and 100 interpolation nodes are employed by default in order to refine the implementation of basic Chebyshev polynomial algebra and calculate truncation errors [152, 153].

Given a piecewise differentiable, univariate function $y = f(x') : \Omega_{x'} \subset \mathbb{R} \to \mathbb{R}$, where $\Omega_{x'} = [x_l, x_u]$. Using the following linear transformation,

$$x = \frac{2x'}{x_u - x_l} - \frac{x_l + x_u}{x_u - x_l}\,, \tag{2.31}$$

the function $y = f(x')$ can be converted into

$$y = f(-\frac{x_l - x_u}{2}x + \frac{x_l + x_u}{2}) = g(x)\,, \tag{2.32}$$

where $\Omega_x = [-1, 1]$. The Chebyshev polynomials of the first kind, defined by the following recurrence relations, are adopted in the interpolation procedure,

$$T_0(x) = 1\,, \quad T_1(x) = x\,, \quad T_{n+1}(x) = 2xT_n(x) - T_{n-1}(x)\,, \tag{2.33}$$

furthermore, the $n+1$ zeros of $T_{n+1}(x)$ are expressed as

$$x_i^* = \cos\frac{(i + 1/2)\pi}{n+1}\,, \quad i = 0, \cdots, n\,. \tag{2.34}$$

The nth-order Chebyshev polynomial can be employed to interpolate the function $g(x)$ in (2.32) at the $n+1$ zeros of $T_{n+1}(x)$ as

$$g(x) = P_{n,1}(x) + R_n(x)\,, \quad P_{n,1}(x) = \sum_{\gamma=0}^{n}{}' a_\gamma T_\gamma(x)\,, \tag{2.35}$$

where the pre-superscript (i.e., prime) denotes that the first term of coefficients $'a_\gamma$ has to be halved. Assume $n > 0$, γ, $\bar{\gamma} \leqslant n$, thus

$$\sum_{i=0}^{n} T_\gamma(x_i)T_{\bar{\gamma}}(x_i) = K_{\bar{\gamma}}\delta_{\gamma\bar{\gamma}}\,, \tag{2.36}$$

where

$$K_{\bar{\gamma}} = \begin{cases} n+1, & \bar{\gamma} = 0\,, \\ \dfrac{1}{2}(n+1), & 1 \leqslant \bar{\gamma} \leqslant n\,, \end{cases}\,, \quad \delta_{\gamma\bar{\gamma}} = \begin{cases} 0 & \gamma \neq \bar{\gamma}\,, \\ 1 & \gamma = \bar{\gamma}\,, \end{cases}\,.$$

Clearly, the function $g(x_i)$ equals $P_{n,1}(x_i)$ exactly at the interpolating nodes. Using (2.36), one obtains

$$\sum_{i=0}^{n} f(x_i)T_\gamma(x_i) = \sum_{\bar{\gamma}=0}^{n}{}' a_{\bar{\gamma}} \sum_{i=0}^{n} T_{\bar{\gamma}}(x_i)T_\gamma(x_i) = \sum_{\bar{\gamma}=0}^{n}{}' a_{\bar{\gamma}} K_{\bar{\gamma}} \delta_{\bar{\gamma}\gamma} = \frac{1}{2}(n+1)a_\gamma \,, \tag{2.37}$$

thus the coefficients a_γ can be calculated by means of (2.37),

$$a_\gamma = \frac{2}{n+1} \sum_{i=0}^{n} f(x_i)T_\gamma(x_i) \,. \tag{2.38}$$

Using (2.38), the coefficient $'a_\gamma$ in (2.35) is further computed as follows,

$$'a_\gamma = \begin{cases} \dfrac{1}{n+1} \displaystyle\sum_{i=0}^{n} f(x_i), & \gamma = 0, \\ \dfrac{2}{n+1} \displaystyle\sum_{i=0}^{n} f(x_i)T_\gamma(x_i), & 1 \leqslant \gamma \leqslant n \,. \end{cases} \tag{2.39}$$

It is apparent that the direct computational cost of (2.39) is $O(n^2)$ operations. However, reference [154] shows that an interval arithmetic adaptation technique of fast Fourier transform enables to accelerate the aforementioned computation to $O(n \log n)$ operations.

Similar to the Taylor expansion case, a repeated application of Rolle's theorem [150] is again employed to compute the truncation error $R_n(x)$, that is, $\forall x \in [-1, 1]$, $\exists \xi(x) \in (-1, 1)$, **s.t.**

$$R_n(x) = \frac{g^{(n+1)}(\xi)}{(n+1)!} W_{n+1,x_i^*}(x) \,, \quad W_{n+1,x_i^*}(x) = \prod_{i=0}^{n}(x - x_i^*) = \frac{1}{2^n} T_{n+1} \,. \tag{2.40}$$

Instead of the interpolation interval $\Omega_x = [-1, 1]$, if $\Omega_{x'} = [x_l, x_u]$ is taken into consideration, (2.31) must be employed to recalculate the new interpolating nodes $x'^*_i \in \Omega_{x'}$ using $x_i^* \in \Omega_x$,

$$x'^*_i = -\frac{x_l - x_u}{2} x_i^* + \frac{x_l + x_u}{2} \,, \tag{2.41}$$

therefore,

$$W_{n+1,x'^*_i}(x') = \frac{(x_u - x_l)^{n+1}}{2^{n+1}} W_{n+1,x_i^*}(x) \,. \tag{2.42}$$

Using (2.32) and (2.41),

$$g^{(n+1)}(\xi) = f^{(n+1)}(-\frac{x_l - x_u}{2}\xi + \frac{x_l + x_u}{2}) = f^{(n+1)}(\xi') \,, \tag{2.43}$$

where $\xi' \in \Omega_{x'}$. We further get $\forall x' \in [x_l, x_u]$, $\exists \xi'(x') \in (x_l, x_u)$, **s.t.**

$$R_n(x') = \frac{f^{(n+1)}(\xi')}{(n+1)!} W_{n+1,x'^*_i}(x') = \frac{f^{(n+1)}(\xi')}{(n+1)!} \frac{(x_u - x_l)^{n+1}}{2^{n+1}} W_{n+1,x_i^*}(x) \,. \tag{2.44}$$

Assume the supermum norm of $f^{(n+1)}(\xi')$ over $\Omega_{x'}$ being M_{n+1}, thus the supermum of the truncation error is,

$$\begin{aligned} |R_n(x')| &= \frac{f^{(n+1)}(\xi')}{(n+1)!}\frac{(x_u - x_l)^{n+1}}{2^{n+1}} W_{n+1,x_i^*}(x) \\ &= \frac{f^{(n+1)}(\xi')}{(n+1)!}\frac{(x_u - x_l)^{n+1}}{2^{2n+1}} T_{n+1} \leqslant \frac{M_{n+1}}{(n+1)!}\frac{(x_u - x_l)^{n+1}}{2^{2n+1}} . \end{aligned} \tag{2.45}$$

Summarizing, (2.35) displays nth-order Chebyshev interpolation polynomials for approximating basic univariate functions, (2.39) and (2.45) give detailed formulae for calculating polynomial coefficients and estimating the supremum of the truncation error. In particular, when the function approximated is piecewise differentiable, the Chebyshev polynomial interpolation technique is preferable since it enables to achieve the uniform convergence, while the Taylor series expansion technique is not available. Note in particular that the remainder of the Chebyshev interpolation polynomial can be obtained in a compact, explicit and closed formula. A straightforward comparison between (2.30) and (2.45) shows that the truncation error of the Chebyshev interpolation method associated with expansion order n, roughly speaking, can be scaled down by a factor of 2^n with comparison to that related to the Taylor series expansion method with the same order. In other words, the Chebyshev interpolation method possesses a much smaller supremum of the truncation error with respect to the Taylor expansion method.

2.2.4 Polynomial approximation to a multi-variable function

Using the approximation methods proposed in the univariate case, Taylor and Chebyshev polynomial approximations of multi-variable functions are achieved in two steps: firstly perform the resulting polynomial expansion of a univariate function and then do the polynomial composition operation. As an illustrative example, a bivariate function $\sin(x_1 + x_2)$ is expanded up to nth-order around the expansion point $\boldsymbol{x} = (0, 0)$ to obtain the resulting polynomial $P_{n,2}(\boldsymbol{x})$ within an acceptable error. A general method is to expand the univariate function $\sin y$ around $y = 0$ as $P_{n,1}(y)$, then do the polynomial composition between $P_{n,1}(y)$ and $y = P_{1,2}(\boldsymbol{x}) = x_1 + x_2$. In other words, the final nth-order polynomial $P_{n,2}(\boldsymbol{x})$ equals the result of polynomial composition $P_{n,1}\left(P_{1,2}(\boldsymbol{x})\right)$. Apparently, it is easy to compute approximation polynomials of basic univariate functions through the algorithms proposed in subsection 2.2.3. Therefore, the key issue is how to perform the algorithm of the polynomial composition operation.

The polynomial composition, denoted as the notation "$\circ$", is defined to evaluate the polynomial ${}^{a}P_{n,d}(\boldsymbol{\beta}(\boldsymbol{x}))$ at a d-dimensional polynomial vector $\boldsymbol{x} = {}^{b}P_{\tilde{n},\tilde{d}}(\boldsymbol{\beta}(\boldsymbol{\xi}))$. Note that, in the polynomial composition procedure, there is no restriction on the degree and variable number of two polynomials, i.e., n and d can be respectively less, equal, or bigger than $\tilde{n}$ and $\tilde{d}$. The polynomial composition rule is

$${}^{a}P_{n,d}(\boldsymbol{\beta}(\boldsymbol{x})) \circ \left[{}^{b}P_{\tilde{n},\tilde{d}}(\boldsymbol{\beta}(\boldsymbol{\xi}))\right]^{d} \to {}^{r}P_{n,d}(\boldsymbol{\beta}(\boldsymbol{\xi}))\,, \tag{2.46}$$

where $[\cdot]^d$ indicates d-dimension polynomial vector. Based on the combination of the aforementioned algebra defined in the polynomial space, (2.46) can be easily achieved and applied to both Taylor and Chebyshev polynomial cases. Similar to the polynomial multiplication and division operations, due to the necessity of the basis transformation, the polynomial composition in the Chebyshev case is a little more complex than that in the Taylor case. Besides, a key point needs to be emphasized is that, in the composition procedure of Chebyshev polynomials, the range of $\boldsymbol{x} = {}^bP_{\tilde{n},\tilde{d}}(\boldsymbol{\beta}(\boldsymbol{\xi}))$, calculated over the domain $\boldsymbol{\xi} \in \Omega_\xi \in [-1,1]^d$, should satisfy the condition $\boldsymbol{x} \in \Omega_x \in [-1,1]^d$. However, it is hard to guarantee this requirement. As such, (2.2) is used to map the polynomial vector $\boldsymbol{x} = {}^bP_{\tilde{n},\tilde{d}}(\boldsymbol{\beta}(\boldsymbol{\xi}))$ from the generic hyper-rectangle domain Ω_x to the standard hyper-rectangle domain $[-1,1]^d$. In this process, the key issue is to bound the domain $\boldsymbol{x} \in [min_{\boldsymbol{\xi}\in\Omega_\xi}\ {}^bP_{\tilde{n},\tilde{d}}(\boldsymbol{\beta}(\boldsymbol{\xi})), max_{\boldsymbol{\xi}\in\Omega_\xi}\ {}^bP_{\tilde{n},\tilde{d}}(\boldsymbol{\beta}(\boldsymbol{\xi}))]$. Generally speaking, the overestimation in the $\boldsymbol{x}$ domain often decreases the accuracy of the Chebyshev approximation method. Some valid approaches aiming at alleviating this issue has been proposed [95, 149].

2.2.5 Numerical simulations

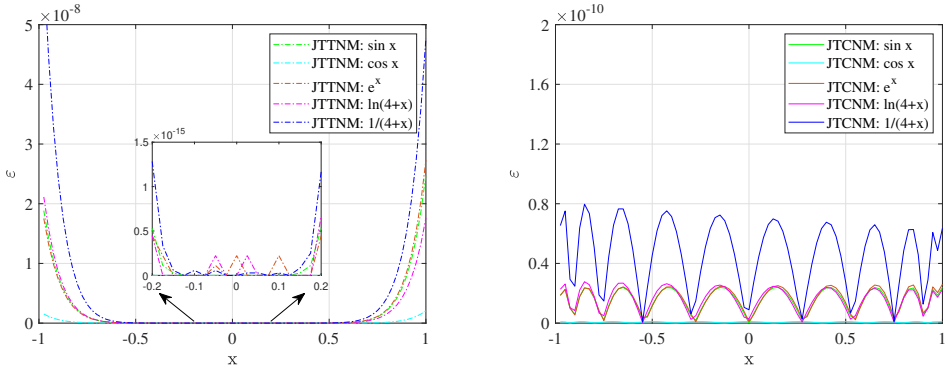
The performance and feasibility of automatic Taylor and Chebyshev polynomial approximation methods for approximating basic functions or their composite functions are assessed by numerical simulations. As an illustrative example, both JTTNM-n and JTCNM-n methods are employed to approximate several basic functions and their composite functions at different expansion orders.

2.2.5.1 Taylor and Chebyshev approximations to basic functions

In this simulation scenario, the newly proposed JTTNM-n and JTCNM-n methods are implemented at different expansion orders in order to approximate basic univariate functions, which contain $y = \sin x$, $y = \cos x$, $y = e^x$, $y = \ln(4+x)$ and $y = \frac{1}{4+x}$. Totally 80 sampling points are evenly distributed on the interval of interest $\Omega_x = [-1,1]$, at which the accuracy of both JTTNM-n and JTCNM-n methods are evaluated.

The absolute error is defined as the difference between the real value, computed by the basic function itself at the sampling points taken from the expansion interval $\Omega_x = [-1,1]$, and the approximate value, evaluating the polynomial results of both JTTNM-n and JTCNM-n methods at the same sampling points. To explore absolute error distributions, both JTTNM-10 and JTCNM-10 methods are respectively adopted to approximate the selected univariate functions. Figure 2.2 shows the absolute error distribution evaluating the polynomial result computed by either JTTNM-10 or JTCNM-10 at the preassigned 80 sampling points. The maximum absolute error is defined as the supremum of absolute errors. The comparison between Fig. 2.2a and Fig. 2.2b exhibits that the maximum absolute error of the JTCNM-10 method is three orders of magnitude smaller than that of the JTTNM-10 method. Figure 2.2a also shows that the approximation error of the

JTTNM-10 method increases as the sampling point drifts away from the expansion point, such that the maximum absolute error is obtained at the margins of the expansion interval $\Omega_x = [-1, 1]$. Per contra, a JTCNM-10 implementation possesses a much more uniform error distribution on the expansion interval, as shown in Fig. 2.2b.



(a) Absolute errors of JTTNM-10 method

(b) Absolute errors of JTCNM-10 method

Figure 2.2: Absolute errors of both JTTNM-10 and JTCNM-10 for approximating basic functions.

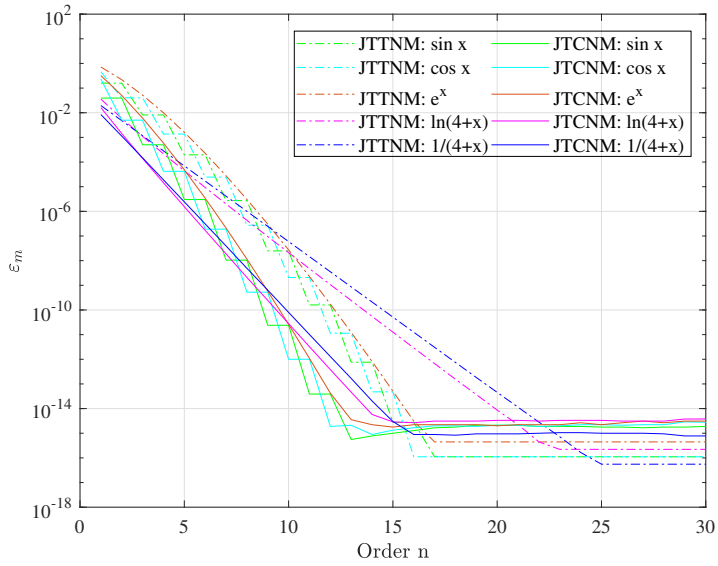


Figure 2.3: Maximum absolute errors of both JTTNM-n and JTCNM-n vs. the expansion order n for approximating basic functions.

Furthermore, it is clear that the adopted expansion order n has a significant influence on the performance of both JTTNM-n and JTCNM-n methods. Figure 2.3 illustrates maximum absolute errors of both JTTNM-n and JTCNM-n methods versus the expansion

order n in the approximation of selected univariate functions. The result reveals that the accuracy of both methods improves as the increase of the expansion order n until that the approximation error and the computer's rounding error are of the same order of magnitude. In this simulation scenario, the critical value of the expansion order is $n = 15$. On the other hand, Fig. 2.3 also illustrates that the approximation error of the JTCNM-n method implemented at the order $n \leqslant 15$ is much smaller than that of the JTTNM-n method implemented at the same order. On the contrary, if the adopted order $n > 15$, both methods are accurate enough for approximating almost all selected basic functions since their approximation errors are much less than the computer's rounding error. Therefore, if the expansion order n exceeds 15, even if the approximation accuracy of the JTTNM-n method is a little better than that of the JTCNM-n method, it almost makes little sense.

2.2.5.2 Taylor and Chebyshev approximations to complicated functions

As stated above, both Taylor series expansion method and Chebyshev polynomial interpolation approach (i.e., JTTNM-n and JTCNM-n methods) have been employed to approximate five basic univariate functions. In this simulation scenario, the feasibility of both methods is further validated through the approximation of a more complicated univariate function $\frac{e^{\sin x}}{4+\cos x}$. Simulation results are compared with real results obtained by pointwise numerical calculations at same sampling points, which are uniformly taken from the interval $\Omega_x = [-1, 1]$.

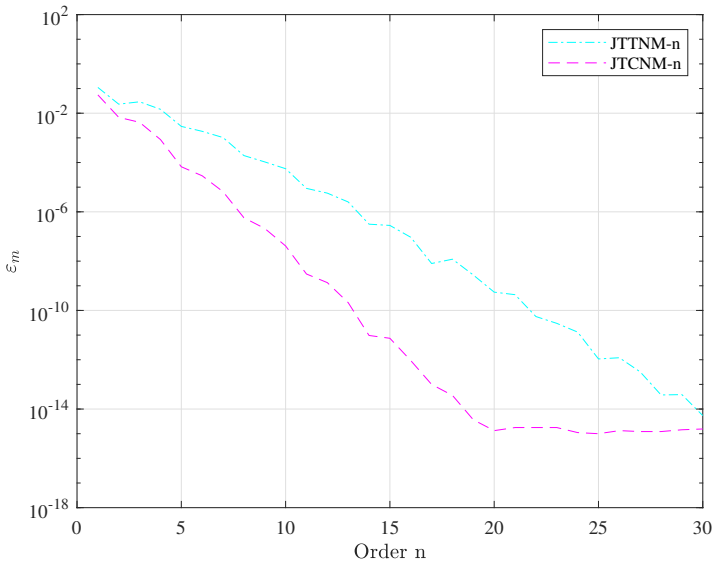


Figure 2.4: Maximum absolute errors of both JTTNM-n and JTCNM-n vs. the expansion order n for approximating a complicated univariate function $\frac{e^{\sin x}}{4+\cos x}$.

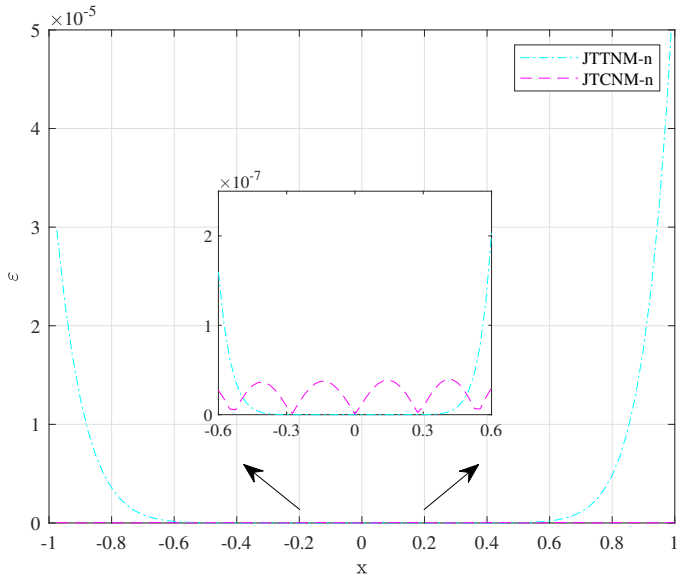


Figure 2.5: Absolute errors of both JTTNM-10 and JTCNM-10 for approximating a complicated univariate function $\frac{e^{\sin x}}{4+\cos x}$.

Figure 2.4 illustrates maximum absolute errors obtained by both JTTNM-n and JTCNM-n methods implemented at different expansion orders n. The simulation result displays that the accuracy of the JTCNM-n implementation is much better than that of the JTTNM-n implementation at all orders less than 30, where the approximation error reaches the magnitude of the rounding error of the computer calculation. Meanwhile, the approximation accuracy of both methods increases as the expansion order n also does. Figure 2.5 exhibits absolute error distributions of both JTTNM-10 and JTCNM-10. Similar to the conclusion obtained when approximate basic univariate functions, the JTTNM-10 method possesses a very good accuracy at a small neighborhood around the expansion point for approximating $\frac{e^{\sin x}}{4+\cos x}$, while has a much bigger error at the edges of the expansion interval. Conversely, the approximation error of the JTCNM-10 method evaluating at the preselected 80 sampling points on the interval $\Omega_x = [-1, 1]$ is tiny, which again illustrates the error distribution of the JTCNM-n method is more uniform as compared to that of the JTTNM-n method. Furthermore, Fig. 2.5 also proves that the JTCNM-10 method occupies much faster rate of convergence relative to the JTTNM-10 method.

2.3 Polynomial-based flow propagation

Generally, it is difficult to obtain a closed form solution of an ODE system, except perhaps for solving a simplified dynamical system. As an alternative, numerical integration

methods, such as Euler's method and Runge-Kutta methods, are often employed to accurately transport the state along the flow of a complicated vector field. In particular, these numerical integration methods enable to solve general dynamical systems without any restriction on their complexity. The unique price to pay for this superior performance is the high computational cost due to the need of small integral step-sizes adopted in the numerical integrating process [33]. Besides, in order to extract the statistical information of an ODE system, a MC simulation has to carry out a mass of repetitive numerical integrals over the same time interval $[t_0\,, t]$ at different initial random states, which makes it very inefficient, especially when the number of sampling points is huge. For instance, to perform accurate covariance analysis of initial trajectory errors, a general MC simulation often needs to propagate at least hundreds of thousands of satellite trajectories, such that a great deal of computational burden is requisite. This section is devoted to proposing a new semi-analytical method for the ordinal computation of the state image under the flow map in an accurate and efficient way.

To address this methodology, we consider an initial value problem,

$$\dot{\boldsymbol{x}} = \boldsymbol{f}\left(\boldsymbol{x}, t\right), \quad \boldsymbol{x}(t_0) = \boldsymbol{x}_0\,, \tag{2.47}$$

where $\boldsymbol{x}_0 \in \Omega_0 \subset \mathbb{R}^d$, $t \in \mathbb{R}$. The usual numerical integration methods only enable to propagate one trajectory with the initial state $\boldsymbol{x}_0 \in \Omega_0$ from t_0 to t each time and obtain the solution of (2.47), i.e., $\boldsymbol{x}_t = \boldsymbol{\Phi}\left(t; t_0, \boldsymbol{x}_0\right)$. On the contrary, instead of the pointwise propagation of $\boldsymbol{x}_0$, this section is expected to transport an initial state set Ω_0 around $\boldsymbol{x}_0$ along the flow of (2.47) from time t_0 to t in order to efficiently obtain the final state set $\Omega_t = \Phi\left(t; t_0, \Omega_0\right)$. To achieve this goal, a parameterization method is employed to represent the state set Ω with the usage of a set of polynomial variables, which are often referred to as JT variables since they are the basic elements operated in the JT framework. Thereafter, a set of polynomial-based numerical integration methods are further proposed to transport the JT state from t_0 to t, such that a semi-analytical polynomial approximation solution of (2.47), parameterizing the final state set Ω_t, is obtained in the JT framework. Finally, a polynomial evaluation method is investigated to compute the JT state set Ω_t at different initial states. Generally, a semi-analytical polynomial solution is much faster to extract statistical information from an ODE system than the classical MC method. Therefore, there is a great potential for this semi-analytical method for solving general ODE systems.

2.3.1 Parameterization method

In principle, a set of state vectors $\boldsymbol{x}$ in a neighborhood $\mathcal{N}$ around $\bar{\boldsymbol{x}}$ can be parameterized by a JT vector $[\boldsymbol{x}] = \bar{\boldsymbol{x}} + \delta\boldsymbol{x}$, such that the local nature of state vectors $\boldsymbol{x}$ over the neighborhood $\mathcal{N}$ can be represented by the statistical information of the deviation vector $\delta\boldsymbol{x}$. The notation $[\boldsymbol{x}]$ indicates that it is a JT vector, which must be described by a polynomial vector in its deviation vector $\delta\boldsymbol{x}$. We further consider a nonlinear multivariable vector field $\boldsymbol{f}(\boldsymbol{x}) : \boldsymbol{\Omega_x} \subset \mathbb{R}^d \to \mathbb{R}^d$. Using the algebra defined in the polynomial space $[\mathcal{P}_{n,d}(\boldsymbol{\beta_\gamma}(\boldsymbol{x})), \odot]$ (see subsection 2.2), the function vector $\boldsymbol{f}(\boldsymbol{x})$ can be parameterized by a set of nth-order Taylor or Chebyshev polynomials in d-dimensional variables.

2.3.1.1 Taylor case

The nth-order Taylor expansion (i.e., JTTNM-n) of a d-dimensional multivariable vector field $\boldsymbol{f}(\boldsymbol{x})$ can be expressed in the monomial basis

$$ {}_i\boldsymbol{f}(\boldsymbol{x}) = {}_i\boldsymbol{P}_{n,d}(\boldsymbol{\delta x}^{\boldsymbol{\gamma}}) + {}_i\boldsymbol{R}_{n,d}(\boldsymbol{\delta x}) = \sum_{|\boldsymbol{\gamma}|\leq n} {}_i\boldsymbol{c}_{\boldsymbol{\gamma}} \boldsymbol{\delta x}^{\boldsymbol{\gamma}} + {}_i\boldsymbol{R}_{n,d}(\boldsymbol{\delta x}) \,, \tag{2.48} $$

where ${}_i\boldsymbol{f}(\boldsymbol{x}), i = 1, \cdots, d$ indicates the ith element of the function vector $\boldsymbol{f}(\boldsymbol{x})$, and

$$ \boldsymbol{\delta x}^{\boldsymbol{\gamma}} = \prod_{j=1}^{d} \delta x_j^{\gamma_j} \,, \quad {}_i\boldsymbol{c}_{\boldsymbol{\gamma}} = \frac{1}{\gamma_1! \cdots \gamma_d!} \frac{\partial^{\boldsymbol{\gamma}} {}_i\boldsymbol{f}(\boldsymbol{x})}{\partial x_1^{\gamma_1} \cdots \partial x_d^{\gamma_d}}. \tag{2.49} $$

2.3.1.2 Chebyshev case

In contrast to the local approximation achieved by the JTTNM-n, the Chebyshev polynomial interpolation method(i.e., JTCNM-n) attempts a global approximation of the nonlinear multi-variable vector field $\boldsymbol{f}(\boldsymbol{x})$ over a prescribed neighborhood $\mathcal{N}$. The nth-order Chebyshev approximation polynomial is

$$ {}_i\boldsymbol{f}(\boldsymbol{x}) = {}_i\boldsymbol{P}_{n,d}(\boldsymbol{T}_{\boldsymbol{\gamma}}(\boldsymbol{\delta x})) + {}_i\boldsymbol{R}_{n,d}(\boldsymbol{\delta x}) = \sum_{|\boldsymbol{\gamma}|\leq n} {}_i\boldsymbol{c}_{\boldsymbol{\gamma}} \boldsymbol{T}_{\boldsymbol{\gamma}}(\boldsymbol{\delta x}) + {}_i\boldsymbol{R}_{n,d}(\boldsymbol{\delta x}) \,, \tag{2.50} $$

where

$$ \boldsymbol{T}_{\boldsymbol{\gamma}}(\boldsymbol{\delta x}) = \prod_{j=1}^{d} T_{\gamma_j}(\delta x_j) \,, $$

and the corresponding coefficient set ${}_i\boldsymbol{c}_{\boldsymbol{\gamma}}$ can be initially computed by means of the Chebyshev polynomial interpolation at the selected Chebyshev nodes.

2.3.1.3 Comparisons between Taylor and Chebyshev methods

Note that JTTNM-n and JTCNM-n are typical representations of a local derivative-based method and a hyper-interpolation-based method. As stated above, the JTTNM-n method adopts a local derivative-based function approximation in the vicinity of the expansion point, generally selecting the central point of the neighborhood $\mathcal{N}$, while the JTCNM-n method attempts a hyper-interpolation technique occupying a global convergence over the whole neighborhood $\mathcal{N}$. Furthermore, the JTCNM-n method only requires the function approximated being piecewise smooth and continuous, while the JTTNM-n method requires it possessing $(n+1)$th-order continuous differentiable [94].

Note that nth-order polynomials in (2.48) obtained by the JTTNM-n method can be further written as

$$ {}_i\boldsymbol{P}_{n,d}(\delta\boldsymbol{x}^{\boldsymbol{\gamma}}) = {}_i\boldsymbol{P}_{n,d}(\boldsymbol{0}) + {}_i\mathcal{T}_{n,d}(\delta\boldsymbol{x}) = {}_i\boldsymbol{a}_0 + \sum_{1\leq|\boldsymbol{\gamma}|\leq n} {}_i\boldsymbol{a}_{\gamma_1\ldots\gamma_d}\delta x_1^{\gamma_1}\ldots\delta x_d^{\gamma_d}, \tag{2.51} $$

where ${}_i\mathcal{T}_{n,d}(\delta\boldsymbol{x})$, depending on the deviation vector $\delta\boldsymbol{x}$, is the Taylor polynomial representation of the ith component of the deviation vector $\delta\boldsymbol{f}$, ${}_i\boldsymbol{a}_0 = {}_i\boldsymbol{f}(\bar{\boldsymbol{x}})$ is the zeroth-order term, and ${}_i\boldsymbol{a}_{\gamma_1\ldots\gamma_d}$ are coefficients of the Taylor polynomial. In contrast to the JTTNM-n method, the JTCNM-n method carries out the Chebyshev polynomial interpolation over a hyper-rectangle neighborhood $\mathcal{N}$, thus the corresponding nth-order polynomial (2.50) can be expressed as

$$ \begin{aligned} {}_i\boldsymbol{P}_{n,d}(\boldsymbol{T}_{\boldsymbol{\gamma}}(\delta\boldsymbol{x})) &= {}_i\boldsymbol{P}_{n,d}(\boldsymbol{T}_{\boldsymbol{\gamma}}(0)) + {}_i\mathcal{C}_{n,d}(\boldsymbol{T}_{\boldsymbol{\gamma}}(\delta\boldsymbol{x})) \\ &= {}_i\boldsymbol{b}_0 + \sum_{1\leq|\boldsymbol{\gamma}|\leq n} {}_i\boldsymbol{b}_{\gamma_1\ldots\gamma_d} T_{\gamma_1}(\delta x_1)\ldots T_{\gamma_d}(\delta x_d)\,, \end{aligned} \tag{2.52} $$

where ${}_i\mathcal{C}_{n,d}(\boldsymbol{T}_{\boldsymbol{\gamma}}(\delta\boldsymbol{x}))$, again depending on the deviation vector $\delta\boldsymbol{x}$, denotes the Chebyshev expansion of the ith component of the deviation vector $\delta\boldsymbol{f}$, $b_0 = {}_i\boldsymbol{P}_{n,d}(\boldsymbol{T}_{\boldsymbol{\gamma}}(0))$ is the zeroth-order term, and ${}_i\boldsymbol{b}_{\gamma_1\ldots\gamma_d}$ are coefficients of the Chebyshev polynomial. Note that if $\delta\boldsymbol{x} = 0$, the approximation error of the JTTNM-n method, defined as the difference between the real value $\boldsymbol{f}(\bar{\boldsymbol{x}})$ and its approximate value $\boldsymbol{P}_{n,d}(\boldsymbol{0})$, only results from the tiny rounding error. On the contrary, if $\delta\boldsymbol{x} = 0$ is taken into consideration in the JTCNM-n method, the approximation error, defined as the difference between the real value $\boldsymbol{f}(\bar{\boldsymbol{x}})$ and its approximate value $\boldsymbol{P}_{n,d}(\boldsymbol{T}_{\boldsymbol{\gamma}}(0))$, roots in both the high-order truncation error and rounding error. Clearly, even if both methods adopt the same expansion order, the approximation errors evaluating at the point $\bar{\boldsymbol{x}}$ do not coincide, as well as the coefficients ${}_i\boldsymbol{a}_{\gamma_1\ldots\gamma_d}$ and ${}_i\boldsymbol{b}_{\gamma_1\ldots\gamma_d}$.

2.3.2 Design of polynomial-based integrators

To perform a JT-based numerical integration, all arithmetic operations of the usual numerical propagator must be replaced with their polynomial counterparts in the JT framework. Using the parameterization method, an initial state set and the vector field of an ODE system $\dot{\boldsymbol{x}} = \boldsymbol{f}(\boldsymbol{x}, t)$ could be expressed as a JT vector consisting of nth-order Taylor or Chebyshev polynomials. At this point, it is worth to remark a particularity of JT-based numerical integrators with adaptive step-size control, such as Runge-Kutta-Fehlberg procedures in polynomial form.

The Runge-Kutta (RK) methods implemented in the JT scheme (JTRK) are the only ones considered in this thesis. These methods can be divided in two categories, including fixed or variable integral step-size cases. In general, a RK with variable integral step-size, such as the well-known RK45 and RK78 of respective orders 5 and 8, gives a better performance but, some tips have to be taken in the integral step-size control when doing JT counterparts. In what follows the implementation of a JTRK78 algorithm is discussed.

Other implementations, like the one of JTRK45, can be performed using the analogous error estimation and integral step-size control strategy.

It is worth mentioning that the nth-order polynomial $\boldsymbol{P}_{n,d}(*)$ in (2.51) and (2.52) can be trivially regarded as a JT variable, where the asterisk $*$ indicates either Taylor or Chebyshev basis. Therefore, for the sake of simplification, the notation of JT variables $[\boldsymbol{x}]^n_*$ can be used to represent the nth-order polynomial $\boldsymbol{P}_{n,d}(*)$. To explain the polynomial integration procedure of an ODE system (2.47), a well approximated JT state vector $[\boldsymbol{x}_j]^n_*$ is assumed being known at time t_j. For a given time step h, two estimates of the real state $\boldsymbol{x}(t_{j+1})$ at $t_{j+1} = t_j + h$, are provided by both 7th and 8th-order JTRK algorithms, respectively

$$[\check{\boldsymbol{x}}_{j+1}]^n_* = [\boldsymbol{x}_j]^n_* + h\,\boldsymbol{\phi}_7(t_j, [\boldsymbol{x}_j]^n_*, h), \qquad [\hat{\boldsymbol{x}}_{j+1}]^n_* = [\boldsymbol{x}_j]^n_* + h\,\boldsymbol{\phi}_8(t_j, [\boldsymbol{x}_j]^n_*, h). \tag{2.53}$$

The rationale underlying a RK method with variable integral step-size control is that when the two estimates are close enough (this is, a suitable norm of their difference is below a selected threshold value $\boldsymbol{\epsilon}$), the propagation with integral step-size h is considered successful, while when they differ dramatically, the integral step-size must be adjusted and recomputed. A trade-off between a fixed accuracy threshold and the computational burden, requires also to look for the maximum integral step-size suitable to obtain a good estimate of the real state $\boldsymbol{x}(t_{j+1})$ when the estimators differ, as well as for the computation of $\boldsymbol{x}(t_{j+2})$ at time t_{j+2} that will follow in the propagation. Note that, at time t_{j+1}, the difference between the two estimates of the state vector are in the Taylor polynomial form,

$$[\check{\boldsymbol{x}}_{j+1}]^n_{\mathcal{T}} - [\hat{\boldsymbol{x}}_{j+1}]^n_{\mathcal{T}} = \sum_{|\boldsymbol{\gamma}|\leq n} \boldsymbol{c}^j_{\boldsymbol{\gamma}} \boldsymbol{\delta x}^{\boldsymbol{\gamma}} = \sum_{|\boldsymbol{\gamma}|\leq n} \boldsymbol{c}^j_{\gamma_1\ldots\gamma_d} \delta x_1^{\gamma_1} \ldots \delta x_d^{\gamma_d}\,, \tag{2.54}$$

or in the Chebyshev polynomial form,

$$[\check{\boldsymbol{x}}_{j+1}]^n_{\mathcal{C}} - [\hat{\boldsymbol{x}}_{j+1}]^n_{\mathcal{C}} = \sum_{|\boldsymbol{\gamma}|\leq n} \boldsymbol{c}^j_{\boldsymbol{\gamma}} \boldsymbol{T}_{\boldsymbol{\gamma}}(\boldsymbol{\delta x}) = \sum_{|\boldsymbol{\gamma}|\leq n} \boldsymbol{c}^j_{\gamma_1\ldots\gamma_d} T_{\gamma_1}(\delta x_1) \ldots T_{\gamma_d}(\delta x_d)\,. \tag{2.55}$$

The classical pointwise propagation algorithm estimates this difference considering some norm of the d-dimension coefficient vector $\boldsymbol{c}^j_{0\ldots0}$, but in our case the differences (2.54) and (2.55) are nth-order polynomials in d variables and all terms must be taken into account. Assuming that an acceptable error threshold vector is $\boldsymbol{\epsilon} = (\epsilon_1 \ldots \epsilon_n)$, each component corresponding to the threshold for the coefficients of the polynomial terms occupying order $|\boldsymbol{\gamma}|$. We further define a norm by[1]

$$C^j_{|\boldsymbol{\gamma}|} = \max_{|\boldsymbol{\gamma}|} \|\boldsymbol{c}^j_{\gamma_1\ldots\gamma_d}\|, \qquad \|\boldsymbol{c}^j_{\gamma_1\ldots\gamma_d}\| = \max_{1\leq i\leq d} |c^j_{i,\gamma_1\ldots\gamma_d}|. \tag{2.56}$$

The condition for a successful integral step-size h at time t_j would be $C^j_{|\boldsymbol{\gamma}|} < \epsilon_{|\boldsymbol{\gamma}|}$ for $|\boldsymbol{\gamma}| \leq n$. In order to further determine the optimal integral step-size for a given accuracy threshold $\boldsymbol{\epsilon}$, let us first introduce

[1] Other definitions of $C^j_{|\boldsymbol{\gamma}|}$ like scaled Euclidean norms or average of absolute values among components could be also considered.

$$\boldsymbol{\Delta}(t,\boldsymbol{x},h)=\frac{\boldsymbol{x}(t+h)-\boldsymbol{x}(t)}{h}. \tag{2.57}$$

Since the JTRK7 integrator has a 7th-order local truncation error, thus

$$-\boldsymbol{\Delta}(t_j,\boldsymbol{x}_j,h)+\boldsymbol{\phi}_7(t_j,[\boldsymbol{x}_j]_*^n,h)=\boldsymbol{N}_7(t_j)\,h^7+O(h^8). \tag{2.58}$$

Analogously, the JTRK8 integrator has an 8th-order accuracy,

$$-\boldsymbol{\Delta}(t_j,\boldsymbol{x}_j,h)+\boldsymbol{\phi}_8(t_j,[\boldsymbol{x}_j]_*^n,h)=\boldsymbol{N}_8(t_j)\,h^8+O(h^9). \tag{2.59}$$

Subtracting the estimates given by (2.53), and using (2.58-2.59), one gets

$$[\check{\boldsymbol{x}}_{j+1}]_*^n-[\hat{\boldsymbol{x}}_{j+1}]_*^n=h\,(\boldsymbol{\phi}_7-\boldsymbol{\phi}_8)=\boldsymbol{N}_7(t_j)\,h^8+O(h^9). \tag{2.60}$$

Neglecting the $O(h^9)$ term, the estimate of the leading coefficient is obtained as follows

$$\boldsymbol{N}_7(t_j)\simeq\frac{[\check{\boldsymbol{x}}_{j+1}]_*^n-[\hat{\boldsymbol{x}}_{j+1}]_*^n}{h^8}, \tag{2.61}$$

and, in terms of (2.56), we have the following estimation for the size of different orders of $\boldsymbol{N}_7(t_j)$

$$N_{7,|\boldsymbol{\gamma}|}^j=\frac{C_{|\boldsymbol{\gamma}|}^j}{h^8},\qquad |\boldsymbol{\gamma}|\le n. \tag{2.62}$$

Let us now assume a new optimal integral step-size h_N at time t_{j+1}. According to (2.62) together with the successful condition, we get

$$C_{|\boldsymbol{\gamma}|}^{j+1}=h_N^8N_{7,|\boldsymbol{\gamma}|}^{j+1}\le\epsilon_{|\boldsymbol{\gamma}|},\qquad |\boldsymbol{\gamma}|\le n. \tag{2.63}$$

Expanding $\boldsymbol{N}_7(t_{j+1})$ around t_j yields $\boldsymbol{N}_7(t_{j+1})=\boldsymbol{N}_7(t_j+h_N)=\boldsymbol{N}_7(t_j)+O(h_N)$, such that (2.63) can be written as

$$C_{|\boldsymbol{\gamma}|}^{j+1}=h_N^8N_{7,|\boldsymbol{\gamma}|}^j+O(h_N^9)\le\epsilon_{|\boldsymbol{\gamma}|},\qquad |\boldsymbol{\gamma}|\le n. \tag{2.64}$$

Then, neglecting the term $O(h_N^9)$, and substituting (2.62) into (2.64), we obtain

$$h_{N,|\boldsymbol{\gamma}|}\le\sqrt[8]{\frac{\epsilon_{|\boldsymbol{\gamma}|}}{N_{7,|\boldsymbol{\gamma}|}^j}}=h\sqrt[8]{\frac{\epsilon_{|\boldsymbol{\gamma}|}}{C_{|\boldsymbol{\gamma}|}^j}},\qquad |\boldsymbol{\gamma}|\le n. \tag{2.65}$$

Clearly, there are $n+1$ values $h_{N,|\boldsymbol{\gamma}|}$ which correspond to $|\boldsymbol{\gamma}|=1,\cdots,n$. The value of h_N advised for the next integral step is the minimum one of these $n+1$ values. Usually h_N is multiplied by a safety factor of 0.9 (in practice, this avoids re-computations of the integral step-size and increases performance).

2.3.3 Semi-analytical polynomial propagation

An appealing application of the JT technique is to provide an arbitrary high-order polynomial representation of the flow of an ODE system w.r.t. the initial state deviation vector $\delta\boldsymbol{x}_0$. Let us consider an initial value problem,

$$\begin{cases} \dot{\boldsymbol{x}} = \boldsymbol{f}\left(\boldsymbol{x}, t\right), \\ \boldsymbol{x}(t_0) = \left[\boldsymbol{x}_0\right], \end{cases} \tag{2.66}$$

where the d-dimensional initial state is defined as a JT vector $[\boldsymbol{x}_0] = \bar{\boldsymbol{x}}_0 + \delta\boldsymbol{x}_0$, intended to describe a set Ω_0 around the nominal state $\bar{\boldsymbol{x}}_0$, $\delta\boldsymbol{x}_0$ describes initial state uncertainties. The aim is to integrate the usual ODE system (2.66) from epoch t_0 to t with the usage of a variable step-size JTRK78 method (see in subsection 2.3.2), or by means of any other numerical integrator, such that the initial JT state vector can be transported along the flow of the dynamical system, i.e., $\boldsymbol{\Phi}\left(t; t_0, \bar{\boldsymbol{x}}_0 + \boldsymbol{\delta x_0}\right)$. Consequently, at time t, the state set $\Omega_t : \boldsymbol{x}(t) = \boldsymbol{\Phi}_t\left(t; t_0, \bar{\boldsymbol{x}}_0 + \delta\boldsymbol{x}_0\right)$ can be represented by a set of n-th Taylor expansion polynomials. This is,

$$[\boldsymbol{x}(t)] = \bar{\boldsymbol{x}}_t + \delta\boldsymbol{x}_t = \bar{\boldsymbol{x}}_t + \mathcal{T}_{n,d}(\delta\boldsymbol{x}_0^{\boldsymbol{\gamma}}) = \bar{\boldsymbol{x}}_t + \sum_{1\le|\boldsymbol{\gamma}|\le n} \boldsymbol{c}_{\gamma_1\dots\gamma_d}\delta x_{0,1}^{\gamma_1}\dots\delta x_{0,d}^{\gamma_d}, \tag{2.67}$$

with

$$\bar{\boldsymbol{x}}_t = \boldsymbol{\Phi}_t\left(t; t_0, \bar{\boldsymbol{x}}_0\right), \quad \delta\boldsymbol{x}_t = \mathcal{T}_{n,d}(\delta\boldsymbol{x}_0^{\boldsymbol{\gamma}}), \quad \boldsymbol{c}_{\gamma_1\dots\gamma_d} = \frac{1}{\gamma_1!\cdots\gamma_d!}\frac{\partial^{\gamma}\boldsymbol{\Phi}_t}{\partial\delta x_{0,1}^{\gamma_1}\cdots\partial\delta x_{0,d}^{\gamma_d}}$$

where $\boldsymbol{c}_{\gamma_1\dots\gamma_d}$ indicates the coefficients of Taylor expansion polynomials. It is worth to mention that $\delta\boldsymbol{x}_t = \mathcal{T}_{n,d}(\delta\boldsymbol{x}_0^{\boldsymbol{\gamma}})$ indicates the so-called STTs [79], which is too complex to be calculated manually if the ODE system at issue possesses high dimensionality. Compared with the hard manual calculation, the JT technique enables to accelerate its computation significantly. In principle, for a given uncertainty region, the polynomial approximation result is capable of reaching the arbitrary desired accuracy by adjusting expansion orders. Besides, if $n = 1$, (2.67) degrades into the linear approximation.

On the other hand, instead of using the Taylor expansion method, if a Chebyshev interpolation technique is employed in the JT implementation, the state set $\Omega_t : \boldsymbol{x}(t) = \boldsymbol{\Phi}_t\left(t; t_0, \bar{\boldsymbol{x}}_0 + \delta\boldsymbol{x}_0\right)$ can be represented as follows,

$$\begin{aligned} [\boldsymbol{x}(t)] &= \bar{\boldsymbol{x}}_t + \delta\boldsymbol{x}_t = \mathcal{C}_{n,d}(\boldsymbol{T}_{\boldsymbol{\gamma}}(0)) + \mathcal{C}_{n,d}(\boldsymbol{T}_{\boldsymbol{\gamma}}(\boldsymbol{\delta x_0})) \\ &= \mathcal{C}_{n,d}(\boldsymbol{T}_{\boldsymbol{\gamma}}(0)) + \sum_{1\le|\boldsymbol{\gamma}|\le n} \boldsymbol{c}_{\gamma_1\dots\gamma_d} T_{\gamma_1}(\delta x_{0,1})\dots T_{\gamma_d}(\delta x_{0,d}), \end{aligned} \tag{2.68}$$

where $\boldsymbol{c}_{\gamma_1\dots\gamma_d}$ indicates the coefficients of Chebyshev interpolation polynomials.

2.3.4 Polynomial evaluation

Note that, using the polynomial composition operation (2.46), a polynomial $P_{n,d}(\delta\boldsymbol{x})$ can be easily evaluated at polynomials $\delta\boldsymbol{x} = P_{\tilde{n},\tilde{d}}(\delta\tilde{\boldsymbol{x}})$. Observe that if the polynomial results in (2.67) and (2.68) are evaluated at real vectors instead of at polynomials, the JT state vector $[\boldsymbol{x}(t)]$ enables to produce the relevant approximate value of $\boldsymbol{x}(t) = \boldsymbol{\Phi}_t\,(t; t_0, \bar{\boldsymbol{x}}_0 + \delta\boldsymbol{x}_0)$ for each deviation vector $\delta\boldsymbol{x}_0$. Based on these semi-analytical polynomial solutions, an interesting application is to develop an *Advanced Monte Carlo method* for improving the computational efficiency of the classical MC simulation. Traditionally, in order to extract statistical information from a complicated ODE system, the classical MC simulation carries out several hundreds or thousands of different realizations, each time with the usage of different random samples. Based on the discussion in subsection 2.3.3, the complicated numerical integration procedure of an ODE system can be approximated by polynomials, such that the classical MC simulation for a designated system can be converted into the polynomial evaluation at the same initial random state. Compared with the classical MC simulation, this advanced MC simulation possesses much higher computational efficiency since the evaluation of polynomials at the real vector is much faster than the numerical integration procedure.

In order to minimize the number of power computations and accelerate the evaluation of polynomials at real numbers, either the usual Horner's method or its modification into multivariate case is adopted in accordance with the variable dimension d [68]. In other words, when $d = 1$, the usual Horner's method is preferable, otherwise, its variant suitable to multiple variable case is adopted. For an nth-order univariate polynomial

$$P_{n,1}(\delta x) = \sum_{i=0}^{n} a_i \delta x^i = a_0 + a_1\delta x + a_2\delta x^2 + \cdots + a_n\delta x^n\,, \tag{2.69}$$

the renowned Horner's method enables to employ the minimum number of operations to calculate the value of the polynomial. It extracts as many δx as possible as common factor to obtain the following unified polynomial form

$$P_{n,1}(\delta x) = a_0 + \delta x\,(a_1 + \delta x\,(\cdots + \delta x\,(a_{n-1} + a_n\delta x)))\,, \tag{2.70}$$

then, the calculation procedure starts from the innermost parenthesis to the outermost one. In the multivariate case, the variant of Horner's method is further investigated to evaluate nth-order polynomials. In principle, a polynomial $P_{n,d}(\delta\boldsymbol{x})$ of order n in d variables can be regarded as a univariate polynomial $P_{n,1}(\delta x_1)$ whose coefficients are polynomials $P_{n,d-1}(\delta\boldsymbol{x}_{\ominus})$ without variable δx_1, i.e.,

$$P_{n,d}(\delta\boldsymbol{x}) = P_{n,1}(\delta x_1) = \sum_{i=0}^{n} P_{n,d-1}(\delta\boldsymbol{x}_{\ominus})\delta x_1^i\,, \quad \text{with} \quad \delta\boldsymbol{x} = [\delta x_1\,, \delta\boldsymbol{x}_{\ominus}]\,, \tag{2.71}$$

therefore, the Horner's method is used to evaluate the polynomial $P_{n,d}(\delta\boldsymbol{x})$ at δx_1. The same procedure is again employed to evaluate the polynomial coefficients by the powers of δx_1 at the second variable δx_2 and further obtains the corresponding polynomial coefficients. The Horner's method is recursively employed until there is only one remaining variable, thus the usual Horner's method is used. Similarly, the real calculation procedure starts from the innermost parenthesis to the outermost one.

2.3.5 Numerical simulations

This subsection validates the feasibility and performance of both JTTNM-n and JTCNM-n methods for computing the semi-analytical polynomial solution of an ODE system. An advanced MC simulation using the polynomial results computed by both JTTNM-n and JTCNM-n methods are employed to efficiently and accurately propagate the trajectory uncertainties. As an illustrative example, the dimensionless dynamics of a satellite orbiting the Earth in a Kepler orbit is considered,

$$\begin{cases} \dot{\breve{\boldsymbol{r}}} &= \breve{\boldsymbol{v}}\,, \\ \dot{\breve{\boldsymbol{v}}} &= -\dfrac{\breve{\boldsymbol{r}}}{\breve{r}^3}\,, \end{cases} \tag{2.72}$$

where $\breve{\boldsymbol{r}} = [\breve{x}\,, \breve{y}]^T$ and $\breve{\boldsymbol{v}} = [\breve{v}_x\,, \breve{v}_y]^T$ respectively indicate the satellite position and velocity vectors in the orbital plane, and $\breve{r}$ denotes the Euclidean norm of the position vector $\breve{\boldsymbol{r}}$. The units are normalized in such a way that the semi-major axis a, the Earth gravitational parameter μ, and the orbital angular velocity $\Omega = \sqrt{\mu/a^3}$ are equal to 1, thus the related dimensionless time $\breve{t}$, orbital angular velocity $\breve{\omega}$, position vector $\breve{\boldsymbol{r}}$ and velocity vector $\breve{\boldsymbol{v}}$ are defined as,

$$\breve{t} = \Omega t, \quad \breve{\omega} = \frac{\omega}{\Omega}, \quad \breve{\boldsymbol{r}} = \frac{\boldsymbol{r}}{a}, \quad \breve{\boldsymbol{v}} = \frac{\boldsymbol{v}}{a\Omega}\,. \tag{2.73}$$

Furthermore, the initial nominal dimensionless state is chosen as

$$\begin{cases} \breve{x} &= 0.828\,, \\ \breve{y} &= 0.478\,, \\ \breve{v}_x &= -0.501\,, \\ \breve{v}_y &= 0.917\,. \end{cases} \tag{2.74}$$

In these units the nominal Kepler orbital period is $T = 2\pi$. The variable step-size JTRK78 algorithm is employed to integrate the ODE system (2.72) forward. The polynomial results, computed by implementing both JTTNM-n and JTCNM-n methods at different order n from 1 to 8, are employed to map a set of initial state vectors, taken from an uncertainty circle with radius $\delta\breve{r} = 0.005$. To assess the performance of both JTTNM-n and JTCNM-n methods, the results computed by a multiple pointwise integration of same samples (i.e., a MC simulation) are considered as the reference values. Note in particular that the uncertainty circle adopted herein is a rather large uncertainty set, which has been exaggerated for illustrative purposes. The evolution of this initial state uncertainty circle is investigated by propagating 80 sampling points uniformly distributed on the boundary.

To be more specific, in the illustrative simulation example, both JTTNM-n and JTCNM-n methods implemented at expansion order $n = 1, 2, 8$ are employed to transport a series of initial state vectors along the flow of the ODE system (2.72) from $t = 0$ to $t = 0.25T$, $0.5T$, $0.75T$. The relevant polynomial results are evaluated at the preselected 80 sampling

points on the boundary of the initial state uncertainty circle, such that the evolved sets $\breve{\boldsymbol{x}}_t$ at time $t = 0.25T$, $0.5T$, $0.75T$ can be obtained. In order to assess the performance, the true state set $\breve{\boldsymbol{x}}_t^r$ is computed by a repetitive pointwise numerical integration. The approximation error is defined as $\boldsymbol{\varepsilon} = \breve{\boldsymbol{x}}_t - \breve{\boldsymbol{x}}_t^r = [\varepsilon_{\breve{x}}, \varepsilon_{\breve{y}}, \varepsilon_{\breve{v}_x}, \varepsilon_{\breve{v}_y}]^T$. Thus the dimensionless position and velocity errors are denoted as

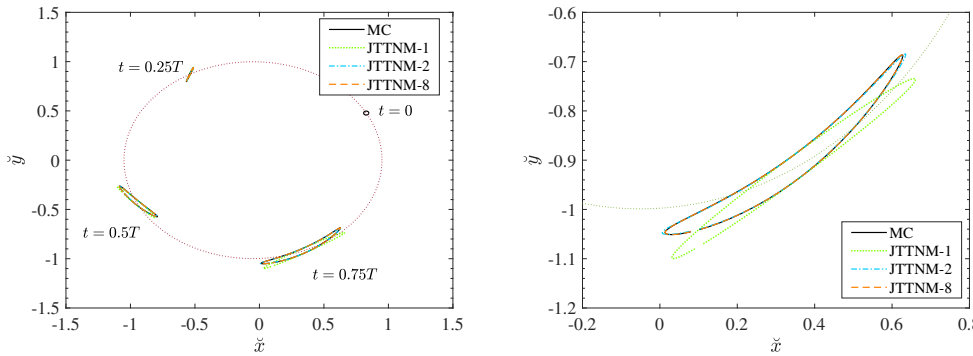
$$\varepsilon_{\breve{r}} = \sqrt{\varepsilon_{\breve{x}}^2 + \varepsilon_{\breve{y}}^2}, \quad \varepsilon_{\breve{v}} = \sqrt{\varepsilon_{\breve{v}_x}^2 + \varepsilon_{\breve{v}_y}^2}\,. \tag{2.75}$$

Furthermore, for a MC simulation with sample of cardinality N, the root-mean square error (RMSE, $\bar{\varepsilon}_\Upsilon$) is defined as

$$\bar{\varepsilon}_\Upsilon^N = \sqrt{\frac{1}{N}\sum_{j=1}^{N}\left(\varepsilon_\Upsilon^j\right)^2}, \tag{2.76}$$

where ε_Υ^j, with $\Upsilon = \breve{r}, \breve{v}$ indicate the dimensionless position and velocity errors of the j-th MC run.

For the sake of clarity, Figs. 2.6a and 2.7a respectively display the approximate results computed by the JTTNM-n and JTCNM-n methods implemented at expansion order $n = 1, 2, 8$, as compared against the evolved state set $\breve{\boldsymbol{x}}_t^r$ computed by a multiple pointwise forward integration of the samples (black line). To further exhibit the accuracy, the state sets obtained at the evolution time $t = 0.75T$ are enlarged and reported in Figs. 2.6b and 2.7b. The results reveal that both JTTNM-8 and JTCNM-8 methods are accurate enough to approximate the real orbit propagation results, which can be computed by a classical MC simulation.

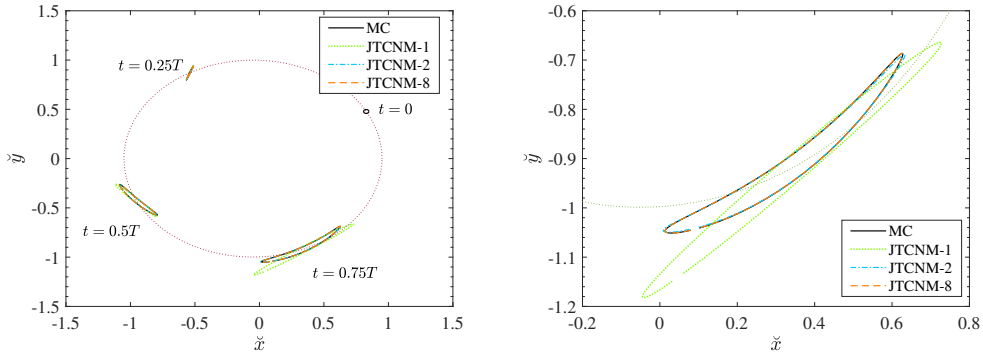


(a) Evolved sets of an initial uncertainty circle at $t = 0.25T$, $0.5T$, $0.75T$.

(b) Enlarged view of the evolved sets at time $t = 0.75T$.

Figure 2.6: Results of JTTNM-1, JTTNM-2, JTTNM-8, and classical MC methods for propagating an initial state uncertainty circle with radius $\delta\breve{r} = 0.005$.

To analyze the effects of the propagation time t and the expansion order n on the performance of both JTTNM-n and JTCNM-n methods, Figs. 2.8 and 2.9 display the RMSEs in



(a) Evolved sets of an initial uncertainty circle at $t = 0.25T$, $0.5T$, $0.75T$.

(b) Enlarged view of the evolved sets at time $t = 0.75T$.

Figure 2.7: Results of JTCNM-1, JTCNM-2, JTCNM-8, and classical MC methods for propagating an initial state uncertainty circle with radius $\delta\breve{r} = 0.005$.

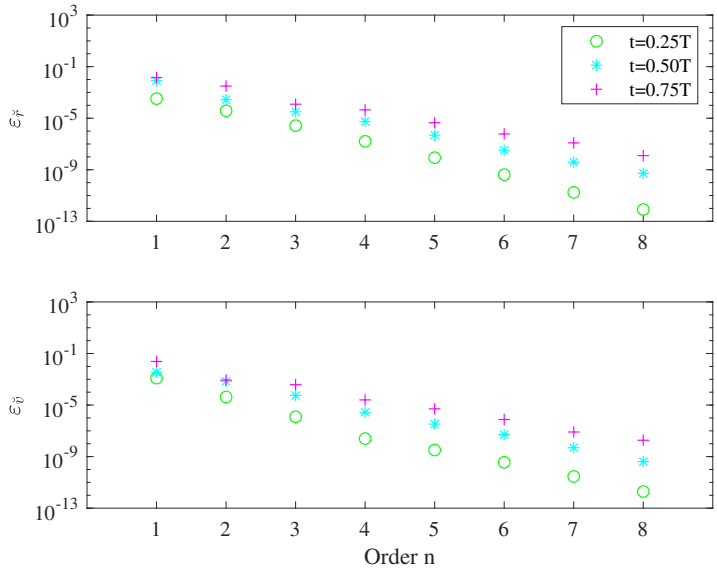


Figure 2.8: RMSEs in dimensionless position and velocity computed by the JTTNM-n method versus the expansion order n.

dimensionless position and velocity at time $t = 0.25T$, $0.5T$, $0.75T$ versus the expansion order $n = 1, \cdots, 8$. The simulation results indicate that for a fixed expansion order n, the accuracy of both methods is on the same order of magnitude and always reduces as the propagation time grows. In general, a longer propagation time results in much more integration steps and associated polynomial operations, such that a larger truncation error is accumulated. Besides, for a selected propagation time t, the RMSEs in dimensionless

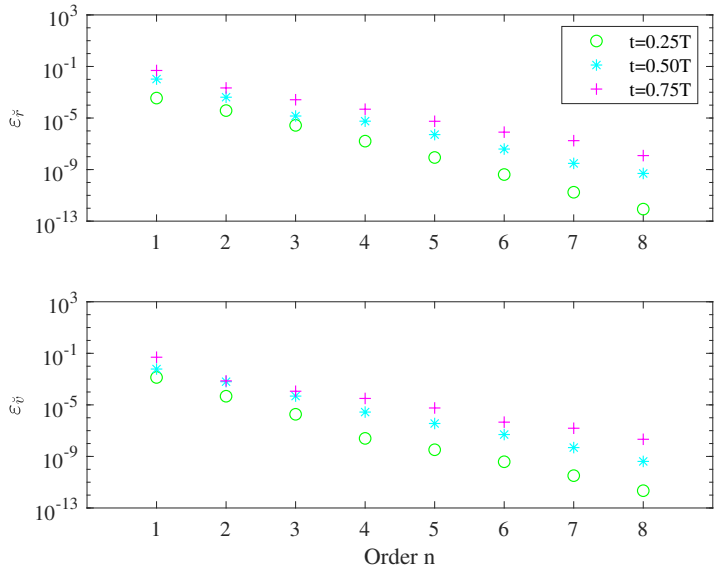


Figure 2.9: RMSEs in dimensionless position and velocity computed by the JTCNM-n method versus the expansion order n.

position and velocity computed by both JTTNM-n and JTCNM-n implementations significantly decrease when the expansion order n increases. Tables 2.1 and 2.2 give the detailed RMSEs in dimensionless position and velocity at propagation time $t = 0.75T$, corresponding to different expansion orders from $n = 1$ to 8. The results again validate that both methods almost enable to yield one order of magnitude accuracy gain via the increase of the expansion order by 1. Note in particular that the RMSEs in dimensionless position and velocity computed by the JTTNM-8 method are 1.3×10^{-8} (0.10 m) and 1.9×10^{-8} (1.28×10^{-4} m/s), while those obtained by the JTCNM-8 method are 1.2×10^{-8} (0.11 m) and 2.1×10^{-8} (1.41×10^{-4} m/s). These tiny dimensionless errors indicate that the advanced MC methods based on the JTTNM-8 and JTCNM-8 are sufficient to achieve an accurate Keplerian orbit propagation of a pretty large initial state uncertainty set.

Table 2.1: RMSEs in dimensionless position and velocity computed by the JTTNM-n at propagation time $t = 0.75T$, implemented at expansion orders from $n = 1$ to 8.

RMSE	JTTNM-1	JTTNM-2	JTTNM-3	JTTNM-4	JTTNM-5	JTTNM-6	JTTNM-7	JTTNM-8
$\varepsilon_{\check{r}}$	1.4×10^{-2}	3.1×10^{-3}	1.2×10^{-4}	4.5×10^{-5}	4.5×10^{-6}	6.0×10^{-7}	1.2×10^{-7}	1.3×10^{-8}
$\varepsilon_{\check{v}}$	2.4×10^{-2}	8.7×10^{-4}	3.8×10^{-4}	2.5×10^{-5}	5.1×10^{-6}	7.6×10^{-7}	8.1×10^{-8}	1.9×10^{-8}

Finally, to compare the computational efficiency of different methods, all codes are written in C plus plus compiled with gcc version 5.5.0 and implemented with a laptop processor Intel(R) Core(TM) i5-7300HQ under Linux (note that this is the default computer configuration to perform all simulations in what follows). Therefore, the CPU time consumed in either the advanced MC simulation or the classical MC simulation enables to reveal

Table 2.2: RMSEs in dimensionless position and velocity computed by the JTCNM-n at propagation time $t = 0.75T$, implemented at expansion orders from $n = 1$ to 8.

RMSE	JTTNM-1	JTTNM-2	JTTNM-3	JTTNM-4	JTTNM-5	JTTNM-6	JTTNM-7	JTTNM-8
$\varepsilon_{\tilde{r}}$	4.9×10^{-2}	2.2×10^{-3}	2.6×10^{-4}	4.8×10^{-5}	5.6×10^{-6}	8.0×10^{-7}	1.8×10^{-7}	1.2×10^{-8}
$\varepsilon_{\tilde{v}}$	5.0×10^{-2}	7.3×10^{-4}	1.1×10^{-4}	3.2×10^{-5}	5.7×10^{-6}	4.6×10^{-7}	1.5×10^{-7}	2.1×10^{-8}

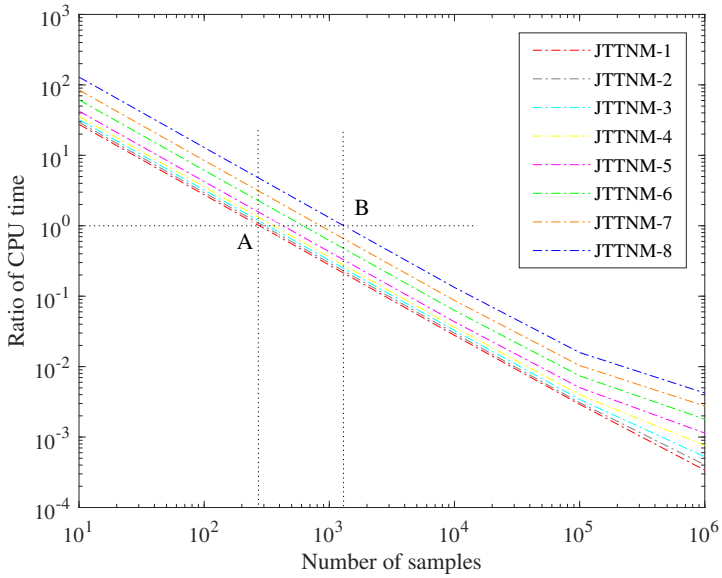


Figure 2.10: Ratio of CPU time versus the number of samples, indicating the computational efficiency of the JTTNM-n relative to a classical MC simulation.

the computational efficiency. Figures 2.10 and 2.11 respectively display the ratio of the CPU time, which is defined as the computational cost consumed in the advanced MC simulation using either the JTTNM-n or the JTCNM-n divided by that consumed in the classical MC simulation. The simulation results in Figs. 2.10 and 2.11 show several interesting conclusions: 1) For both JTTNM-n and JTCNM-n methods, the ratio of the CPU time decreases as the number of sampling points increases, that is, the computational efficiency of both JTTNM-n and JTCNM-n methods significantly enhances when the number of sampling points also does; 2) The computational efficiency of both JTTNM-n and JTCNM-n methods decrease as the expansion order increases; 3) For a MC simulation with sample of cardinality $N = 10^6$, the ratio of CPU time is between 10^{-2} and 10^{-3} in both JTTNM-n and JTCNM-n cases, which indicates that both methods achieve at least two orders of magnitude of computational efficiency improvement; 4) At point A (i.e., $N = 280$ in Fig. 2.10 and $N = 360$ in in Fig. 2.11), the advanced MC simulation based on either the JTTNM-1 or the JTCNM-1 occupies the same computational efficiency with a classical MC simulation, while at point B (i.e., $N = 1200$ in Fig. 2.10 and $N = 2100$ in Fig. 2.11), the computational efficiency of the advanced MC simulation using either the

JTTNM-8 or the JTCNM-8 method is identical to that of a classical MC simulation.

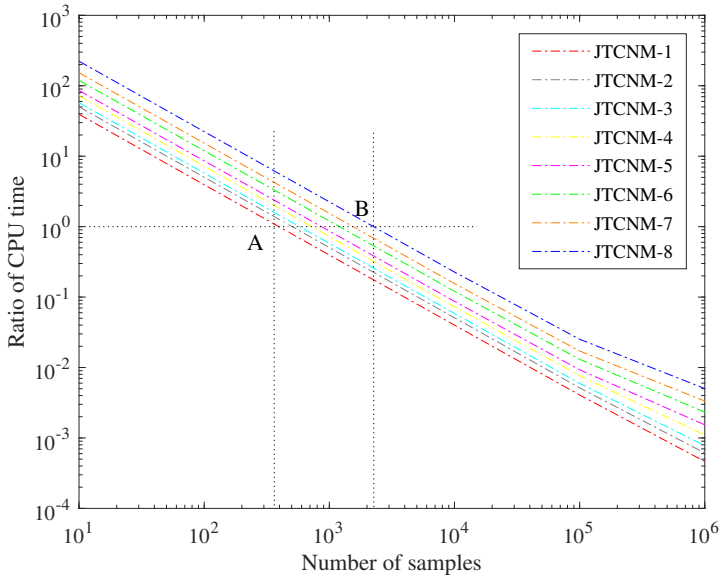


Figure 2.11: Ratio of CPU time versus the number of samples, indicating the computational efficiency of the JTCNM-n relative to a classical MC simulation.

2.4 Remarks and conclusions

This chapter introduces and constructs a useful polynomial operation tool, i.e., JT technique. This tool defines a series of polynomial operations in the modern computer, such as polynomial storage, addition, subtraction, multiplication, division, differentiation, integration and composition, as well as some useful algorithms, for instance, the JTRK78 algorithm and polynomial evaluation. Then, based on Taylor series expansion and Chebyshev polynomial interpolation technique, both JTTNM-n and JTCNM-n methods are developed to approximate practical state propagation problems addressed by a set of nonlinear functions or ordinary differential equations.

It is worth to mention that the JTTNM-n method adopts a local derivative-based function approximation in the vicinity of the expansion point, thus displays the better accuracy at the expansion point than at the margins of the expansion interval. On the contrary, the JTCNM-n method attempts a hyper-interpolation technique occupying a global convergence over the whole expansion domain, thus possesses a uniform error distribution. Besides, the JTTNM-n method requires the nonlinear functions approximated to be $n+1$ times continuously differentiable over the expansion domain, while the JTCNM-n method only requires it to be piecewise smooth and continuous.

In order to validate the effectiveness and performance, both JTTNM-n and JTCNM-n methods are carried out to approximate some usual basic functions, composition functions and orbit propagation of a spacecraft Keplerian motion. The results show that both JTTNM-n and JTCNM-n methods enable to achieve an accurate approximation to general nonlinear functions and the solution of an ODE system. In particular, as the expansion order increases, the approximation error decreases, while the required computation time significantly increases. Besides, both JTTNM-n and JTCNM-n methods possess higher computational efficiency relative to the classical MC method in the orbit propagation problem, especially when the number of sampling trajectories is huge.

3

Chapter 3

Analysis of high order orbit uncertainty propagation

The Geostationary Earth Orbit regime, capable of maintaining a spacecraft nearly fixed in an ECEF frame, is the best option to provide continuous observations, broadcast and communication services, on the Earth's surface. However, the growing amount of controlled satellites and uncontrolled debris aggravates the congestion and increases collision risks. Moreover, new emerging applications of GEO satellites requiring an unprecedented accuracy are being considered. Examples are the future GEO SAR mission [11] and the GEO signal-based mezzo-scale sea altimetry measurement [13]. Therefore, an accurate and efficient orbit propagation procedure for predicting GEO trajectories is not only justified from the point of view of collision avoidance analysis, but also for an efficient orbit prediction and estimation.

In this chapter, both JTTNM-n and JTCNM-n methods are employed to accurately propagate the state uncertainties of geostationary satellite trajectories under four dominant perturbations: SRP, Earth's non-spherical gravity potential, and luni-solar gravitational attractions. The feasibility of both methods is validated by implementing the orbit propagation in three coordinate representations, and in two polynomial forms: Taylor and Chebyshev expansions, i.e., JTTNM-n and JTCNM-n methods. Taking into account the size of the uncertainty neighborhood, as well as the expansion order and the time step-size of JTRK integrators, a large quantity of combinations of the above factors are simulated,

and a series of look-up tables with recommendations on the best option to address the GEO trajectory propagation are given.

3.1 Dynamical models

In general, the accuracy and efficiency of trajectory propagation methods closely depend on the choice of coordinate representations. Many types of coordinate representations have been widely investigated to describe near-Earth satellite orbits: classic Cartesian coordinates, Keplerian orbital elements, synchronous orbit elements, modified equinoctial elements, GEO orbit elements, Hill's variables, cylindrical coordinates, and Deprit's ideal elements [155–158]. The traditional Cartesian coordinates (i.e., inertial position and velocity vectors) are capable of easily modelling the dominant perturbations in the GEO region and intuitively providing physical insight into the dynamics. However, the fast changes of these coordinates over time obscure the beneficial characteristic that GEO spacecrafts remain almost steady in the ECEF frame, which results in small integration step-sizes and heavy computational costs. Keplerian orbital elements overcome this shortcoming and mitigate the computational cost by increasing integration time step-sizes but, unfortunately, they encounter poor definitions in the case of low-inclination or low-eccentricity orbits. Similarly, coordinate representations including synchronous orbit elements, modified equinoctial elements, Hill's variables and Deprit's ideal elements are likely to encounter the singularities when they are employed to describe GEO orbits. To avoid this issue, only Cartesian coordinates, GEO orbit elements, and cylindrical coordinates are employed in what follows.

3.1.1 Cartesian dynamic model

The motion of a spacecraft near the Earth is governed by a set of three second order ODEs in Cartesian coordinates (CAC),

$$\begin{cases} \ddot{x} &= -\dfrac{\mu x}{r^3} + a_x, \\ \ddot{y} &= -\dfrac{\mu y}{r^3} + a_y, \\ \ddot{z} &= -\dfrac{\mu z}{r^3} + a_z, \end{cases} \tag{3.1}$$

where $\boldsymbol{a}_f = (a_x, a_y, a_z)^T$ indicates the total perturbation acceleration described in the Earth-centered inertial reference frame (ECI), $\boldsymbol{r} = (x, y, z)^T$ is the spacecraft position vector, $r = |\boldsymbol{r}|$ indicates its Euclidean norm, and $\mu = G\,M_e$ is the gravitational coefficient of the Earth. We can also write these equations in a compact and unified form as [31],

$$\ddot{\boldsymbol{r}} = -\frac{\mu \boldsymbol{r}}{r^3} + \boldsymbol{a}_f\,. \tag{3.2}$$

3.1.2 Cylindrical coordinates

Cylindrical coordinates (CYC) are another alternative to describe GEO orbits without singularities. It consists of the distance ρ from the z-axis, the azimuth angle φ, and the height z along the z-axis, together with their time derivatives $(\dot{\rho}, \dot{\varphi}, \dot{z})$. The equations of motion are [64]

$$\begin{cases} \dfrac{d\rho}{dt} &= \dot{\rho}, \quad \dfrac{d\varphi}{dt} = \dot{\varphi}, \quad \dfrac{dz}{dt} = \dot{z}, \\ \dfrac{d\dot{\rho}}{dt} &= \rho\dot{\varphi}^2 - \dfrac{\mu}{r^3}\rho + a_\rho, \\ \dfrac{d\dot{\varphi}}{dt} &= \dfrac{-2\,\dot{\rho}\,\dot{\varphi}}{\rho} + \dfrac{a_\varphi}{\rho}, \\ \dfrac{d\dot{z}}{dt} &= -\dfrac{\mu}{r^3}z + a_z, \end{cases} \tag{3.3}$$

where $\boldsymbol{a}_f = (a_\rho, a_\varphi, a_z)^T$ are the components of the total perturbation acceleration projected into the reference frame CYC with unit axes along the radial, azimuthal, and axial directions (RAA).

3.1.3 GEO dynamical model

In [157] Tombasco introduced a novel non-dimensional GEO element set (GES) in terms of the classical Keplerian elements. It consists of the Earth-fixed sub-spacecraft longitude λ, the longitudinal drift rate $\delta\bar{a}$, two eccentricity vector components (e_x, e_y), and two equinoctial elements (Q_1, Q_2) defined by

$$\begin{cases} \lambda &\triangleq (\omega + \Omega + \theta) - GA\,(t)\,, \\ \delta\bar{a} &\triangleq \dfrac{a - A_n}{A_n}, \\ e_x &\triangleq e\,\cos\,(\omega + \Omega)\,, \\ e_y &\triangleq e\,\sin\,(\omega + \Omega)\,, \\ Q_1 &\triangleq \tan\left(\dfrac{i}{2}\right)\sin\,(\Omega)\,, \\ Q_2 &\triangleq \tan\left(\dfrac{i}{2}\right)\cos\,(\Omega)\,, \end{cases} \tag{3.4}$$

here, $\{a, e, i, \omega, \Omega, \theta\}$ refer to the classical Keplerian elements, $\delta\bar{a}$ is non-dimensional value of the semi-major axis deviation with respect to a nominal GEO semi-major axis $A_n = 42164.2$ km, and the time dependent function $GA(t)$ stands for the Greenwich sidereal angle at t, i.e., $GA\,(t) = GA(t_0) + \omega_e\,(t - t_0)$, where $\omega_e = 7.292115 \times 10^{-5}$ rad/s is the average angular speed of the Earth's rotation. Using the Poisson brackets method, Tombasco deduced the GEO dynamic model in terms of the GEO element set [157], that can be written as

$$\begin{cases} \dot{\lambda} &= \dfrac{h}{r^2} + \dfrac{r}{h}\left[Q_2\,\sin s - Q_1\,\cos s\right] a_h - \omega_e, \\ \delta\dot{\bar{a}} &= \dfrac{2\left(\delta\bar{a}+1\right)^2}{h a_n}\left[\left(e_x\,\sin s - e_y\,\cos s\right) a_r + \dfrac{p}{r} a_\theta\right], \\ \dot{e}_x &= \dfrac{r}{h}\left\{\dfrac{p}{r}\,\sin s \cdot a_r + \left[e_x + \left(1+\dfrac{p}{r}\right)\,\cos s\right] a_\theta \right. \\ & \quad \left. + e_y\left[Q_1\,\cos s - Q_2\,\sin s\right] a_h\right\}, \\ \dot{e}_y &= \dfrac{r}{h}\left\{-\dfrac{p}{r}\,\cos s \cdot a_r + \left[e_y + \left(1+\dfrac{p}{r}\right)\,\sin s\right] a_\theta \right. \\ & \quad \left. - e_x\left[Q_1\,\cos s - Q_2\,\sin s\right] a_h\right\}, \\ \dot{Q}_1 &= \dfrac{r}{2h}\left(1+Q_1^2+Q_2^2\right)\,\sin s \cdot a_h, \\ \dot{Q}_2 &= \dfrac{r}{2h}\left(1+Q_1^2+Q_2^2\right)\,\cos s \cdot a_h, \end{cases} \tag{3.5}$$

where the components of the perturbation acceleration $\boldsymbol{a}_f = (a_r, a_\theta, a_h)^T$ are given in the local vertical local horizontal reference frame (LVLH), r is the radial distance of the spacecraft to the center of the Earth, s is the spacecraft sidereal angle, p is the semi-latus rectum of the orbit, and h is the modulus of the angular momentum. Using the definitions of r, s, p, and h, they can be written in terms of GEO elements as

$$\begin{cases} r &= \dfrac{A_n\left(\delta\bar{a}+1\right)\left(1-e_x^2-e_y^2\right)}{1+e_x\,\cos s + e_y\,\sin s}, \\ s &= \lambda + GA\left(t\right) = \omega + \Omega + \theta, \\ p &= a_n\left(\delta\bar{a}+1\right)\left(1-e_x^2-e_y^2\right), \\ h &= \sqrt{p\mu}\,. \end{cases} \tag{3.6}$$

3.1.4 Coordinate transformations

To project perturbation accelerations among the aforementioned frames swimmingly, it is necessary to study the coordinate transformation matrices. The direction cosines matrices between the frames: ECI and LVLH, ECEF and ECI, and RAA and ECI, are respectively expressed as,

$$\mathcal{T}_{\mathcal{I}}^{\mathcal{L}} = \begin{pmatrix} \boldsymbol{I}\cdot\boldsymbol{i} & \boldsymbol{I}\cdot\boldsymbol{j} & \boldsymbol{I}\cdot\boldsymbol{k} \\ \boldsymbol{J}\cdot\boldsymbol{i} & \boldsymbol{J}\cdot\boldsymbol{j} & \boldsymbol{J}\cdot\boldsymbol{k} \\ \boldsymbol{K}\cdot\boldsymbol{i} & \boldsymbol{K}\cdot\boldsymbol{j} & \boldsymbol{K}\cdot\boldsymbol{k} \end{pmatrix}, \tag{3.7}$$

$$\mathcal{T}_{\mathcal{F}}^{\mathcal{I}} = \begin{pmatrix} \cos\left(\omega_e t\right) & -\sin\left(\omega_e t\right) & 0 \\ \sin\left(\omega_e t\right) & \cos\left(\omega_e t\right) & 0 \\ 0 & 0 & 1 \end{pmatrix}, \tag{3.8}$$

$$\mathcal{T}_{\mathcal{C}}^{\mathcal{I}} = \begin{pmatrix} \cos\varphi & -\sin\varphi & 0 \\ \sin\varphi & \cos\varphi & 0 \\ 0 & 0 & 1 \end{pmatrix}, \tag{3.9}$$

where $\mathcal{I}$, $\mathcal{F}$, $\mathcal{L}$ and $\mathcal{C}$ respectively indicate the ECI, ECEF, LVLH and RAA reference frames, $\boldsymbol{I}, \boldsymbol{J}, \boldsymbol{K}$ represent the unitary vectors defining the ECI frame, while $\boldsymbol{i}, \boldsymbol{j}, \boldsymbol{k}$ are along the unitary vectors defining the LVLH frame. Note that $\boldsymbol{i}, \boldsymbol{j}, \boldsymbol{k}$ can be computed by

$$\boldsymbol{i} = \frac{\boldsymbol{r}}{\|\boldsymbol{r}\|}, \qquad \boldsymbol{k} = \frac{\boldsymbol{r} \times \boldsymbol{v}}{\|\boldsymbol{r} \times \boldsymbol{v}\|}, \qquad \boldsymbol{j} = \boldsymbol{k} \times \boldsymbol{i}. \tag{3.10}$$

Note that the perturbation accelerations in the equations of motion (3.2), (3.3) and (3.5) can be easily computed using the satellite's Cartesian position, but in order to propagate trajectories in the other two sets of coordinate representations, the perturbations must be described as a function of GES elements and CYC coordinates. The conversion procedure is straightforward: at a certain epoch GES elements or CYC coordinates must be transformed into CAC coordinates, then compute the considered perturbation acceleration, and finally project it into the LVLH or RAA reference frames. The transformation relation from GES elements into CAC coordinates can be addressed as the following explicit functions (see Appendix A.1)

$$\begin{cases} x &= \dfrac{r}{1+Q_1^2+Q_2^2}\left[2Q_1Q_2\,\sin\,s + \left(1-Q_1^2+Q_2^2\right)\,\cos\,s\right], \\ y &= \dfrac{r}{1+Q_1^2+Q_2^2}\left[2Q_1Q_2\,\cos\,s + \left(1+Q_1^2-Q_2^2\right)\,\sin\,s\right], \\ z &= \dfrac{2\,r}{1+Q_1^2+Q_2^2}\left[Q_2\,\sin\,s - Q_1\,\cos\,s\right], \\ \dot{x} &= \dfrac{\mu}{h\left(1+Q_1^2+Q_2^2\right)}\left[2Q_1Q_2\,\left(\cos\,s+e_x\right)\right. \\ & \quad \left.+\left(-1+Q_1^2-Q_2^2\right)\,\left(\sin\,s+e_y\right)\right], \\ \dot{y} &= \dfrac{\mu}{h\left(1+Q_1^2+Q_2^2\right)}\left[-2Q_1Q_2\,\left(\sin\,s+e_y\right)\right. \\ & \quad \left.+\left(1+Q_1^2-Q_2^2\right)\,\left(\cos\,s+e_x\right)\right], \\ \dot{z} &= \dfrac{\mu}{h\left(1+Q_1^2+Q_2^2\right)}\left[Q_1\,\left(\sin\,s+e_y\right)+Q_2\left(\cos\,s+e_x\right)\right]. \end{cases} \tag{3.11}$$

Besides, the explicit conversion formulae from CYC to CAC coordinates is expressed as (see Appendix A.2)

$$\begin{cases} x = \rho\cos\varphi, & y = \rho\sin\varphi, & z = z, \\ \dot{x} = \dot{\rho}\cos\varphi - \rho\dot{\varphi}\sin\varphi, & \dot{y} = \dot{\rho}\sin\varphi + \rho\dot{\varphi}\cos\varphi, & \dot{z} = \dot{z}\,. \end{cases} \tag{3.12}$$

3.2 Analysis of dominant perturbation accelerations

3.2.1 Perturbation modelling

The dominant perturbations considered in the precise GEO orbit propagation problem include SRP, Earth's non-spherical gravity, and luni-solar gravitational attraction. The detailed models can be found in [31, 159]. For convenience, their brief formulae are reviewed in what follows.

3.2.1.1 Solar radiation pressure perturbation

The adopted SRP acceleration model is expressed in the ECI reference frame as [31]

$$\boldsymbol{a}_{p,\mathcal{I}} = -\nu P\, C_r \frac{A}{m} \frac{\boldsymbol{r}_\odot}{r_\odot^3}\, AU^2\,, \tag{3.13}$$

where $\boldsymbol{r}_\odot$ indicates the relative position vector from the spacecraft to the Sun, $r_\odot = |\boldsymbol{r}_\odot|$ represents the Euclidean norm of the relative position vector $\boldsymbol{r}_\odot$, $P = 4.56 \times 10^{-6}\ \mathrm{Nm}^{-2}$ is the solar radiation pressure at a distance of 1 AU, ν is a shadow function (its computation is also given in [31]). In particular, the shape of the spacecraft is assumed as a sphere. The radiation pressure coefficient C_r, the mass m, and the cross-sectional area A are related to the individual spacecraft properties.

3.2.1.2 Luni-solar gravitational perturbation

The gravitational acceleration of a point mass M (Moon or Sun) on the spacecraft can be expressed in the ECI reference frame as

$$\boldsymbol{a}_{M,\mathcal{I}} = GM \left(\frac{\boldsymbol{r}_M - \boldsymbol{r}}{|\boldsymbol{r}_M - \boldsymbol{r}|^3} - \frac{\boldsymbol{r}_M}{|r_M|^3} \right), \tag{3.14}$$

where $\boldsymbol{r}_M$ and $\boldsymbol{r}$ represent the Earth-centered coordinates of the mass M and of the spacecraft, respectively. The position vector $\boldsymbol{r}_M$ can be computed by the method given in [31] or using any other kind of ephemerides. Note in particular that $M = s$ indicates the solar gravity, while $M = m$ indicates the lunar gravity.

3.2.1.3 Earth's non-spherical gravity perturbation

The Earth's potential has a significant impact on the evolution of GEO spacecraft trajectories. Its detailed computation is given in [31]. For convenience, a brief summary is given

in what follows. In the ECEF reference frame, the Earth's gravitational acceleration can be written as

$$\boldsymbol{a}_{g,\mathcal{F}} = \left(\sum_{n,m} a_{x,nm} \, , \sum_{n,m} a_{y,nm} \, , \sum_{n,m} a_{z,nm} \right). \tag{3.15}$$

The components $a_{x,nm}$, $a_{y,nm}$, and $a_{z,nm}$ can be expressed as

$$\begin{cases} a_{x,nm} &= \dfrac{\mu}{2R_{\oplus}^2}\{(-C_{nm}V_{n+1,m+1} - S_{nm}W_{n+1,m+1}) + \\ & \quad \dfrac{(n-m+2)!}{(n-m)!}\,(C_{nm}V_{n+1,m-1} + S_{nm}W_{n+1,m-1})\}, \\ a_{y,nm} &= \dfrac{\mu}{2R_{\oplus}^2}\{(-C_{nm}W_{n+1,m+1} + S_{nm}V_{n+1,m+1}) + \\ & \quad \dfrac{(n-m+2)!}{(n-m)!}\,(-C_{nm}W_{n+1,m-1} + S_{nm}V_{n+1,m-1})\}, \\ a_{z,nm} &= \dfrac{\mu}{R_{\oplus}^2}\left\{(n-m+1)\,(-C_{nm}V_{n+1,m} - S_{nm}W_{n+1,m})\right\}, \end{cases} \tag{3.16}$$

where μ is the gravitational coefficient of the Earth, $R_{\oplus}$ is the Earth's equatorial radius, and C_{nm}, S_{nm} are geopotential coefficients given by the Earth gravity model EGM96S. Note that in the following numerical simulations, $0 \leq m \leq n \leq 5$ is considered. Besides, the case $m = n = 0$ corresponds to the gravitational acceleration of an ideal spherical Earth. The values of V_{nm} and W_{nm} satisfy the recurrence relations

$$\begin{cases} V_{mm} &= (2m-1)\left\{\dfrac{xR_{\oplus}}{r^2}V_{m-1,m-1} - \dfrac{yR_{\oplus}}{r^2}W_{m-1,m-1}\right\}, \\ W_{mm} &= (2m-1)\left\{\dfrac{xR_{\oplus}}{r^2}W_{m-1,m-1} + \dfrac{yR_{\oplus}}{r^2}V_{m-1,m-1}\right\}, \\ V_{nm} &= \left(\dfrac{2n-1}{n-m}\right)\dfrac{zR_{\oplus}}{r^2}V_{n-1,m} - \left(\dfrac{n+m-1}{n-m}\right)\dfrac{R_{\oplus}^2}{r^2}V_{n-2,m}, \\ W_{nm} &= \left(\dfrac{2n-1}{n-m}\right)\dfrac{zR_{\oplus}}{r^2}W_{n-1,m} - \left(\dfrac{n+m-1}{n-m}\right)\dfrac{R_{\oplus}^2}{r^2}W_{n-2,m}, \end{cases} \tag{3.17}$$

where r denotes the spacecraft distance with respect to the Earth's mass center. To compute V_{nm} and W_{nm}, the following initial values are required:

$$V_{00} = \frac{R_{\oplus}}{r}, \quad W_{00} = 0, \quad V_{m-1,m} = 0, \quad W_{m-1,m} = 0\,. \tag{3.18}$$

3.2.2 Perturbation analysis

In both JTTNM-n and JTCNM-n implementations, the higher expansion order employs, the smaller approximation error obtains, but the more CPU time consumes. Targeting towards an intensive and accurate propagation, the trade-off between accuracy and CPU

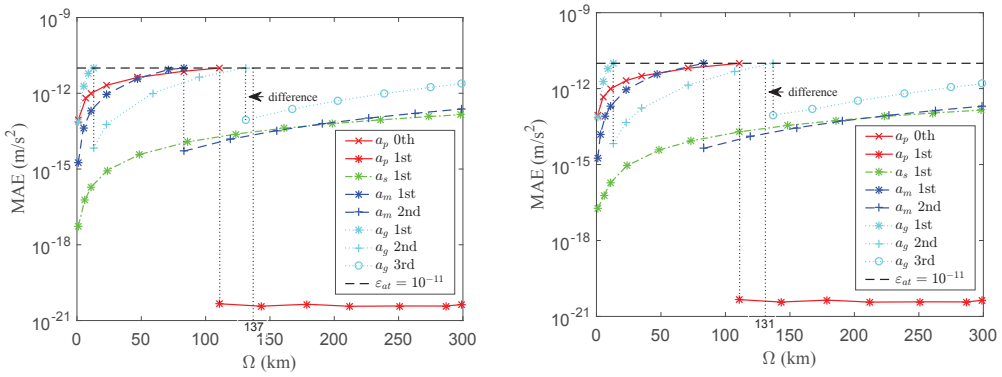
time must be analyzed in order to explore the *optimal expansion order* in the approximation of perturbations, that is, to find the lowest expansion order satisfying a prescribed accuracy threshold. In the assessment of propagation accuracy, the simulation results computed by both JTTNM-n and JTCNM-n methods are compared against those calculated by a classical MC simulation with a cardinality of $N = 10^4$. Note that the samples can be generated by means of a Latin Hyper-square generator. The maximum absolute error (MAE, $\mathcal{E}^m$) is defined as

$$\mathcal{E}^m = \underset{1 \le i \le N}{max} \left| \boldsymbol{a}_{f,i} - \boldsymbol{a}_{f,i}^r \right|, \tag{3.19}$$

where $\boldsymbol{a}_{f,i}^r$ indicates the true perturbation acceleration computed by a numerical method at the i-th sampling point, while $\boldsymbol{a}_{f,i}$ indicates its approximate value computed by evaluating the polynomial result of either JTTNM-n or JTCNM-n at the same sampling point.

The nonlinearity measure for algebraic functions, such as models of dominant perturbations, has been described by a nonlinearity index [64], which shows that the region encountering the fastest position variation along the expected orbit is the unique place to explore the optimal expansion order. During this process, an uncertainty cube shaped domain about the GEO satellite position is taken into account. The prescribed accuracy threshold adopted equals to $\varepsilon_{at} = 10^{-11}$ m/s^2, which is three orders of magnitude smaller than the smallest one among the dominant perturbations in the GEO regime, i.e., the SRP acceleration [31], such that the truncation error smaller than this particular threshold is practically negligible. Based on this accuracy threshold, the relation between the optimal expansion order and the size of the uncertainty domain is explored straightforward. First, a Latin Hyper-square generator is employed to produce a set of samples in terms of the size of the position uncertainty domain. Then, the metric MAE is adopted to assess the approximation accuracy of both JTTNM-n and JTCNM-n methods. Finally, the choice of the lowest order for a given position uncertainty domain requires that the MAE should be always less than the prescribed accuracy threshold ε_{at}.

Figures 3.1a and 3.1b respectively illustrate the optimal expansion order adopted by either the JTTNM-n or the JTCNM-n for approximating four dominant GEO perturbations when different sizes of position uncertainty domains are considered. It is clear that both JTTNM-n and JTCNM-n methods provide almost the same performance when an uncertainty domain is small. The performance difference appears when the nonlinear approximation polynomial must be employed, as shown by the "difference" mark in Fig. 3.1, which reveals that the JTCNM-2 still possesses a good accuracy for approximating the Earth's non-spherical gravity over $131 \le \Omega \le 137$ km, while the JTTNM-2 is no longer accurate enough. Figure 3.1 also gives a variety of useful recommendations about the choice of optimal expansion orders for both JTTNM-n and JTCNM-n methods in terms of the magnitude of the uncertainty domain. For instance, the SRP acceleration is almost constant when $\Delta \le 113$ km. As a consequence, for the SRP acceleration, one can take the measure on the central point of the uncertainty domain, a constant value, when $\Delta \le 113$ km, while the first order polynomial must be applied when $\Delta \in (113, 300]$ km.



(a) JTTNM-n method **(b)** JTCNM-n method

Figure 3.1: Optimal expansion orders for approximating four dominant GEO perturbations vs. the size of the position uncertainty domain.

3.3 Geostationary orbit propagation

3.3.1 Selection of fixed step integrator

Generally, due to the slow-varying vector field in the GEO region, the integration step-size of the JTRK integrator varies slightly in the accurate propagation of GEO trajectories. Therefore, fixed step JTRK integrators are preferred on account to a higher computational efficiency when compared to variable step integrators. Although for all that, variable step JTRK integrators are useful to provide a suitable value for exploring the optimal integration step-sizes that keep the required accuracy at the least computational cost. The optimal integration step-size is defined as the largest integration step-size adopted in a particular implementation within the allowable accuracy threshold, meanwhile its slight change would not affect the performances of JTTNM-n and JTCNM-n methods. An a priori study of the optimal integration step-sizes is carried out by considering different coordinate representations and fixed step JTRK integrators. Unless otherwise stated, the initial conditions given in Table 3.1 are employed in the following simulations. Table 3.2 displays the optimal integration step-sizes employed in both fixed step integrators JTRK4 and JTRK8 for integrating three dynamical models respectively formulated in CAC, CYC and GES coordinates. In a general way, the JTRK8 is more efficient than the JTRK4, such that it will be adopted in all the following simulations.

3.3.2 Polynomial-based orbit propagation

To provide an accurate and efficient nonlinear orbit propagation in the GEO regime, the optimal combination of the expansion methods and coordinate representations are

Table 3.1: Nominal initial values of spacecraft states and parameters

$\boldsymbol{r}_0$	Nominal value	$\boldsymbol{v}_0$	Nominal value	Parameter	Nominal value
x_0	24487.8 km	v_{x0}	-2.50298 km/s	S	74 m^2
y_0	34324.4 km	v_{y0}	1.78568 km/s	m	3300 kg
z_0	0 km	v_{z0}	0 km/s	C_r	1.3

Note: Epoch 15 November 2015, 0:0:0 UTC.

Table 3.2: Selection of integration step-sizes.

Method	Integrator	Coordinate representations		
		CAC	CYC	GES
JTTNM-n	JTRK4	150	754	1623
	JTRK8	1554	7240	10800
JTCNM-n	JTRK4	150	754	1623
	JTRK8	1503	7200	10800

Note: the unit of the integration step-size is seconds (s).

explored. Both JTTNM-n and JTCNM-n methods are considered to test the propagation of an initial state set. The performance comparisons between coordinate representations and between expansion methods are described in all cases by means of the RMSE, MAE, CPU time and the standard deviation (SD) of the RMSEs. Note that the adopted CPU time measure contains the initialization and propagation of both JTTNM-n and JTCNM-n implementations.

3.3.2.1 Evaluation metrics

To assess the accuracy of the newly proposed methods, true state vectors $\boldsymbol{x}_i^r$ computed by a classical MC simulation with a cardinality of N samples (i.e., point-wise numerical integration) are considered as the reference values. The approximate state vectors $\boldsymbol{x}_i$ are obtained by evaluating the polynomial results of either JTTNM-n or JTCNM-n at the same sampling points. The estimation error vector of the i-th sample is defined as $\boldsymbol{\varepsilon}_i = \boldsymbol{x}_i - \boldsymbol{x}_i^r = [\varepsilon_{x,i}, \varepsilon_{y,i}, \varepsilon_{z,i}, \varepsilon_{v_x,i}, \varepsilon_{v_y,i}, \varepsilon_{v_z,i}]^T$, thus the position and velocity errors can be further expressed as $\varepsilon_r^i = \sqrt{\varepsilon_{x,i}^2 + \varepsilon_{y,i}^2 + \varepsilon_{z,i}^2}$ and $\varepsilon_v^i = \sqrt{\varepsilon_{v_x,i}^2 + \varepsilon_{v_y,i}^2 + \varepsilon_{v_z,i}^2}$.

In addition, the RMSEs ($\bar{\varepsilon}_\Upsilon$) and MAEs ($\mathcal{E}_\Upsilon^m$) in position and velocity can be defined as

$$\bar{\varepsilon}_\Upsilon = \sqrt{\frac{1}{N}\sum_{i=1}^{N}\left(\varepsilon_\Upsilon^i\right)^2}, \quad \mathcal{E}_\Upsilon^m = \underset{1\le i\le N}{max}\left|\varepsilon_\Upsilon^i\right|. \tag{3.20}$$

where $\Upsilon = r, v$. The standard deviation of the RMSEs is another alternative option to measure the uniformity of the convergence,

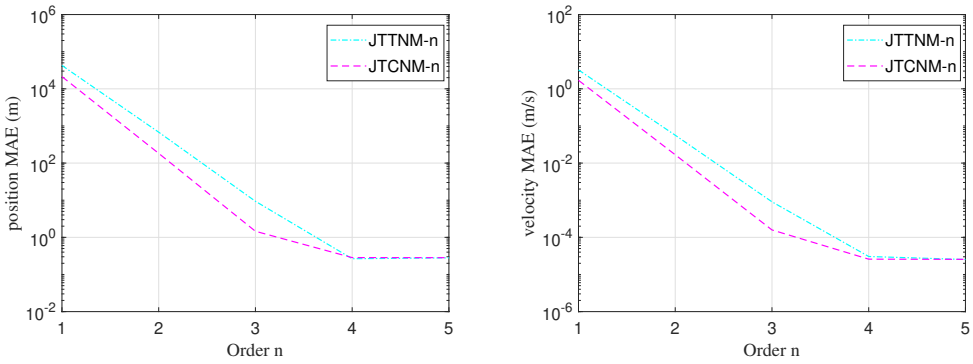
$$\sigma_{\Upsilon} = \sqrt{\frac{1}{N-1}\sum_{i=1}^{N}(\varepsilon_{\Upsilon}^{i} - \bar{\varepsilon}_{\Upsilon})^2}\,, \tag{3.21}$$

where i indicate the i-th sampling trajectory.

As underlined in Chapter 2, all codes are written in C plus plus compiled with gcc version 5.5.0 and implemented with a laptop processor Intel(R) Core(TM) i5-7300HQ under Linux. Therefore, the CPU time is an elegant measure for assessing the computational efficiency of the proposed methods.

3.3.2.2 Comparison between Taylor and Chebyshev implementations

In the following, JT-based orbit propagations from different initial state vectors are implemented using both Taylor and Chebyshev expansions. The goal is to explore the pros and cons of the JTTNM-n and JTCNM-n methods. The simulation is conducted by propagating an uncertainty position set $\Omega_r : u_r(0) = (u_x, u_y, u_z) = (50, 50, 50)$ km in the GEO regime for a given time interval of 2 days. Without loss of generality, the equations of motion formulated in CAC coordinates are adopted.

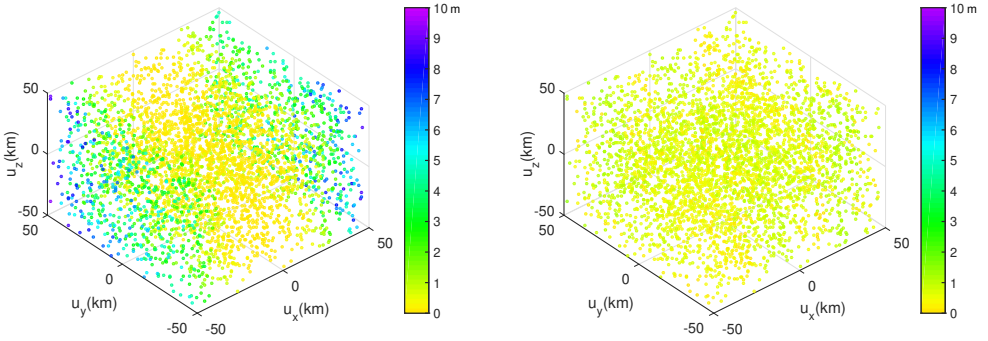


(a) MAE in position **(b)** MAE in velocity

Figure 3.2: MAEs in position and velocity vs. the expansion order for propagating an initial set $\Omega_r : u_r(0) = (u_x, u_y, u_z) = (50, 50, 50)$ km using both JTTNM-n and JTCNM-n for 2 days.

Figure 3.2 shows the MAEs in position and velocity versus the expansion order n, calculated by both JTTNM-n and JTCNM-n methods. Clearly, in a natural way, the accuracy improves with the increase of the expansion orders up to a certain value, for instance, order 4 in this test case. In particular, the JTCNM-n method is more accurate than the JTTNM-n one. In order to clarify this conclusion, a detailed comparison between the real state vectors computed by a MC simulation with a cardinality of 10^4 samples and the approximate state vectors obtained by the evaluation of the polynomial results of both

JTTNM-3 and JTCNM-3 methods at the same initial states. The resulting position and velocity error distributions are reported in Figs. 3.3 and 3.4, which again display that the JTCNM-3 method delivers a better position and velocity accuracy relative to the JTTNM-3 method, especially on the margins of the uncertainty domain. In essence, the JTTNM-n method is a local expansion method, and for a particular dynamics, its accuracy mostly depends on the expansion order and on the distance between the mapped point and the expansion point. However, the JTCNM-n method employs a hyper-interpolation technique occupying a global convergence over the uncertainty domain, its accuracy mostly depends on the expansion order and on the a priori information of the size of the uncertainty domain. As expected, the JTCNM-3 method provides an uniform convergence over the whole uncertainty domain and performs quite well, even on the margins, in a much more subtle fashion. In contrast, the JTTNM-3 method yields a sensational approximation in the vicinity of the central point, and gradually loses accuracy as the mapping point drifts away from the expansion point and gets close to the distal sections of the uncertainty domain (shown by the blue points on the margin of plots in Figs. 3.3 and 3.4).



(a) JTTNM-3 **(b)** JTCNM-3

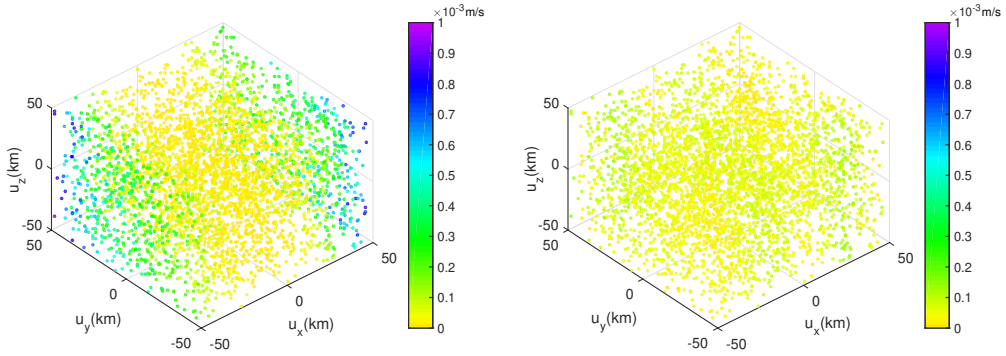
Figure 3.3: Position error distribution of both JTTNM-3 and JTCNM-3 for propagating an initial set $\Omega_r : u_r(0) = (u_x, u_y, u_z) = (50, 50, 50)$ km for 2 days.

Table 3.3: Performance comparison between both JTTNM-3 and JTCNM-3.

	$\bar{\varepsilon}_r$ (m)	$\bar{\varepsilon}_v$ (m/s)	$\mathcal{E}_r^m$ (m)	$\mathcal{E}_v^m$ (m/s)	σ_r (m)	σ_v (m/s)	t (s)
JTTNM-3	1.46	1.14×10^{-4}	9.36	9.06×10^{-4}	1.94	1.6×10^{-4}	6.75
JTCNM-3	0.62	5.48×10^{-5}	1.46	1.58×10^{-4}	0.28	2.85×10^{-5}	8.51

Note: $\bar{\varepsilon}_r$, $\bar{\varepsilon}_v$, $\mathcal{E}_r^m$, $\mathcal{E}_v^m$ respectively indicate RMSEs and MAEs in position and velocity, σ_r and σ_v are SDs of the RMSEs, t is the CPU time. Propagation results of $\Omega_r(0) = (u_x, u_y, u_z) = (50, 50, 50)$ km for 2 days.

Apart from the above illustration, the overall performance comparison between both JTTNM-3 and JTCNM-3 methods, including RMSE, MAE, SD and CPU time, are reported in Table 3.3. It is clear that the JTCNM-3 method has smaller RMSEs, MAEs and SDs in the test case, at the cost of a slight increase in CPU time. The influence of the size of the uncertainty domain on the accuracy of both JTTNM-3 and JTCNM-3



(a) JTTNM-3 **(b)** JTCNM-3

Figure 3.4: Velocity error distribution of both JTTNM-3 and JTCNM-3 for propagating an initial set $\Omega_r : u_r(0) = (u_x, u_y, u_z) = (50, 50, 50)$ km for 2 days.

Table 3.4: Influence of the size of the initial uncertainty domain on the performances of JTTNM-3 and JTCNM-3.

$\Omega_r(0)$		$\bar{\varepsilon}_r$ (m)	$\bar{\varepsilon}_v$ (10^{-5} m/s)	σ_r (m)	σ_v (10^{-5} m/s)
3 km	JTTNM-3	0.0010	0.0073	0.0003	0.0020
	JTCNM-3	0.0010	0.0073	0.0003	0.0020
5 km	JTTNM-3	0.0011	0.0080	0.0004	0.0030
	JTCNM-3	0.0010	0.0073	0.0004	0.0030
10 km	JTTNM-3	0.0030	0.0239	0.0029	0.0244
	JTCNM-3	0.0016	0.0121	0.0005	0.0050
25 km	JTTNM-3	0.0916	0.7160	0.1211	1.0084
	JTCNM-3	0.0393	0.0348	0.0177	0.1756
50 km	JTTNM-3	1.4562	11.378	1.9391	16.135
	JTCNM-3	0.6222	5.4751	0.2849	2.8457

Note: $\bar{\varepsilon}_r$ and $\bar{\varepsilon}_v$, σ_r and σ_v respectively indicate the RMSEs in position and velocity, as well as the associated SDs.

methods is shown in Table 3.4, where five different sizes of initial uncertainty domains are considered: 3, 5, 10, 25, and 50 km. Table 3.4 clearly illustrates that the JTTNM-3 method possesses the same accuracy as the JTCNM-3 method when the size of the initial uncertainty is less than 5 km. In this case, the JTTNM-3 method should be preferred due to somewhat less CPU time needs while providing the same accuracy. However, when the size of the initial uncertainty is bigger than 5 km, the JTCNM-3 not only possesses a much smaller RMSEs, but also smaller SDs in position and velocity, showing that the JTCNM-3 method outperforms the JTTNM-3 method. Besides, Fig. 3.5 shows the CPU time consumed in both JTTNM-n and JTCNM-n implementations at different expansion orders and concluding that the JTTNM-n method has a higher computational speed, although its accuracy is somewhat worse than that of the JTCNM-n method.

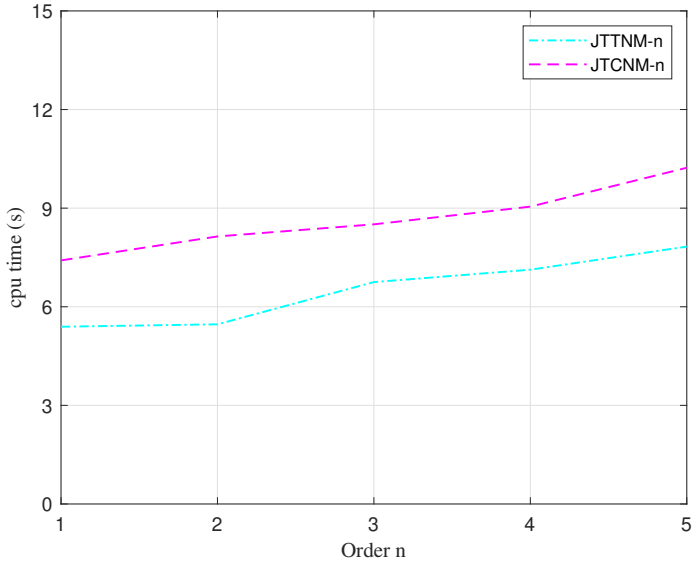
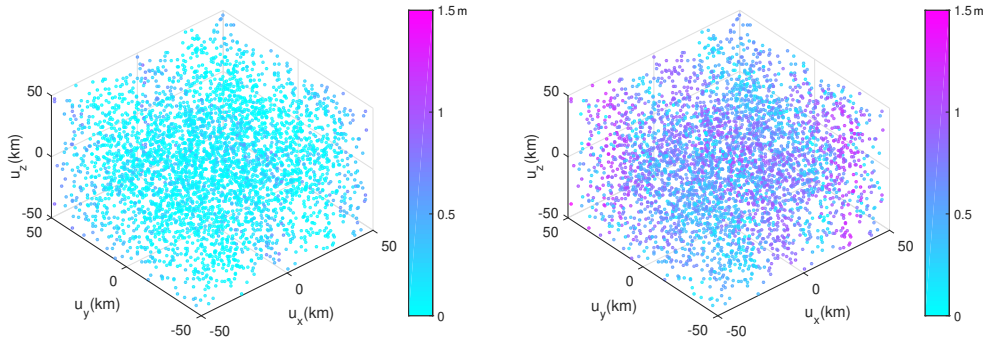


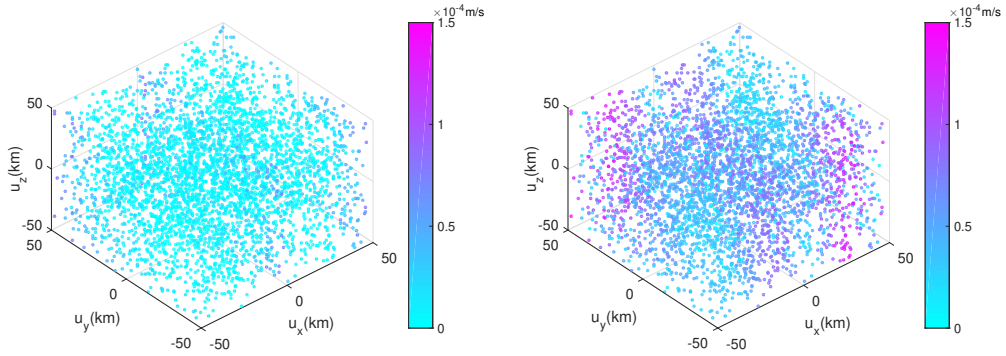
Figure 3.5: The CPU time vs. the expansion order for both JTTNM-n and JTCNM-n.



(a) ADS-based JTTNM-3 **(b)** JTCNM-3

Figure 3.6: Position error distribution of the ADS-based JTTNM-3 and normal JTCNM-3 for propagating an initial set $\Omega_r : u_r(0) = (u_x, u_y, u_z) = (50, 50, 50)$ km for 2 days.

Due to the lower CPU consumption of the JTTNM-n method, it is of interest to explore the possibility of replacing the JTCNM-n method with multiple implementations of the JTTNM-n method with the aim of obtaining similar accuracy and CPU cost when large initial uncertainty domains are considered. The ADS technique [91, 92] tailored for the JTTNM-n method is adopted to test this idea with prescribed position and velocity errors less than 1.5 m and 1.5×10^{-4}m/s. The polynomial evaluation results, obtained by the ADS-based JTTNM-3 method and normal JTCNM-3 method, are compared against that of a classical MC simulation. Figures 3.6 and 3.7 displays position and velocity



(a) ADS-based JTTNM-3 **(b)** JTCNM-3

Figure 3.7: Velocity error distribution of the ADS-based JTTNM-3 and normal JTCNM-3 for propagating an initial set $\Omega_r : u_r(0) = (u_x, u_y, u_z) = (50, 50, 50)$ km for 2 days.

error distributions. Note that just only one split occurs on the component u_x in the implementation of the ADS-based JTTNM-3 method. Table 3.5 again exhibits the overall performance comparison of both ADS-based JTTNM-3 and JTCNM-3 methods, including RMSE, MAE, SD and CPU time. The results illustrate that the ADS-based JTTNM-3 method outperforms the JTCNM-3 method since it possesses smaller RMSEs, MAEs and SDs in position and velocity.

Table 3.5: Performance comparison between ADS-based JTTNM-3 and JTCNM-3.

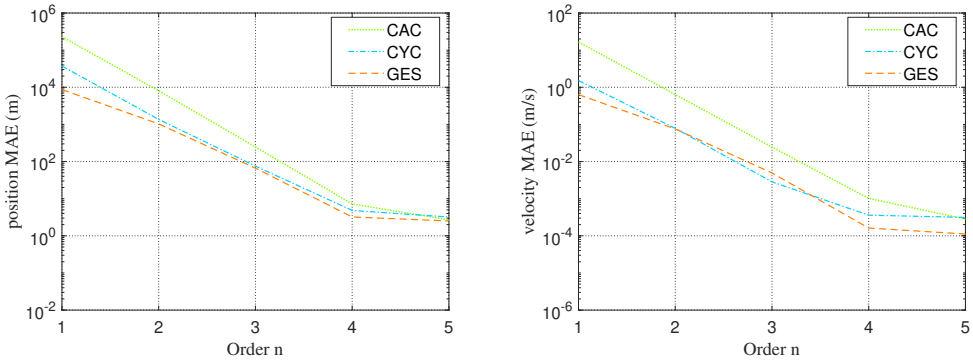
	$\bar{\varepsilon}_r$ (m)	$\bar{\varepsilon}_v$ (m/s)	$\mathcal{E}_r^m$ (m)	$\mathcal{E}_v^m$ (m/s)	σ_r (m)	σ_v (m/s)	t (s)
ADS JTTNM-3	0.12	1.07×10^{-5}	0.83	8.98×10^{-5}	0.15	1.43×10^{-5}	11.1
JTCNM-3	0.62	5.48×10^{-5}	1.46	1.58×10^{-4}	0.28	2.85×10^{-5}	8.51

Note: $\bar{\varepsilon}_r$, $\bar{\varepsilon}_v$, $\mathcal{E}_r^m$, $\mathcal{E}_v^m$ respectively indicate RMSEs and MAEs in position and velocity, σ_r and σ_v are SDs of RMSEs, t is the CPU time. Propagation results of $\Omega_r(0) = (u_x, u_y, u_z) = (50, 50, 50)$ km for 2 days.

These two discussions show that the JTCNM-3 method has smaller RMSEs, MAEs and SDs in position and velocity than the JTTNM-3 method in the test case with a large uncertainty domain, but at the cost of a slight increase of the CPU time. The inverse conclusion is obtained when comparing the JTCNM-3 with the ADS-based JTTNM-3. Therefore, there is a trading-off among the selection of these three methods. In addition, more attention should be paid to the feature of the dynamics, such as the continuity of the vector field. For instance, if the vector fields were not $n+1$-th order continuous differentiable, a JTCNM-n method would be the necessary choice, at least near the discontinuity points.

3.3.3 Comparison between different coordinate representations

The coordinate representation has also a remarkable impact on the propagation accuracy and efficiency of both JTTNM-n and JTCNM-n methods. The preferred choice is the one enabling to describe the dynamics in a near linear way, without neglecting nonlinear terms. The goal is to seek the optimal coordinate representation for the JT-based set propagation of GEO trajectories. Without loss of generality, the JTTNM-n method is adopted. The implementation for propagating an initial uncertainty domain, described by $u_r(0) = (50, 50, 50)$ km and $u_v(0) = (5, 5, 5)$ m/s, is carried out through a forward integration for 2 days of the GEO dynamics formulated in CAC, CYC and GES coordinates.



(a) MAE in position **(b)** MAE in velocity

Figure 3.8: MAEs in position and velocity vs. the expansion order for propagating an initial set $\Omega_r : u_r(0) = (u_x, u_y, u_z) = (50, 50, 50)$ km, $u_v(0) = (u_{v_x}, u_{v_y}, u_{v_z}) = (5, 5, 5)$ m/s using three different coordinate representations.

The MAEs and CPU time are employed to carry out the performance comparisons among the JTTNM-n implementations of the three aforementioned models. Figure 3.8 shows the MAEs in position and velocity computed by the JTTNM-n implementations of the CAC, CYC and GES dynamical models versus the expansion order n. The results illustrate that the equations of motion formulated in the GES coordinates possess the best performance at all expansion orders, followed by far by the one formulated in CYC coordinates, while the one formulated in CAC coordinates is the worst. Using the JTTNM-n with the same expansion order n, the accuracy of both CYC and GES representations are much better than that of the CAC representation, since all CAC coordinates vary quickly along the nominal trajectory. Besides, the approximation error decreases when the expansion order n increases. Figure 3.9 shows the CPU time that a JTTNM-n method spends in the propagation of a specific initial set for two days. We can see the way the CPU time consumed in the implementations in all three coordinate representations increases when the expansion order n also does. On the other hand, the CPU time consumed in the CYC and GES implementations is less than that consumed in the CAC implementation, due to larger integration step-sizes they can use. Figure 3.9 also reveals that, for the same expansion order n, the forward integration of the CAC dynamical model is almost three

times slower than that of the CYC and GES dynamical models. In summary, the GES coordinate representation should be preferred in practical GEO applications due to both the high computational efficiency and the small approximation error.

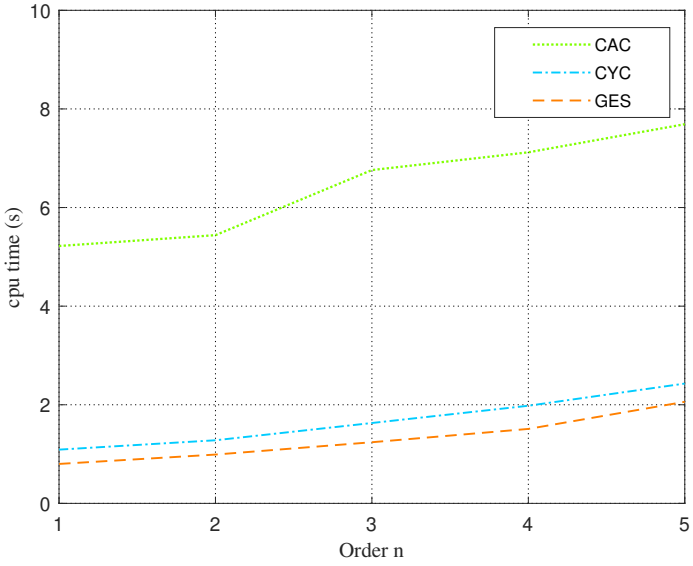


Figure 3.9: The CPU time vs. the expansion order for the JTTNM-n implementation of three different models.

3.4 Remarks

Using the JT technique, this chapter studies the propagations of geostationary trajectory uncertainties subjected to the four dominant perturbations towards efficient orbit determination purposes. The results presented correspond to several simulation scenarios based on different coordinate representations, expansion orders, and uncertainty neighborhoods.

It can be concluded that the JTCNM-n method has a better accuracy and a more uniform error distribution than the JTTNM-n method implemented at the same expansion order, but at the cost of a little more CPU time. Furthermore, the improved JTTNM-n method, by means of using an automatic domain splitting technique, has been compared with the JTCNM-n method under the restrictions of a prescribed accuracy. The comparison reveals that it is possible for the JTTNM-n method to reach the same prescribed performance of the JTCNM-n method through the usage of an automatic domain splitting technique, except for a slight increase of the CPU time. In addition, as a global approximation method, the accuracy of the JTCNM-n method strongly depends on the size of the uncertainty domain, thus, an a priori information about this domain is necessary. On the contrary, the JTTNM-n method is a local approximation method and is not

going to be affected by the size of the uncertainty domain. Its accuracy in the vicinity of the central point is fairly good even in some divergent situations. Therefore, a trading-off should be done for selecting the expansion forms according to many factors, such as the accuracy, CPU time, and the feature of the dynamics.

The performance comparisons among the JT-based propagation of orbit uncertainties using the equations of motion formulated in CAC, CYC and GES coordinates show that the model formulated in the GES coordinates is preferred in practical GEO missions due to its low computational cost and high accuracy. The CPU time consumed in the propagation of the GES representation is a little less than that of the CYC representation and three times less than that of the CAC representation. Furthermore, the propagation accuracy of the GES representation is somewhat better than that of the CYC representation and much better than that of the CAC representation.

4

Chapter 4

Jet transport-based joint orbit and parameter estimation

To simultaneously estimate spacecraft orbits and uncertain parameters, either physically related with the spacecraft or with the measurement procedure, this chapter applies the developed JTTNM-n method into the state prediction step of a Bayesian filtering framework, thus proposes a novel augmented high order extended Kalman filter in the JT framework (JT-AHEKF-n). Two different model representations, including Cartesian coordinates and GEO elements, are used in the implementation of the JT-AHEKF-n estimators and the main dominant perturbations in the GEO regime are taken into consideration. The performance and sensitivity analyses of the proposed JT-AHEKF-n estimators are assessed by means of numerical simulations and compared with those of the classical EKF approach.

4.1 Augmented high order extended Kalman filter

Consider a general nonlinear discretized dynamical system,

$$\boldsymbol{x}_{k+1} = \boldsymbol{\Phi}_{x,k+1}\left(t_{k+1}; t_k, \boldsymbol{x}_k + \delta\boldsymbol{x}_k, \boldsymbol{p}_k + \delta\boldsymbol{p}_k\right) + \boldsymbol{w}_k\,, \tag{4.1}$$

where $\boldsymbol{x}_k \in \mathbb{R}^d$ and $\boldsymbol{p}_k \in \mathbb{R}^l$ respectively indicate nominal spacecraft state and parameter vectors at time t_k, $\delta\boldsymbol{x}_k \in \mathbb{R}^d$, $\delta\boldsymbol{p}_k \in \mathbb{R}^l$ represent initial uncertainty deviation vectors, and $\boldsymbol{w}_k \in \mathbb{R}^d$ is the process noise vector, assumed as a Gaussian white noise process, i.e., $E\{\boldsymbol{w}_k\} = \boldsymbol{0}$, $E\{\boldsymbol{w}_k\boldsymbol{w}_k^T\} = \boldsymbol{Q}_k^x \in \mathbb{R}^{d\times d}$, $k \in \mathbb{N}$. In the above equalities $E\{\cdot\}$ denotes the expectation operator. The initial state estimates are assumed to be a multivariable Gaussian distribution with mean $\boldsymbol{x}_{+,0}$ and covariance $E\{(\boldsymbol{x}_0-\boldsymbol{x}_{+,0})(\boldsymbol{x}_0-\boldsymbol{x}_{+,0})^T\} = \boldsymbol{P}_{+,0}^{\boldsymbol{x}}$. Moreover, it is reasonable to suppose that spacecraft physical parameters, or uncertain parameters related to the ground tracking station, only depend on time or on parameters themselves, such that the discretized parameter equation can be expressed as

$$\boldsymbol{p}_{k+1} = \boldsymbol{\Phi}_{p,k+1}\left(t_{k+1}; t_k, \boldsymbol{p}_k + \delta\boldsymbol{p}_k\right) + \boldsymbol{v}_k\,, \tag{4.2}$$

where $\boldsymbol{v}_k \in \mathbb{R}^l$ denotes the process noise vector, assumed again to be a Gaussian white noise process, i.e., $E\{\boldsymbol{v}_k\} = \boldsymbol{0}$, $E\{\boldsymbol{v}_k\boldsymbol{v}_k^T\} = \boldsymbol{Q}_k^p \in \mathbb{R}^{l\times l}$. The process noise vectors $\boldsymbol{w}_k$ and $\boldsymbol{v}_k$ are further assumed to be non-correlated, i.e. $E\{\boldsymbol{w}_k, \boldsymbol{v}_k\} = \boldsymbol{0}$. Clearly, the parameter vector can be merged into the spacecraft state vector, thus forming an augmented state vector $\boldsymbol{X} = [\boldsymbol{x}, \boldsymbol{p}] \in \mathbb{R}^{d+l}$. It is worth to mention that the full state covariance matrix $\boldsymbol{P}_{+,0} \in \mathbb{R}^{(d+l)\times(d+l)}$ consists of the state covariance sub-matrix $\boldsymbol{P}_{+,0}^{x} \in \mathbb{R}^{d\times d}$, the parameter covariance sub-matrix $\boldsymbol{P}_{+,0}^{p} \in \mathbb{R}^{l\times l}$, and the covariance sub-matrix $\boldsymbol{P}_{+,0}^{xp} \in \mathbb{R}^{d\times l}$ associated to the state and parameter vectors, i.e.,

$$\boldsymbol{P}_{+,0} = \begin{pmatrix} \boldsymbol{P}_{+,0}^{\boldsymbol{x}} & \boldsymbol{P}_{+,0}^{\boldsymbol{xp}} \\ \boldsymbol{P}_{+,0}^{\boldsymbol{px}} & \boldsymbol{P}_{+,0}^{\boldsymbol{p}} \end{pmatrix}, \qquad \text{with} \quad \boldsymbol{P}_{+,0}^{\boldsymbol{px}} = (\boldsymbol{P}_{+,0}^{\boldsymbol{xp}})^T\,. \tag{4.3}$$

Furthermore, (4.1) and (4.2) are augmented as

$$\boldsymbol{X}_{k+1} = \boldsymbol{\Phi}_{k+1}\left(t_{k+1}; t_k, \boldsymbol{X}_k + \delta\boldsymbol{X}_k\right) + \hat{\boldsymbol{w}}_k\,, \tag{4.4}$$

where $\hat{\boldsymbol{w}}_k = [\boldsymbol{w}_k^T, \boldsymbol{v}_k^T]^T \in \mathbb{R}^{d+l}$ indicates the process noise vector, while the full covariance matrix of the process noise $\boldsymbol{Q}_k \in \mathbb{R}^{(d+l)\times(d+l)}$ is a block diagonal matrix consisting of $\boldsymbol{Q}_k^{\boldsymbol{x}} \in \mathbb{R}^{d\times d}$ and $\boldsymbol{Q}_k^{\boldsymbol{p}} \in \mathbb{R}^{l\times l}$, this is

$$\boldsymbol{Q}_k = \begin{pmatrix} \boldsymbol{Q}_k^{\boldsymbol{x}} & \boldsymbol{0} \\ \boldsymbol{0} & \boldsymbol{Q}_k^{\boldsymbol{p}} \end{pmatrix}. \tag{4.5}$$

We assume that the measurement vector at time t_{k+1} can be obtained from the ground tracking station. The nonlinear measurement equation is written as

$$\boldsymbol{z}_{k+1} = \boldsymbol{\Xi}_{k+1}\left(t_{k+1}; t_k, \boldsymbol{X}_{k+1}\right) + \boldsymbol{u}_{k+1}\,, \tag{4.6}$$

where $\boldsymbol{z}_{k+1} \in \mathbb{R}^m$ represents the measurement vector, and the measurement noise vector $\boldsymbol{u}_{k+1} \in \mathbb{R}^m$ is still assumed to be a Gaussian white noise with zero mean $E\{\boldsymbol{u}_{k+1}\} = \boldsymbol{0}$ and covariance $E\{\boldsymbol{u}_{k+1}\boldsymbol{u}_{k+1}^T\} = \boldsymbol{R}_{k+1} \in \mathbb{R}^{m\times m}$. The process and measurement noises are assumed to be non-correlated, i.e., $E\{\boldsymbol{w}_k, \boldsymbol{u}_k\} = \boldsymbol{0}$, $E\{\boldsymbol{v}_k, \boldsymbol{u}_k\} = \boldsymbol{0}$.

We consider a nonlinear filtering problem with the dynamics (4.4) and measurement (4.6), the detailed recursive process of a JT-AHEKF-n can be described as:

Prediction step: If the state vector $\boldsymbol{X}_{+,k}$ and the associated state error covariance matrix $\boldsymbol{P}_{+,k}$ are given at time t_k, using (2.67), the a priori estimates of the state mean $\boldsymbol{X}_{-,k+1}$ and measurement mean $\boldsymbol{z}_{-,k+1}$, as well as the state error covariance matrix $\boldsymbol{P}_{-,k+1}$ at time t_{k+1}, are predicted by means of expanding the state transition function $\boldsymbol{\Phi}_{k+1}$ and the measurement function $\boldsymbol{\Xi}_{k+1}$ around the state mean $\boldsymbol{X}_{+,k}$ at time t_k, up to order n,

$$
\left\{\begin{array}{rcl}
\boldsymbol{X}^i_{-,k+1} &=& E\{\boldsymbol{\Phi}^i_{k+1}\left(t_{k+1};t_k,\boldsymbol{X}_{+,k}+\delta\boldsymbol{X}_k\right)+\hat{\boldsymbol{w}}^i_k\} \\
&=& \boldsymbol{X}^i_{k+1}+\sum\limits_{1\le|\boldsymbol{\gamma}|\le n} a^i_{\gamma_1\ldots\gamma_{d+l}} E\left\{\delta X^{\gamma_1}_{k,1}\cdots\delta X^{\gamma_{d+l}}_{k,d+l}\right\}, \\
\boldsymbol{z}^j_{-,k+1} &=& E\{\boldsymbol{\Xi}^j_{k+1}\left(t_{k+1};t_k,\boldsymbol{X}_{-,k+1}\right)+\boldsymbol{u}^j_{k+1}\} \\
&=& \boldsymbol{z}^j_{k+1}+\sum\limits_{1\le|\boldsymbol{\gamma}|\le n} b^j_{\gamma_1\ldots\gamma_{d+l}} E\left\{\delta X^{\gamma_1}_{k,1}\cdots\delta X^{\gamma_{d+l}}_{k,d+l}\right\}, \\
\boldsymbol{P}^{i_1i_2}_{-,k+1} &=& E\{[\boldsymbol{\Phi}^{i_1}_{k+1}\left(t_{k+1};t_k,\boldsymbol{X}_{+,k}+\delta\boldsymbol{X}_k\right)-\boldsymbol{X}^{i_1}_{-,k+1}+\hat{\boldsymbol{w}}^{i_1}_k]\cdot \\
&& \quad[\boldsymbol{\Phi}^{i_2}_{k+1}\left(t_{k+1};t_k,\boldsymbol{X}_{+,k}+\delta\boldsymbol{X}_k\right)-\boldsymbol{X}^{i_2}_{-,k+1}+\hat{\boldsymbol{w}}^{i_2}_k]\}, \\
&=& \sum\limits_{1\le|\boldsymbol{\gamma}|\le n}\sum\limits_{1\le|\tilde{\boldsymbol{\gamma}}|\le n} a^{i_1}_{\gamma_1\ldots\gamma_{d+l}} a^{i_2}_{\tilde{\gamma}_1\ldots\tilde{\gamma}_{d+l}} E\left\{\delta X^{\gamma_1+\tilde{\gamma}_1}_{k,1}\cdots\delta X^{\gamma_{d+l}+\tilde{\gamma}_{d+l}}_{k,d+l}\right\} \\
&& -\delta\boldsymbol{m}^{i_1}_{k+1}\delta\boldsymbol{m}^{i_2}_{k+1}+\boldsymbol{Q}^{i_1i_2}_{k+1},
\end{array}\right. \tag{4.7}
$$

where

$$
|\boldsymbol{\gamma}|=\sum_{i=1}^{d+l}\gamma_i, \qquad |\tilde{\boldsymbol{\gamma}}|=\sum_{i=1}^{d+l}\tilde{\gamma}_i,
$$

$$
\delta\boldsymbol{m}^i_{k+1}=\boldsymbol{X}^i_{k+1}-\boldsymbol{X}^i_{-,k+1}, \qquad \boldsymbol{Q}^{i_1i_2}_k=E\left\{\hat{\boldsymbol{w}}^{i_1}_k\hat{\boldsymbol{w}}^{i_2}_k\right\},
$$

$$
1\le i,i_1,i_2\le d+l, \quad 1\le j\le m, \quad \delta\boldsymbol{X}_k=[\delta\boldsymbol{x}^T_k,\delta\boldsymbol{p}^T_k]^T,
$$

$$
\boldsymbol{X}^i_{k+1}=\boldsymbol{\Phi}^i_{k+1}\left(t_{k+1};t_k,\boldsymbol{X}_{+,k}\right), \quad \boldsymbol{z}^j_{k+1}=\boldsymbol{\Xi}^j_{k+1}\left(t_{k+1};t_k,\boldsymbol{X}_{-,k+1}\right),
$$

$$
a^i_{\gamma_1\ldots\gamma_{d+l}}=\frac{1}{\gamma_1!\cdots\gamma_{d+l}!}\frac{\partial^\gamma\boldsymbol{\Phi}^i_{k+1}}{\partial X^{\gamma_1}_{k,1}\cdots\partial X^{\gamma_{d+l}}_{k,d+l}}, \quad b^j_{\gamma_1\ldots\gamma_{d+l}}=\frac{1}{\gamma_1!\cdots\gamma_{d+l}!}\frac{\partial^\gamma\boldsymbol{\Xi}^j_{k+1}}{\partial X^{\gamma_1}_{k,1}\cdots\partial X^{\gamma_{d+l}}_{k,d+l}}.
$$

Note that $a^i_{\gamma_1\ldots\gamma_{d+l}}$, $b^j_{\gamma_1\ldots\gamma_{d+l}}$ in (4.7) indicate the coefficients of Taylor expansions of $\boldsymbol{\Phi}^i_{k+1}$ and $\boldsymbol{\Xi}^j_{k+1}$, respectively. Besides, $\boldsymbol{X}^i_{-,k+1}$, $\boldsymbol{P}^{i_1i_2}_{-,k+1}$ indicate the components of the mean and covariance matrix , while $\boldsymbol{z}^j_{-,k+1}$ denotes the components of the mean of the measurement vector. Besides, although the estimated parameter vector $\boldsymbol{p}$ has been merged into the augmented state $\boldsymbol{X}$, it is worth to mention two special cases for its prediction. In the first case the estimated parameter vector $\boldsymbol{p}$ is assumed to be constant, this is, (4.2) is invariable and degraded into

$$
\boldsymbol{p}^j_{-,k+1}=\boldsymbol{p}^j_{+,k}. \tag{4.8}
$$

To reduce the dimension of the estimation problem and save the computational cost, it is not necessary to propagate the parameter vector in this case, which will be further discussed when the position vector of the ground tracking station is estimated. In the

second case, the parameter equation (4.2) is considered to be nonlinear, such that the full equations (4.7) are employed.

Update step: Using the a priori estimates of the state mean $\boldsymbol{X}_{-,k+1}$ and measurement mean $\boldsymbol{z}_{-,k+1}$, the components of the measurement error covariance matrix $\boldsymbol{P}_{zz,k+1}$ and the ones of the cross correlation matrix $\boldsymbol{P}_{Xz,k+1}$ between the state and measurement vectors are computed by

$$\left\{\begin{array}{rcl} \boldsymbol{P}^{j_1 j_2}_{zz,k+1} & = & E\{[\boldsymbol{\Xi}^{j_1}_{k+1}\left(t_{k+1};t_k,\boldsymbol{X}_{-,k+1}\right)-\boldsymbol{z}^{j_1}_{-,k+1}]\cdot \\ & & \quad [\boldsymbol{\Xi}^{j_2}_{k+1}\left(t_{k+1};t_k,\boldsymbol{X}_{-,k+1}\right)-\boldsymbol{z}^{j_2}_{-,k+1}\}\,, \\ & = & \displaystyle\sum_{1\le|\boldsymbol{\gamma}|\le n}\sum_{1\le|\tilde{\boldsymbol{\gamma}}|\le n} b^{j_1}_{\gamma_1\ldots\gamma_{d+l}} b^{j_2}_{\tilde{\gamma}_1\ldots\tilde{\gamma}_{d+l}} E\left\{\delta x_{k,1}^{\gamma_1+\tilde{\gamma}_1}\cdots\delta x_{k,d+l}^{\gamma_{d+l}+\tilde{\gamma}_{d+l}}\right\} \\ & & -\delta\boldsymbol{n}^{j_1}_{k+1}\delta\boldsymbol{n}^{j_2}_{k+1}\,, \\ \boldsymbol{P}^{ij}_{Xz,k+1} & = & E\{[\boldsymbol{\Phi}^{i}_{k+1}\left(t_{k+1};t_k,\boldsymbol{X}_{+,k}+\delta\boldsymbol{X}_k\right)-\boldsymbol{X}^{i}_{-,k+1}+\hat{\boldsymbol{w}}^{i_1}_{k}]\cdot \\ & & \quad [\boldsymbol{\Xi}^{j}_{k+1}\left(t_{k+1};t_k,\boldsymbol{X}_{-,k+1}\right)-\boldsymbol{z}^{j}_{-,k+1}+\boldsymbol{u}^{j}_{k+1}]\} \\ & = & \displaystyle\sum_{1\le|\boldsymbol{\gamma}|\le n}\sum_{1\le|\tilde{\boldsymbol{\gamma}}|\le n} a^{i}_{\gamma_1\ldots\gamma_{d+l}} b^{j}_{\tilde{\gamma}_1\ldots\tilde{\gamma}_{d+l}} E\left\{\delta x_{k,1}^{\gamma_1+\tilde{\gamma}_1}\cdots\delta x_{k,d+l}^{\gamma_{d+l}+\tilde{\gamma}_{d+l}}\right\} \\ & & -\delta\boldsymbol{m}^{i}_{k+1}\delta\boldsymbol{n}^{j}_{k+1}\,, \end{array}\right. \tag{4.9}$$

where $1\le i\le d+l$, $1\le j, j_1, j_2\le m$, $\delta\boldsymbol{n}^{j}_{k+1}=\boldsymbol{z}^{j}_{k+1}-\boldsymbol{z}^{j}_{-,k+1}$. $\boldsymbol{P}^{j_1 j_2}_{zz,k+1}$ indicates the components of the measurement error covariance matrix and $\boldsymbol{P}^{ij}_{Xz,k+1}$ denotes the components of the cross-covariance matrix between the state and measurement vectors.

Thus, the innovation covariance matrix $\boldsymbol{P}_{\nu\nu,k+1}$ is further defined as

$$\boldsymbol{P}^{j_1 j_2}_{\nu\nu,k+1}=\boldsymbol{P}^{j_1 j_2}_{zz,k+1}+\boldsymbol{R}^{j_1 j_2}_{k+1},\quad 1\le j_1, j_2\le m, \tag{4.10}$$

where $\boldsymbol{R}^{j_1 j_2}_{k+1}=E\{\boldsymbol{u}^{j_1}_{k+1}\boldsymbol{u}^{j_2}_{k+1}\}$ represents the components of the measurement noise covariance matrix $\boldsymbol{R}_{k+1}$.

When a new measurement $\boldsymbol{z}^{r}_{k+1}$ is obtained at time t_{k+1}, the measurement innovation vector $\boldsymbol{\nu}_{k+1}$ is defined as,

$$\boldsymbol{\nu}_{k+1}=\boldsymbol{z}^{r}_{k+1}-\boldsymbol{z}_{-,k+1}\,, \tag{4.11}$$

and finally, the updated mean $\boldsymbol{X}_{+,k+1}$ and error covariance matrix $\boldsymbol{P}_{+,k+1}$ at time t_{k+1} can be obtained from

$$\left\{\begin{array}{rcl} \boldsymbol{K}_{k+1} & = & \boldsymbol{P}_{Xz,k+1}\left(\boldsymbol{P}_{vv,k+1}\right)^{-1}\,, \\ \boldsymbol{X}_{+,k+1} & = & \boldsymbol{X}_{-,k+1}+\boldsymbol{K}_{k+1}\boldsymbol{\nu}_{k+1}\,, \\ \boldsymbol{P}_{+,k+1} & = & \boldsymbol{P}_{-,k+1}-\boldsymbol{K}_{k+1}\boldsymbol{P}_{vv,k+1}\boldsymbol{K}^{T}_{k+1}\,, \end{array}\right. \tag{4.12}$$

where $\boldsymbol{K}_{k+1}$ is the Kalman filter gain.

Note that the proposed nonlinear JT-AHEKF-n filter is based on the JTTNM-n method and implemented in the JT scheme. Its main advantages contain: 1) nonlinear information extraction of the dynamical and measurement models in the filtering process; 2) save computation time. Apparently, the JT-AHEKF-n filter is a generalization of the EKF in which only first-order Taylor expansion polynomials are used to approximate the functions $\boldsymbol{\Phi}_{k+1}\left(t_{k+1}; t_k, \boldsymbol{x}_{+,k} + \delta\boldsymbol{x}_k\right)$ and $\boldsymbol{\Xi}_{k+1}\left(t_{k+1}; t_k, \boldsymbol{x}_{-,k+1}\right)$, so, one can easily derive the EKF algorithm by taking $n = 1$ in (4.7) and (4.9). Since the JT-AHEKF-n filter (excluding JT-AHEKF-1) enables to extract the nonlinear information from the system, the distribution of the augmented state vector is no longer Gaussian at $t = t_{k+1}$, even if it was Gaussian at $t = t_k$. However, reference [123] showed that it is accurate enough to assume the state distribution being Gaussian in the proposed JT-AHEKF-n filter for OD applications. Under the Gaussian assumption, Isserlis's formula [160] provides an analytical way to compute the associated expectation values in the implementation of the JT-HEKF-n.

Finally, a brief summary of the JT-AHEKF-n implementation is given in Algorithm 4.1.

Algorithm 4.1 JT-AHEKF-n algorithm

1: Initialize the state $\boldsymbol{X}_{+,0}$, state covariance matrix $\boldsymbol{P}_{+,0}$, expansion order n, process and measurement noise covariance matrices $\boldsymbol{Q}$, $\boldsymbol{R}$, final time t_f, $t_k = 0$, $k = 0$;
2: k is increased by 1;
3: Use (4.7) to predict the a priori estimates of the state mean $\boldsymbol{X}_{-,k+1}$ and measurement mean $\boldsymbol{z}_{-,k+1}$, as well as the state error covariance matrix $\boldsymbol{P}_{-,k+1}$ at time t_{k+1};
4: Incorporate the new measurement $\boldsymbol{z}^r_{k+1}$ into the update step (4.9)-(4.12), thus obtain the a posterior estimates of the state mean $\boldsymbol{X}_{+,k+1}$ and state covariance matrix $\boldsymbol{P}_{+,k+1}$ at time t_{k+1};
5: Execute steps from 2 to 4 until $t_k = t_f$.

4.2 Joint orbit and parameter estimation

The purpose of this chapter is to investigate how parameters can be estimated jointly with the spacecraft state. The nature of the parameters can be very different, from physical ones related with the spacecraft to the uncertain ones associated with the measurement procedure. Without loss of generality, and in order to show the behavior of the developed JT-AHEKF-n method, here only two illustrative cases are presented: one physically affected by the time-varying area-to-mass ratio, and another one related with an uncertain position of the tracking station.

4.2.1 Equations of motion

To accurately predict the motion of GEO satellites, the dynamical models formulated in the Cartesian coordinates and GEO elements are adopted in the OD procedures (see

in subsections 3.1.1 and 3.1.3). The dominant perturbations considered contain SRP, Earth's non-spherical gravity, and luni-solar gravitational attraction, whose formulae are shown in subsection 3.2.1.

4.2.2 Measurement model

Currently, optical telescopes are typically exploited to measure the topocentric right ascension (RA, α) and declination (δ) of space objects orbiting the Earth. As a passive data collection technique, optical telescopes enable to efficiently track numerous objects surreptitiously. The recent development of more powerful wide-field-of-view optical sensors possess sub-arcsecond observation accuracy for the near-geosynchronous objects [156]. In particular, the state-of-the-art accuracy of the most sophisticated optical sensors enables to reach 20 milli-arcseconds [161]. In the following simulations, the measurement errors on the topocentric RA (α) and declination (δ) are assumed to be Gaussian white noises with the same standard deviation 66.6 milli-arcseconds (corresponding to $\sigma_m = 3.232 \times 10^{-7}$ rad and 14 m in the GEO regime). The position of the ground tracking station is known by specifying its east longitude Λ, geodetic latitude ϕ, and an elevation H above the WGS84 ellipsoidal surface. The local sidereal time of the ground tracking station is $s_o = \Lambda + GA(t)$. After the coordinate transformation, the position vector of the ground tracking station in the ECI frame can be expressed as $\boldsymbol{r_o} = (x_o, y_o, z_o) = (R_c \cos\phi \cos s_o,\ R_c \cos\phi \sin s_o,\ R_s \sin\phi)$, where $\boldsymbol{r_o}$ indicates the inertial position vector of the tracking station, while R_c and R_s are two intermediate values depending on the equatorial radius of the Earth, R_e, flattening, f, and height of the tracking station, H. At time t, the spacecraft inertial position vector is denoted as $\boldsymbol{r} = (x, y, z)$, thus the relative position vector from the ground tracking station to the spacecraft can be expressed as $\boldsymbol{\varrho} = \boldsymbol{r} - \boldsymbol{r}_o = (x - x_o\,, y - y_o\,, z - z_o)$. Therefore, the measurement model is written as,

$$\begin{cases} \alpha &= \arctan\left(\dfrac{y - y_o}{x - x_o}\right) + u_1\,, \\ \delta &= \arcsin\left(\dfrac{z - z_o}{||\boldsymbol{\varrho}||_2}\right) + u_2\,, \end{cases} \tag{4.13}$$

where $\boldsymbol{z} = (\alpha, \delta)^T \in \mathbb{R}^2$ represents the measurement vector.

It is worth mentioning that, in practice, the real angular measurement vector, i.e., right ascension and declination, is provided by optical telescopes or radars. To simulate such a measurement vector in the following discussions, a numerical simulator is employed: firstly a true nominal trajectory is generated through the forward integration of a higher fidelity dynamical model from time t_0 to t_k, with consideration of 10×10 EGM96S gravity model, SRP, albedo, solar and lunar gravitational perturbations; then, the measurement vector including topocentric RA (α) and declination (δ) at time t_k is obtained by evaluating the measurement model at the nominal trajectory; finally, the measurement simulator adds some Gaussian white noises into the aforementioned values of RA (α) and declination (δ) and forms the sequential measurement vectors.

4.2.3 Case A: spacecraft state and physical parameter estimation

Although many spacecraft physical parameters are pre-designed and precisely manufactured for concrete missions, still some of them must be determined or adjusted during practical missions, and need to be estimated in real time. Examples of such ones could be the spacecraft mass and the illuminated cross sectional area. The uncertainty on the mass mainly originates from inaccurate operations of the propulsion system, while the uncertainty on the illuminated cross sectional area can be affected by the shape and attitude of the spacecraft, as well as the relative position between the spacecraft and the Sun.

In general, the modelling error of the SRP acceleration is the largest dynamical error source in the GEO regime [162]. Moreover, due to maneuvers, the area-to-mass ratio $\eta = A/m$ changes with time and needs refitting. In this case study the simulation is assumed inside a period of time without maneuvers, but a time-varying η is employed because the non-spherical GEO spacecraft is rotating with respect to the Sun in a diurnal basis, and so, its cross sectional area varies accordingly. The nonlinear variation law considered in the simulation is assumed to be

$$\dot{\eta} = \eta_0 \sin(\omega_e t)\,, \tag{4.14}$$

where ω_e stands for the average angular speed of the Earth's rotation.

4.2.4 Case B: spacecraft state and tracking station position estimation

For usual data collection techniques, underlying systematic biases and unmodeled observation errors significantly degrade the measurement quality. These factors limit the accuracy improvement of the OD tasks, especially in the GEO regime, where one arcsecond of the angular measurement error results in approximately 200 meters of the observational error in the spacecraft position. Therefore, the precise estimation of these systematic biases and unmodeled observation errors, as well as the offset to remove them, is of significant interest and importance. In particular, an accurate estimation of the height of a removable ground tracking station is very meaningful, since it is difficult to be precisely determined via GPS services.

In this case study the uncertain position of a ground tracking station and spacecraft orbit are jointly estimated. In general, the position of the ground tracking station is provided by specifying its east longitude Λ, geodetic latitude ϕ, and an elevation H above the WGS84 ellipsoidal surface, and then, the position vector has to be converted into the ECI frame. Unlike the case study A, the parameters are now invariable and (4.2) can be simplified into (4.8). Therefore, assuming that the height of the ground tracking station is denoted by H and its corresponding uncertainty by δH, the parameter equation can be expressed as

$$H_{k+1} = H_k + \delta H_k + v_k\,. \tag{4.15}$$

4.2.5 Evaluation metrics

The performance of the proposed JT-AHEKF-n filter can be evaluated by the estimation error, the error dispersion and the computational cost. The estimation error vector at time t_k is generally defined as the difference between the estimated state and parameter vectors (i.e., $\boldsymbol{x}_{+,k}$, $\boldsymbol{p}_{+,k}$), computed by the JT-AHEKF-n, and the true state and parameter vectors (i.e., $\boldsymbol{x}_k^r$, $\boldsymbol{p}_k^r$), obtained by the numerical integration of the state and parameter equations. Thus the state and parameter errors can be expressed as

$$\boldsymbol{\varepsilon}_{\boldsymbol{x},k} = \boldsymbol{x}_{+,k} - \boldsymbol{x}_k = (\varepsilon_{x,k}, \varepsilon_{y,k}, \varepsilon_{z,k}, \varepsilon_{v_x,k}, \varepsilon_{v_y,k}, \varepsilon_{v_z,k})^T \,, \tag{4.16}$$

and

$$\boldsymbol{\varepsilon}_{\boldsymbol{p},k} = \boldsymbol{p}_{+,k} - \boldsymbol{p}_k \,. \tag{4.17}$$

Using the state error vector $\boldsymbol{\varepsilon}_{\boldsymbol{x},k}$, the estimation errors in position and velocity can be written as

$$\varepsilon_{r,k} = \sqrt{\varepsilon_{x,k}^2 + \varepsilon_{x,y}^2 + \varepsilon_{x,z}^2} \,, \quad \varepsilon_{v,k} = \sqrt{\varepsilon_{v_x,k}^2 + \varepsilon_{v_y,k}^2 + \varepsilon_{v_z,k}^2} \,. \tag{4.18}$$

To assess the accuracy of the JT-AHEKF-n filter, the RMSEs in position, velocity and parameter at the filtering steady stage $t \in [t_{k_s}, t_{k_e}]$ are defined as

$$\varepsilon_\Upsilon = \sqrt{\frac{1}{k_e - k_s + 1} \sum_{k=k_s}^{k_e} \varepsilon_{\Upsilon,k}^2} \,, \quad \Upsilon = r, v, p \,, \tag{4.19}$$

where t_{k_s} and t_{k_e} respectively indicate the start and end time of the filtering steady stage. Besides, a MC simulation with a cardinality of N samples is further employed to assess the performance of the proposed filters. The RMSEs in position and velocity for a MC simulation are defined as

$$\bar{\varepsilon}_\Upsilon = \sqrt{\frac{1}{N} \sum_{i=1}^{N} \left(\varepsilon_\Upsilon^i\right)^2} \,, \quad \Upsilon = r, v, p \,, \tag{4.20}$$

where ε_Υ^i indicates the RMSE of the i-th MC run at the filtering steady stage $t \in [t_{k_s}, t_{k_e}]$. Furthermore, the SD of the RMSEs is also a vital important metric to measure the error dispersion,

$$\sigma_\Upsilon = \sqrt{\frac{1}{N-1} \sum_{i=1}^{N} (\varepsilon_\Upsilon^i - \bar{\varepsilon}_\Upsilon)^2} \,, \quad \Upsilon = r, v, p \,. \tag{4.21}$$

Furthermore, to compare the performance between two JT-AHEKF-n implementations at different expansion orders n_1 and n_2, two elegant statistical indices are defined as follows,

$$\tau_{n_1}^{n_2} = \frac{n_2 \bar{\varepsilon}_\Upsilon}{n_1 \bar{\varepsilon}_\Upsilon} \,, \quad \zeta_{n_1}^{n_2} = \frac{n_2 \sigma_\Upsilon}{n_1 \sigma_\Upsilon} \,, \tag{4.22}$$

where ${}_n\bar{\varepsilon}_\Upsilon$ and ${}_n\sigma_\Upsilon$ denote the RMSEs and its SDs computed by the JT-AHEKF-n algorithm accounting for the expansion order n. Note that $\tau_{n_1}^{n_2}$ reveals the ratio of the RMSEs computed by the JT-AHEKF-n_2 and JT-AHEKF-n_1. If $0 < \tau_{n_1}^{n_2} < 1$, the JT-AHEKF-n_2 algorithm features a better accuracy than the JT-AHEKF-n_1 algorithm, otherwise is worse. The ratio $\zeta_{n_1}^{n_2}$ shows the ratio of the SDs of the RMSEs computed by two MC simulations of both JT-AHEKF-n_2 and JT-AHEKF-n_1 implementations. If $\zeta_{n_1}^{n_2} < 1$, the estimation errors computed by the JT-AHEKF-n_2 algorithm possess the smaller error dispersion than those obtained by the JT-AHEKF-n_1 algorithm, showing that the JT-AHEKF-n_2 algorithm delivers more consistent filtering results. That is, the JT-AHEKF-n_2 algorithm outperforms the JT-AHEKF-n_1 algorithm. Besides, the computational efficiency of the proposed JT-AHEKF-n filter is still assessed by the CPU time consumed since all simulations are performed in the same computing environment.

4.3 Numerical simulations

The performance of a JT-AHEKF-n filter includes the computational burden, estimation error and its dispersion. The computational burden can be measured through the CPU time-keeping if all codes are carried out in the identical computing environment. The estimation error can be addressed by the RMSEs in position, velocity and parameters. To assess the error dispersion of the JT-AHEKF-n filter, the sensitivity analyses with respect to initial state deviations, measurement acquisition frequencies, and the observation geometry between the spacecraft and the ground tracking station (i.e., $\Delta\Lambda_r = \lambda - \Lambda$) are discussed. Besides, for reasons of space and clarity, only the sensitivity analysis results for case study A are given, the same conclusions are also valid for case study B.

4.3.1 Spacecraft physical parameter estimation

In this study case all the equations of the spacecraft physical parameter vector, state vector and measurement vector are considered to be nonlinear. The estimation performances of the JT-AHEKF-n implemented in both Cartesian and GEO representations with the same simulation conditions, are assessed by a series of numerical simulations. Table 4.1 shows the true initial spacecraft state vector and parameters, as well as the true position vector of the considered ground tracking station. It is worth mentioning that the simulations considering small initial state and parameter deviations are omitted since both usual EKF and JT-AHEKF-n proposed herein are capable of achieving the joint GEO orbit and parameter estimation in an accurate way. The relative large errors are assumed in the initial spacecraft state vector and the area-to-mass ratio. The initial deviation of the area-to-mass ratio η is assumed to be 0.1 m^2/kg off from the true value given in Table 4.1, and its initial standard deviation is taken as $\sigma_\eta = 0.1$ m^2/kg. The initial state deviations are of 100 km in the position vector components, and of 0.5 $\mathrm{m/s}^2$ in the velocity vector components. The adopted initial covariance matrix is

$$\boldsymbol{P}_{+,0} = \begin{pmatrix} \boldsymbol{P}^x_{+,0} & \boldsymbol{0} \\ \boldsymbol{0} & \sigma^2_\eta \end{pmatrix}, \quad \boldsymbol{P}^x_{+,0} = \begin{pmatrix} 10^{10}\,\boldsymbol{I}_{3\times3} & \boldsymbol{0} \\ \boldsymbol{0} & 0.25\,\boldsymbol{I}_{3\times3} \end{pmatrix}. \tag{4.23}$$

Table 4.1: True initial values of spacecraft state, parameters, and the ground tracking station position

Parameter	Nominal value	Initial state	Nominal value
Epoch	15 November 2015, 0:0:0 UTC	x_0	24487.8 km
$\eta = A/m$	74/3300 m^2/kg	y_0	34324.4 km
C_r	1.3	z_0	0 km
Λ	42.0516528°	$\dot{x}_0$	-2.50298 km/s
ϕ	0.7293472°	$\dot{y}_0$	1.78568 km/s
H	1620 m	$\dot{z}_0$	0 km/s

The total simulation time is 4 days, provided that 7 measurements can be done per night (10 hours), separated by regular time intervals. Let the angular measurement noise be a Gaussian white noise with standard deviation $\sigma_m = 3.232 \times 10^{-7}$ rad. The measurement noise covariance matrix $\boldsymbol{R}$ is

$$\boldsymbol{R} = \begin{pmatrix} \sigma^2_m & 0 \\ 0 & \sigma^2_m \end{pmatrix}. \tag{4.24}$$

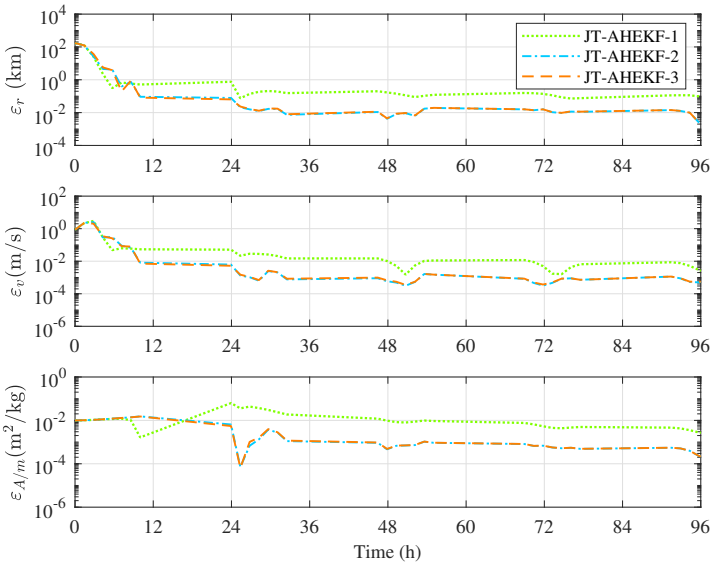


Figure 4.1: Case A: Profiles of estimation errors in position, velocity and area-to-mass ratio with a measurement frequency 7 times/night. Implemented in the GEO representation.

Figures 4.1 and 4.2 exhibit the estimation errors in position, velocity and area-to-mass ratio for a joint GEO orbit determination and parameter estimation problem, respectively

implemented in the Cartesian and GEO representations. Both figures show that the JT-AHEKF-2 filter outputs a better accuracy than the JT-AHEKF-1 one (i.e., the classic EKF filter). For both JT-AHEKF-2 and JT-AHEKF-3 filters, the estimation errors in position and velocity are respectively less than 10 m and 5×10^{-4} m/s. The estimation error of the area-to-mass ratio converges to 2×10^{-4} m^2/kg, which is around 0.1% of the initial deviation of the area-to-mass ratio. In contrast, the JT-AHEKF-1 filter produces the worse estimation errors in position, velocity, and area-to- mass ratio: respectively, 100 m, 3×10^{-3} m/s and 3×10^{-3} m^2/kg. In essence, the JT-AHEKF-1 fails in the estimation of the area-to-mass ratio, as is shown in the bottom line sub-figures of Figs. 4.1 and 4.2. Both sub-figures illustrate that the estimation error of the area-to-mass ratio is almost 14% of the initial deviation of the area-to-mass ratio, so the filter does not work properly. From this simulation, it follows that one can at least obtain one order of magnitude accuracy gain replacing the EKF (i.e., the JT-AHEKF-1 filter) by the JT-AHEKF-2 filter. Furthermore, due to the normality hypothesis, the expectations of the third order terms vanish, thus no accuracy gain is obtained replacing the JT-AHEKF-2 by the JT-AHEKF-3.

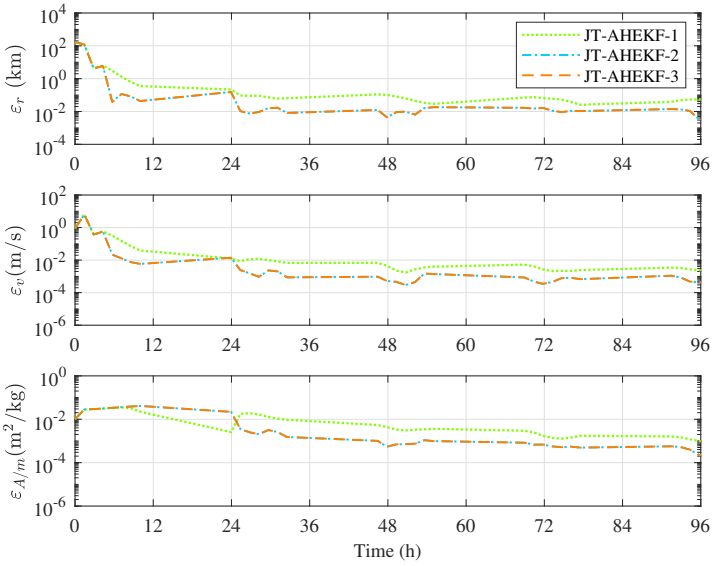


Figure 4.2: Case A: Profiles of estimation errors in position, velocity and area-to-mass ratio with a measurement frequency 7 times/night. Implemented in the Cartesian representation.

Comparing Fig. 4.1 with Fig. 4.2, it follows that the accuracy of both representations is of the same order of magnitude but, in general, the accuracy of GEO representation is a little better than that of the Cartesian representation if the algorithms are convergent (i.e., JT-AHEKF-2 and JT-AHEKF-3). This phenomenon originates from the more accurate prediction achieved by the GEO representation in the state prediction step. Although the Cartesian representation behaves better than the GEO representation in the JT-AHEKF-1 implementation, it makes no sense since the first order algorithm is essentially failing.

Another important issue to be pointed out here is related with the computational efficiency. Table 4.2 shows the computational cost of the simulations corresponding to Figs. 4.1 and 4.2 (without including the computational cost for generating the measurements). It underlines that the implementation of the GEO representation is much faster than that of the Cartesian one, because the GEO elements vary much slower than the classical Cartesian ones in the GEO regime. Thus, a larger integration step-size can be adopted in the prediction process, resulting in less computation time and in a higher efficiency. Therefore, one can conclude that the GEO representation possesses a better performance (including a superior estimation accuracy and less computational burden) than the Cartesian representation. This conclusion is also validated in the case study B. But, for brevity, in what follows we only show the results of the GEO representation for the case study B, while the ones corresponding to the Cartesian representation are omitted.

Table 4.2: Computation time (s)

Coordinate representation	JT-AHEKF-n		
	$n = 1$	$n = 2$	$n = 3$
GEO elements	24	44	114
Cartesian coordinates	144	149	197

4.3.1.1 Sensitivity analysis relative to the observational geometry

The geometry of a GEO spacecraft relative to the ground tracking station is of importance in the estimation process of spacecraft state and area-to-mass ratio. Obviously, the GEO spacecraft must be in the visible region from the ground tracking station. With the consideration of a fixed position of the ground tracking station, the relative observational geometry is generally described by the difference in longitude between the spacecraft and the tracking station, denoted by $\Delta\Lambda_r$. To study the influence of the relative observational geometry on the performance of the JT-AHEKF-n, a MC simulation considering 25 uniform sampling points in the interval $\Delta\Lambda_r \in [-60^\circ, -60^\circ]$ is carried out. Apart from the different initial longitude of the spacecraft, same initial conditions and measurements are used.

Table 4.3 provides a deep insight into the result of the sensitivity analysis of the JT-AHEKF-n relative to the observational geometry, which displays the RMSEs in spacecraft position, velocity and area-to-mass ratio computed by the JT-AHEKF-1 and JT-AHEKF-2. The estimation results of the JT-AHEKF-3 filter are omitted, since no accuracy improvement is obtained relative to the JT-AHEKF-2 filter. It is clear that the accuracy of the JT-AHEKF-2 filter is better than that of the JT-AHEKF-1 filter, also the SD of the RMSEs computed by the JT-AHEKF-2 filter at the steady stage $t \in [48, 96]$ h is much smaller than that calculated by the JT-AHEKF-1 filter. For the sake of intuition, Fig. 4.3 exhibits all the error curves obtained in the MC simulation, which again shows that the results computed by the JT-AHEKF-2 filter possess a small error dispersion. Therefore,

it concludes that the accuracy of the JT-AHEKF-1 filter is more strongly dependent on the observational geometry than the one of the JT-AHEKF-2 filter, being less robust.

Table 4.3: Case A: Sensitivity analysis relative to the observational geometry

Filter	Metrics	$\Upsilon = r$ (m)	$\Upsilon = v$ (10^{-4} m/s)	$\Upsilon = \eta$ (10^{-4} m^2/kg)
JT-AHEKF-1	${}_1\bar{\varepsilon}_\Upsilon$	36.7	27.6	7.96
	${}_1\sigma_\Upsilon$	24.1	421.9	438.5
JT-AHEKF-2	${}_2\bar{\varepsilon}_\Upsilon$	7.88	6.45	1.50
	${}_2\sigma_\Upsilon$	2.08	76.8	81.7
Ratio	$\tau_1^2 = {}_2\bar{\varepsilon}_\Upsilon / {}_1\bar{\varepsilon}_\Upsilon$	0.215	0.234	0.188
	$\zeta_1^2 = {}_2\sigma_\Upsilon / {}_1\sigma_\Upsilon$	0.086	0.182	0.186

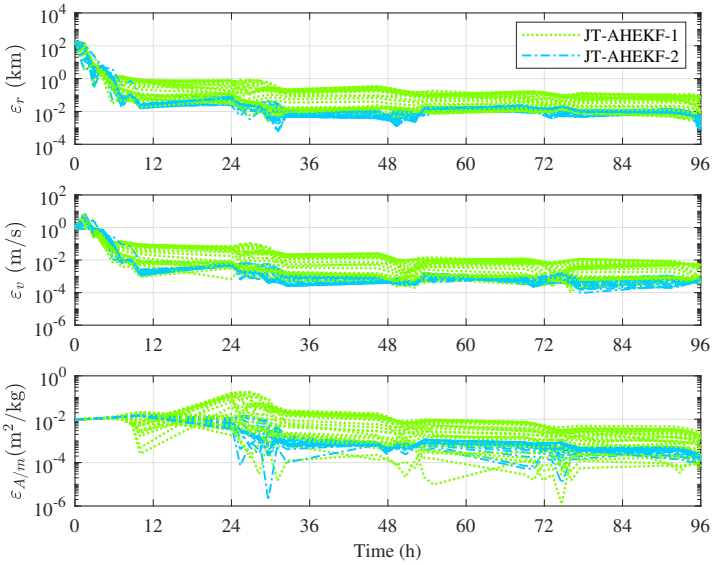


Figure 4.3: Case A: Sensitivity analysis relative to the observational geometry with a measurement frequency of 7 times/night. Implemented in the GEO representation.

4.3.1.2 Sensitivity analysis relative to initial state deviations

To analyze the sensitivity of the JT-AHEKF-n algorithms with respect to initial state deviations, a MC simulation is carried out with the consideration of 2000 initial state

vectors generated around the initial nominal state, subjected to a multivariable Gaussian distribution with a large standard deviation. Table 4.4 displays the RMSEs and the associated SDs computed by means of the JT-AHEKF-1 and JT-AHEKF-2. Note that the JT-AHEKF-2 filter outputs a better accuracy than the JT-AHEKF-1 filter and at least one order of magnitude accuracy gain can be obtain in position, velocity and area-to-mass ratio by adjusting the order n from 1 to 2. Meanwhile, Table 4.4 also shows that a tiny value of the ratio ζ_1^2 is obtained, revealing that the RMSEs in position, velocity and area-to-mass ratio computed by the JT-AHEKF-2 filter possess a smaller error dispersion than those calculated by the JT-AHEKF-1 filter. In other words, the performance of the JT-AHEKF-2 filter displays a much more insensitive with respect to initial state deviations. Besides, Fig. 4.4 detailedly exhibits the profiles of 25 MC runs occupying the largest estimation errors, where the nonlinearities play a prominent role. The results again prove that the JT-AHEKF-2 filter outperforms the JT-AHEKF-1 one since the JT-AHEKF-2 filter possesses not only lower RMSEs but also a smaller error dispersion.

Table 4.4: Case A: Sensitivity analysis relative to initial state deviations

Filter	Metrics	$\Upsilon = r$ (m)	$\Upsilon = v$ (10^{-4} m/s)	$\Upsilon = \eta$ (10^{-4} m^2/kg)
JT-AHEKF-1	${}_1\bar{\varepsilon}_\Upsilon$	296.9	79.0	101.2
	${}_1\sigma_\Upsilon$	117.7	317.6	315.5
JT-AHEKF-2	${}_2\bar{\varepsilon}_\Upsilon$	7.76	7.25	1.82
	${}_2\sigma_\Upsilon$	0.176	71.8	77.4
Ratio	$\tau_1^2 = {}_2\bar{\varepsilon}_\Upsilon / {}_1\bar{\varepsilon}_\Upsilon$	0.026	0.092	0.018
	$\zeta_1^2 = {}_2\sigma_\Upsilon / {}_1\sigma_\Upsilon$	0.001	0.226	0.245

Furthermore, to explore the influence of more measurement information on the performance of the JT-AHEKF-n method, the relative distance from the spacecraft to the ground tracking station is assumed to be available. Therefore, the measurement model should be expressed as

$$\begin{cases} \alpha &= \arctan\left(\dfrac{y-y_o}{x-x_o}\right) + u_1\,, \\ \delta &= \arcsin\left(\dfrac{z-z_o}{||\boldsymbol{\varrho}||_2}\right) + u_2\,, \\ ||\boldsymbol{\varrho}||_2 &= \sqrt{(x-x_o)^2+(y-y_o)^2+(z-z_o)^2} + u_3\,. \end{cases} \tag{4.25}$$

The standard deviation of the measurement noise in the relative distance is supposed to be 1 meter. A MC simulation is implemented to assess the influence of the supplemented relative distance information on the performance of the JT-AHEKF-n method. To compare the results with those displayed in Fig. 4.4, the JT-AHEKF-n is fed with the identical

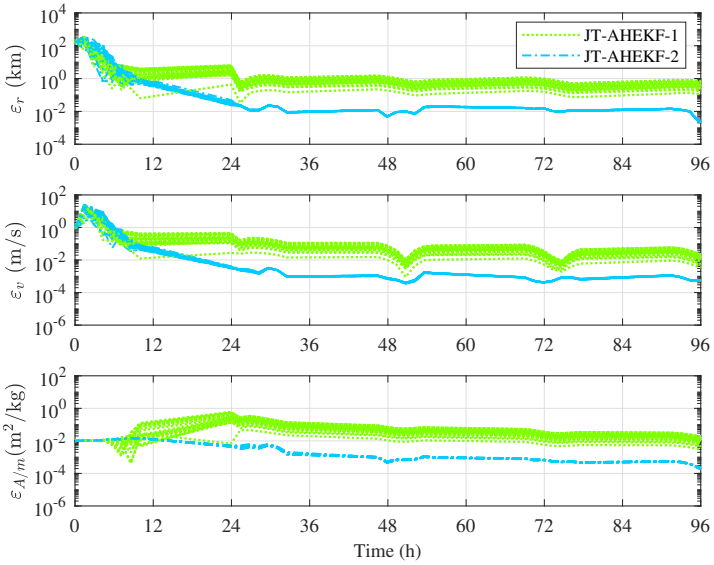


Figure 4.4: Case A: Sensitivity analysis relative to initial state deviations with a measurement frequency of 7 times/night. Implemented in the GEO representation.

Table 4.5: Case A: Sensitivity analysis relative to initial state deviations incorporating the relative distance information

Filter	Metrics	$\Upsilon = r$ (m)	$\Upsilon = v$ (10^{-3} m/s)	$\Upsilon = \eta$ (10^{-4} m^2/kg)
JT-AHEKF-1	${}_1\bar{\varepsilon}_\Upsilon$	204.5	288.0	83.2
	${}_1\sigma_\Upsilon$	77.5	1953.0	2147.0
JT-AHEKF-2	${}_2\bar{\varepsilon}_\Upsilon$	1.22	0.80	0.18
	${}_2\sigma_\Upsilon$	0.07	11.64	12.27
Ratio	$\tau_1^2 = {}_2\bar{\varepsilon}_\Upsilon / {}_1\bar{\varepsilon}_\Upsilon$	0.006	0.003	0.002
	$\zeta_1^2 = {}_2\sigma_\Upsilon / {}_1\sigma_\Upsilon$	0.001	0.006	0.006

measurements (excluding the relative distance information), and initialized with same initial values of the state and area-to-mass ratio. Figure 4.5 shows that the supplement of the measurement information about the relative distance deteriorates the performance of the JT-AHEKF-1 filter, while improve the performance of the JT-AHEKF-2 filter. The comparison between Tables 4.4 and 4.5 further reveals this phenomenon, that is, the supplement of the measurement information about the relative distance worsens the original divergent algorithm diverging more severely (i.e., for JT-AHEKF-1), but it enhances the

estimation accuracy of the original convergent methods (i.e., for JT-AHEKF-2).

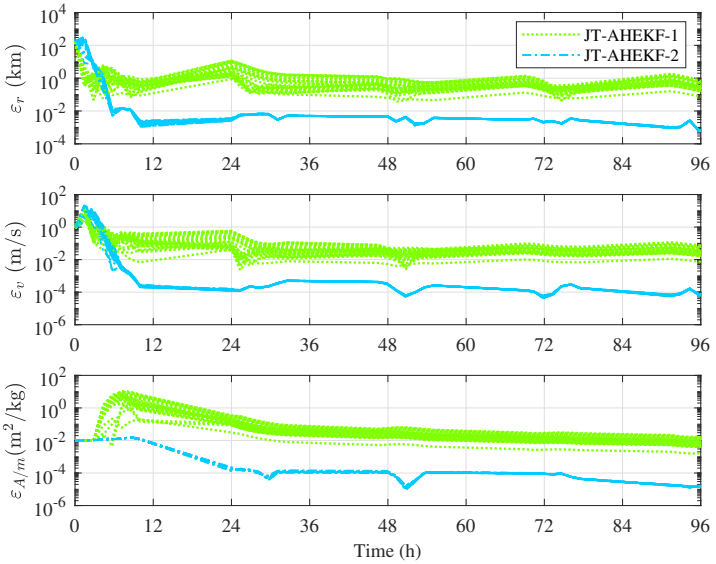


Figure 4.5: Case A: Sensitivity analysis relative to initial state deviations with a measurement frequency of 7 times/night. Implemented in the GEO representation and incorporating the relative distance information.

4.3.1.3 Sensitivity analysis relative to the measurement acquisition frequency

Note that the magnitude of measurement acquisition frequencies is a main factor that affects the performance of the JT-AHEKF-n method. To illustrate this influence, three different measurement acquisition frequencies are considered, including 7, 14 and 21 times per night. In principle, a high measurement acquisition frequency is in favor of the accuracy improvement of the JT-AHEKF-n, especially for the JT-AHEKF-1 since the linearization of the state equations always tends to lose accuracy for long integration time spans (i.e., in the low frequency acquisition case). In contrast, the high order nonlinear approximation of the state equations already achieves a highly precise uncertainty propagation over long time intervals, such that the high order JT-AHEKF-n can not significantly improve the accuracy through the increase of the measurement frequency.

Figures 4.6 and 4.7 respectively display the simulation results with measurement frequencies of 14 and 21 times per night, separated by regular time intervals. The comparison among Figs. 4.1, 4.6, and 4.7 illustrates that a high measurement acquisition frequency benefits for the accuracy improvement in the JT-AHEKF-n implementations at all orders. Note in particular that the low order JT-AHEKF-n filter can obtain a larger accuracy gain than the high order ones. This is, high order JT-AHEKF-n algorithms have a stronger robustness when the tracking sensors encounter the functional degradation (i.e.,

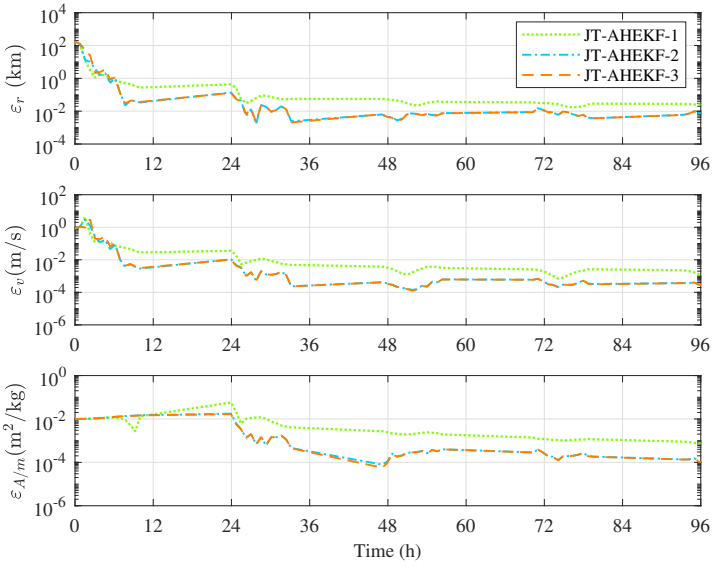


Figure 4.6: Case A: Profiles of estimation errors in position, velocity and area-to-mass ratio with a measurement frequency 14 times/night. Implemented in the GEO representation.

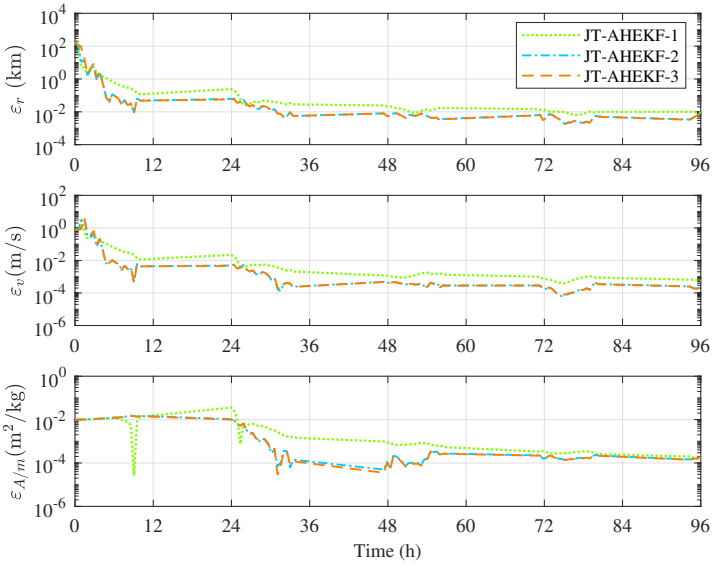


Figure 4.7: Case A: Profiles of estimation errors in position, velocity and area-to-mass ratio with a measurement frequency 21 times/night. Implemented in GEO representation.

when the measurement acquisition frequency decreases significantly). Finally, it is worth to underline that the corresponding sensitivity analyses implemented in the Cartesian representation produce similar results, but they omitted here for brevity.

4.3.2 Tracking station position estimation

The preceding discussions indicate that the JT-AHEKF-n algorithms are capable of simultaneously estimating the spacecraft state and area-to-mass parameter. In this subsection the feasibility of the JT-AHEKF-n algorithms for estimating the underlying systematic biases or unmodeled observational errors is discussed. Clearly, these factors always lead to a significant degeneration of the measurement quality. Therefore, the estimation and compensation of uncertain biases in the measurement equations is of importance to improve the accuracy of the OD tasks. In general, GPS services enable to provide very accurate values of the longitude Λ and geodetic latitude ϕ of a ground tracking station, but not so accurate for the height H. Consequently, the estimation of the underlying height deviation of a ground tracking station is specially meaningful in cases where the tracking stations are mobile. In the following simulation, both the spacecraft state $\boldsymbol{x}$ and an uncertain height H of the ground tracking station are jointly estimated. The true initial state and parameters of the spacecraft, as well as the precise position of the tracking station, are again given in Table 4.1.

Assuming that the initial estimation error of the ground tracking station height is 50 m off from the true height given in Table 4.1, that is, $\delta H_0 = 50$ m, and that the standard deviation of the initial height error is taken as $\sigma_H = 50$ m. The initial state deviations are of 100 km in all position components, and of 0.5 m/s in velocity components. No error on the longitude and latitude of the tracking station is considered. The adopted initial covariance matrix is,

$$\boldsymbol{P}_{+,0} = \begin{pmatrix} \boldsymbol{P}^x_{+,0} & \boldsymbol{0} \\ \boldsymbol{0} & \sigma_H^2 \end{pmatrix}, \quad \boldsymbol{P}^x_{+,0} = \begin{pmatrix} 10^{10}\,\boldsymbol{I}_{3\times 3} & \boldsymbol{0} \\ \boldsymbol{0} & 0.25\,\boldsymbol{I}_{3\times 3} \end{pmatrix}. \tag{4.26}$$

The measurement acquisition frequency is taken again as 7 times per night separated by regular time intervals. The measurement equations (4.13) are still adopted and the associated measurement noise covariance matrix is (4.24). The accuracy comparison of the first, second and third orders JT-AHEKF-n filters, for estimating the spacecraft state and the uncertain height of the ground tracking station, are shown in Fig. 4.8. The result shows that the nonlinear JT-AHEKF-n algorithms ($n > 1$) work very well in the estimation process of both spacecraft state and height of the ground tracking station. The estimation errors in position and velocity computed by the JT-AHEKF-2 and JT-AHEKF-3 filters are respectively around 5 m and 3×10^{-4} m/s, meanwhile the estimation error of the uncertain height converges to 2 m, which is around 4% of the initial height deviation $\delta H_0 = 50$ m. Again, the JT-AHEKF-1 filter produces the worse estimation errors in position, velocity, and height, respectively: 70 m, 0.003 m/s and 5 m (10%). In principle, the JT-AHEKF-2 and JT-AHEKF-3 algorithms are capable of extracting more

nonlinear information from the dynamical system to improve the estimation accuracy, at least one order of magnitude, relative to the JT-AHEKF-1 filter. In addition, the results implemented in the Cartesian representation produce the same conclusions as in the GEO representation, but they are omitted for brevity.

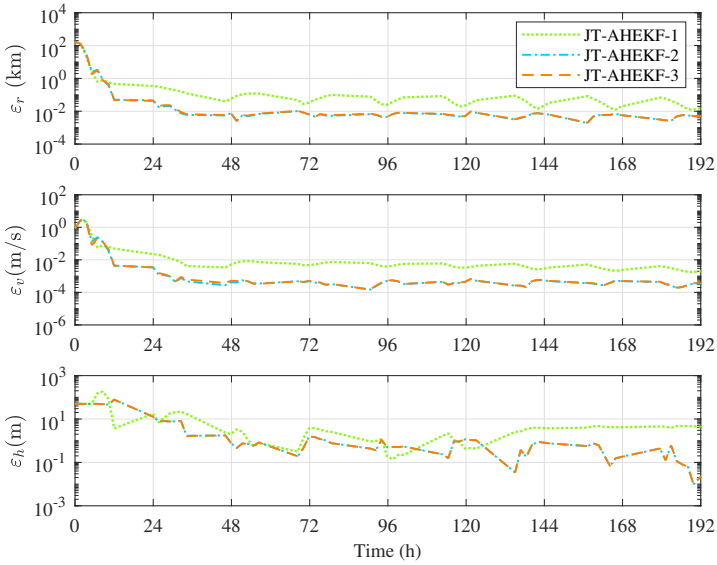


Figure 4.8: Case B: Profiles of estimation errors in position, velocity and height of the ground tracking station with a measurement frequency 7 times/night. Implemented in the GEO representation.

4.4 Remarks and conclusions

This chapter studies augmented JT-AHEKF-n filtering algorithms for the simultaneous estimation of spacecraft state and additional parameters in the GEO regime. These parameters describe either the spacecraft physical features or the position information of the ground tracking station. The effectiveness of the proposed nonlinear estimators has been validated considering two case examples.

The comparison between two adopted dynamical models, formulated in Cartesian coordinates and GEO elements, reveals that the model formulated in GEO elements not only possesses better estimation accuracy, but also needs of less computational cost than the one formulated in Cartesian coordinates. Letting aside the performance comparisons between two coordinate representations, the simulations also confirm that both dynamical models are effective to achieve a real-time joint estimation via the proposed JT-AHEKF-n method.

The performance comparison among the JT-AHEKF-n implementations at different orders is also investigated. The results show that higher order filters provide a superior estimation accuracy, and usually, just by tuning the order of the JT-AHEKF-n from 1 to 2, one order of magnitude accuracy gain can be easily obtained in two case studies considered. Furthermore, a series of detailed sensitivity analyses of the proposed nonlinear filters with respect to the observational geometry, magnitudes of initial estimation errors and measurement acquisition frequencies, has been carried out. It concludes that the JT-AHEKF-n filters (i.e. with $n > 1$) output a smaller estimation error dispersion relative to the classical EKF, pointing to that high order JT-AHEKF-n filters are more robust than the usual EKF. Finally, it should be mentioned that the proposed JT-AHEKF-n method could be suitable to achieve the needs of future applications related with high precise geostationary or geosynchronous orbit determination.

5

Chapter 5

Autonomous orbit determination and fault-tolerant designs

With the fast development of the space technology, an increasing number of artificial satellites have been launched and deployed in near Earth orbits. This is a really big number relative to the small number of ground observation stations for actively tracking them implying that the satellite OD task is very likely to encounter a data starved issue. In other words, OD techniques for current or future space missions inevitably have to employ increasingly sparse observations provided by either ground observation stations or on board measuring sensors [48]. Note that the usage of sparse space-based measurements is appealing due to both its autonomy improvement and reduced loads on measuring equipment, motivating the exploration of advanced OD algorithms.

The tasks developed in this chapter contain: 1) the autonomy improvement of the OD procedure; 2) the robustness enhancement of the OD procedure with respect to the measurement faults. To achieve these two tasks, the augmented JT-AHEKF-n developed in Chapter 4 can be degraded into the standard JT-HEKF-n algorithm if no parameter should be estimated. This chapter further employs the JT-HEKF-n to achieve autonomous satellite orbit estimation via the usage of sparse inter-satellite relative measurement information. Three fault-tolerant algorithms are further put forward to suppress measurement faults. In the light of the insensitiveness of the proposed JT-HEKF-n to the measurement frequency, the first strategy directly discards the identified false mea-

surements, being someway equivalent to the decrease of the measurement frequency. The second and third strategies employ a single scale factor and an adaptive scale matrix, respectively, as a multiplier to the measurement noise covariance matrix in order to alleviate the pollution of faulty measurements. The effectiveness of the JT-HEKF-n are firstly demonstrated with respect to diverse initial state deviations and measurement errors, while the behavior exploration of fault-tolerant nonlinear filters are compared with the standard JT-HEKF-n method considering different measurement faults.

5.1 Standard high order extended Kalman filter

Considering a general nonlinear discretized dynamical system,

$$\begin{cases} \boldsymbol{x}_{k+1} &= \boldsymbol{\Phi}_{k+1}\left(t_{k+1};t_k,\boldsymbol{x}_k\right)+\boldsymbol{w}_k, \\ \boldsymbol{z}_{k+1} &= \boldsymbol{\Xi}_{k+1}\left(t_{k+1};t_k,\boldsymbol{x}_{k+1}\right)+\boldsymbol{u}_{k+1}, \end{cases} \tag{5.1}$$

where $k\in\mathbb{N}$, $\boldsymbol{x}\in\mathbb{R}^d$ and $\boldsymbol{z}\in\mathbb{R}^m$ indicate the state and measurement vectors, with $\boldsymbol{\Phi}$ and $\boldsymbol{\Xi}$ defining the state transition and the measurement functions. $\boldsymbol{w}_k\in\mathbb{R}^d$ and $\boldsymbol{u}_k\in\mathbb{R}^m$ are uncorrelated zero mean Gaussian random noise sequences, i.e. $E\{\boldsymbol{w}_k\}=E\{\boldsymbol{u}_k\}=\boldsymbol{0}$, $E\{\boldsymbol{w}_k\cdot\boldsymbol{w}_k^T\}=\boldsymbol{Q}_k$, $E\{\boldsymbol{u}_k\cdot\boldsymbol{u}_k^T\}=\boldsymbol{R}_k$, $E\{\boldsymbol{w}_k\cdot\boldsymbol{u}_k^T\}=\boldsymbol{0}$. $\boldsymbol{Q}_k\in\mathbb{R}^{d\times d}$ and $\boldsymbol{R}_k\in\mathbb{R}^{m\times m}$ are the symmetric real matrices addressing the process and measurement noises respectively.

A detailed recursive process of a JT-HEKF-n algorithm can be obtained by simplifying the derivation of the augmented JT-AHEKF-n algorithm of section 4.1 without the consideration of an uncertain parameter vector:

Prediction step: If the state vector $\boldsymbol{x}_{+,k}$ and the associated state error covariance matrix $\boldsymbol{P}_{+,k}$ are given at time t_k, the a priori estimates of the state mean $\boldsymbol{x}_{-,k+1}$ and measurement mean $\boldsymbol{z}_{-,k+1}$, as well as the state error covariance matrix $\boldsymbol{P}_{-,k+1}$ at time t_{k+1}, are predicted by means of expanding the state transition function $\boldsymbol{\Phi}_{k+1}$ and the measurement function $\boldsymbol{\Xi}_{k+1}$ around the state mean $\boldsymbol{x}_{+,k}$ at time t_k, up to order n,

$$\begin{cases} \boldsymbol{x}^i_{-,k+1} &= E\{\boldsymbol{\Phi}^i_{k+1}\left(t_{k+1};t_k,\boldsymbol{x}_{+,k}+\delta\boldsymbol{x}_k\right)+\boldsymbol{w}^i_k\} \\ &= \boldsymbol{x}^i_{k+1}+\sum\limits_{1\le|\boldsymbol{\gamma}|\le n} a^i_{\gamma_1\dots\gamma_d}E\left\{\delta x^{\gamma_1}_{k,1}\cdots\delta x^{\gamma_d}_{k,d}\right\}, \\ \boldsymbol{z}^j_{-,k+1} &= E\{\boldsymbol{\Xi}^j_{k+1}\left(t_{k+1};t_k,\boldsymbol{x}_{-,k+1}\right)+\boldsymbol{u}^j_{k+1}\} \\ &= \boldsymbol{z}^j_{k+1}+\sum\limits_{1\le|\boldsymbol{\gamma}|\le n} b^j_{\gamma_1\dots\gamma_d}E\left\{\delta x^{\gamma_1}_{k,1}\cdots\delta x^{\gamma_d}_{k,d}\right\}, \\ \boldsymbol{P}^{i_1i_2}_{-,k+1} &= E\{[\boldsymbol{\Phi}^{i_1}_{k+1}\left(t_{k+1};t_k,\boldsymbol{x}_{+,k}+\delta\boldsymbol{x}_k\right)-\boldsymbol{x}^{i_1}_{-,k+1}+\boldsymbol{w}^{i_1}_k]\cdot \\ &\qquad [\boldsymbol{\Phi}^{i_2}_{k+1}\left(t_{k+1};t_k,\boldsymbol{x}_{+,k}+\delta\boldsymbol{x}_k\right)-\boldsymbol{x}^{i_2}_{-,k+1}+\boldsymbol{w}^{i_2}_k]\}, \\ &= \sum\limits_{1\le|\boldsymbol{\gamma}|\le n}\sum\limits_{1\le|\tilde{\boldsymbol{\gamma}}|\le n} a^{i_1}_{\gamma_1\dots\gamma_d}a^{i_2}_{\tilde{\gamma}_1\dots\tilde{\gamma}_d}E\left\{\delta x^{\gamma_1+\tilde{\gamma}_1}_{k,1}\cdots\delta x^{\gamma_d+\tilde{\gamma}_d}_{k,d}\right\} \\ &\qquad -\delta\boldsymbol{m}^{i_1}_{k+1}\delta\boldsymbol{m}^{i_2}_{k+1}+\boldsymbol{Q}^{i_1i_2}_{k+1}, \end{cases} \tag{5.2}$$

where

$$1 \le i, i_1, i_2 \le d\,, \quad 1 \le j \le m\,, \quad |\boldsymbol{\gamma}| = \sum_{i=1}^{d} \gamma_i\,, \quad |\tilde{\boldsymbol{\gamma}}| = \sum_{i=1}^{d} \tilde{\gamma}_i\,,$$

$$\boldsymbol{x}_{k+1}^{i} = \boldsymbol{\Phi}_{k+1}^{i}\left(t_{k+1}; t_k, \boldsymbol{x}_{+,k}\right)\,, \quad \boldsymbol{z}_{k+1}^{j} = \boldsymbol{\Xi}_{k+1}^{j}\left(t_{k+1}; t_k, \boldsymbol{x}_{-,k+1}\right)\,,$$

$$\delta\boldsymbol{m}_{k+1}^{i} = \boldsymbol{x}_{k+1}^{i} - \boldsymbol{x}_{-,k+1}^{i}\,, \quad \boldsymbol{Q}_{k+1}^{i_1 i_2} = E\left\{\boldsymbol{w}_{k+1}^{i_1}\boldsymbol{w}_{k+1}^{i_2}\right\}\,,$$

$$a_{\gamma_1\ldots\gamma_d}^{i} = \frac{1}{\gamma_1!\cdots\gamma_d!}\frac{\partial^{\gamma}\boldsymbol{\Phi}_{k+1}^{i}}{\partial x_{k,1}^{\gamma_1}\cdots\partial x_{k,d}^{\gamma_d}}\,, \quad b_{\gamma_1\ldots\gamma_d}^{j} = \frac{1}{\gamma_1!\cdots\gamma_d!}\frac{\partial^{\gamma}\boldsymbol{\Xi}_{k+1}^{j}}{\partial x_{k,1}^{\gamma_1}\cdots\partial x_{k,d}^{\gamma_d}}\,.$$

Update step: Using the a priori estimates of the state mean $\boldsymbol{x}_{-,k+1}$ and measurement mean $\boldsymbol{z}_{-,k+1}$, the components of the measurement error covariance matrix $\boldsymbol{P}_{zz,k+1}$, and the ones of the cross correlation matrix $\boldsymbol{P}_{xz,k+1}$ between the state and measurement vectors are

$$\left\{\begin{array}{rcl}
\boldsymbol{P}_{zz,k+1}^{j_1 j_2} & = & E\{[\boldsymbol{\Xi}_{k+1}^{j_1}\left(t_{k+1}; t_k, \boldsymbol{x}_{-,k+1}\right) - \boldsymbol{z}_{-,k+1}^{j_1}]\cdot \\
& & \quad [\boldsymbol{\Xi}_{k+1}^{j_2}\left(t_{k+1}; t_k, \boldsymbol{x}_{-,k+1}\right) - \boldsymbol{z}_{-,k+1}^{j_2}]\}, \\
& = & \displaystyle\sum_{1\le|\boldsymbol{\gamma}|\le n}\sum_{1\le|\tilde{\boldsymbol{\gamma}}|\le n} b_{\gamma_1\ldots\gamma_d}^{j_1} b_{\tilde{\gamma}_1\ldots\tilde{\gamma}_d}^{j_2} E\left\{\delta x_{k,1}^{\gamma_1+\tilde{\gamma}_1}\cdots\delta x_{k,d}^{\gamma_d+\tilde{\gamma}_d}\right\} \\
& & -\delta\boldsymbol{n}_{k+1}^{j_1}\delta\boldsymbol{n}_{k+1}^{j_2}\,, \\
\boldsymbol{P}_{xz,k+1}^{ij} & = & E\{[\boldsymbol{\Phi}_{k+1}^{i}\left(t_{k+1}; t_k, \boldsymbol{x}_{+,k} + \delta\boldsymbol{x}_k\right) - \boldsymbol{x}_{-,k+1}^{i} + \boldsymbol{w}_k^{i}]\cdot \\
& & \quad [\boldsymbol{\Xi}_{k+1}^{j}\left(t_{k+1}; t_k, \boldsymbol{x}_{-,k+1}\right) - \boldsymbol{z}_{-,k+1}^{j} + \boldsymbol{u}_{k+1}^{j}]\} \\
& = & \displaystyle\sum_{1\le|\boldsymbol{\gamma}|\le n}\sum_{1\le|\tilde{\boldsymbol{\gamma}}|\le n} a_{\gamma_1\ldots\gamma_d}^{i} b_{\tilde{\gamma}_1\ldots\tilde{\gamma}_d}^{j} E\left\{\delta x_{k,1}^{\gamma_1+\tilde{\gamma}_1}\cdots\delta x_{k,d}^{\gamma_d+\tilde{\gamma}_d}\right\} \\
& & -\delta\boldsymbol{m}_{k+1}^{i}\delta\boldsymbol{n}_{k+1}^{j}\,,
\end{array}\right. \tag{5.3}$$

where $1 \le i \le d$, $1 \le j,\, j_1,\, j_2 \le m$, $\delta\boldsymbol{n}_{k+1}^{j} = \boldsymbol{z}_{k+1}^{j} - \boldsymbol{z}_{-,k+1}^{j}$.

When a new measurement $\boldsymbol{z}_{k+1}^{r}$ is obtained at time t_{k+1}, the measurement innovation vector $\boldsymbol{\nu}_{k+1}$ is defined as,

$$\boldsymbol{\nu}_{k+1} = \boldsymbol{z}_{k+1}^{r} - \boldsymbol{z}_{-,k+1}\,. \tag{5.4}$$

Furthermore, the innovation covariance matrix $\boldsymbol{P}_{\nu\nu,k+1}$ is further defined as

$$\boldsymbol{P}_{\nu\nu,k+1}^{j_1 j_2} = \boldsymbol{P}_{zz,k+1}^{j_1 j_2} + \boldsymbol{R}_{k+1}^{j_1 j_2}, \quad 1 \le j_1, j_2 \le m\,. \tag{5.5}$$

where $\boldsymbol{R}_{k+1}^{j_1 j_2} = E\{\boldsymbol{u}_{k+1}^{j_1}\boldsymbol{u}_{k+1}^{j_2}\}$ indicate the components of the measurement noise covariance matrix $\boldsymbol{R}_{k+1}$. Finally, the updated mean $\boldsymbol{x}_{+,k+1}$ and error covariance matrix $\boldsymbol{P}_{+,k+1}$ at time t_{k+1} can be obtained from

$$\left\{\begin{array}{rcl}
\boldsymbol{K}_{k+1} & = & \boldsymbol{P}_{xz,k+1}\left(\boldsymbol{P}_{vv,k+1}\right)^{-1}\,, \\
\boldsymbol{x}_{+,k+1} & = & \boldsymbol{x}_{-,k+1} + \boldsymbol{K}_{k+1}\boldsymbol{\nu}_{k+1}\,, \\
\boldsymbol{P}_{+,k+1} & = & \boldsymbol{P}_{-,k+1} - \boldsymbol{K}_{k+1}\boldsymbol{P}_{vv,k+1}\boldsymbol{K}_{k+1}^{T}\,,
\end{array}\right. \tag{5.6}$$

where $\boldsymbol{K}_{k+1}$ is the Kalman filter gain.

A brief summary of the JT-HEKF-n filtering algorithm is arranged in Algorithm 5.1.

Algorithm 5.1 JT-HEKF-n algorithm

1: Initialize the state $\boldsymbol{x}_{+,0}$, state covariance matrix $\boldsymbol{P}_{+,0}$, expansion order n, process and measurement noise covariance matrices $\boldsymbol{Q}$, $\boldsymbol{R}$, final time t_f, $t_k = 0$, $k = 0$;
2: k is increased by 1;
3: Use (5.2) to predict the a priori estimates of the state mean $\boldsymbol{x}_{-,k+1}$ and measurement mean $\boldsymbol{z}_{-,k+1}$, as well as the state error covariance matrix $\boldsymbol{P}_{-,k+1}$ at time t_{k+1};
4: Incorporate the new measurement $\boldsymbol{z}^r_{k+1}$ into the update step (5.3)-(5.6), thus obtain the a posterior estimates of the state mean $\boldsymbol{x}_{+,k+1}$ and state covariance matrix $\boldsymbol{P}_{+,k+1}$ at time t_{k+1};
5: Execute steps from 2 to 4 until $t_k = t_f$.

5.2 Fault-tolerant variants of the JT-HEKF-n filter

5.2.1 Fault measurement detection

The key point in the design of a fault-tolerant variant of the JT-HEKF-n algorithm is to timely detect false measurements and alleviate their pollution on the filtering performance. The comparison between the real and theoretical values of the innovation covariance matrix provides an intuitive and effective approach to identify faults. In principle, if a measurement fault occurs, the measurement estimation error must exceed the theoretical one, causing a mismatch.

Using the measurement innovation sequence $\{\boldsymbol{\nu}_{k+1}\}$ and the innovation covariance matrix $\boldsymbol{P}_{\nu\nu,k+1}$, the normalized innovations squared (NIS) is defined as [163]

$$q_{k+1} = \boldsymbol{\nu}^T_{k+1}\, \boldsymbol{P}^{-1}_{\nu\nu,k+1}\, \boldsymbol{\nu}_{k+1}\,, \tag{5.7}$$

thus, the time-average NIS can be further expressed as

$$\bar{q}_{k+1} = \frac{1}{L} \sum_{j=k-L+2}^{k+1} q_j\,, \tag{5.8}$$

where L is the width of a moving window, guaranteeing that the detection of the test statistics is sufficient [164]. The appropriate value of L can be taken between 1 and $k+1$. Note that L is herein set to 3 in order to save the computational cost.

If there are no measurement faults, the time-averaging NIS function has a χ^2 distribution with m degrees of freedom, m being the dimension of the measurement innovation vector.

The fault detection is done by means of a statistical analysis of the time-average NIS function $\bar{q}_{k+1}$. Two hypotheses are presented as follows,

- $\mathbb{H}_0$: the filter does not encounter a measurement fault.
- $\mathbb{H}_1$: the filter encounters a measurement fault.

For a given level of significance, i.e. $\alpha = 99\%$, the criterion is defined as

$$P(\bar{q}_{k+1} > \chi^2_{m,\alpha}) = \alpha\,, \tag{5.9}$$

where the threshold value $\chi^2_{m,\alpha}$ is given by the chi-square probability distribution function with the known values of m and α [163]. If $\bar{q}_{k+1} > \chi^2_{m,\alpha}$, the hypothesis $\mathbb{H}_1$ is accepted, proving that a measurement malfunction occurs in the JT-HEKF-n implementation, otherwise $\mathbb{H}_0$ holds true and the JT-HEKF-n is assumed to work well.

5.2.2 False measurement-discarding based JT-HEKF-n filter

In general, the usage of high-order JTTNM-n method enables to extract more nonlinear information from the dynamics and achieve a good state prediction accuracy over large integration time intervals, such that the performance of a high order JT-HEKF-n implementation is only slightly affected by the decrease of the measurement frequency. This appealing behavior will be detailedly discussed and tested in section 5.4. Note in particular that it is the rationale underlying the adaptive false measurement-discarding based JT-HEKF-n filter (JT-DHEKF-n). In this procedure, when the chi-square test (5.9) detects and identifies a false measurement at time t_k, the state updating step would be canceled. That is, the propagation time interval is enlarged, and the state vector at time $t = t_{k+1}$ is predicted instead of at time $t = t_k$. This is equivalent to reduce the measurement frequency of the JT-HEKF-n method. This strategy is elegant since it makes the JT-DHEKF-n method robust against measurement faults with a slight loss of the accuracy. The JT-DHEKF-n implementation is very close to the one of the standard JT-HEKF-n method. Their unique difference lies at the fault detection mechanism before the updating step. The detailed implementation is given in Algorithm 5.2.

5.2.3 Single and multiple scale factor based JT-HEKF-n filter

When false measurements last over a pretty long time interval, the high order JTTNM-n method employed in the JT-DHEKF-n implementation could fail to approximate the dynamical model. To reduce the dependence of the estimation accuracy on the duration of measurement faults, both a single scale factor and an adaptive scale matrix are successively introduced aiming to systematically correct Kalman gains. In what follows, these two new fault-tolerant filtering algorithms are respectively denoted by JT-SHEKF-n

Algorithm 5.2 JT-DHEKF-n algorithm

1: Initialize the state $\boldsymbol{x}_{+,0}$, state covariance matrix $\boldsymbol{P}_{+,0}$, expansion order n, process and measurement noise covariance matrices $\boldsymbol{Q}$, $\boldsymbol{R}$, final time t_f, $t_k = 0$, $k = 0$;
2: k is increased by 1;
3: Use (5.2) to predict the a priori estimates of the state mean $\boldsymbol{x}_{-,k+1}$ and measurement mean $\boldsymbol{z}_{-,k+1}$, as well as the state error covariance matrix $\boldsymbol{P}_{-,k+1}$ at time t_{k+1};
4: Acquire a new measurement $\boldsymbol{z}^r_{k+1}$ at time t_{k+1}, calculate the measurement innovation $\boldsymbol{\nu}_{k+1}$ using (5.4), as well as the innovation covariance matrix $\boldsymbol{P}_{\nu\nu,k+1}$ via (5.3) and (5.5);
5: Compute the time-averaging NIS function using (5.8) and, for a given level of significance, perform the test hypothesis (5.9). If $\mathbb{H}_0$ is true, jump to step 6; otherwise, cancel the *updating* process and jump to step 2;
6: Incorporate the new measurement $\boldsymbol{z}^r_{k+1}$ into the update step (5.3)-(5.6), thus obtain the a posterior estimates of the state mean $\boldsymbol{x}_{+,k+1}$ and state covariance matrix $\boldsymbol{P}_{+,k+1}$ at time t_{k+1};
7: Execute steps from 2 to 6 until $t_k = t_f$.

(single scale factor based JT-HEKF-n) and JT-MHEKF-n (multiple scale factors based JT-HEKF-n).

In absence of measurement faults, the true innovation covariance matrix $\boldsymbol{\nu}_{k+1}\boldsymbol{\nu}^T_{k+1}$ underestimates the one computed in (5.5), i.e., $\boldsymbol{P}_{\nu\nu,k+1}$, thus

$$\boldsymbol{\nu}_{k+1}\boldsymbol{\nu}^T_{k+1} \leq \boldsymbol{P}_{\nu\nu,k+1}\,, \tag{5.10}$$

and in presence of measurement malfunctions, $\boldsymbol{\nu}_{k+1}\boldsymbol{\nu}^T_{k+1}$ mismatches $\boldsymbol{P}_{\nu\nu,k+1}$, that is, $\boldsymbol{\nu}_{k+1}\boldsymbol{\nu}^T_{k+1} > \boldsymbol{P}_{\nu\nu,k+1}$.

To assure the robustness of the JT-SHEKF-n in the case of measurement malfunctions, a single scale factor λ_{k+1} should be introduced as a corrective term into the innovation covariance matrix,

$$\boldsymbol{P}_{\nu\nu,k+1} = \boldsymbol{P}_{zz,k+1} + \lambda_{k+1}\boldsymbol{R}_{k+1}\,. \tag{5.11}$$

The single scale factor is determined in terms of the condition (5.10), requiring that

$$\boldsymbol{\nu}_{k+1}\boldsymbol{\nu}^T_{k+1} \leq \boldsymbol{P}_{zz,k+1} + \lambda_{k+1}\boldsymbol{R}_{k+1}\,, \tag{5.12}$$

so, it is given by

$$\lambda_{k+1} = \max_{i=1,\cdots,m} \frac{\left[\boldsymbol{\nu}_{k+1}\boldsymbol{\nu}^T_{k+1} - \boldsymbol{P}_{zz,k+1}\right]_i}{\left[\boldsymbol{R}_{k+1}\right]_i}\,, \tag{5.13}$$

where $[A]_i$ denotes the i-th diagonal element of the matrix A. Note that, in this case, the Kalman gain becomes

$$\boldsymbol{K}_{k+1} = \boldsymbol{P}_{xz,k+1}\left[\boldsymbol{P}_{zz,k+1} + \lambda_{k+1}\boldsymbol{R}_{k+1}\right]^{-1}\,. \tag{5.14}$$

If there are no malfunctions in the measurement system, the single scale factor is very close to one, since the real and theoretical innovation covariance matrices already fulfill the condition (5.10). However, in case of measurement faults, according to (5.13) and (5.14), measurement faults produce an increase of the scale factor and a decrease of the Kalman gain.

In the above procedure, the single scale factor not only performs the correction of the most mismatched term in the real and computed innovation covariance matrices $\boldsymbol{P}_{\nu\nu,k+1}$, but also modifies the remaining terms. However, not all the terms require to be corrected since it is very likely that some of the sensors will work well even that some others fail. Note that the redundant correction of the proper terms in the innovation covariance matrix also leads to the accuracy loss. To avoid this defect, it is better to introduce a suitable diagonal matrix $\boldsymbol{G}_{k+1}$, instead of a single scale factor λ_{k+1}, as a multiplier factor of the measurement noise covariance matrix $\boldsymbol{R}_{k+1}$. The measurement noise scale matrix $\boldsymbol{G}_{k+1}$ is determined requiring that

$$\frac{1}{L}\sum_{i=k-L+2}^{k+1}\boldsymbol{\nu}_i\boldsymbol{\nu}_i^T = \boldsymbol{P}_{zz,k+1} + \boldsymbol{G}_{k+1}\boldsymbol{R}_{k+1}\,, \tag{5.15}$$

where L is the width of the phase moving window, which usually takes values between 2 and 5. $L = 3$ is set by default to smooth the filtering process. From the above requirement, it follows that

$$\boldsymbol{G}_{k+1} = \left[\frac{1}{L}\sum_{i=k-L+2}^{k+1}\boldsymbol{\nu}_i\boldsymbol{\nu}_i^T - \boldsymbol{P}_{zz,k+1}\right]\boldsymbol{R}_{k+1}^{-1}\,. \tag{5.16}$$

For a given sensor experiencing measurement malfunctions, its accuracy may be worse than its design performance, but it is not reasonable to assume that it would be better. Therefore, in the case of certain measurement malfunctions, the corresponding terms in $\boldsymbol{G}_{k+1}$ become larger than one, while in the normal operation case of the measurement system, the diagonal components of $\boldsymbol{G}_{k+1}$ should be one. However, due to the approximation and rounding errors in the computation of (5.16), it is possible to cause the measurement noise scale matrix $\boldsymbol{G}_{k+1}$ to be off-diagonal, or its diagonal components to be less than one. To avoid these issues, a new measurement noise scale matrix is defined as

$$\boldsymbol{G}_{k+1}^{*} = \mathrm{diag}\left\{\max\left(1, G_{k+1,11}\right), \max\left(1, G_{k+1,22}\right), \cdots, \max\left(1, G_{k+1,mm}\right)\right\}. \tag{5.17}$$

Besides, in the case of measurement faults, the measurement noise scale matrix $\boldsymbol{G}_{k+1}^{*}$ must be further used to correct the original Kalman gain, i.e.,

$$\boldsymbol{K}_{k+1} = \boldsymbol{P}_{xz,k+1}\left[\boldsymbol{P}_{zz,k+1} + \boldsymbol{G}_{k+1}^{*}\boldsymbol{R}_{k+1}\right]^{-1}\,. \tag{5.18}$$

Note that the scale matrix $\boldsymbol{G}_{k+1}^{*}$ amplifies the variance of the measurement noise in presence of measurement faults in some sensors. This leads to a smaller Kalman gain related to the corresponding sensors. This operation effectively reduces the weights of

false measurements and avoids their pollution in the state estimation , further ensuring the robustness of the JT-MHEKF-n against measurement malfunctions.

The detailed JT-SHEKF-n and JT-MHEKF-n implementations are in Algorithm 5.3.

Algorithm 5.3 JT-SHEKF-n and JT-MHEKF-n algorithms

1: Initialize the state $\boldsymbol{x}_{+,0}$, state covariance matrix $\boldsymbol{P}_{+,0}$, expansion order n, process and measurement noise covariance matrices $\boldsymbol{Q}$, $\boldsymbol{R}$, final time t_f, $t_k = 0$, $k = 0$;
2: k is increased by 1;
3: Use (5.2) to predict the a priori estimates of the state mean $\boldsymbol{x}_{-,k+1}$ and measurement mean $\boldsymbol{z}_{-,k+1}$, as well as the state error covariance matrix $\boldsymbol{P}_{-,k+1}$ at time t_{k+1};
4: Acquire a new measurement $\boldsymbol{z}^r_{k+1}$ at time t_{k+1}, calculate the measurement innovation $\boldsymbol{\nu}_{k+1}$ using (5.4), as well as the innovation covariance matrix $\boldsymbol{P}_{\nu\nu,k+1}$ via (5.3) and (5.5);
5: Compute the time-averaging NIS function using (5.8) and, for a given level of significance, perform the hypothesis test (5.9). If $\mathbb{H}_0$ is true, keep the Kalman gain in (5.6) unchanged; otherwise, compute the single scale factor or the adaptive scale matrix via (5.13) or (5.16). Then, replace the original Kalman gain in (5.6) with the corrected one calculated by (5.14) or (5.18);
6: Incorporate the new measurement $\boldsymbol{z}^r_{k+1}$ into the update step (5.3)-(5.6), thus obtain the a posterior estimates of the state mean $\boldsymbol{x}_{+,k+1}$ and state covariance matrix $\boldsymbol{P}_{+,k+1}$ at time t_{k+1};
7: Execute steps from 2 to 6 until $t_k = t_f$.

5.3 Model description

5.3.1 Equations of motion

To improve the autonomy of the spacecraft OD problem, the newly proposed nonlinear filters employ the space-based inter-satellite relative measurement information. In such cases, the orbit of the beacon is accurately described by its ephemeris, while the uncertain orbit of the estimated satellite is the one to be determined. The dynamical model adopted in this chapter must be convenient to address orbits with different altitudes, thus the one formulated in the Cartesian coordinates has been adopted (see in subsection 3.1.1). Besides, the dominant perturbations considered contain SRP, Earth's non-spherical gravity, and luni-solar gravitational attraction, whose formulations are shown in subsection 3.2.1.

5.3.2 Measurement model

Conventional space-based autonomous OD methods often acquire the measurement information from some nature beacons using certain special sensors, such as X-ray sensors [165], magnetometers [166], gravity gradiometers [167], and thermal sensors [168]. Although it significantly lessens the dependence on ground facilities, a worse OD accuracy is often obtained due to the inferior measurement performance of these special sensors, and, generally, it has been considered as an aided approach. However, certain highly valuable space missions require a much more reliable and autonomous sensing method, that is, a stronger anti-interference signal ability, and a less dependency on ground facilities. A novel array signal detection technique has been proposed to determine the angles-of-arrival of signal propagation paths in the body-fixed frame attached to the signal receiving satellite [169, 170]. A space-based optical (SBO) sensor is another alternative to produce similar orientation information relative to beacon satellites [171–173]. In what follows we avoid the discussion about the detection mechanisms of antenna arrays or SBO sensors, nowadays being a hot research field, and assume that the relative angle information is being acquired directly via the inter-satellite relative tracking framework.

Assuming that the relative right ascension α and declination β between two satellites can be measured, thus the measurement model is

$$\begin{cases} \alpha &= \arctan \dfrac{y_b - y}{x_b - x}, \\ \beta &= \arcsin \dfrac{z_b - z}{\sqrt{(x_b - x)^2 + (y_b - y)^2 + (z_b - z)^2}}, \end{cases} \tag{5.19}$$

where the position vector $\boldsymbol{r}_b(t) = [x_b, y_b, z_b]^T$ is determined using the accurate ephemeris of the beacon satellite at time t, and the position vector of the observed satellite $\boldsymbol{r}(t) = [x, y, z]^T$ is predicted through the forward polynomial integration of (3.2).

The measurement accuracy is a key factor of the OD procedure. According to ESA's "Assessment Study for Space Based Space Surveillance Demonstration System", the general random observational angular noise of spacecraft-based sensors is of the order of 0.64 arc-sec [171]. Besides, the root mean square residual of typical angular observations in Gaia mission is about 0.2-0.3 milli-arc-sec. Therefore, observational noises of both 0.64 arc-sec and 0.2 arc-sec in angular measurement are considered in the following simulations since they may be achieved in current or near future space missions.

5.4 Numerical simulations

The proposed standard JT-HEKF-n, together with three fault-tolerant algorithms, i.e., JT-DHEKF-n, JT-SHEKF-n, and JT-MHEKF-n, will be tested in autonomous GEO OD problems. To improve the autonomy of the OD procedure, an inter-satellite relative angular measurement procedure is considered to provide the relative right ascension α and

declination β. Without loss of generality, two observation configurations are employed. The feasibility and effectiveness of the standard JT-HEKF-n, implemented at different expansion orders n, are firstly illustrated by various numerical simulations using different magnitudes of initial state deviations, measurement noises and measurement acquisition periods. Secondly, three scenarios, with different types of measurement faults, have been considered to test the robustness of the introduced fault-tolerant variants. It is worth to mention that some metrics proposed in subsection 4.2.5, such as RMSE, SD of RMSEs and CPU time, are again employed to evaluate the performance of the proposed standard JT-HEKF-n filter and its three fault-tolerant variants.

5.4.1 Autonomous nonlinear orbit determination

In this case study, the effects of filter orders, initial state deviations, measurement noises, and measurement acquisition frequencies on the performance of the standard JT-HEKF-n are explored. The GEO satellite SES-6 (TLE number 39172) is used as the beacon, while the GEO satellite Hispasat-1D (TLE number 27528) is regarded as the estimated satellite, as is shown schematically in Fig. 5.1. The orbit of the beacon is accurately described by its ephemeris, and the orbit of the estimated satellite is the one that must be determined. The accurate initial state vectors and area-to-mass ratios of two satellites are given in Table 5.1. To observe the beacon with the usage of the sensors installed on the estimated satellite, it requires that the angle formed by connecting the Sun, beacon and estimated satellites must be less than 80° [174]. The visible tracklets of SES-6 observed by Hispasat-1D are displayed in Fig. 5.2, which shows that a continuous visible tracklet of 12 hours per day is available. In the simulations, the measurement noise is described as a zero mean normal distribution with a standard deviation $\sigma_m = 0.64$ arc-sec [171], thus the measurement noise covariance matrix is set as $\boldsymbol{R} = \mathrm{diag}[\sigma_m^2, \sigma_m^2]$.

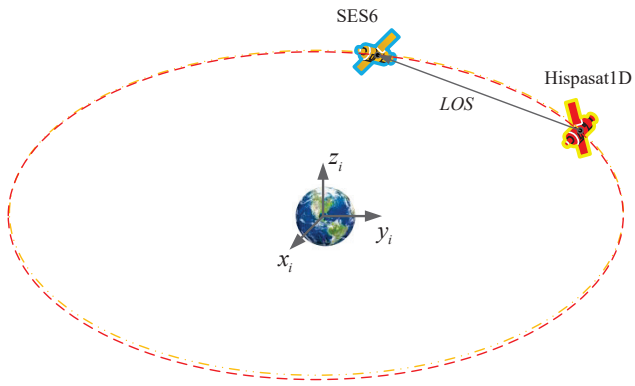


Figure 5.1: Schematic representation of the GEO-GEO observation configuration.

The first simulation is employed to analyze the effect of the adopted order n on the performance of the standard JT-HEKF-n for achieving the autonomous orbit determination.

Table 5.1: State and area-to-mass ratio at initial epoch July 4th 2019, 03:20:00.000 UTC.

TLE	x_0 (km)	y_0 (km)	z_0 (km)	v_{x0} (m/s)	v_{y0} (m/s)	v_{z0} (m/s)	η (m^2/kg)
39172	15631.6	−39276.4	−11.6	2862.9	1119.6	0.86	0.02000
27528	22170.3	−35886.3	−93.1	2614.6	1615.4	−12.5	0.02242

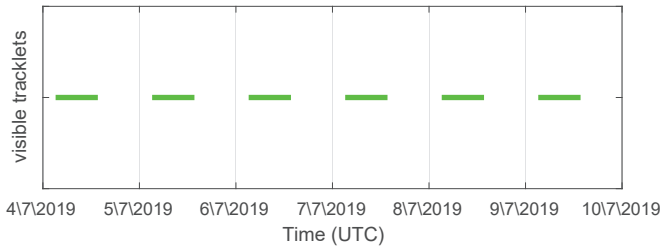


Figure 5.2: Visible tracklets of the GEO-GEO observation configuration.

In this case study, the initial state error $\delta\boldsymbol{x}_{+,0}$ has been taken equal to 100 km for all position components, and 5 m/s for all velocity components. The initial state covariance matrix is set as $\boldsymbol{P}_{+,0} = \delta\boldsymbol{x}_{+,0} \cdot \delta\boldsymbol{x}_{+,0}^T$. The process noise can be represented by a zero mean Gaussian distribution with a standard deviation 5×10^{-10}m/s^2. The total simulation time is set as 6 days, and the measurement acquisition period is set as one hour within visible tracklets. Using all these parameters, the JT-HEKF-n is implemented at expansion orders $n = 1, 2, 3$. Figure 5.3 exhibits the profiles of estimation errors in position and velocity, showing that JT-HEKF-2 and JT-HEKF-3 algorithms enable to provide an almost same estimation accuracy and achieve one order of magnitude accuracy gain relative to the JT-HEKF-1. Table 5.2 reveals the RMSEs in position and velocity over the time interval $t \in [96, 144]$ h, computed by the JT-HEKF-n at order $n = 1, 2, 3$. To be specific, at the filtering steady stage $t \in [96, 144]$ h, JT-HEKF-2 and JT-HEKF-3 algorithms enable to obtain the RMSEs of about 21 m in position and 1.4×10^{-3} m/s in velocity, while the JT-HEKF-1 produces worse results, i.e., the RMSEs are approximately of 735 m in position and 0.054 m/s in position. Again, Table 5.2 reveals that one order of magnitude accuracy gain is obtained by adjusting the order from 1 to 2, but no significant accuracy improves by switching the order from 2 to 3. One key point should be underlined is that the convergence rate of both JT-HEKF-2 and JT-HEKF-3 is somewhat slow since only one measurement per hour is available (a pretty low measurement frequency), as shown in Fig. 5.3. Even so, compared to the divergent result obtained by the JT-HEKF-1 (i.e., EKF), it shows that both JT-HEKF-2 and JT-HEKF-3 perform very well since they enable to achieve good orbit determination under these poor conditions. To verify the reliability of this last conclusion, a MC simulation considering 2000 runs of JT-HEKF-1 and JT-HEKF-2 is carried out and the results of 25 runs with largest estimation errors are given in Fig. 5.4, from which it follows that the JT-HEKF-2 clearly again outperforms the JT-HEKF-1 since it possesses a smaller error dispersion.

The estimation failure of the JT-HEKF-1 can be analyzed by a filter consistency test.

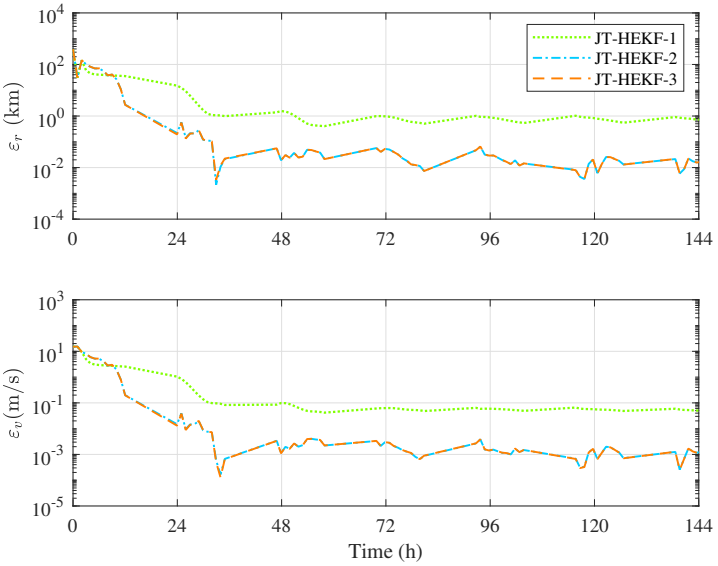


Figure 5.3: Profiles of estimation errors in position and velocity with measurement acquisition period 1 hour and measurement noise 0.64 arc-sec in the GEO-GEO observation configuration.

Table 5.2: RMSEs in position and velocity of the JT-HEKF-n at the steady stage $t \in [96, 144]$ h and the associated CPU time.

	JT-HEKF-1	JT-HEKF-2	JT-HEKF-3
$\bar{\varepsilon}_r$ (km)	0.7350	0.0210	0.0210
$\bar{\varepsilon}_v$ (m/s)	0.0538	0.0014	0.0014
CPU time (s)	21.69	31.75	125.98

In principle, if 95% of the innovations $\boldsymbol{\nu}_k$ are bounded by $\pm 2\sqrt{\boldsymbol{P}_{vv,k}}$, one can assume that the filter is consistent and performs correctly, otherwise the filter is inconsistent and does not. Figure 5.5 shows that more innovations than expected (i.e., $> 5\%$) fall outside the $\pm 2\sqrt{\boldsymbol{P}_{vv,k}}$ bound in the JT-HEKF-1 implementation, revealing its divergence in the OD application. On the contrary, both JT-HEKF-2 and JT-HEKF-3 filters work much better, since more than 95% of the innovation values lie within the $\pm 2\sqrt{\boldsymbol{P}_{vv,k}}$ bound. The last line of Table 5.2 gives the CPU time consumed in the JT-HEKF-n implementation at orders $n = 1, 2, 3$. Clearly, the computation time increases when the expansion order also does. Therefore, from the perspective of the computation time and estimation accuracy, the JT-HEKF-2 filter seems to be the best option.

To further analyze the sensitivity of the filtering performance with respect to initial state deviations, both JT-HEKF-1 and JT-HEKF-2 filters are further employed to estimate the GEO trajectory whose initial state vectors are taken from four Gaussian error distribution

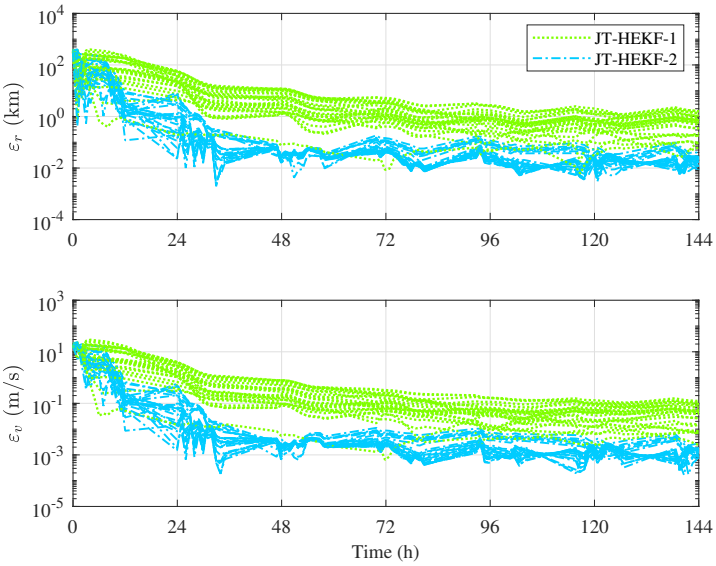


Figure 5.4: Accuracy profiles of 25 MC runs with the worst estimation errors obtained by JT-HEKF-1 and JT-HEKF-2 implementations. Measurement acquisition period 1 hour and measurement noise 0.64 arc-sec.

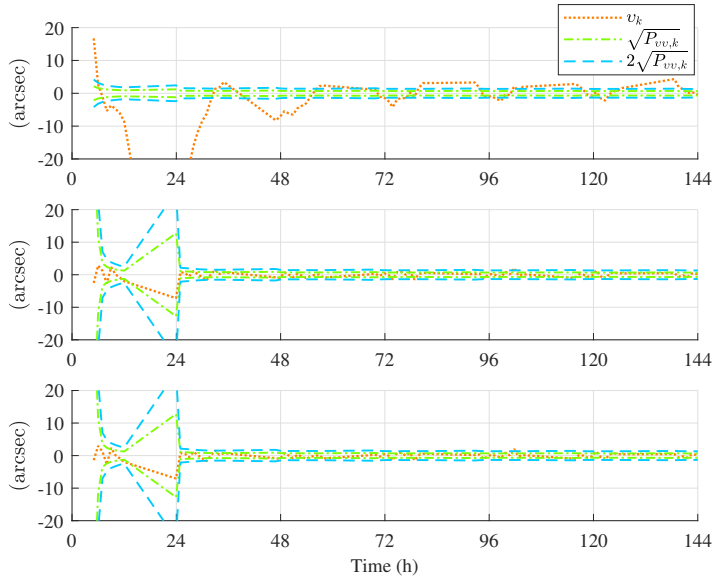


Figure 5.5: Innovation and innovation standard deviation bounds of the JT-HEKF-n implemented at orders $n = 1$ (top), 2 (middle), and 3 (bottom).

with different standard deviations. For each case, 2000 random samples are generated around the true initial state in terms of the specified initial state error distribution. Then, 25 samples with largest estimation errors are chosen, assuring this way that the worst circumstances, where the nonlinearity plays a prominent role, are investigated. Table 5.3 shows the position and velocity RMSEs of both JT-HEKF-1 and JT-HEKF-2 at the steady state (i.e., $t \in [96, 144]$ h). Clearly, in all study cases, the JT-HEKF-2 achieves a better accuracy than the JT-HEKF-1. Besides, the RMSEs in position and velocity of the JT-HEKF-1 gradually increase when the initial state standard deviation also does, while the JT-HEKF-2 enables to maintain its accuracy even if the initial state standard deviation remarkably increases. Note in particular that the JT-HEKF-1 diverges when the initial state standard deviation are too large (i.e., $\hat{\sigma}_r = 2000$ km, $\hat{\sigma}_v = 50$ m/s). However, quite good RMSEs in position and velocity can be still obtained with the usage of a JT-HEKF-2 filter. The tiny ratio of RMSEs computed by both JT-HEKF-1 and JT-HEKF-2 further shows that approximately one order of magnitude accuracy gain can be obtained in these cases. On the other hand, Table 5.3 also gives the ratio of SDs of the RMSEs computed by both JT-HEKF-1 and JT-HEKF-2 filters, which reveals that a smaller error dispersion can be obtained by a JT-HEKF-2 filter.

Table 5.3: Sensitivity analysis of the JT-HEKF-n with respect to initial state deviations. ($\hat{\sigma}_r$ km, $\hat{\sigma}_v$ m/s) indicates the standard deviation of the Gaussian distribution describing the initial spacecraft state vector. The unit of ${}_n\bar{\varepsilon}_\Upsilon$ and ${}_n\sigma_\Upsilon$ are kilometers and meters per second. The - symbol indicates filtering divergence.

	Metric	($\hat{\sigma}_r$ km, $\hat{\sigma}_v$ m/s)			
		(50, 2)	(100, 5)	(200, 10)	(2000, 50)
JT-HEKF-1	${}_1\bar{\varepsilon}_r$	0.1269	0.7270	4.3060	-
	${}_1\bar{\varepsilon}_v$	0.0091	0.0510	0.2871	-
	${}_1\sigma_r$	0.1043	0.5868	3.4243	-
	${}_1\sigma_v$	0.1592	0.9062	5.3463	-
JT-HEKF-2	${}_2\bar{\varepsilon}_r$	0.0245	0.0297	0.0290	0.0370
	${}_2\bar{\varepsilon}_v$	0.0017	0.0020	0.0020	0.0025
	${}_2\sigma_r$	0.0138	0.0194	0.0152	0.0244
	${}_2\sigma_v$	0.0275	0.0347	0.0318	0.0433
Ratio	${}^2_1\tau_r$	0.1931	0.0409	0.0067	-
	${}^2_1\tau_v$	0.1868	0.0392	0.0069	-
	${}^2_1\zeta_r$	0.1323	0.0331	0.0044	-
	${}^2_1\zeta_v$	0.1727	0.0383	0.0059	-

Another key point is the sensitivity analysis of the JT-HEKF-n with respect to the measurement noise and acquisition period. Five different measurement acquisition periods (10, 20, 60, 120, 180 minutes), and two levels of measurement noises (0.2 and 0.64 arcsec) have been considered in the simulations. As in the preceding MC simulation, 2000 samples are generated around the true initial state, with $\hat{\sigma}_r = 100$ km and $\hat{\sigma}_v = 5$ m/s. Table 5.4 gives the position and velocity RMSEs computed by both JT-HEKF-1 and JT-HEKF-2 filters, as well as the associated ratios at the steady state (i.e., $t \in [96, 144]$ h). From which it follows that the position and velocity estimation errors of the JT-HEKF-

1 increase when the measurement acquisition period also does. A larger measurement acquisition period produces a worse first-order state prediction in the JT-HEKF-1, thus deteriorates its estimation accuracy. In contrast, the nonlinear state prediction used in the JT-HEKF-2 is accurate enough and almost not affected by the choice of measurement acquisition periods (less than 180 minutes), such that the JT-HEKF-2 keeps good estimation properties even in the case of using large measurement acquisition periods. Apparently, the JT-HEKF-2 is more robust relative to the measurement acquisition period than the JT-HEKF-1. In particular, it provides the theoretical basis for the design of the fault-tolerant filter introduced in section 4.1, which implies a decrease of the measurement frequency (i.e., discarding the fault measurements) when it is used.

Table 5.4: Position and velocity RMSEs in the sensitivity analysis of the JT-HEKF-n with respect to the measurement noise and acquisition period. The units of ${}_n\bar{\varepsilon}_r$ and ${}_n\bar{\varepsilon}_v$ are kilometers and meters per second.

		Metric	Measurement acquisition period (min)				
			10	20	60	120	180
Measurement noise (arc-sec)	0.2	${}_1\bar{\varepsilon}_r$	0.2816	0.4935	0.7206	1.4538	2.8543
		${}_1\bar{\varepsilon}_v$	0.0205	0.0361	0.0505	0.1046	0.2012
		${}_2\bar{\varepsilon}_r$	0.0080	0.0118	0.0085	0.0084	0.0122
		${}_2\bar{\varepsilon}_v$	0.0005	0.0008	0.0006	0.0006	0.0009
		${}^2_1\tau_r$	0.0284	0.0239	0.0118	0.0057	0.0042
		${}^2_1\tau_v$	0.0244	0.0222	0.0119	0.0057	0.0044
	0.64	${}_1\bar{\varepsilon}_r$	0.2986	0.5014	0.7270	1.4549	2.8564
		${}_1\bar{\varepsilon}_v$	0.0220	0.0367	0.0510	0.1047	0.2013
		${}_2\bar{\varepsilon}_r$	0.0268	0.0369	0.0297	0.0261	0.0416
		${}_2\bar{\varepsilon}_v$	0.0018	0.0026	0.0020	0.0018	0.0032
		${}^2_1\tau_r$	0.0898	0.0736	0.0409	0.0179	0.0145
		${}^2_1\tau_v$	0.0818	0.0708	0.0392	0.0171	0.0158

Table 5.4 also shows that there is no performance improvement in the JT-HEKF-1 implementation when decreasing the measurement noise from 0.64 to 0.2 arc-sec. This result, in principle not expected, is due to the poor state prediction provided by the JT-HEKF-1 in the case of a large initial state error. In contrast, the nonlinear prediction employed in the JT-HEKF-2 is more accurate, such that the estimation accuracy can be improved by decreasing the measurement noise. This fact shows that only when the state prediction is accurate enough, the measurement accuracy improvement has a significant influence on the performance of the JT-HEKF-n algorithm.

Table 5.5 gives the results for the SDs of the position and velocity RMSEs, and the associated ratios at the filter steady stage $t \in [96, 144]$ h considering measurement noises of 0.2 and 0.64 arc-sec. The tiny ratios reveal that a smaller error dispersion is obtained by the JT-HEKF-2 when compared with the JT-HEKF-1. A lower measurement noise produces a smaller estimation error dispersion if the state prediction is accurate enough, for instance when the JT-HEKF-2 is used. Furthermore, the estimation error dispersion of the JT-HEKF-2 remains fixed when the measurement acquisition period enlarges.

Table 5.5: SDs of the position and velocity RMSEs in the sensitivity analysis of the JT-HEKF-n with respect to the measurement noise and acquisition period. The units of ${}_n\sigma_r$ and ${}_n\sigma_v$ are kilometers and meters per second.

		Metric	Measurement acquisition period (min)				
			10	20	60	120	180
Measurement noise (arc-sec)	0.2	${}_1\sigma_r$	0.2749	0.4789	0.5861	1.4185	2.6339
		${}_1\sigma_v$	0.3829	0.6688	0.9011	1.9810	3.7844
		${}_2\sigma_r$	0.0018	0.0011	0.0046	0.0030	0.0009
		${}_2\sigma_v$	0.0079	0.0112	0.0094	0.0087	0.0115
		${}^2_1\zeta_r$	0.0065	0.0023	0.0078	0.0021	0.0003
		${}^2_1\zeta_v$	0.0206	0.0167	0.0104	0.0043	0.0030
	0.64	${}_1\sigma_r$	0.2879	0.4818	0.5868	1.4217	2.6359
		${}_1\sigma_v$	0.4033	0.6762	0.9061	1.9842	3.7871
		${}_2\sigma_r$	0.0063	0.0066	0.0194	0.0075	0.0062
		${}_2\sigma_v$	0.0264	0.0357	0.0347	0.0261	0.0340
		${}^2_1\zeta_r$	0.0218	0.0137	0.0331	0.0052	0.0023
		${}^2_1\zeta_v$	0.0655	0.0523	0.0383	0.0131	0.0089

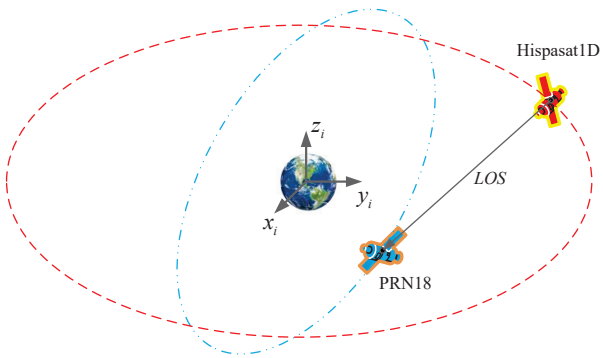


Figure 5.6: Schematic representation of the MEO-GEO observation configuration.

Note that similar results can be obtained when other measurement performance and orbit determination configurations are employed. Figure 5.6 illustrates an observation configuration where a medium earth orbit (MEO) satellite, named as PRN18 (TLE number 22877) is chosen as the beacon, while the same GEO satellite Hispasat-1D (TLE number 27528) is regarded as the estimated satellite. Table 5.6 shows the initial state vectors and area-to-mass ratios. The visible tracklets are displayed in Fig. 5.7, which shows that a continuous tracklet of 14 hours per day is available. The simulations consider a time interval of 6 days with a measurement acquisition period of 1 hour. The measurement noise is described as a zero mean normal distribution with a standard deviation $\sigma_m = 0.2$ arc-sec. Figure 5.8 shows that the estimation errors computed by the JT-HEKF-2 and JT-HEKF-3 almost coincide, of about 20 m in position and 0.001 m/s in velocity. On the contrary, the

JT-HEKF-1 produces much worse estimation errors in both position and velocity: 100 m and 0.01 m/s. Again, as in the GEO-GEO configuration and due to the same arguments, the JT-HEKF-2 is also preferred to solve the autonomous orbit determination.

Table 5.6: State and area-to-mass ratio at initial epoch July 3rd 2019, 07:20:00.000 UTC.

TLE	x_0 (km)	y_0 (km)	z_0 (km)	v_{x0} (m/s)	v_{y0} (m/s)	v_{z0} (m/s)	η (m^2/kg)
22877	−7475.1	−25354.9	−3691.9	2264.2	−273.9	−3108.0	0.02000
27528	22170.3	−35886.3	−93.1	2614.6	1615.4	−12.5	0.02242

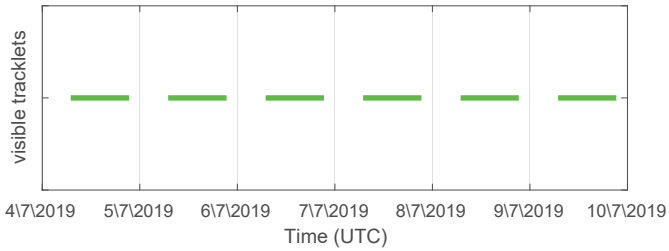


Figure 5.7: Visible tracklets of the MEO-GEO observation configuration.

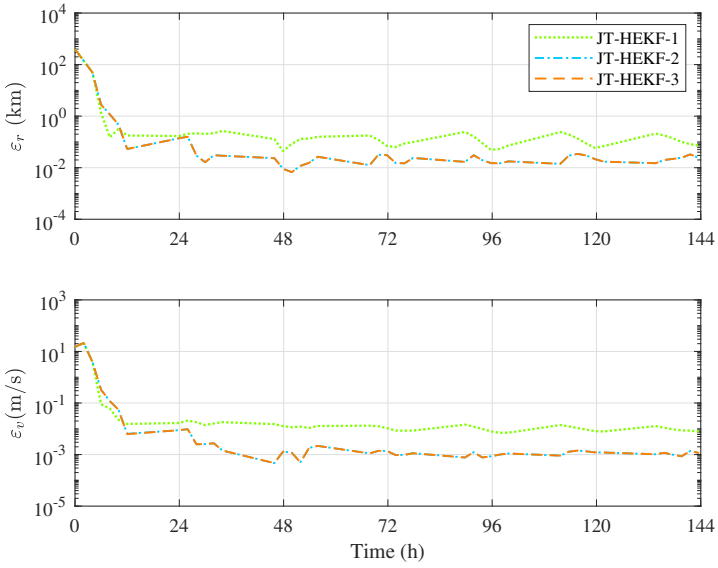


Figure 5.8: Profiles of estimation errors in position and velocity with measurement acquisition period of 1 hour and measurement noise 0.2 arc-sec in the MEO-GEO observing configuration.

5.4.2 Autonomous fault-tolerant orbit determination

To assess the performances of three fault-tolerant filters, different types of measurement faults have been considered, including measurement faults with a continuous constant bias, a random bias and an instantaneous zero output. The remaining simulation parameters are identical to the ones employed in subsection 5.4.1 for the GEO-GEO configuration. In particular, $\chi^2_{m,\alpha} = 9.21$ is taken in the fault detection procedure, corresponding to degrees of freedom $m = 2$ and the reliability level $\alpha = 99\%$ [163]. For comparison purposes, same simulations also have been done using the standard JT-HEKF-n. Finally, MC simulations based on repetitive implementations of the standard JT-HEKF-n and its three fault-tolerant variants are carried out to assess the sensitivity with respect to initial state deviations.

5.4.2.1 Measurements with a continuous constant bias

In this study case, the behaviors of the standard JT-HEKF-2 and its three fault-tolerant variants are investigated when a constant measurement bias is added to the declination. The simulation is performed during a time interval of six days (144 hours). The measurement bias of 8×10^{-5} rad (almost twenty times the nominal measurement noise) is active inside the time interval $t \in [70.8, 76.8]$ h, that is, the fault lasts for 6 hours. In this way the measurement model in this time interval becomes

$$\begin{cases} \alpha &= \arctan \dfrac{\bar{y}_b - \bar{y}}{\bar{x}_b - \bar{x}} + u_1, \\ \beta &= \arcsin \dfrac{\bar{z}_b - \bar{z}}{\sqrt{(\bar{x}_b - \bar{x})^2 + (\bar{y}_b - \bar{y})^2 + (\bar{z}_b - \bar{z})^2}} + u_2 + 8 \times 10^{-5}. \end{cases} \tag{5.20}$$

The position and velocity estimation errors for the JT-HEKF-2, JT-DHEKF-2, JT-SHEKF-2 and JT-MHEKF-2 are displayed in Fig. 5.9. Obviously, the results obtained by a standard JT-HEKF-2 are unreliable in the case of a continuous constant bias fault inside the time interval $t \in [70.8, 76.8]$ h due to the filter inconsistency. In fact, the detractive influence of measurement faults on the standard JT-HEKF-2 is not only important over the period in which the faults happened, but also lasts for a much longer time span. In contrast, the fault-tolerant JT-DHEKF-2, JT-SHEKF-2 and JT-MHEKF-2 enable to produce an excellent estimation accuracy without being affected from continuous constant measurement faults. Therefore, it can be concluded that the JT-DHEKF-2, JT-SHEKF-2 and JT-MHEKF-2 outperform the JT-HEKF-2 when the measurement is polluted by a continuous constant bias.

From the perspective of quantitative analysis, the position and velocity RMSEs computed by four newly proposed filters at steady state $t \in [96, 144]$ h are given in Table 5.7. The results show that the JT-DHEKF-2, JT-SHEKF-2 and JT-MHEKF-2 perform better than the JT-HEKF-2 and stand robust against constant measurement faults. More concretely, when the measurements are false, the JT-DHEKF-2 completely guards against

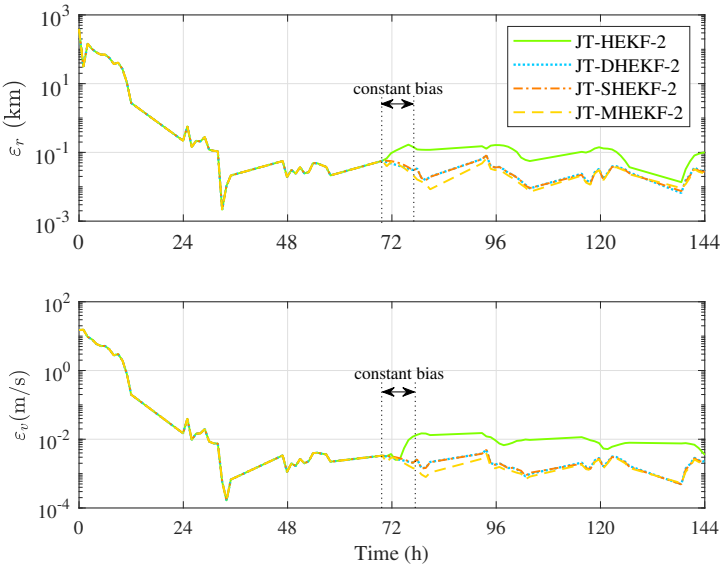


Figure 5.9: Profiles of estimation errors in position and velocity computed by the JT-HEKF-2, JT-DHEKF-2, JT-SHEKF-2 and JT-MHEKF-2 in the case of a constant measurement bias inside the time interval $t \in [70.8, 76.8]$ h.

Table 5.7: RMSEs in position and velocity of the JT-HEKF-2, JT-DHEKF-2, JT-SHEKF-2, and JT-MHEKF-2 at the steady stage $t \in [96, 144]$ h in the case of a constant measurement bias and the associated CPU time.

RMSE	JT-HEKF-2	JT-DHEKF-2	JT-SHEKF-2	JT-MHEKF-2
$\bar{\varepsilon}_r$ (km)	0.0915	0.0283	0.0294	0.0256
$\bar{\varepsilon}_v$ (m/s)	0.0069	0.0021	0.0021	0.0018
CPU time (s)	36.30	36.43	36.51	36.53

the deteriorative effects of constant measurement faults, since it discards them during the associated state prediction steps. The JT-SHEKF-2 decreases the Kalman filter gain via the increment of the single scale factor and, in this way, suppresses the pollution from measurement faults. The JT-MHEKF-2 stands robust, and keeps a better estimation result for the whole process by independently increasing the related diagonal elements of the measurement noise scale matrix. In principle, an increment of the related diagonal element in the measurement noise scale matrix results in a decrement of the related element in the Kalman gain, so as to isolate and correct the negative effect of the innovation sequences stemming from measurement faults. Compared with the usage of the single scale factor used in the JT-SHEKF-2, the measurement noise scale matrix generally makes the JT-MHEKF-2 more advantageous. Table 5.7 clearly exhibits such an advantage, that is, the JT-MHEKF-2 affords a better estimation result than the JT-SHEKF-2. This table also displays the CPU time consumed in the implementation of four consid-

ered filters, showing that three fault-tolerant algorithms only have a slight increase of the computational burden relative to the standard JT-HEKF-n. Summarizing, the JT-DHEKF-2, JT-SHEKF-2 and JT-MHEKF-2 outperform well the JT-HEKF-2 in the case of a continuous constant measurement bias.

In order to illustrate that the robustness of the JT-DHEKF-2, JT-SHEKF-2 and JT-MHEKF-2 does not depend on the choice of the initial state, a MC simulation considering 1000 different initial state vectors, generated around the initial nominal state in terms of the initial error distribution, is carried out. The 10 cases with the worst estimation accuracy are shown in Fig. 5.10. From this figure it follows that the results of these 10 runs are consistent, that is, the results obtained by three fault-tolerant filters are not affected by the selection of initial state deviations.

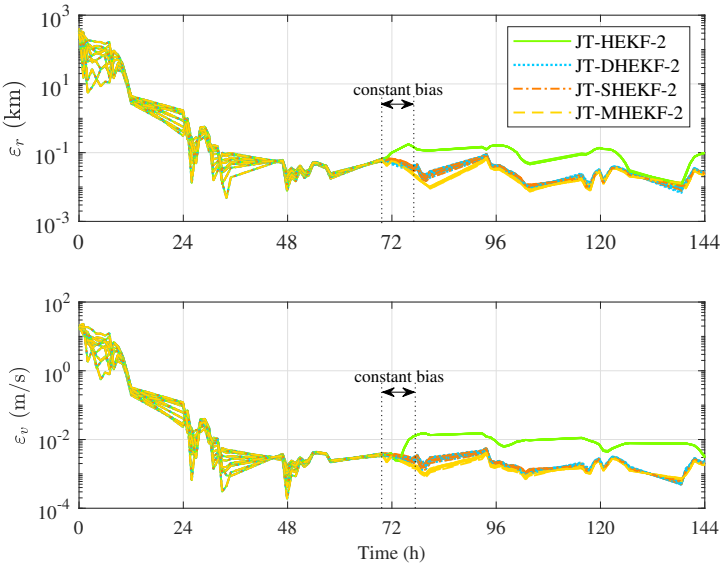


Figure 5.10: 10 MC runs with the worst estimation accuracy for the JT-HEKF-2, JT-DHEKF-2, JT-SHEKF-2 and JT-MHEKF-2 in the case of a constant measurement bias inside the time interval $t \in [70.8, 76.8]$ h.

5.4.2.2 Measurements with a continuous random bias

A second type of failures considered consists in adding a Gaussian random bias, with mean 8×10^{-5} rad and a standard deviation 8×10^{-5} rad, to the declination measurements inside the time interval $t \in [70.8, 76.8]$ h. These failures can be due, for instance, to a random bias in the orientation error of the optical sensor. In this case, and during the time interval $t \in [70.8, 76.8]$ h, the measurement model is,

$$\begin{cases} z_1 &= \arctan \dfrac{\bar{y}_b - \bar{y}}{\bar{x}_b - \bar{x}} + u_1 \,, \\ z_2 &= \arcsin \dfrac{\bar{z}_b - \bar{z}}{\sqrt{(\bar{x}_b - \bar{x})^2 + (\bar{y}_b - \bar{y})^2 + (\bar{z}_b - \bar{z})^2}} + u_2 + v \,, \end{cases} \tag{5.21}$$

where v indicates the random bias polluting the measurements.

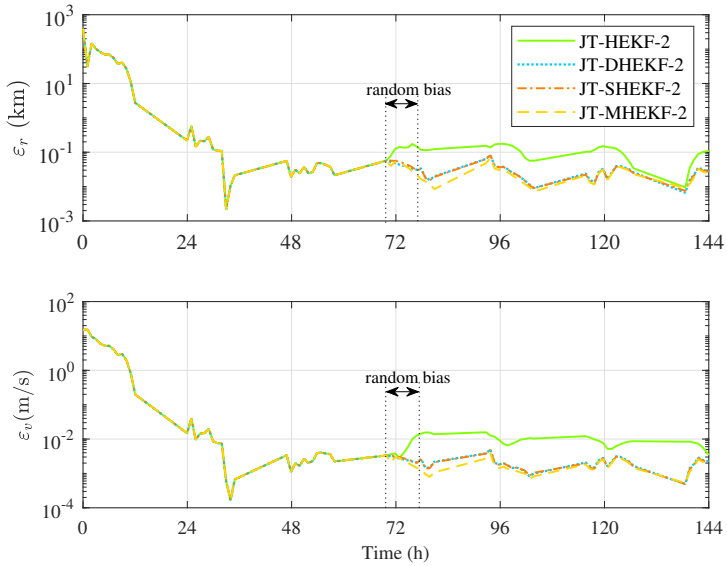


Figure 5.11: Position and velocity errors of the JT-HEKF-2, JT-DHEKF-2, JT-SHEKF-2 and JT-MHEKF-2 in the case of a random measurement bias inside the time interval $t \in [70.8, 76.8]$ h.

Table 5.8: RMSEs in position and velocity of the JT-HEKF-2, JT-DHEKF-2, JT-SHEKF-2, and JT-MHEKF-2 at the steady stage $t \in [96, 144]$ h in the case of a random measurement bias and the associated CPU time.

RMSE	JT-HEKF-2	JT-DHEKF-2	JT-SHEKF-2	JT-MHEKF-2
$\bar{\varepsilon}_r$ (km)	0.0965	0.0283	0.0293	0.0256
$\bar{\varepsilon}_v$ (m/s)	0.0073	0.0021	0.0021	0.0017
CPU time (s)	36.39	36.50	36.52	36.59

Figure 5.11 illustrates the estimation errors in position and velocity computed using the standard JT-HEKF-2, as well as the fault-tolerant JT-DHEKF-2, JT-SHEKF-2 and JT-MHEKF-2. Again, the standard JT-HEKF-2 fails to accurately estimate the GEO trajectory due to a random bias fault in the declination measurements. The effect of a continuous random measurement bias is significant on the JT-HEKF-2 algorithm, even when the measurement fault disappears. On the contrary, the fault-tolerant JT-DHEKF-2, JT-SHEKF-2 and JT-MHEKF-2 are almost not affected over the whole estimation procedure. The RMSEs in position and velocity and the CPU time are given in Table 5.8,

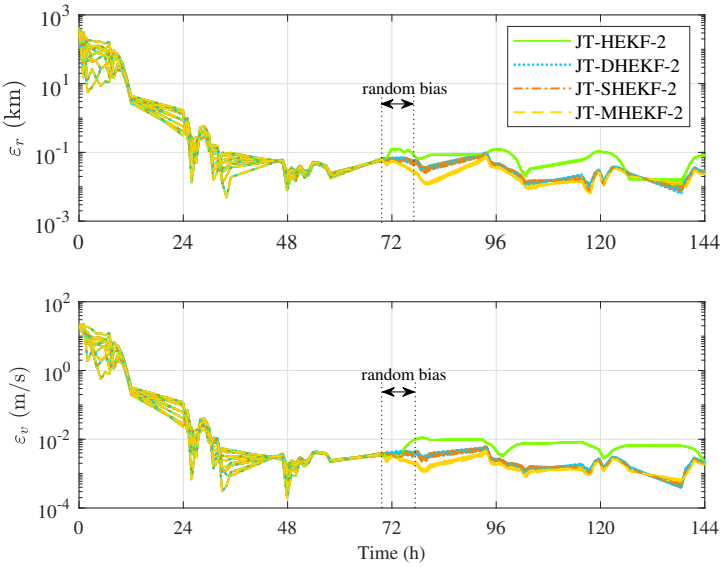


Figure 5.12: 10 MC runs with the worst estimation accuracy for the JT-HEKF-2, JT-DHEKF-2, JT-SHEKF-2 and JT-MHEKF-2 in the case of a random measurement bias inside the time interval $t \in [70.8, 76.8]$ h.

which again reveal the robustness improvement of the JT-DHEKF-2, JT-SHEKF-2 and JT-MHEKF-2, relative to the JT-HEKF-2, at the cost of a slight increase of the computational burden. Also in this case, 1000 MC runs have been done to test the influence of initial state deviations. The worst 10 simulation results are shown in Fig. 5.12, which shows that three fault-tolerant filters are still very robust when dealing with a random measurement bias.

5.4.2.3 An instantaneous zero output measurement

The last measurement failure considered consists in setting an instantaneous zero output to the declination measurement at a certain epoch ($t = 70.8$ h). In this way, the measurement at $t = 70.8$ h is given by

$$\begin{cases} z_1 &= \arctan \dfrac{\bar{y}_b - \bar{y}}{\bar{x}_b - \bar{x}} + u_1, \\ z_2 &= u_2. \end{cases} \tag{5.22}$$

The estimation errors for the satellite position and velocity computed by the adopted four filters are displayed in Fig. 5.13. Similar to the previous two cases, the JT-DHEKF-2, JT-SHEKF-2 and JT-MHEKF-2 are capable of affording accurate estimation results, even if they confront with an instantaneous zero measurement output. However, the performance

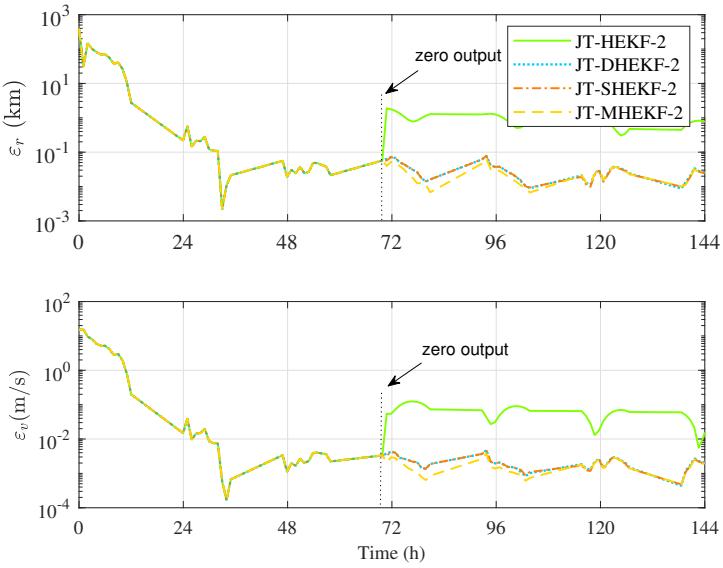


Figure 5.13: Position and velocity errors for the JT-HEKF-2, JT-DHEKF-2, JT-SHEKF-2 and JT-MHEKF-2 in the case of a zero sensor output at $t = 70.8$ h.

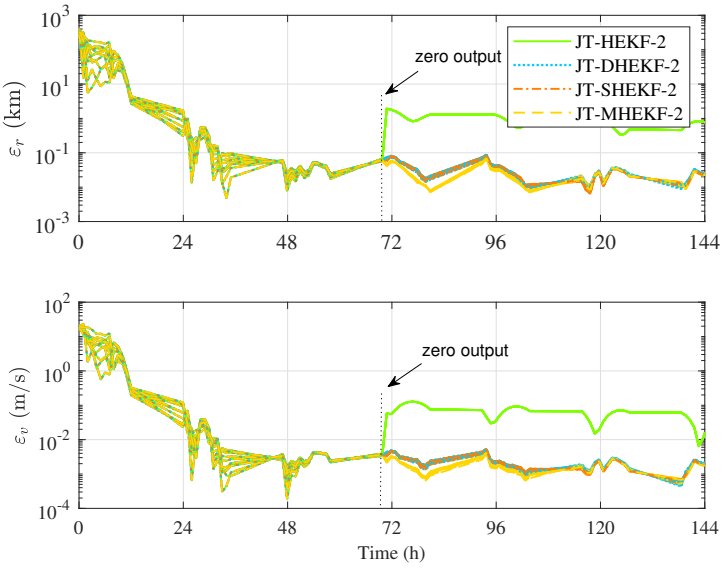


Figure 5.14: 10 MC runs with the worst estimation accuracy for the JT-HEKF-2, JT-DHEKF-2, JT-SHEKF-2 and JT-MHEKF-2 in the case of a zero sensor output at $t = 70.8$ h.

Table 5.9: RMSEs in position and velocity of the JT-HEKF-2, JT-DHEKF-2, JT-SHEKF-2, and JT-MHEKF-2 at the steady stage $t \in [96, 144]$ h in the case of an instantaneous zero sensor output and the associated CPU time.

RMSE	JT-HEKF-2	JT-DHEKF-2	JT-SHEKF-2	JT-MHEKF-2
$\bar{\varepsilon}_r$ (km)	0.7407	0.0291	0.0286	0.0247
$\bar{\varepsilon}_v$ (m/s)	0.0534	0.0021	0.0020	0.0017
CPU time (s)	36.67	36.81	36.85	36.93

of the JT-HEKF-2 is clearly affected from the fault, it fails giving accurate estimation results after $t = 70.8$ h and finally diverges. Table 5.9 displays the RMSEs in position and velocity obtained in this case. As in the previous cases, the JT-MHEKF-2 again achieves the best estimation accuracy. Figure 5.14 gives the results of the sensitivity analysis, with respect to initial state deviations, displaying once more the good accuracy and robustness of the JT-DHEKF-2, JT-SHEKF-2 and JT-MHEKF-2.

5.5 Remarks and conclusions

This chapter proposes a standard JT-HEKF-n for the autonomous orbit determination problem using the inter-satellite relative line of sight information. To validate the procedure, various scenarios have been considered and the results obtained indicate that the JT-HEKF-n enables to outperform the usual EKF. In fact, at least one order of magnitude accuracy gain is obtained by replacing the JT-HEKF-1 with the JT-HEKF-2. Also, large initial state deviations and long measurement acquisition periods generally tend to deteriorate the estimation performance of the usual EKF, or even make it to diverge. However, the JT-HEKF-n takes the advantage of the accurate nonlinear mapping in the synthesis of the filter, mitigating the sensitivity with respect to the sizes of initial state deviations and measurement acquisition periods. As a conclusion, an autonomous orbit determination problem can be perfectly addressed by the JT-HEKF-2, even when only a poor initial state guess or a sparse measurement is available. That is, the performance of the JT-HEKF-2 is quite robust and insensitive relative to initial state deviations and measurement acquisition periods.

Three fault-tolerant variants of the JT-HEKF-n, that keep good estimation characteristics in the case of measurement faults, have also been developed. An efficient measurement fault detection strategy has been first introduced using an innovation-based chi-square test. To deal with measurement faults, the JT-DHEKF-n completely guards against the deteriorative effects, just discarding false measurements and cancels the state updating steps. Besides, both JT-SHEKF-n and JT-MHEKF-n decrease the Kalman gain via the increment of the single scale factor, and the scale factor matrix, respectively. They suppress and correct the negative effect of measurement faults in the filter procedure.

In order to validate the effectiveness of fault-tolerant filters, various simulation scenarios,

with three different type of measurement faults, have been tested and compared against the results of a standard JT-HEKF-n. The comparisons show that three fault-tolerant variants have a superior estimation performance when measurement malfunctions are the points at issue. The JT-MHEKF-n possesses the best estimation performance due to the usage of multiple scale factors, and in contrast, the estimation accuracy of the standard JT-HEKF-n deteriorates, or even diverges in the case of measurement faults. Moreover, MC simulations have been carried out to stress consistent results of the proposed JT-HEKF-n and its three fault-tolerant variants with respect to initial state deviations. Finally, a further discussion on the required CPU time also indicates that the robustness of fault-tolerant algorithms significantly improve just at a slight increase on the cost of the computational burden.

6

Chapter 6

An efficient nonlinear statistical adaptive order-switching filter

As discussed in Chapters 4 and 5, based on the usage of the Taylor expansion technique, the common EKF has been extended into its high order forms in the JT framework, such as the standard JT-HEKF-n and augmented JT-AHEKF-n. Furthermore, to avoid the pollution of false measurements and guarantee the robustness of the standard JT-HEKF-n, the fault-tolerant JT-DHEKF-n, JT-SHEKF-n and JT-MHEKF-n have been integrally discussed. These filters are capable of extracting more nonlinear information from the dynamical and measurement models to provide a much better performance, especially in the case of sparse measurements and poor initial state guesses. From the perspective of the estimation accuracy, high order filters are preferred to solve OD problems. However, all variants of the EKF using high order state transition tensors are likely to encounter the heavy computational cost, especially when the system dimension is large. Note in particular that the antagonism between the estimation accuracy delivered and the computational burden associated always persists and, in general, the estimation error can be reduced by increasing the order of the filter, but the computational cost likewise increases in a significant way.

At present, the computational speed has been remarkably improved via the usage of automatic differentiation and integration techniques, such as the DA [138,139] and JT [83, 159] softwares, making nowadays possible the practical applications of these high order

variants. However, the continuous usage of a high order variant of the EKF in a whole estimation process is quite uneconomic. A high order filter always possesses a superior estimation accuracy but, if possible, for instance in a steady regime, a low order filter is much preferable due to its light computational burden.

Targeting to solve the OD problem with a high accuracy and computational efficiency, this chapter investigates a new adaptive order-switching variant of the standard JT-HEKF-n (named JT-OSHEKF). Note that this order-switching strategy is general and of straightforward application into other variants of the EKF. The rationale underlying the JT-OSHEKF filter is to design an adaptive switching strategy for automatically tuning the order of a JT-HEKF-n within one single run. At each filter step, an innovation-based function, accounting for the filter consistency, is put forward and estimated to determine the necessity of an order-switching operation. The JT technique is again used to accelerate the calculations by means of an advanced polynomial algebra proposed in Chapter 2, but it is decoupled from the adaptive filter algorithm.

6.1 Order-switching based JT-HEKF-n filter

The simulations in Chapters 4 and 5 exhibit that the JT-HEKF-1 algorithm often loses the estimation accuracy, or even diverges, when it encounters some hard situations with one or more of the following factors: strong system nonlinearities, sparse measurements, or large initial state deviations. In these cases, a high order JT-HEKF-n achieves much better estimations at the cost of increasing the computation time. This section is devoted to designing an effective order-switching strategy in terms of a filter consistency test within one single standard JT-HEKF-n implementation. Using this adaptive order-switching strategy, a new JT-OSHEKF filter is developed. It automatically employs a high order prediction process to deal with adverse simulation conditions, while it uses a low order prediction approach to handle the general mild simulation conditions. The main aim of developing the JT-OSHEKF filter is to solve OD problems within an allowable accuracy at the lowest computational cost.

6.1.1 Design of the adaptive order-switching strategy

The filter consistency not only implies the convergence of the estimates towards the true values, but also assesses the filter optimality [163]. The JT-OSHEKF filter algorithm is called consistent only if it satisfies the following statistical properties: 1) the estimates are unbiased, and 2) the actual innovation vector matches the innovation covariance matrix computed inside the JT-OSHEKF filter. Note that the NIS value is a common criterion adopted to test the filter consistency, since it enables to simultaneously verify both aforementioned properties [163].

The improvements of the JT-OSHEKF relative to the JT-HEKF-n can be summarized

as the following two parts: 1) the filter consistency test; 2) the design of an adaptive order-switching strategy. In the filter consistency test, the formulae (5.4) and (5.5) define the measurement innovation vector $\boldsymbol{\nu}_{k+1}$ and its associated innovation covariance matrix $\boldsymbol{P}_{\nu\nu,k+1}$ as follows

$$\boldsymbol{\nu}_{k+1} = \boldsymbol{z}^r_{k+1} - \boldsymbol{z}_{-,k+1}\,, \quad \boldsymbol{P}_{\nu\nu,k+1} = \boldsymbol{P}_{zz,k+1} + \boldsymbol{R}_{k+1}\,, \tag{6.1}$$

where $\boldsymbol{z}^r_{k+1}$ and $\boldsymbol{z}_{-,k+1}$ respectively indicate the real and predicted measurement vectors, $\boldsymbol{P}_{zz,k+1}$ and $\boldsymbol{R}_{k+1}$ respectively represent the measurement error covariance matrix and measurement noise covariance matrix. Furthermore, the NIS and time-average NIS can be further defined as

$$q_{k+1} = \boldsymbol{\nu}^T_{k+1}\,\boldsymbol{P}^{-1}_{\nu\nu,k+1}\,\boldsymbol{\nu}_{k+1}\,, \quad \bar{q}_{k+1} = \frac{1}{L}\sum_{j=k-L+2}^{k+1} q_j\,, \tag{6.2}$$

where L is the width of a moving window, indicating the sensitivity of an order-switching strategy relative to the result of the current chi-square test. An empirical analysis indicates that appropriate values of L can be taken between 2 and 5. By default, this value is set to 3 in the following illustrative example.

In order to design an adaptive order-switching strategy, two hypotheses are presented:

- $\mathbb{H}_0$: the JT-OSHEKF filter is consistent;
- $\mathbb{H}_1$: the JT-OSHEKF filter is not consistent.

Under the $\mathbb{H}_0$ hypothesis, the time-average NIS $\bar{q}_{k+1}$ is chi-square distributed with m degrees of freedom (this is χ^2_m, where m indicates the dimension of the measurement vector). If a level of significance α is selected (assume $\alpha = 99\%$), then the threshold value $\chi^2_{m,\alpha}$ is obtained by means of

$$P(\bar{q}_{k+1} > \chi^2_{m,\alpha}) = \alpha\,, \tag{6.3}$$

in terms of the probability values of chi-square distributions. At each estimation step, if $\bar{q}_{k+1} > \chi^2_{m,\alpha}$ the hypothesis $\mathbb{H}_1$ is accept, requiring the JT-OSHEKF algorithm to increase the order n for satisfying the filter consistency condition; otherwise, if $\bar{q}_{k+1} \leq \chi^2_{m,\alpha}$, the hypothesis $\mathbb{H}_0$ holds true showing the filter is consistent. Of course, in this latter situation one needs to further test whether the JT-OSHEKF filter enables to reduce the order with the consideration of improving the computational efficiency. Similarly, the following two hypotheses are put forward:

- $\bar{\mathbb{H}}_0$: the JT-OSHEKF filter can reduce the order n;
- $\bar{\mathbb{H}}_1$: the JT-OSHEKF filter needs to keep the order n unchanged.

Note that a very rigorous interval $[\chi^2_{m,\alpha_l}, \chi^2_{m,\alpha_u}]$ around $E(\bar{q}_{k+1}) = m$ should be preset before testing $\bar{\mathbb{H}}_0$ and $\bar{\mathbb{H}}_1$. It is worth to mention that the choices of $\alpha_l = 10\%$ and

$\alpha_u = 75\%$ used in this chapter depend on an empirical analysis. In essence, they should be always less than the α value adopted in the foregoing test of $\mathbb{H}_0$ and $\mathbb{H}_1$. If the statistical value $\bar{q}_{k+1} \in [\chi^2_{m,\alpha_l}, \chi^2_{m,\alpha_u}]$, it means that the current filter consistency is quite good. Therefore, the hypothesis $\bar{\mathbb{H}}_0$ is assumed to be true, which recommends the JT-OSHEKF filter to reduce the order n by one at the next estimation step, otherwise the order n is unchanged.

In general, the increment of the order in the JT-OSHEKF implementation enables to improve its convergence, even so, it is almost impossible to completely avoid the filter divergence issue. For instance, when the measurement is too sparse, the JT-OSHEKF filter is very likely to lose efficacy. Under these adverse circumstances, the JT-OSHEKF filter is possible to get stuck in an endless loop due to the unbounded increment of the order of a JT-OSHEKF filter. To avoid this situation, in the case of filter inconsistency a special index is proposed

$$\Pi_{n_1}^{n_2} = \frac{\left\| \bar{q}_{k+1}^{n_2} - \bar{q}_{k+1}^{n_1} \right\|}{\bar{q}_{k+1}^{n_1}}, \tag{6.4}$$

where the adaptive order-switching strategy takes n_1 and n_2 consecutively from the vector $[1, 2, 4, 6, \ldots, n_{max}]$ containing potential orders to be used, and n_{max} is a prescribed positive even number. If $\Pi_{n_1}^{n_2} < \varrho = 5\%$, it indicates that the JT-OSHEKF filter is not only inconsistent at orders n_1 and n_2, but also that it is not possible to improve the filter consistency through the increment of the order. Therefore, from the perspective of improving the computational efficiency, the increment of the order is ceased meanwhile the lower order n_1 will be adopted at the next estimation step. If $\Pi_{n_1}^{n_2} \geq 5\%$, it makes sense that the filter consistency can be improved by means of increasing the order from n_1 to n_2. In this case order n_2 will be adopted.

6.1.2 Detailed implementation of a JT-OSHEKF-n filter

Similar to the implementations of the other variants, such as JT-HEKF-n and JT-AHEKF-n filters, the JT technique is again employed to accelerate the evaluation of the high order Taylor expansions adopted in the prediction and updating steps of a JT-OSHEKF filter.

In the implementation of the developed JT-OSHEKF filter, it is necessary to check the filter optimality and automatically tuning the order to satisfy the filter consistency at each step. The detailed implementation of a JT-OSHEKF filter is summarized in Algorithm 6.1.

Algorithm 6.1 JT-OSHEKF algorithm

1: Initialize state $\boldsymbol{x}_{+,0}$, state covariance matrix $\boldsymbol{P}_{+,0}$, process and measurement noise covariance matrices $\boldsymbol{Q}$, $\boldsymbol{R}$, final time t_f, $t_k = 0$, $k = 0$;
2: Initialize the expansion order vector $n_v = [1, 2, 4, 6, \ldots, n_{max}]$, where n_{max} is selected empirically as well as the order $n = n_v[i]$, $i = 1$;

3: Initialize $\alpha_l = 10\%$, $\alpha_u = 75\%$, $\alpha = 99\%$ and determine χ^2_{m,α_l}, χ^2_{m,α_u}, $\chi^2_{m,\alpha}$ in terms of the probability tables of chi-square distributions;
4: k is increased by 1;
5: Use (5.2) to predict the a priori estimates of the state mean $\boldsymbol{x}_{-,k+1}$ and measurement mean $\boldsymbol{z}_{-,k+1}$, as well as the state error covariance matrix $\boldsymbol{P}_{-,k+1}$ at time t_{k+1};
6: Acquire a new measurement $\boldsymbol{z}^r_{k+1}$ at time t_{k+1}, calculate measurement innovation $\boldsymbol{\nu}_{k+1}$ using (5.4), meanwhile compute the innovation covariance matrix $\boldsymbol{P}_{\nu\nu,k+1}$ and cross-covariance matrix $\boldsymbol{P}_{xz,k+1}$ via (5.3) and (5.5);
7: Execute the order-switching strategy in terms of a chi-square test:
   1) Calculate the time-average NIS $\bar{q}_{k+1}$ using (6.2);
   2) Test the hypotheses $\mathbb{H}_0$ and $\mathbb{H}_1$:
      – If $\bar{q}_{k+1} < \chi^2_{m,\alpha}$ ($\mathbb{H}_0$ holds true), then test the hypotheses $\bar{\mathbb{H}}_0$ and $\bar{\mathbb{H}}_1$;
         * If $\bar{q}_{k+1} \in [\chi^2_{m,\alpha_l}, \chi^2_{m,\alpha_u}]$ ($\bar{\mathbb{H}}_0$ holds true), meanwhile if $n > 1$, set $i = i - 1$ and $n = n_v[i]$, otherwise set $n = 1$;
         * If $\bar{q}_{k+1} \notin [\chi^2_{m,\alpha_l}, \chi^2_{m,\alpha_u}]$ ($\bar{\mathbb{H}}_1$ holds true), keep the order n unchanged;
      – If $\bar{q}_{k+1} \geq \chi^2_{m,\alpha}$ ($\mathbb{H}_1$ holds true), set $i = i + 1$, $n = n_v[i]$, use t_{k-1}, $\boldsymbol{x}_{+,k-1}$, $\boldsymbol{P}_{+,k-1}$ to recompute $\bar{q}_{k+1}$ using (6.2) and calculate $\Pi^{n_2}_{n_1}$ using (6.4). Judge
         * If $\Pi^{n_2}_{n_1} > 5\%$, return to *2) in Step 7*;
         * If $\Pi^{n_2}_{n_1} \leq 5\%$ or $n = n_{max}$, keep the order n unchanged; (Generally, the estimation loses accuracy in this case).
8: Update $\boldsymbol{K}_{k+1}$, $\boldsymbol{x}_{+,k+1}$, $\boldsymbol{P}_{+,k+1}$ using (5.3)-(5.6);
9: Execute steps from 4 to 8 until $t_k = t_f$.

6.2 Model description

To validate the feasibility and assess the performance of the proposed JT-OSHEKF algorithm, the space-based autonomous geosynchronous OD problem based on a GEO-GEO observation configuration is taken as an illustrative example. Consider a pair of geosynchronous satellites, referred as GeoSat-A and GeoSat-B, where the GeoSat-A is assumed to be a beacon satellite with given accurate ephemeris, while the orbit of GeoSat-B has to be estimated from the information measured relative to the GeoSat-A.

6.2.1 Equations of motion

For a spacecraft operating in GEO orbits, the equations of motion formulated in the Cartesian coordinates are (see in subsection 3.1.1)

$$\begin{cases} \dot{x} = v_x, & \dot{y} = v_y, & \dot{z} = v_z, \\ \dot{v}_x = -\dfrac{\mu x}{r^3} + a_{p,x}, & \dot{v}_y = -\dfrac{\mu y}{r^3} + a_{p,y}, & \dot{v}_z = -\dfrac{\mu z}{r^3} + a_{p,z}, \end{cases} \tag{6.5}$$

where $\mu = 3.986004418 \times 10^5$ km^3/s^2 is the Earth's gravitational constant, $\boldsymbol{r} = [x, y, z]^T$, $\boldsymbol{v} = [v_x, v_y, v_z]^T$, and $\boldsymbol{a}_p = [a_{p,x}, a_{p,y}, a_{p,z}]^T$ respectively denote the satellite position, velocity, and perturbation acceleration vectors, $r = \sqrt{x^2 + y^2 + z^2}$ indicates the distance from the center of the Earth to the satellite. Besides, the dominant perturbations considered contain SRP, Earth's non-spherical gravity, and luni-solar gravitational attraction, whose formulae are shown in subsection 3.2.1.

6.2.2 Measurement model

To solve the space-based autonomous OD problem, the relative measurement information is employed to update the a priori estimates of spacecraft state vectors. Instead of the pure relative direction measurement model adopted in the Chapter 4 and 5, both inter-satellite relative distance and angular information are considered in this chapter, whose purpose is to further display the validity of the proposed JT-HEKF-n and JT-OSHEKF considering different types of the measurements. Denoting by α and β the relative right ascension (RA) and declination, and by Δr the relative range between GeoSat-A and GeoSat-B, the measurement model can be expressed as

$$\begin{cases} \alpha &= \arctan \dfrac{\Delta y}{\Delta x}\,, \\ \beta &= \arcsin \dfrac{dz}{\sqrt{\Delta x^2 + \Delta y^2 + \Delta z^2}}\,, \\ \Delta r &= \sqrt{\Delta x^2 + \Delta y^2 + \Delta z^2}\,, \end{cases} \tag{6.6}$$

being

$$\Delta x = x_A - x_B, \quad \Delta y = y_A - y_B, \quad \Delta z = z_A - z_B\,,$$

and where $\boldsymbol{r}_A = [x_A, y_A, z_A]^T$, $\boldsymbol{r}_B = [x_B, y_B, z_B]^T$ indicate the position vectors of GeoSat-A and GeoSat-B.

6.3 Numerical simulations

To compare the accuracy and efficiency of the proposed JT-OSHEKF algorithm with those computed by the standard JT-HEKF-n filter, a pair of geosynchronous satellites, i.e., GeoSat-A and GeoSat-B, are considered to construct an inter-satellite observation configuration, whose initial orbit elements and area-to-mass ratios $\eta = A/m$ are given in Table 6.1. It is worth to mention that the metrics proposed in subsection 4.2.5, such as RMSE and CPU time, are again employed here to evaluate the performance of the JT-OSHEKF filter. The process noise $\boldsymbol{w}_k$ and the measurement noise $\boldsymbol{u}_k$ are assumed to be uncorrelated zero mean white noises, whose covariance satisfy

$$\boldsymbol{Q}_k = diag\{0_{3\times3}, Q_v \boldsymbol{I}_{3\times3}\}\,, \quad \boldsymbol{R}_k = diag\{\sigma_\alpha^2, \sigma_\beta^2, \sigma_{dr}^2\}\,, \tag{6.7}$$

Table 6.1: Initial states and area-to-mass ratio of GeoSat-A and GeoSat-B.

	x (km)	y (km)	z (km)	v_x (m/s)	v_y (m/s)	v_z (m/s)	η (m^2/kg)
GeoSat-A	15631.6	−39276.4	−11.6	2862.9	1119.6	0.86	0.02000
GeoSat-B	22170.3	−35886.3	−93.1	2614.6	1615.4	−12.5	0.02242

where, $Q_v = 10^{-8}$ m/s^2, $\sigma_\alpha = \sigma_\beta = 0.64$ arc-sec, $\sigma_{dr} = 10^{-3}$ km.

In what follows, two simulation scenarios considering different magnitudes of initial state deviations and measurement frequencies are explored. The results computed by the adaptive JT-OSHEKF filter are compared against those computed by the standard JT-HEKF-1 and JT-HEKF-2 implemented with same simulation conditions.

6.3.1 Case study A: adverse simulation scenario

In the first case study (A), the initial position and velocity of GeoSat-B are assumed to be 100 km and 0.01 km/s off from the true state vector given in Table 6.1. Therefore, the initial covariance matrix is taken as a diagonal matrix, i.e., $\boldsymbol{P}_{0,+} = diag\{10^4\, \boldsymbol{I}_{3\times3}, 10^{-4}\, \boldsymbol{I}_{3\times3}\}$. A total simulation time of 6 days with 12 measurements per day, evenly distributed in a continuous visible tracklet of 12 hours is considered. Figure 6.1 illustrates the estimation errors in position and velocity computed by the JT-HEKF-1, JT-HEKF-2 and JT-OSHEKF. The results clearly reveal that both JT-OSHEKF and JT-HEKF-2 perform very well for estimating GEO orbits and they deliver almost the same estimation accuracy at the filter steady stage $t \in [48, 144]$ h: about 4.9 m in position and 0.0004 m/s in velocity. Furthermore, Fig. 6.2 displays the information about the standard deviations of the diagonal components of the state error covariance matrix computed in the JT-HEKF-1, JT-HEKF-2 and JT-OSHEKF implementations, where $\sigma_r = \sqrt{\sigma_x^2 + \sigma_y^2 + \sigma_z^2}$ and $\sigma_v = \sqrt{\sigma_{v_x}^2 + \sigma_{v_y}^2 + \sigma_{v_z}^2}$. Comparing with the position and velocity errors shown in Fig. 6.1 and the associated calculation using the diagonal elements of the filter-calculated state error covariance matrix shown in Fig. 6.2, it can be easily concluded that the JT-HEKF-1 filter significant overestimates the state errors at the initial estimation stage, such that it produces a quite bad estimation both in position and velocity.

Table 6.2: Case study A: RMSEs in position and velocity over $t \in [48, 144]$ h and the required CPU time.

Filters	$\bar{\varepsilon}_r$ (km)	$\bar{\varepsilon}_v$ (m/s)	CPU time (s)
JT-HEKF-1	0.1994	0.0146	24.3
JT-HEKF-2	0.0049	0.0004	38.7
JT-OSHEKF	0.0049	0.0004	26.1

Table 6.2 shows the RMSEs in position and velocity at the filter steady stage $t \in$

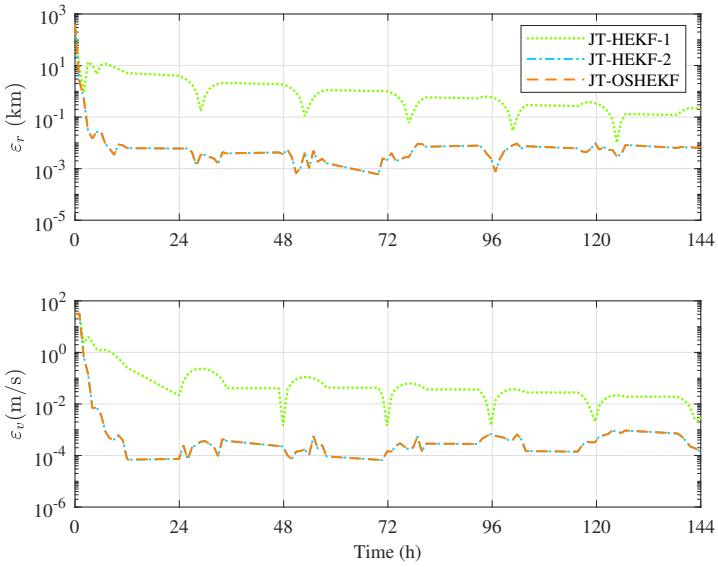


Figure 6.1: Comparison of position and velocity errors for case study A.

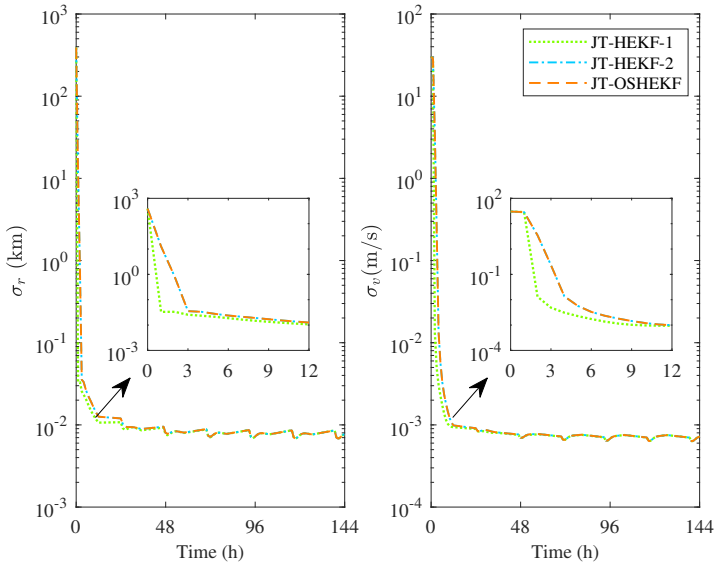


Figure 6.2: Comparison of standard deviations of position and velocity errors for case study A.

[48, 144] h, as well as the CPU time consumed in the JT-HEKF-1, JT-HEKF-2 and JT-OSHEKF implementations. It is worth to remark that the CPU time of the JT-OSHEKF implementation is just a little more than that of the JT-HEKF-1 implementation, but much less than the one required by the JT-HEKF-2 implementation. Meanwhile, Table 6.2 also indicates that the JT-OSHEKF performs as well as the JT-HEKF-2, keeping two orders of magnitude accuracy improvement with respect to the JT-HEKF-1, which further clearly reveals the importance of considering the nonlinearity of the dynamical and measurement models. As a conclusion, by means of the design and usage of the adaptive order-switching strategy, the JT-OSHEKF possesses the advantages of both JT-HEKF-1 and JT-HEKF-2, that is, a high computational efficiency and superior estimation accuracy.

Note that there is only one order-switching procedure carried out in this simulation scenario A, it happens at $t = 4$ h. So, the adaptive order-switching behaviors should be further tested with the re-consideration of the case study A by means of designedly mismatching the actual innovation and the filter-calculated innovation covariance matrix. This has been carried out by considering a noise of 10^{-5} rad in the angular measurement at $t = 53$ h with all other simulation conditions remaining the same. Note that the epoch $t = 53$ h is inside the filter steady stage. The simulation results of the JT-OSHEKF, implemented with and without this angular measurement noise, are compared in Fig. 6.3. Due to the addition of the angular measurement noise, the NIS of the JT-OSHEKF implementation at $t = 53$ h exceeds the threshold $\chi^2_{3,99\%}$, as it can be seen by the upper subgraph in Fig. 6.4. Therefore, two additional order switches appear in the noisy case: one at $t = 53$ h, switching from order 1 to 2 and another one at $t = 56$ h, switching back from order 2 to order 1, as it is displayed in the lower subgraph of Fig. 6.4.

To further analyze the influence of initial state deviations on the performance of the JT-OSHEKF, a MC simulation considering 1000 samplings is carried out. The initial state vectors are taken from a multivariable Gaussian distribution with the state mean the same as the true state and with standard deviations of 100 km in position components and of 0.01 km/s in velocity components. The results showed in a consistent way that both JT-OSHEKF and JT-HEKF-2 have the same filter accuracy, which is much better than that of the JT-HEKF-1. In other words, the performance of the JT-OSHEKF is independent on initial state deviations. For brevity, Fig. 6.5 displays the simulation results of 10 MC runs with the largest RMSEs in position and velocity inside the filter steady stage $t \in [48, 144]$ h, which essentially describe the behavior of the most adverse simulation scenarios in the case study A.

6.3.2 Case study B: mild simulation scenario

In the second case study (B), the mild simulation scenario, considered a smaller initial state deviation and a higher measurement frequency, is employed to further display the performance of both JT-HEKF-n and JT-OSHEKF methods. The smaller initial state deviation is assumed as an offset of 20 km in position components and 0.001 km/s in velocity components. Meanwhile, 36 measurements evenly distributed over each continuous

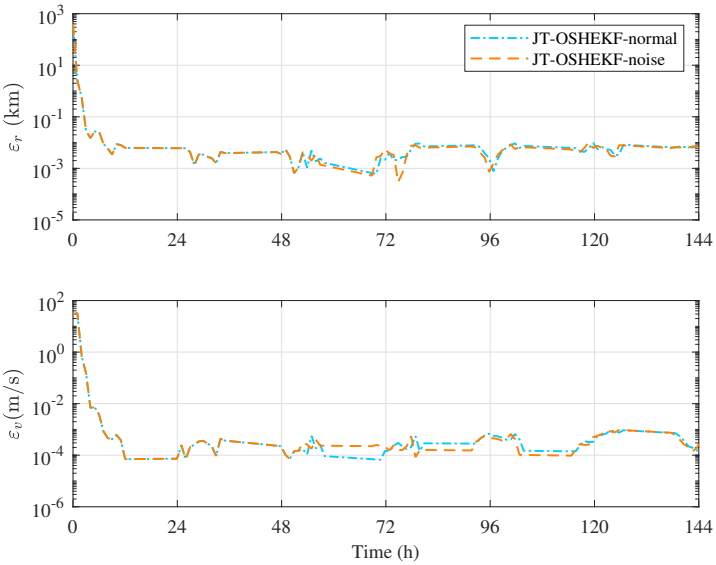


Figure 6.3: Comparison of position and velocity errors for case study A with and without an angular measurement noise at $t = 53$ h.

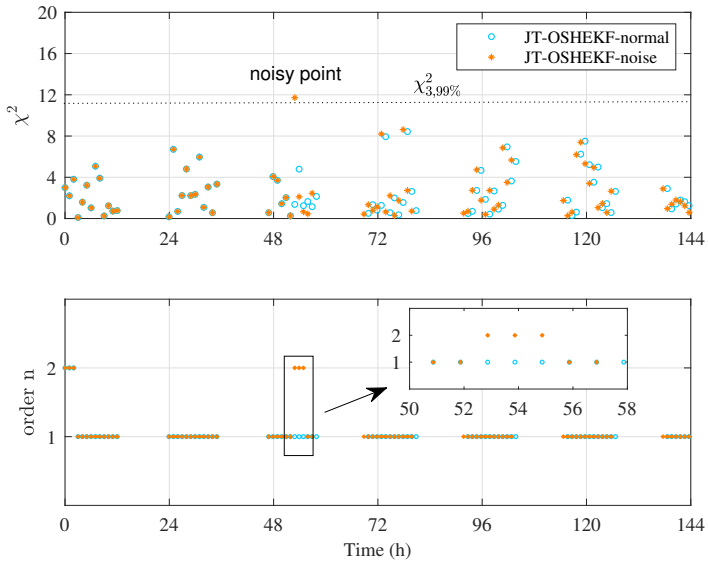


Figure 6.4: Order-switching procedures for the JT-OSHEKF algorithm with and without an angular measurement noise at $t = 53$h.

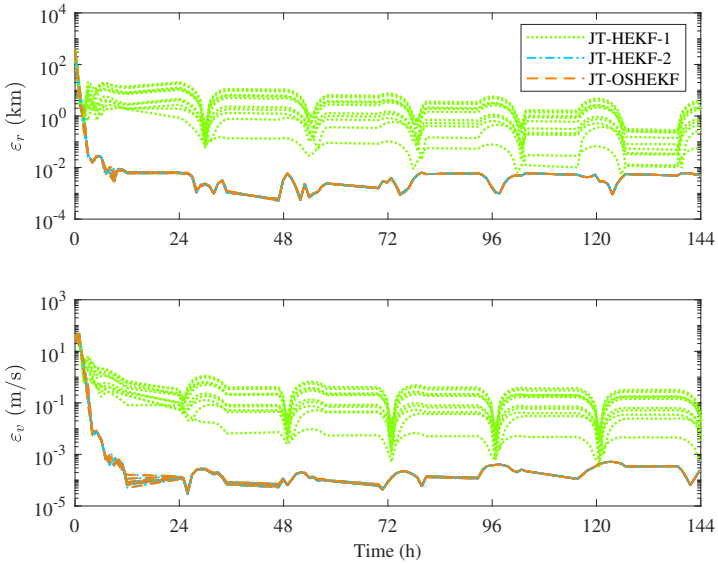


Figure 6.5: Sensitivity analysis relative to initial state deviations for case study A.

visible tracklet (again of 12 hours) are employed, that is, 20 minutes apart per measurement. Figure 6.6 exhibits the magnitude of position and velocity errors for the case study B. Again, this figure shows that there is no much performance difference between both JT-OSHEKF and JT-HEKF-2. Besides, with the consideration of a small initial state deviation and high measurement frequency, the accuracy of the JT-HEKF-1 improves significantly, which is almost identical to those obtained by the JT-OSHEKF and JT-HEKF-2 at the filter steady stage $t \in [48, 144]$ h. The unique difference is that the rates of convergence of the JT-OSHEKF and JT-HEKF-2 are much faster than that of the JT-HEKF-1. From the perspective of quantitative analysis, Table 6.3 gives the RMSEs in position and velocity at the filter steady stage $t \in [48, 144]$ h, as well as the CPU time consumed in the JT-HEKF-1, JT-HEKF-2 and JT-OSHEKF implementations. The results of the RMSEs in position and velocity again prove that the estimation accuracy of the JT-HEKF-1, JT-HEKF-2 and JT-OSHEKF is almost the same. The comparison between the results in Tables 6.2 and 6.3, indicates the JT-HEKF-1 improves its accuracy significantly, while the accuracy of the JT-HEKF-2 and JT-OSHEKF keeps constant since the results obtained in both case studies are optimal.

Table 6.3: Case study B: RMSEs in position and velocity over $t \in [48, 144]$ h and the required CPU time.

Filters	$\bar{\varepsilon}_r$ (km)	$\bar{\varepsilon}_v$ (m/s)	CPU time (s)
JT-HEKF-1	0.0056	0.0005	25.8
JT-HEKF-2	0.0049	0.0004	63.3
JT-OSHEKF	0.0049	0.0004	26.9

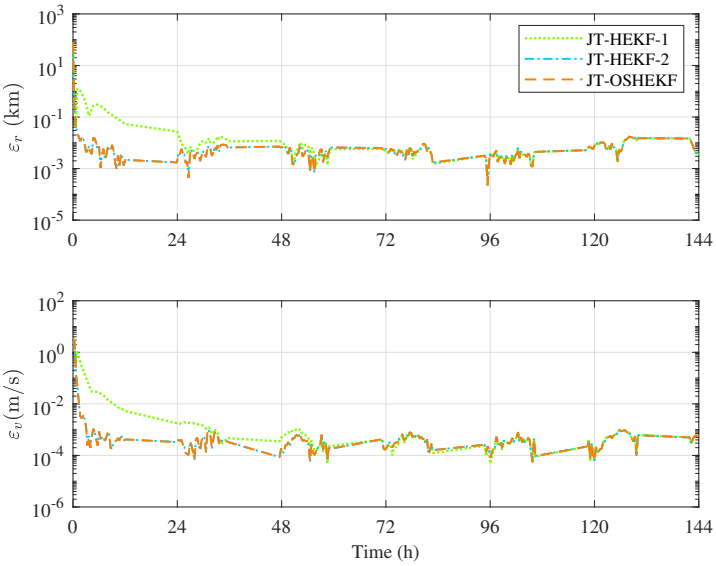


Figure 6.6: Comparison of position and velocity errors for case study B.

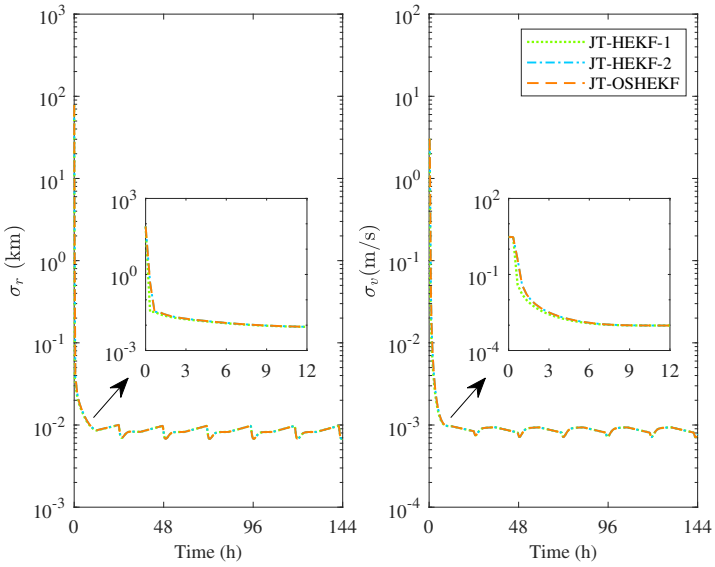


Figure 6.7: Comparison of the position and velocity error estimators for case study B.

On the other hand, Table 6.3 also evidences the significant amount of the CPU time consumed in the JT-HEKF-2 implementation (even compared with that cost in the case study A) due to the number of updating operations (i.e., the usage of a high measurement

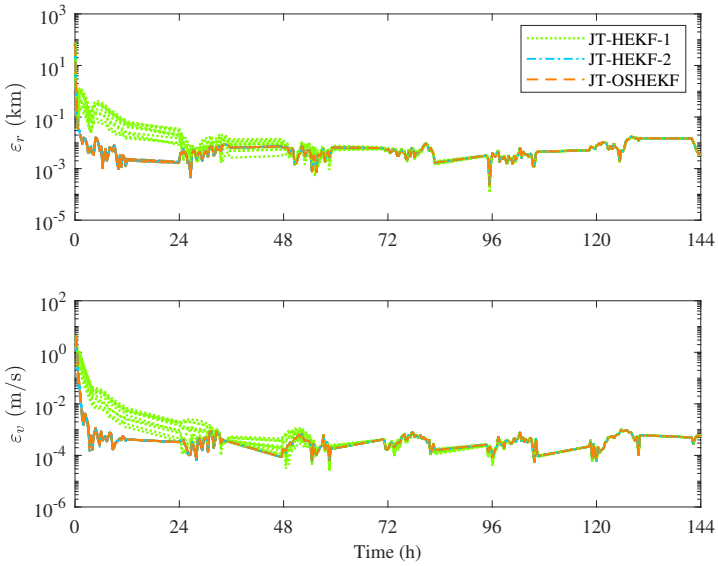


Figure 6.8: Sensitivity analysis relative to initial state deviations for case study B.

frequency), while the computational burden of the JT-OSHEKF and JT-HEKF-1 is low and just rises slightly. It is apparent that the JT-OSHEKF filter again maintains the individual advantages of both JT-HEKF-1 and JT-HEKF-2 in the case study B, through the adaptive adjustment of the order within one single run, that is, the JT-OSHEKF enables to implement the OD process as quick as the JT-HEKF-1 and as accurate as the JT-HEKF-2. Figure 6.7 shows that all these filters, i.e., JT-HEKF-1, JT-HEKF-2 and JT-OSHEKF, are capable of providing a conservative state error prediction in between measurements. This fact accounts for the accuracy improvement of the JT-HEKF-1 in the case study B, that is, first order method is only suitable to approximate the dynamics when initial state deviations are small and the propagation time is short.

Finally, similar to the case study A, the sensitivity analysis of the JT-OSHEKF with respect to initial state deviations is performed by means of a MC simulation with 1000 sampling points, where initial state vectors are taken from a multivariable Gaussian distribution with the state mean the same as the true state and a standard deviation of 20 km in position components and of 0.001 km/s in velocity components. The results of the 10 samples with the largest RMSEs in position and velocity at the filter steady stage $t \in [48, 144]$ h are shown in Fig. 6.8 and evidence that the analysis of behaviors we did for the particular case study B is, in fact, general.

6.4 Remarks and conclusions

This chapter proposes a new adaptive high order extended Kalman filter based on the design of a special order-switching strategy (JT-OSHEKF). The purpose is to ease the opposite balance between the estimation accuracy and the computational burden resulting from common high order extended Kalman filters. In other words, the JT-OSHEKF is designed to carry out a precise estimation at the lowest computational cost. To this aim, an innovation-based function, accounting for the filter consistency, is first introduced to determine the need of an order-switching operation at each estimation step. Moreover, further than that, an adaptive order-switching strategy is put forward to automatically adjust the order within one filter run by means of two innovation-based chi-square tests.

In order to validate the effectiveness of the proposed JT-OSHEKF algorithm, various simulation scenarios are tested in an autonomous orbit determination problem and compared with the standard JT-HEKF-n filter. The results indicate that the new proposed JT-OSHEKF algorithm significantly improve the performance by means of adaptive order-switching operations within one single filter run. To be more specific, in the illustrative examples, the computational complexity of the JT-OSHEKF represents a slight increase relative to the JT-HEKF-1, but its accuracy and rate of convergence are identical to those of the JT-HEKF-2. Besides, a Monte Carlo simulation verified that, within the same measurement accuracy premises, the performance of the JT-OSHEKF is insensible with respect to initial state deviations. Finally, it is worth to emphasize that, although the performance of the JT-OSHEKF has been discussed in an orbit determination problem, it can be easily generalized into any engineering estimation problem where the system nonlinearity or dimensionality are relevant in the robustness and efficiency of the filter.

7

Chapter 7
Conclusions

7.1 Main Conclusions

This dissertation is devoted to investigating accurate and efficient spacecraft orbit propagation and determination methods by means of using either high order Taylor or Chebyshev polynomial approximations. To this end, a series of high order propagators and estimators have been developed, as well as the required software to implement them in an efficient and accurate way. The main contributions and conclusions of this dissertation are be summarized as follows:

- **Accurate and efficient high order orbit propagators**

  In **Chapter 3** (i.e., the work published in [90]) high order nonlinear state mapping methods JTTNM-n and JTCNM-n have been developed using Taylor series expansion and Chebyshev interpolation technique in the JT framework. Both methods are validated by implementing the GEO orbit propagation with the usage of three coordinate representations. Besides the Earth's central Keplerian gravity, the considered vector field contains four main dominant perturbations: solar radiation pressure, Earth's potential and luni-solar gravitational attractions. Taking into account the size of uncertainty neighborhoods as well as the polynomial order and integration time step-size, abundant combinations of these above factors are simulated and a series of look-up tables with recommendations on the best options to address orbit propagations are given. This part of the work not only verifies the

feasibility of the proposed nonlinear propagators, but also provides valuable information on how to choose suitable polynomial forms for subsequent investigation of a progressive JT-based high order extended Kalman filter (JT-HEKF) and a JT-based augmented high order extended Kalman filter (JT-AHEKF).

It can also be concluded that the JTCNP-n method has a better accuracy and a more uniform error distribution than the JTTNP-n method implemented at the same expansion order, but at the cost of a little more CPU time. Furthermore, the improved JTTNP-n method based on an automatic domain splitting technique has been proved capable to achieve a better accuracy than the JTCNP-n method just with a slight decrease of the computational efficiency. The performance comparison among the equations of motion formulated in Cartesian, cylindrical coordinates and GEO hybrid elements shows that the GEO model is preferred in practical GEO missions due to its low computational cost and high accuracy. Generally, the CPU time spent in the propagation of the GEO model is a little less than that of the cylindrical dynamical model and three times less than that of the Cartesian model. Meanwhile, the propagation accuracy of the GEO model is somewhat better than that of the cylindrical dynamical model and much better than that of the Cartesian model.

- **Joint nonlinear orbit and parameter estimation using high order Taylor approximation**

  In **Chapter 4** (i.e., the work published in [159,175]) a novel augmented JT-AHEKF-n for simultaneously estimating the spacecraft orbit and uncertain parameters, either physically related with the spacecraft or with the measurement procedure, has been put forward through applying JTTNM-n method into the state prediction step of a Bayesian filtering framework. Two different model representations, formulated in the Cartesian coordinates and in GEO elements, are used and four dominant perturbations in the GEO regime are taken into consideration. The simulation results underline that the dynamical model formulated in GEO elements not only possesses a better estimation accuracy, but also needs of less computational burden than the Cartesian one. Letting aside the performance comparison between two coordinate representations, numerical simulations also confirm that both coordinate representations are effective to achieve a real-time joint estimation via the proposed JT-AHEKF-n.

  The performance comparison among different order JT-AHEKF-n methods shows that a higher order filter enables to provide a better estimation accuracy, and usually, just by tuning the order from 1st to 2nd, the JT-AHEKF-n obtains one order of magnitude accuracy gain in the case studies considered. Besides, one can conclude that the JT-AHEKF-n filters (i.e., $n > 1$) output a smaller estimation error dispersion relative to the classical EKF, pointing to the fact that the JT-AHEKF-n is more robust than the usual EKF.

- **Three fault-tolerant nonlinear orbit determination techniques**

In **Chapter 5** (i.e., the work published in [176]) the JT-HEKF-n filter is employed to achieve the autonomous satellite orbit estimation via the usage of sparse inter-satellite relative measurement information. Three fault-tolerant algorithms are put forward to suppress measurement faults. In the light of the insensitiveness of the proposed JT-HEKF-n relative to the measurement frequency, the first strategy directly discards the identified false measurements, being someway equivalent to the decrease of the measurement frequency. The second and third strategies employ a single scale factor and an adaptive scale matrix, respectively, as a multiplier to the measurement noise covariance matrix to avoid the pollution of false measurements.

In order to validate the effectiveness of fault-tolerant filters, various simulation scenarios, with three different type of measurement faults, have been tested and compared against a standard JT-HEKF-n filter. The comparison shows that three fault-tolerant variants have a superior estimation performance when measurement malfunctions are the points at issue. In contrast, the accuracy of the standard JT-HEKF-n deteriorates, or even diverges, in the case of false measurements. Moreover, a Monte Carlo simulation has been carried out to stress the robustness of the proposed JT-HEKF-n and its three fault-tolerant variants with respect to initial state deviations. A further discussion on the required CPU time indicates that fault-tolerant filters enable to obtain the robustness relative to false measurements at the cost of a slight increase of the computation time.

- **A statistical adaptive order-switching filter for orbit determination**

In **Chapter 6** (i.e., the work published in [177]) a new adaptive high order extended Kalman filter based on the design of a special order-switching strategy has been proposed in the JT framework (JT-OSHEKF). The improvement further eases the opposite balance between the estimation accuracy delivered and the computational burden associated to the JT-HEKF-n filter and achieves the orbit determination with higher accuracy and computational efficiency. The rationale underlying the JT-OSHEKF is to design an adaptive switching strategy for automatically tuning the order of a JT-HEKF-n within one single run. At each filter step, an innovation-based function, accounting for the filter consistency, is put forward and estimated to determine the necessity of an order-switching operation. Note that this order-switching strategy is general and of straightforward application into other variants of the EKF.

The performance of the JT-OSHEKF filter is tested in various simulation scenarios, considering an autonomous orbit determination problem and comparing with the standard JT-HEKF-n. The results indicate that the developed JT-OSHEKF filter significantly improves the performance by means of using the adaptive order-switching strategy within one single filter run. To be more specific, in the illustrative examples, the computational complexity of the JT-OSHEKF represents a slight increase relative to the JT-HEKF-1, but its accuracy and rate of convergence are identical to those of the JT-HEKF-2.

Finally, it is worth to emphasize that, although this dissertation discussed all high order propagators, estimators and associated variants either in orbit propagation problems or in orbit determination problems, the proposed methodology can be easily generalized and applied in general engineering prediction or estimation problems where the system non-linearity or the state dimensionality are relevant, for sake robustness and computational efficiency.

7.2 Future work

There are several open questions and research lines related to the work explored in this dissertation that deserve further investigations, the key ones are listed as follows:

Polynomial expansion form. Both Taylor and Chebyshev polynomials have been adopted and the associated polynomial algebras have been addressed in Chapter 2. Although both of them are accurate enough to develop the high order orbit propagators and estimators, their shortcomings are obvious. For instance, the Taylor series expansion often encounters the loss of accuracy when large state deviations are considered or the approximated systems are not continuous and differentiable, while the Chebyshev interpolation technique may be adequate for non-smooth systems, but always possesses very low computational efficiency when the dimensionality of the system is large. From the perspective of the computational accuracy and efficiency, the accuracy might be enhanced at specific levels using other interval algebraic methods, such as Fourier series expansion. Therefore, the JT software developed in this Ph.D project is expected to further contain other efficient and precise approximation techniques.

Fault-tolerant variants of the standard high order extended Kalman filter. A thorough analysis of three measurement fault-tolerant design of the standard high order extended Kalman filter has been carried out. However, this dissertation does not explore any fault-tolerant strategy to avoid the loss of accuracy or divergence in the case of the abrupt change of the system dynamics, for instance, an unknown impulse maneuver. In practical engineering missions, a special fault-tolerant design is of vital importance to keep the robustness of the filter relative to sudden change of the system dynamics. Another interesting point is to design fault diagnosis and tolerant strategies considering possible faults in both, dynamical models and measurement procedures.

Applications and missions. Finally, this dissertation focused on the applications of the JT technique into the orbit propagation and determination problems. However, a growing interest has been focused on the usage of JT technique into other applications and missions, such as bounded relative motion design [178], perturbed Lambert problem analysis [179], high order optimal feedback control problem [180]. As an automatic, efficient, and accurate polynomial approximation technique implemented in modern computers, JT technique deserves a further investigation about its potential applications.

Appendix

This appendix details the necessary steps for deducing explicit transformation formulae from GES and CYC elements into CAC coordinates.

Appendix A.1: Conversion formula from GES to CAC

This sub-appendix gives a detailed conversion from GES elements into CAC coordinates. We firstly define the perifocal reference frame as follows: its origin is centered at the focus of the orbit, x axis is directed from the focus to the periapse, z axis is normal to the orbital plane along the direction of the angular momentum vector and the direction of y axis is given by the right-hand rule. In the derivation process, the Keplerian element set $\{a, e, i, \omega, \Omega, \theta\}$ is used as a bridge. The satellite position and velocity vectors can be expressed in the perifocal reference frame as

$$\boldsymbol{r} = [r\cos\theta, r\sin\theta, 0]^T \ , \tag{1}$$

$$\boldsymbol{v} = \left[-\frac{\mu}{h}\sin\theta, \frac{\mu}{h}(e+\cos\theta), 0\right]^T \ . \tag{2}$$

The direction cosine matrix between the ECI and the perifocal reference frames can be obtained using Keplerian elements,

$$\mathcal{T}_p^{\mathcal{I}} = \begin{pmatrix} \cos\Omega\cos\omega - \sin\Omega\sin\omega\cos i & -\cos\Omega\sin\omega - \sin\Omega\cos\omega\cos i & \sin\Omega\sin i \\ \sin\Omega\cos\omega + \cos\Omega\sin\omega\cos i & -\sin\Omega\sin\omega + \cos\Omega\cos\omega\cos i & -\cos\Omega\sin i \\ \sin\omega\sin i & \cos\omega\sin i & \cos i \end{pmatrix} . \tag{3}$$

According to the definition of the GEO elements [157], we obtain,

$$\begin{gathered}\sin\Omega=\frac{Q_1}{\sqrt{Q_1^2+Q_2^2}},\quad \cos\Omega=\frac{Q_2}{\sqrt{Q_1^2+Q_2^2}},\quad 1-\cos i=\frac{2(Q_1^2+Q_2^2)}{1+Q_1^2+Q_2^2},\\ \cos^2\Omega+\sin^2\Omega\cos i=\frac{1-Q_1^2+Q_2^2}{1+Q_1^2+Q_2^2},\quad \sin^2\Omega+\cos^2\Omega\cos i=\frac{1+Q_1^2-Q_2^2}{1+Q_1^2+Q_2^2},\end{gathered} \tag{4}$$

and using, $s=\omega+\Omega+\theta$, we have, $\omega+\theta=s-\Omega$.

Based on the above equations, the satellite position and velocity vectors can be transformed into the ECI reference frame through the direction cosine matrix $\mathcal{T}_p^{\mathcal{I}}$. For instance, the x component of the satellite's position vector is

$$\begin{aligned}x &= r\cdot[\cos\Omega\cos(\omega+\theta)-\sin\Omega\sin(\omega+\theta)\cos i]\\ &= r\cdot\left(\cos^2\Omega\cos s+\sin\Omega\cos\Omega\sin s-\sin\Omega\cos\Omega\sin s\cos i+\sin^2\Omega\cos s\cos i\right)\\ &= r\cdot\left[\cos\Omega\sin\Omega\sin s\,(1-\cos i)+\cos s\left(\cos^2\Omega+\sin^2\Omega\cos i\right)\right]\\ &= \frac{r}{1+Q_1^2+Q_2^2}\left[2Q_1Q_2\sin s+(1-Q_1^2+Q_2^2)\cos s\right],\end{aligned} \tag{5}$$

where r is given by (3.6). The same procedure is carried out to compute other components of the satellite position and velocity vectors, obtaining the following explicit functions

$$\begin{cases} x &= \dfrac{r}{1+Q_1^2+Q_2^2}\left[2Q_1Q_2\ \sin\ s+\left(1-Q_1^2+Q_2^2\right)\ \cos\ s\right],\\ y &= \dfrac{r}{1+Q_1^2+Q_2^2}\left[2Q_1Q_2\ \cos\ s+\left(1+Q_1^2-Q_2^2\right)\ \sin\ s\right],\\ z &= \dfrac{2\,r}{1+Q_1^2+Q_2^2}\left[Q_2\ \sin\ s-Q_1\ \cos\ s\right],\\ \dot{x} &= \dfrac{\mu}{h\,(1+Q_1^2+Q_2^2)}\left[2Q_1Q_2\ (\cos\ s+e_x)\right.\\ &\quad \left.+\left(-1+Q_1^2-Q_2^2\right)\ (\sin\ s+e_y)\right],\\ \dot{y} &= \dfrac{\mu}{h\,(1+Q_1^2+Q_2^2)}\left[-2Q_1Q_2\ (\sin\ s+e_y)\right.\\ &\quad \left.+\left(1+Q_1^2-Q_2^2\right)\ (\cos\ s+e_x)\right],\\ \dot{z} &= \dfrac{\mu}{h\,(1+Q_1^2+Q_2^2)}\left[Q_1\ (\sin\ s+e_y)+Q_2\,(\cos\ s+e_x)\right].\end{cases} \tag{6}$$

Appendix A.2: Conversion formula from CYC to CAC

This sub-appendix gives a detailed derivation of the explicit relation between CYC and CAC coordinates, which is quite straightforward

$$\begin{cases} x = \rho\cos\varphi, & y = \rho\sin\varphi, & z = z, \\ \dot{x} = \dot{\rho}\cos\varphi - \rho\dot{\varphi}\sin\varphi, & \dot{y} = \dot{\rho}\sin\varphi + \rho\dot{\varphi}\cos\varphi, & \dot{z} = \dot{z}\,. \end{cases} \tag{7}$$

Bibliography

[1] G. Maral and M. Bousquet, *Satellite communications systems: systems, techniques and technology*. John Wiley & Sons, 2011. (Cited on page 1.)

[2] S. K. Shrivastava, Orbital perturbations and stationkeeping of communication satellites, *Journal of Spacecraft and Rockets*, vol. 15, no. 2, pp. 67–78, 1978. (Cited on page 1.)

[3] Q. Lu, W. Bell, P. Bauer, N. Bormann, and C. Peubey, An evaluation of FY-3A satellite data for numerical weather prediction, *Quarterly Journal of the Royal Meteorological Society*, vol. 137, no. 658, pp. 1298–1311, 2011. (Cited on page 1.)

[4] P. Gong, X. Li, and W. Zhang, 40-year (1978–2017) human settlement changes in China reflected by impervious surfaces from satellite remote sensing, *Science Bulletin*, vol. 64, no. 11, pp. 756–763, 2019. (Cited on page 1.)

[5] K. Tomiyasu and J. L. Pacelli, Synthetic aperture radar imaging from an inclined geosynchronous orbit, *IEEE Transactions on Geoscience and Remote Sensing*, no. 3, pp. 324–329, 1983. (Cited on pages 1 and 2.)

[6] L. Kou, X. Wang, and M. Xiang, Effects on three-dimensional geosynchronous circular sar imaging by orbit errors, *Journal of the Indian Society of Remote Sensing*, vol. 42, no. 1, pp. 1–12, 2014. (Cited on pages 1 and 3.)

[7] L. Kou, X. Wang, M. Xiang, J. Chong, and M. Zhu, Effect of orbital errors on the geosynchronous circular synthetic aperture radar imaging and interferometric processing, *Journal of Zhejiang University SCIENCE C*, vol. 12, no. 5, pp. 404–416, 2011. (Cited on pages 1 and 3.)

[8] J. A. Kennewell and B. N. Vo, An overview of space situational awareness, *Proceedings of the 16th International Conference on Information Fusion*, IEEE, 2013, pp. 1029–1036. (Cited on page 2.)

[9] M. Jiang, W. Hu, C. Ding, and G. Liu, The effects of orbital perturbation on geosynchronous synthetic aperture radar imaging, *IEEE Geoscience and Remote Sensing*

Letters, vol. 12, no. 5, pp. 1106–1110, 2015. (Cited on pages 2 and 3.)

[10] S. N. Madsen, C. Chen, and W. Edelstein, Radar options for global earthquake monitoring, *IEEE International Geoscience and Remote Sensing Symposium*, vol. 3, pp.1483–1485, IEEE, 2002. (Cited on page 2.)

[11] R. M. Fuster, M. F. Usón, and A. B. Ibars, Interferometric orbit determination for geostationary satellites, *Science China Information Sciences*, vol. 60, no. 6, pp.1-11, 2017. (Cited on pages 2 and 53.)

[12] G. Wadge, A. M. Guarnie, S. Hobbs, and D. Schul, Potential atmospheric and terrestrial aplications of a geosynchronous radar, *2014 IEEE Geoscience and Remote Sensing Symposium*, pp.946–949, IEEE, 2014. (Cited on page 2.)

[13] S. Ribo, J. C. Arco, E. Cardellach, S. Oliveras, A. Rius, and C. Buck, Preliminary results of digital satellite TV opportunity signals scattered on the sea-surface, *Workshop on Reflectometry using GNSS and Other Signals of Opportunity*, Purdue University, West Lafayette, USA, October 10-11, 2012. (Cited on pages 2 and 53.)

[14] A. Moussessian, C. Chen, W. Edelstein, S. Madsen, and P. Rosen, System concepts and technologies for high orbit SAR, *IEEE MTT-S International Microwave Symposium Digest*, IEEE, 2005. (Cited on page 2.)

[15] K. Tomiyasu, Synthetic aperture radar in geosynchronous orbit, *Proceedings of IEEE Antennas and Propagation Society International Symposium*, 1978. (Cited on page 2.)

[16] JPL NASA, Global earthquake satellite system: a 20-year plan to enable earthquake prediction, *Technology Report JPL*, pp. 400–1069, 2003. (Cited on page 2.)

[17] W. N. Edelstein, S. N. Madsen, A. Moussessian, and C. Chen, Concepts and technologies for synthetic aperture radar from MEO and geosynchronous orbits, *Enabling Sensor and Platform Technologies for Spaceborne Remote Sensing*, International Society for Optics and Photonics, vol. 5659, pp. 195–203, 2005. (Cited on page 2.)

[18] D. Bruno, S. E. Hobbs, and G. Ottavianelli, Geosynchronous synthetic aperture radar: Concept design, properties and possible applications, *Acta Astronautica*, vol. 59, no. 1-5, pp. 149–156, 2006. (Cited on page 2.)

[19] S. Hobbs, C. Mitchell, B. Forte, R. Holley, B. Snapir, and P. Whittaker, System design for geosynchronous synthetic aperture radar missions, *IEEE Transactions on Geoscience and Remote Sensing*, vol. 52, no. 12, pp. 7750–7763, 2014. (Cited on page 2.)

[20] Z. Sun, J. Wu, J. Pei, Z. Li, Y. Huang, and J. Yang, Inclined geosynchronous spaceborne–airborne bistatic SAR: Performance analysis and mission design, *IEEE*

Transactions on Geoscience and Remote Sensing, vol. 54, no. 1, pp. 343–357, 2015. (Cited on page 2.)

[21] A. M. Guarnieri, A. Broquetas, A. Recchia, F. Rocca, and J. Ruiz-Rodon, Advanced radar geosynchronous observation system: Argos, *IEEE Geoscience and Remote Sensing Letters*, vol. 12, no. 7, pp. 1406–1410, 2015. (Cited on page 2.)

[22] A. M. Guarnieri and C. Hu, Geosynchronous and geostationary SAR: face to face comparison, *Proceedings of EUSAR 2016: 11th European Conference on Synthetic Aperture Radar*, VDE, pp. 1–4, 2016. (Cited on page 2.)

[23] W. Tian, C. Hu, T. Zeng, and Z. Ding, Several special issues in GEO SAR system, *8th European Conference on Synthetic Aperture Radar*, VDE, pp. 1–4, 2010. (Cited on page 2.)

[24] F. K. Li, D. N. Held, J. C. Curlander, and C. Wu, Doppler parameter estimation for spaceborne synthetic-aperture radars, *IEEE transactions on Geoscience and Remote Sensing*, vol. GE-23, no. 1, pp. 47–56, 1985. (Cited on page 2.)

[25] R. F. Hanssen, *Radar interferometry: data interpretation and error analysis*, Netherlands: Kluwer Academic Publishers, 2002. (Cited on page 3.)

[26] T. Kelso, Analysis of the iridium 33-cosmos 2251 collision, *AIAA/AAS Astrodynamics Specialist Conference*, Pittsburgh, Pennsylvania, August, 2009. (Cited on page 3.)

[27] S. Veniaminov, I. Oleynikov, and E. Melnikov, Indices of growth of danger for space activities from orbital debris and the related mitigation measures, *Kinematics and Physics of Celestial Bodies*, vol. 32, no. 5, pp. 227–232, 2016. (Cited on page 3.)

[28] R. J. Rovetto and T. Kelso, Preliminaries of a space situational awareness ontology, *26th AIAA/AAS Space Flight Mechanics meeting*, Napa, California, 2016. (Cited on pages 3 and 6.)

[29] N. Bobrinsky and L. Del Monte, The space situational awareness program of the European space agency, *Cosmic Research*, vol. 48, no. 5, pp. 392–398, 2010. (Cited on page 4.)

[30] R. D. Coder and M. J. Holzinger, Multi-objective design of optical systems for space situational awareness, *Acta Astronautica*, vol. 128, pp. 669–684, 2016. (Cited on page 4.)

[31] O. Montenbruck and E. Gill, *Satellite orbits: models, methods and applications*, Springer Science & Business Media, 2012. (Cited on pages 6, 8, 13, 54, 58, and 60.)

[32] O. P. L. Maître and O. M. Knio, *Spectral methods for uncertainty quantification: with*

applications to computational fluid dynamics, The Netherlands, chap. 1: Springer, 2010. (Cited on page 6.)

[33] A. Wittig and R. Armellin, High order transfer maps for perturbed Keplerian motion, *Celestial Mechanics and Dynamical Astronomy*, vol. 122, no. 4, pp. 333–358, 2015. (Cited on pages 6, 11, and 38.)

[34] D. A. Vallado, *Fundamentals of astrodynamics and applications*, Springer Science and Business Media, vol. 12, 2001. (Cited on page 6.)

[35] K. Aksnes, A second-order artificial satellite theory based on an intermediate orbit, *The Astronomical Journal*, vol. 75, no. 9, pp.1066–1094, 1970. (Cited on page 6.)

[36] H. Kinoshita, Third-order solution of an artificial-satellite theory, *SAO special report*, vol. 379, 1977. (Cited on page 6.)

[37] L. M. Healy, Orbit propagation with Lie transfer maps in the perturbed Kepler problem, *Celestial Mechanics and Dynamical Astronomy*, vol. 85, no. 2, pp. 175–207, 2003. (Cited on page 6.)

[38] R. A. Broucke, Long-term third-body effects via double averaging, *Journal of Guidance, Control, and Dynamics*, vol. 26, no. 1, pp. 27–32, 2003. (Cited on page 6.)

[39] W. D. McClain, A recursively formulated first-order semianalytic artificial satellite theory based on the generalized method of averaging. volume 1: The generalized method of averaging applied to the artificial satellite problem, *Computer Sciences Corporation*, vol. 1, 1977. (Cited on page 6.)

[40] G. Hori, Theory of general perturbation with unspecified canonical variable, *Publications of the Astronomical Society of Japan*, vol. 18, pp. 287–296, 1966. (Cited on page 6.)

[41] M. Lara, J. F. San-Juan, L. M. López, and P. J. Cefola, On the third-body perturbations of high-altitude orbits, *Celestial Mechanics and Dynamical Astronomy*, vol. 113, no. 4, pp. 435–452, 2012. (Cited on page 6.)

[42] J. L. Junkins, M. R. Akella, and K. T. Alfriend, Non-Gaussian error propagation in orbital mechanics, *Advances in the Astronautical Sciences*, vol. 92, pp. 283–298, 1996. (Cited on pages 7 and 9.)

[43] C. Sabol, K. Hill, K. Alfriend, and T. Sukut, Nonlinear effects in the correlation of tracks and covariance propagation, *Acta Astronautica*, vol. 84, pp. 69–80, 2013. (Cited on page 7.)

[44] A. C. Jesus, M. L. O. Souza, and A. Prado, Statistical analysis of nonimpulsive

orbital transfers under thrust errors, *Nonlinear Dynamics and Systems Theory*, vol. 2, no. 2, pp. 157–172, 2002. (Cited on page 7.)

[45] R. Ghrist and D. Plakalovic, Impact of non-Gaussian error volumes on conjunction assessment risk analysis, *AIAA/AAS Astrodynamics specialist conference*, Minneapolis, Minnesota, USA, pp. 4965–4980, 2012. (Cited on page 7.)

[46] P. S. Maybeck, *Stochastic models, estimation, and control*, New York: Academic press, vol. 3, 1982. (Cited on page 8.)

[47] C. Yang and M. Kumar, On the effectiveness of Monte Carlo for initial uncertainty forecasting in nonlinear dynamical systems, *Automatica*, vol. 87, pp. 301–309, 2018. (Cited on page 8.)

[48] C. Yang and M. Kumar, Closed-loop adaptive Monte Carlo framework for uncertainty forecasting in nonlinear dynamic systems, *Journal of Guidance, Control, and Dynamics*, vol. 42, no. 6, pp. 1218–1236, 2019. (Cited on pages 8 and 91.)

[49] A. Morselli, R. Armellin, P. Di Lizia, and F. B. Zazzera, Computing collision probability using differential algebra and advanced Monte Carlo methods, *Proceedings of the 63rd International Astronautical Congress, Napoli, Italy*, 2010. (Cited on page 8.)

[50] A. Morselli, R. Armellin, P. Di Lizia, and F. B. Zazzera, Computation of collision probabilities based on special perturbations and high order methods, *6th European Conference on Space Debris, ESA/ESOC*, ESA, 2013, pp. 1–7. (Cited on page 8.)

[51] N. Arora, V. Vittaldev, and R. P. Russell, Parallel computation of trajectories using graphics processing units and interpolated gravity models, *Journal of Guidance, Control, and Dynamics*, vol. 38, no. 8, pp. 1345–1355, 2015. (Cited on page 8.)

[52] D. T. Hall, S. J. Casali, L. C. Johnson, B. B. Skrehart, and L. G. Baars, High fidelity collision probabilities estimated using brute force Monte Carlo simulations, *2018 AAS/AIAA Astrodynamics Specialist Conference*, Snowbird, UT, 2018. (Cited on pages 8 and 9.)

[53] B. Schutz, B. Tapley, and G. H. Born, *Statistical orbit determination*, Elsevier, 2004. (Cited on page 8.)

[54] R. H. Battin, *An introduction to the mathematics and methods of astrodynamics, revised edition*, AIAA, 1999. (Cited on page 8.)

[55] D. K. Geller, Linear covariance techniques for orbital rendezvous analysis and autonomous onboard mission planning, *Journal of Guidance, Control, and Dynamics*, vol. 29, no. 6, pp. 1404–1414, 2006. (Cited on page 9.)

[56] D. C. Woffinden and D. K. Geller, Relative angles-only navigation and pose estimation for autonomous orbital rendezvous, *Journal of Guidance, Control, and Dynamics*, vol. 30, no. 5, pp. 1455–1469, 2007. (Cited on page 9.)

[57] D. K. Geller and D. Christensen, Linear covariance analysis for powered lunar descent and landing, *Journal of Spacecraft and Rockets*, vol. 46, no. 6, pp. 1231–1248, 2009. (Cited on page 9.)

[58] D. K. Geller, M. B. Rose, and D. C. Woffinden, Event triggers in linear covariance analysis with applications to orbital rendezvous, *Journal of Guidance, Control, and Dynamics*, vol. 32, no. 1, pp. 102–111, 2009. (Cited on page 9.)

[59] N. Leiter and D. K. Geller, Linear covariance techniques for closed-loop attitude determination and control analysis, *Itzhack Y. Bar-Itzhack Memorial Symposium on Estimation, Navigation, and Spacecraft Control*, Springer, 2012, pp. 453–476. (Cited on page 9.)

[60] K. Jin, D. Geller, and J. Luo, Development and validation of linear covariance analysis tool for atmospheric entry, *Journal of Spacecraft and Rockets*, vol. 56, no. 3, pp. 854–864, 2018. (Cited on page 9.)

[61] D. J. Mook and J. L. Junkins, Minimum model error estimation for poorly modeled dynamic systems, *Journal of Guidance, Control, and Dynamics*, vol. 11, no. 3, pp. 256–261, 1988. (Cited on page 9.)

[62] P. P. Rao and S. C. Bell, Conditional performance error covariance analyses for commercial Titan launch vehicles, *Journal of Guidance, Control, and Dynamics*, vol. 14, no. 2, pp. 398–405, 1991. (Cited on page 9.)

[63] F. L. Markley and J. R. Carpenter, Generalized linear covariance analysis, *The Journal of the Astronautical Sciences*, vol. 57, no. 1-2, pp. 233–260, 2009. (Cited on page 9.)

[64] J. L. Junkins and P. Singla, How nonlinear is it? a tutorial on nonlinearity of orbit and attitude dynamics, *Journal of the Astronautical Sciences*, vol. 52, no. 1-2, pp. 7–60, 2004. (Cited on pages 9, 55, and 60.)

[65] D. J. Gondelach, Orbit prediction and analysis for space situational awareness, Ph.D. dissertation, University of Surrey, 2019. (Cited on page 9.)

[66] S. H. Park, Nonlinear trajectory navigation, Ph.D. dissertation, Pennsylvania State University, 2007. (Cited on page 9.)

[67] A. J. Rosengren, J. Daquin, K. Tsiganis, E. M. Alessi, F. Deleflie, A. Rossi, and G. B. Valsecchi, Galileo disposal strategy: stability, chaos and predictability, *Monthly Notices of the Royal Astronomical Society*, vol. 464, no. 4, pp. 4063–4076, 2016. (Cited

on page 9.)

[68] D. Pérez-Palau, Dynamical transport mechanisms in celestial mechanics and astrodynamics problems, Ph.D. dissertation, Universitat de Barcelona, 2016. (Cited on pages 9, 21, and 44.)

[69] D. Scheeres, F.-Y. Hsiao, R. Park, B. Villac, and J. Maruskin, Fundamental limits on spacecraft orbit uncertainty and distribution propagation, *The Journal of the Astronautical Sciences*, vol. 54, no. 3-4, pp. 505–523, 2006. (Cited on page 9.)

[70] D. J. Scheeres, M. A. de Gosson, and J. Maruskin, Applications of symplectic topology to orbit uncertainty and spacecraft navigation, *The Journal of the Astronautical Sciences*, vol. 59, no. 1-2, pp. 63–83, 2012. (Cited on page 9.)

[71] S. J. Julier, J. K. Uhlmann, and H. F. Durrant-Whyte, A new approach for filtering nonlinear systems, *Proceedings of 1995 American Control Conference-ACC'95*, vol. 3, IEEE, pp. 1628–1632, 1995. (Cited on page 10.)

[72] J. K. Uhlmann, Simultaneous map building and localization for real time applications, Ph.D. dissertation, University of Oxford, 1994. (Cited on pages 10 and 13.)

[73] S. Julier, J. Uhlmann, and H. F. Durrant-Whyte, A new method for the nonlinear transformation of means and covariances in filters and estimators, *IEEE Transactions on Automatic Control*, vol. 45, no. 3, pp. 477–482, 2000. (Cited on pages 10 and 13.)

[74] R. G. Ghanem and P. D. Spanos, *Stochastic finite elements: a spectral approach*, New York: Springer, 1991. (Cited on page 10.)

[75] B. A. Jones, A. Doostan, and G. H. Born, Nonlinear propagation of orbit uncertainty using non-intrusive polynomial chaos, *Journal of Guidance, Control, and Dynamics*, vol. 36, no. 2, pp. 430–444, 2013. (Cited on page 10.)

[76] X. Wan and G. E. Karniadakis, An adaptive multi-element generalized polynomial chaos method for stochastic differential equations, *Journal of Computational Physics*, vol. 209, no. 2, pp. 617–642, 2005. (Cited on page 10.)

[77] F. Nobile, R. Tempone, and C. G. Webster, A sparse grid stochastic collocation method for partial differential equations with random input data, *SIAM Journal on Numerical Analysis*, vol. 46, no. 5, pp. 2309–2345, 2008. (Cited on page 10.)

[78] J. Hampton and A. Doostan, Compressive sampling of polynomial chaos expansions: Convergence analysis and sampling strategies, *Journal of Computational Physics*, vol. 280, pp. 363–386, 2015. (Cited on page 10.)

[79] R. S. Park and D. J. Scheeres, Nonlinear mapping of Gaussian statistics: theory

and applications to spacecraft trajectory design, *Journal of Guidance, Control, and Dynamics*, vol. 29, no. 6, pp. 1367–1375, 2006. (Cited on pages 10 and 43.)

[80] K. Fujimoto, D. J. Scheeres, and K. T. Alfriend, Analytical nonlinear propagation of uncertainty in the two-body problem, *Journal of Guidance, Control, and Dynamics*, vol. 35, no. 2, pp. 497–509, 2012. (Cited on page 10.)

[81] K. Fujimoto, New methods in optical track association and uncertainty mapping of earth-orbiting objects, Ph.D. dissertation, University of Colorado at Boulder, 2013. (Cited on page 10.)

[82] M. Berz, *Modern map methods in particle beam physics*, Academic Press, vol. 108, 1999. (Cited on page 10.)

[83] E. M. Alessi, A. Farres, A. Vieiro, A. Jorba, and C. Simó, Jet transport and applications to NEOs, *Proceedings of the 1st IAA Planetary Defense Conference, Granada, Spain*, 2009. (Cited on pages 10, 11, 14, and 117.)

[84] R. Barrio, M. Rodríguez, A. Abad, and S. Serrano, Uncertainty propagation or box propagation, *Mathematical and Computer Modelling*, vol. 54, no. 11-12, pp. 2602–2615, 2011. (Cited on page 10.)

[85] A. Abad, R. Barrio, M. Marco-Buzunariz, and M. Rodríguez, Automatic implementation of the numerical Taylor series method: A mathematica and sage approach, *Applied Mathematics and Computation*, vol. 268, pp. 227–245, 2015. (Cited on page 10.)

[86] N. Nakhjiri and B. Villac, Modified picard integrator for spaceflight mechanics, *Journal of Guidance, Control, and Dynamics*, vol. 37, no. 5, pp. 1625–1637, 2014. (Cited on page 10.)

[87] À. Jorba and M. Zou, A software package for the numerical integration of ODEs by means of high-order Taylor methods, *Experimental Mathematics*, vol. 14, no. 1, pp. 99–117, 2005. (Cited on page 11.)

[88] D. J. Gondelach and R. Armellin, Element sets for high-order Poincaré mapping of perturbed Keplerian motion, *Celestial Mechanics and Dynamical Astronomy*, vol. 130, no. 10, pp. 65–100, 2018. (Cited on page 11.)

[89] M. Valli, R. Armellin, P. di Lizia, and M. R. Lavagna, Nonlinear mapping of uncertainties in celestial mechanics, *Journal of Guidance, Control, and Dynamics*, vol. 36, no. 1, pp. 48–63, 2012. (Cited on page 11.)

[90] J. Chen, J. J. Masdemont, G. Gómez, and J. Yuan, Analysis of jet transport-based geostationary trajectory uncertainty propagation, *Journal of Guidance, Control, and Dynamics*, vol. 43, no. 6, pp. 1210–1219, 2020. (Cited on pages 11 and 131.)

[91] A. Wittig, P. Di Lizia, R. Armellin, K. Makino, F. Bernelli-Zazzera, and M. Berz, Propagation of large uncertainty sets in orbital dynamics by automatic domain splitting, *Celestial Mechanics and Dynamical Astronomy*, vol. 122, no. 3, pp. 239–261, 2015. (Cited on pages 11 and 66.)

[92] D. Pérez-Palau, G. Gómez, and J. J. Masdemont, A new subdivision algorithm for the flow propagation using polynomial algebras, *Communications in Nonlinear Science and Numerical Simulation*, vol. 61, pp. 37–53, 2018. (Cited on pages 11 and 66.)

[93] A. Morselli, R. Armellin, P. Di Lizia, and F. B. Zazzera, A high order method for orbital conjunctions analysis: Monte Carlo collision probability computation, *Advances in Space Research*, vol. 55, no. 1, pp. 311–333, 2015. (Cited on page 11.)

[94] A. Riccardi, C. Tardioli, and M. Vasile, An intrusive approach to uncertainty propagation in orbital mechanics based on tchebycheff polynomial algebra, *Proceedings of the AAS/AIAA Astrodynamics Specialist Conference*, Vail, Colorado: American Astronautical Society, August 2015. (Cited on pages 11, 26, and 39.)

[95] M. Vasile, C. O. Absil, and A. Riccardi, Set propagation in dynamical systems with generalised polynomial algebra and its computational complexity, *Communications in Nonlinear Science and Numerical Simulation*, vol. 75, pp. 22–49, 2019. (Cited on pages 11, 26, and 34.)

[96] M. Kumar, S. Chakravorty, and J. L. Junkins, A homotopic approach to domain determination and solution refinement for the stationary Fokker–Plank equation, *Probabilistic engineering mechanics*, vol. 24, no. 3, pp. 265–277, 2009. (Cited on page 11.)

[97] M. Kumar, S. Chakravorty, and J. L. Junkins, A semianalytic meshless approach to the transient Fokker–Plank equation, *Probabilistic Engineering Mechanics*, vol. 25, no. 3, pp. 323–331, 2010. (Cited on page 11.)

[98] Y. Sun and M. Kumar, Numerical solution of high dimensional stationary Fokker–Plank equations via tensor decomposition and Chebyshev spectral differentiation, *Computers & Mathematics with Applications*, vol. 67, no. 10, pp. 1960–1977, 2014. (Cited on page 11.)

[99] Y. Sun and M. Kumar, A numerical solver for high dimensional transient Fokker–Plank equation in modeling polymeric fluids, *Journal of Computational Physics*, vol. 289, pp. 149–168, 2015. (Cited on page 11.)

[100] Y. Sun and M. Kumar, Uncertainty propagation in orbital mechanics via tensor decomposition, *Celestial Mechanics and Dynamical Astronomy*, vol. 124, no. 3, pp. 269–294, 2016. (Cited on page 11.)

[101] D. Alspach and H. Sorenson, Nonlinear Bayesian estimation using Gaussian sum

approximations, *IEEE Transactions on Automatic Control*, vol. 17, no. 4, pp. 439–448, 1972. (Cited on page 11.)

[102] G. Terejanu, P. Singla, T. Singh, and P. D. Scott, Uncertainty propagation for nonlinear dynamic systems using Gaussian mixture models, *Journal of Guidance, Control, and Dynamics*, vol. 31, no. 6, pp. 1623–1633, 2008. (Cited on page 11.)

[103] K. Vishwajeet, P. Singla, and M. Jah, Nonlinear uncertainty propagation for perturbed two-body orbits, *Journal of Guidance, Control, and Dynamics*, vol. 37, no. 5, pp. 1415–1425, 2014. (Cited on page 11.)

[104] K. Vishwajeet and P. Singla, Adaptive split/merge-based Gaussian mixture model approach for uncertainty propagation, *Journal of Guidance, Control, and Dynamics*, vol. 41, no. 3, pp. 603–617, 2018. (Cited on page 11.)

[105] K. Tuggle and R. Zanetti, Automated splitting Gaussian mixture nonlinear measurement update, *Journal of Guidance, Control, and Dynamics*, vol. 41, no. 3, pp. 725–734, 2018. (Cited on page 11.)

[106] K. Fujimoto and D. Scheeres, Tractable expressions for nonlinearly propagated uncertainties, *Journal of Guidance, Control, and Dynamics*, vol. 38, no. 6, pp. 1146–1151, 2015. (Cited on page 11.)

[107] I. Park and D. J. Scheeres, Hybrid method for uncertainty propagation of orbital motion, *Journal of Guidance, Control, and Dynamics*, vol. 41, no. 1, pp. 240–254, 2018. (Cited on page 11.)

[108] H. Peng and X. Bai, Comparative evaluation of three machine learning algorithms on improving orbit prediction accuracy, *Astrodynamics*, vol. 3, no. 4, pp. 325–343, 2019. (Cited on page 12.)

[109] H. Peng and X. Bai, Machine learning approach to improve satellite orbit prediction accuracy using publicly available data, *The Journal of the Astronautical Sciences*, vol. 67, no. 2, pp. 762–793, 2020. (Cited on page 12.)

[110] H. Peng and X. Bai, Artificial neural network–based machine learning approach to improve orbit prediction accuracy, *Journal of Spacecraft and Rockets*, vol. 55, no. 5, pp. 1248–1260, 2018. (Cited on page 12.)

[111] H. Peng and X. Bai, Gaussian processes for improving orbit prediction accuracy, *Acta Astronautica*, vol. 161, pp. 44–56, 2019. (Cited on page 12.)

[112] R. E. Kalman, A new approach to linear filtering and prediction problems, *Journal of basic Engineering*, vol. 82, no. 1, pp. 35–45, 1960. (Cited on pages 12 and 13.)

[113] R. E. Kalman and R. S. Bucy, New results in linear filtering and prediction theory, *Journal of basic Engineering*, vol. 83, no. 1, pp. 95–108, 1961. (Cited on page 12.)

[114] D. Simon and T. L. Chia, Kalman filtering with state equality constraints, *IEEE Transactions on Aerospace and Electronic Systems*, vol. 38, no. 1, pp. 128–136, 2002. (Cited on page 12.)

[115] D. Simon, *Optimal state estimation: Kalman, H infinity, and nonlinear approaches*, John Wiley & Sons, 2006. (Cited on page 12.)

[116] V. Pesce, Autonomous navigation for close proximity operations around uncooperative space objects, Ph.D. dissertation, Politecnico di Milano, 2019. (Cited on page 12.)

[117] G. Einicke, *Smoothing, filtering and prediction: Estimating the past, present and future*, InTech, 2012. (Cited on page 12.)

[118] F. Auger, M. Hilairet, J. M. Guerrero, E. Monmasson, T. Orlowska-Kowalska, and S. Katsura, Industrial applications of the Kalman filter: A review, *IEEE Transactions on Industrial Electronics*, vol. 60, no. 12, pp. 5458–5471, 2013. (Cited on page 12.)

[119] J. Bellantoni and K. Dodge, A square root formulation of the Kalman-Schmidt filter. *AIAA journal*, vol. 5, no. 7, pp. 1309–1314, 1967. (Cited on page 13.)

[120] I. Arasaratnam and S. Haykin, Cubature Kalman filters, *IEEE Transactions on Automatic Control*, vol. 54, no. 6, pp. 1254–1269, 2009. (Cited on page 13.)

[121] X. Zhang and C. Guo, Cubature Kalman filters: Derivation and extension, *Chinese Physics B*, vol. 22, no. 12, p. 128401, 2013. (Cited on page 13.)

[122] R. Madankan, P. Singla, T. Singh, and P. D. Scott, Polynomial-chaos-based Bayesian approach for state and parameter estimations, *Journal of Guidance, Control, and Dynamics*, vol. 36, no. 4, pp. 1058–1074, 2013. (Cited on pages 13 and 14.)

[123] R. S. Park and D. J. Scheeres, Nonlinear semi-analytic methods for trajectory estimation, *Journal of Guidance, Control, and Dynamics*, vol. 30, no. 6, pp. 1668–1676, 2007. (Cited on pages 13, 14, and 75.)

[124] M. Majji, J. L. Junkins, and J. D. Turner, A high order method for estimation of dynamic systems, *The Journal of the Astronautical Sciences*, vol. 56, no. 3, pp. 401–440, 2008. (Cited on pages 13 and 14.)

[125] A. Germani, C. Manes, and P. Palumbo, State estimation of stochastic systems with switching measurements: a polynomial approach, *International Journal of Robust and Nonlinear Control: IFAC-Affiliated Journal*, vol. 19, no. 14, pp. 1632–1655, 2009. (Cited on pages 13 and 14.)

[126] J. T. Horwood, N. D. Aragon, and A. B. Poore, Gaussian sum filters for space surveillance: theory and simulations, *Journal of Guidance, Control, and Dynamics*, vol. 34, no. 6, pp. 1839–1851, 2011. (Cited on pages 13 and 14.)

[127] Y. Sun and M. Kumar, Nonlinear Bayesian filtering based on Fokker–Planck equation and tensor decomposition, *2015 18th International Conference on Information Fusion*, IEEE, 2015, pp. 1483–1488. (Cited on pages 13 and 14.)

[128] X. Ning, X. Ma, C. Peng, W. Quan, and J. Fang, Analysis of filtering methods for satellite autonomous orbit determination using celestial and geomagnetic measurement, *Mathematical Problems in Engineering*, vol. 2012, 2012. (Cited on page 13.)

[129] G. Biondi, Fault-tolerant feature-based estimation of space debris motion and inertial properties, Ph.D. dissertation, Politecnico di Milano, 2019. (Cited on page 13.)

[130] D. J. Lee, Nonlinear Bayesian filtering with applications to estimation and navigation, Ph.D. dissertation, Texas A&M University, 2005. (Cited on page 13.)

[131] E. A. Wan and R. Van Der Merwe, The unscented Kalman filter for nonlinear estimation, *Proceedings of the IEEE 2000 Adaptive Systems for Signal Processing, Communications, and Control Symposium*, IEEE, pp. 153–158, 2000. (Cited on page 13.)

[132] L. Zhang, T. Li, H. Yang, S. Zhang, H. Cai, and S. Qian, Unscented Kalman filtering for relative spacecraft attitude and position estimation," *The Journal of Navigation*, vol. 68, no. 3, pp. 528–548, 2015. (Cited on page 13.)

[133] H. Nourmohammadi and J. Keighobadi, Decentralized INS/GNSS system with MEMS-grade inertial sensors using QR-factorized CKF, *IEEE Sensors Journal*, vol. 17, no. 11, pp. 3278–3287, 2017. (Cited on page 13.)

[134] S. Wang, J. Feng, and K. T. Chi, Novel Cubature Kalman filtering for systems involving nonlinear states and linear measurements, *AEU-International Journal of Electronics and Communications*, vol. 69, no. 1, pp. 314–320, 2015. (Cited on page 13.)

[135] L. Chang, B. Hu, A. Li, and F. Qin, Transformed unscented Kalman filter, *IEEE Transactions on Automatic Control*, vol. 58, no. 1, pp. 252–257, 2012. (Cited on page 14.)

[136] P. G. Medewar, M. Yadav, and H. G. Patel, A comparison between nonlinear estimation based algorithms for mobile robot localizations, *2019 IEEE 1st International Conference on Energy, Systems and Information Processing (ICESIP)*, IEEE, pp. 1–6, 2019. (Cited on page 14.)

[137] R. Garcia, P. Pardal, H. Kuga, and M. Zanardi, Nonlinear filtering for sequential spacecraft attitude estimation with real data: Cubature Kalman filter, unscented

Kalman filter and extended Kalman filter, *Advances in Space Research*, vol. 63, no. 2, pp. 1038–1050, 2019. (Cited on page 14.)

[138] M. Valli, R. Armellin, P. di Lizia, and M. R. Lavagna, Nonlinear filtering methods for spacecraft navigation based on differential algebra, *Acta Astronautica*, vol. 94, no. 1, pp. 363 – 374, 2014. (Cited on pages 14 and 117.)

[139] F. Cavenago, P. Di Lizia, M. Massari, and A. Wittig, On-board spacecraft relative pose estimation with high-order extended Kalman filter, *Acta Astronautica*, vol. 158, pp. 55–67, 2019. (Cited on pages 14 and 117.)

[140] N. J. Gordon, D. J. Salmond, and A. F. Smith, Novel approach to nonlinear/non-Gaussian Bayesian state estimation, *IEE proceedings F (radar and signal processing)*, vol. 140, no. 2, IET, 1993, pp. 107–113. (Cited on page 14.)

[141] M. S. Sharifian, A. Rahimi, and N. Pariz, Classifying the weights of particle filters in nonlinear systems, *Communications in Nonlinear Science and Numerical Simulation*, vol. 31, no. 1-3, pp. 69–75, 2016. (Cited on page 14.)

[142] A. Doucet, N. De Freitas, and N. Gordon, An introduction to sequential Monte Carlo methods, *Sequential Monte Carlo methods in practice*, Springer, 2001, pp. 3–14. (Cited on page 15.)

[143] M. Kiani and S. H. Pourtakdoust, Adaptive square-root cubature-quadrature Kalman particle filter via KLD-sampling for orbit determination, *Aerospace Science and Technology*, vol. 46, pp. 159–167, 2015. (Cited on page 15.)

[144] F. Weimer, M. Frangenberg, and W. Fichter, Pipelined particle filter with nonobservability measure for attitude and velocity estimation, *Journal of Guidance, Control, and Dynamics*, vol. 38, no. 3, pp. 506–518, 2015. (Cited on page 15.)

[145] C. Ma, Z. Zheng, J. Chen, and J. Yuan, Jet transport particle filter for attitude estimation of tumbling space objects, *Aerospace Science and Technology*, vol. 107, no. 1, 2020. (Cited on page 15.)

[146] A. Jorba, A methodology for the numerical computation of normal forms, centre manifolds and first integrals of Hamiltonian systems, *Experimental Mathematics*, vol. 8, no. 2, pp. 155–195, 1999. (Cited on page 21.)

[147] K. Makino and M. Berz, Taylor models and other validated functional inclusion methods, *International Journal of Pure and Applied Mathematics*, vol. 6, pp. 239–316, 2003. (Cited on page 26.)

[148] P. Giorgi, On polynomial multiplication in Chebyshev basis, *IEEE Transactions on Computers*, vol. 61, no. 6, pp. 780–789, 2011. (Cited on page 26.)

[149] N. Brisebarre and M. Joldeş, Chebyshev interpolation polynomial-based tools for rigorous computing, *Proceedings of the 2010 International Symposium on Symbolic and Algebraic Computation*, ACM, pp. 147–154, 2010. (Cited on pages 26, 30, and 34.)

[150] E. W. Cheney, *Introduction to approximation theory*, McGraw-Hill New York, vol. 6, 1966. (Cited on pages 30 and 32.)

[151] R. E. Moore, *Methods and applications of interval analysis*, SIAM, 1979. (Cited on page 30.)

[152] J. P. Boyd, *Chebyshev and Fourier spectral methods*, Courier Corporation, 2001. (Cited on page 31.)

[153] C. Epstein, W. Miranker, and T. Rivlin, Ultra-arithmetic I: function data types, *Mathematics and Computers in Simulation*, vol. 24, no. 1, pp. 1–18, 1982. (Cited on page 31.)

[154] R. B. Platte and L. N. Trefethen, Chebfun: a new kind of numerical computing, *Progress in industrial mathematics at ECMI 2008*, Springer, pp. 69–87, 2010. (Cited on page 32.)

[155] E. M. Soop, *Handbook of geostationary orbits*, Kluwer Academic, Dordrecht, The Netherlands, chap. 2 & 3, 1994. (Cited on page 54.)

[156] J. Tombasco, Orbit estimation of geosynchronous objects via ground-based and space-based optical tracking, Ph.D. dissertation, University of Colorado at Boulder, 2011. (Cited on pages 54 and 76.)

[157] J. Tombasco, P. Axelrad, and M. Jah, Specialized coordinate representation for dynamic modeling and orbit estimation of geosynchronous orbits, *Journal of Guidance, Control, and Dynamics*, vol. 33, no. 6, pp. 1824–1836, 2010. (Cited on pages 54, 55, and 136.)

[158] D. J. Gondelach and R. Armellin, Element sets for high-order Poincaré mapping of perturbed Keplerian motion, *Celestial Mechanics and Dynamical Astronomy*, vol. 130, no. 10, p. 65, 2018. (Cited on page 54.)

[159] J. Chen, J. J. Masdemont, G. Gómez, and J. Yuan, Jet transport-based nonlinear state and parameter estimation for geostationary spacecraft, *Acta Astronautica*, vol. 164, pp. 321–333, 2019. (Cited on pages 58, 117, and 132.)

[160] L. Isserlis, On a formula for the product-moment coefficient of any order of a normal frequency distribution in any number of variables, *Biometrika*, vol. 12, no. 1/2, pp. 134–139, 1918. (Cited on page 75.)

[161] A. T. Tokunaga, New generation ground-based optical/infrared telescopes, in *Encyclopedia of the Solar System (Third Edition)*, Elsevier, pp. 1089–1105, 2014. (Cited on page 76.)

[162] C. Tang, X. Hu, S. Zhou, R. Guo *et al.*, Improvement of orbit determination accuracy for Beidou navigation satellite system with two-way satellite time frequency transfer, *Advances in Space Research*, vol. 58, no. 7, pp. 1390–1400, 2016. (Cited on page 77.)

[163] Y. Bar-Shalom, X. Li, and T. Kirubarajan, *Estimation with applications to tracking and navigation: theory algorithms and software*, John Wiley & Sons, 2004. (Cited on pages 94, 95, 108, and 118.)

[164] P. Teunissen, Quality control in integrated navigation systems, *IEEE Aerospace and Electronic Systems Magazine*, vol. 5, no. 7, pp. 35–41, 1990. (Cited on page 94.)

[165] D. Feng, H. Guo, X. Wang, and X. Yuan, Autonomous orbit determination and its error analysis for deep space using x-ray pulsar, *Aerospace Science and Technology*, vol. 32, no. 1, pp. 35–41, 2014. (Cited on page 99.)

[166] J. Deutschmann, R. Harman, and I. Bar-Itzhack, An innovative method for low cost, autonomous navigation for low earth orbit satellites, *AIAA/AAS Astrodynamics Specialist Conference and Exhibit*, 1998. (Cited on page 99.)

[167] M. Kiani, Robust integrated orbit and attitude estimation using geophysical data, *Aerospace Science and Technology*, vol. 93, p. 105307, 2019. (Cited on page 99.)

[168] F. N. Gourabi, M. Kiani, and S. H. Pourtakdoust, Autonomous temperature-based orbit estimation, *Aerospace Science and Technology*, vol. 86, pp. 671–682, 2019. (Cited on page 99.)

[169] P. Stoica and K. C. Sharman, Maximum likelihood methods for direction-of-arrival estimation, *IEEE Transactions on Acoustics, Speech, and Signal Processing*, vol. 38, no. 7, pp. 1132–1143, 1990. (Cited on page 99.)

[170] X. Zhang, M. Zhou, H. Chen, and J. Li, Two-dimensional DOA estimation for acoustic vector-sensor array using a successive music, *Multidimensional Systems and Signal Processing*, vol. 25, no. 3, pp. 583–600, 2014. (Cited on page 99.)

[171] J. Silha, T. Schildknecht, A. Hinze, J. Utzmann, A. Wagner, P. Willemsen, F. Teston, and T. Flohrer, Capability of a space-based space surveillance system to detect and track objects in GEO, MEO and LEO orbits, *65th International Astronautical Congress*, IAF, 2014. (Cited on pages 99 and 100.)

[172] T. Flohrer, H. Krag, H. Klinkrad, and T. Schildknecht, Feasibility of performing

space surveillance tasks with a proposed space-based optical architecture, *Advances in Space Research*, vol. 47, no. 6, pp. 1029–1042, 2011. (Cited on page 99.)

[173] Y. Hu, X. Bai, L. Chen, and H. Yan, A new approach of orbit determination for LEO satellites based on optical tracking of GEO satellites, *Aerospace Science and Technology*, vol. 84, pp. 821–829, 2019. (Cited on page 99.)

[174] C. Sabol, K. T. Alfriend, and D. Wiese, Angles-only orbit updates for low-earth-orbit satellites, *Journal of Guidance, Control, and Dynamics*, vol. 30, no. 2, pp. 314–321, 2007. (Cited on page 100.)

[175] J. Chen, J. J. Masdemont, G. Gómez, and J. Yuan, High accuracy state and parameter estimation of geostationary satellites using jet transport based nonlinear filtering algorithm, *IAC Papers Archive*, pp. 1–12, 2018. (Cited on page 132.)

[176] J. Chen, J. J. Masdemont, G. Gómez, and J. Yuan, A polynomial-based autonomous orbit determination technique and its three fault-tolerant designs, *Submitted to Aerospace Science and Technology*. (Cited on page 133.)

[177] J. Chen, J. J. Masdemont, G. Gómez, and J. Yuan, An efficient statistical adaptive order-switching methodology for Kalman filters, *Communications in Nonlinear Science and Numerical Simulation*, vol. 93, p. 105539, 2020. (Cited on page 133.)

[178] A. Weisskopf, R. Armellin, and M. Berz, Bounded motion design in the earth zonal problem using differential algebra based normal form methods, *Celestial Mechanics and Dynamical Astronomy*, vol. 132, no. 2, pp. 1–32, 2020. (Cited on page 134.)

[179] R. Armellin, D. Gondelach, and J. F. San Juan, Multiple revolution perturbed Lambert problem solvers, *Journal of Guidance, Control, and Dynamics*, vol. 41, no. 9, pp. 2019–2032, 2018. (Cited on page 134.)

[180] P. Di Lizia, R. Armellin, A. Morselli, and F. B. Zazzera, High order optimal feedback control of space trajectories with bounded control, *Acta Astronautica*, vol. 94, no. 1, pp. 383–394, 2014. (Cited on page 134.)