TU Delft

Delft University of Technology

Quantification of flyby effects in the three-body problem using the Gaussian process method

Liu, Y.

DOI
10.4233/uuid:20dd1357-2c56-446d-9635-d60edf2c0bd1

Publication date
2021

Document Version
Final published version

Citation (APA)
Liu, Y. (2021). *Quantification of flyby effects in the three-body problem using the Gaussian process method*. [Dissertation (TU Delft), Delft University of Technology]. https://doi.org/10.4233/uuid:20dd1357-2c56-446d-9635-d60edf2c0bd1

Important note
To cite this publication, please use the final published version (if applicable).
Please check the document version above.

Copyright
Other than for strictly personal use, it is not permitted to download, forward or distribute the text or part of it, without the consent of the author(s) and/or copyright holder(s), unless the work is under an open content license such as Creative Commons.

Takedown policy
Please contact us and provide details if you believe this document breaches copyrights.
We will remove access to the work immediately and investigate your claim.

This work is downloaded from Delft University of Technology.
For technical reasons the number of authors shown on this cover page is limited to a maximum of 10.

QUANTIFICATION OF FLYBY EFFECTS IN THE THREE-BODY PROBLEM USING THE GAUSSIAN PROCESS METHOD

QUANTIFICATION OF FLYBY EFFECTS IN THE THREE-BODY PROBLEM USING THE GAUSSIAN PROCESS METHOD

Dissertation

for the purpose of obtaining the degree of doctor
at Delft University of Technology,
by the authority of the Rector Magnificus prof. dr. ir. T.H.J.J. van der Hagen,
chair of the Board for Doctorates
to be defended publicly on
Monday 1 November 2021 at 17:30 o'clock

by

Yuxin LIU

Master of Engineering in Aeronautical and Astronautical Science & Technology
Beijing Institute of Technology, China
born in Harbin, China

This dissertation has been approved by the promotor.

composition of the doctoral committee:

Rector Magnificus	Chairperson
Prof. dr. ir. P.N.A.M. Visser,	Delft University of Technology, promotor
Ir. R. Noomen,	Delft University of Technology, supervisor

Independent members:

Dr. B. Jones	The University of Texas at Austin
Dr. S. Campagnola	JPL, NASA
Prof. dr. M. Vasile	University of Strathclyde
Prof. dr. G.C.H.E. de Croon	Delft University of Technology
Prof. dr. ir. G. Jongbloed	Delft University of Technology
Prof. dr. E.K.A. Gill	Delft University of Technology, reserve member

The realization of the dissertation was partially sponsored by the China Scholarship Council.

Keywords: Gravity Assists; Circular Restricted Three-Body Problem; Gaussian Process Method; Gravity Assist Mapping; Jacobi Constant;

Printed by: Ipskamp

Copyright © 2021 by Yuxin LIU

ISBN 000-00-0000-000-0

An electronic version of this dissertation is available at
http://repository.tudelft.nl/.

CONTENTS

Summary vii

Samenvatting xi

1 Introduction 1
- 1.1 Gravity Assist . . . 1
- 1.2 Circular Restricted Three-Body Problem . . . 3
- 1.3 Methods to Study Flybys . . . 6
  - 1.3.1 Semi-analytical methods . . . 6
  - 1.3.2 Numerical methods . . . 8
- 1.4 Machine Learning based on Gaussian Process . . . 9
- 1.5 Research Motivation . . . 12
- 1.6 Outline . . . 14

2 A Gravity Assist Mapping Based on Gaussian Process Regression 17
- 2.1 Introduction . . . 17
- 2.2 Planar Circular Restricted Three-Body Problem . . . 19
- 2.3 Gravity Assist Mapping . . . 19
  - 2.3.1 GPR fundamentals . . . 20
  - 2.3.2 GPR Covariance Function . . . 22
  - 2.3.3 Training Samples . . . 24
- 2.4 Performance of GPR-based Gravity Assist Mapping . . . 26
  - 2.4.1 Accuracy . . . 28
  - 2.4.2 Flybys outside the sphere of influence of the secondary . . . 31
  - 2.4.3 Flybys outside the Hill sphere of the secondary . . . 33
- 2.5 Conclusions . . . 35
- 2.6 Acknowledgement . . . 36
- 2.7 Appendix A: Minimum distance between spacecraft and secondary . . . 36
- 2.8 Appendix B: Values of hyper-parameters . . . 37

3 A Gravity Assist Mapping for the Circular Restricted Three-Body Problem using Gaussian Processes 39
- 3.1 Introduction . . . 39
- 3.2 Circular Restricted Three-Body Problem . . . 41
  - 3.2.1 Jacobi Constant . . . 42
- 3.3 Gravity Assist Mapping . . . 43
  - 3.3.1 Mapping Function . . . 43
  - 3.3.2 Configuration of flyby . . . 43

3.4 Gaussian Process Regression Model . . . 45
3.4.1 Mean Function . . . 47
3.4.2 Covariance Function . . . 47
3.4.3 Training Samples . . . 49
3.4.4 Accuracy Evaluation and Training Data Size . . . 49
3.5 Performance of GPR-based Gravity Assist Mapping . . . 50
3.5.1 Different Covariance Functions . . . 50
3.5.2 Comparison with Semi-Analytical Method . . . 52
3.5.3 Robustness when Changing Training Dataset . . . 53
3.5.4 Robustness when Changing Test Dataset . . . 56
3.5.5 Low-energy Cases . . . 57
3.6 Conclusions . . . 59
3.7 Acknowledgement . . . 60

4 A Machine-Learning Design Tool for Quantifying Gravity Assist Effects in the Circular Restricted Three-Body Problem 61
4.1 Introduction . . . 62
4.2 Circular Restricted Three-Body Problem . . . 63
4.3 Gravity Assist Mapping . . . 64
4.3.1 Mapping Function . . . 64
4.3.2 Gaussian Process Models . . . 66
4.3.3 Generation of Training Dataset . . . 68
4.3.4 Accuracy Assessment . . . 68
4.4 Performance of Gravity Assist Mapping . . . 69
4.4.1 Sun-(Earth+Moon)-spacecraft . . . 69
4.4.2 Jupiter-Callisto-spacecraft . . . 73
4.4.3 Sun-Jupiter-spacecraft . . . 78
4.4.4 Time efficiency of GAM models . . . 80
4.5 Quality Assessment . . . 83
4.6 Conclusions . . . 85
4.7 Acknowledgement . . . 86

5 Conclusions and recommendations 87
5.1 Conclusions . . . 87
5.2 Recommendations . . . 90

References 93

Acknowledgements 99

Curriculum Vitae 101

List of Publications 103

SUMMARY

The gravity assist (GA) plays an important role in space missions since it was first applied by the Luna 3 vehicle in 1959. For preliminary trajectory design, the so-called patched-conics model provides a simple model for a gravity assist. This approach, based on two-body formulations, splits a multi-body problem into a succession of two-body problems. This model has a fundamental assumption: the trajectory of the spacecraft is driven by one celestial body only. A boundary for switching the driving bodies is defined by the Sphere of Influence (SoI) of the GA body. The patched conics model cannot be used to study low-energy trajectories. Moreover, it fails to describe special dynamics existing in the multi-body regime, such as the invariant manifolds. The three-body formulation is a logical choice to study the dynamics in the multi-body problem. In order to reduce its inherent difficulty, the circular restricted three-body problem (CR3BP) formulation is developed to study the behavior of the motion of a particle influenced by two massive bodies simultaneously. Flybys in the CR3BP have been studied by many researchers, using a numerical or semi-analytical approach, e.g. the Flyby map (FM) and Keplerian map (KM), respectively. Inspired by these approaches and the idea of artificial intelligence, this thesis focuses on the investigation of flybys from a machine-learning perspective.

With this background, this thesis presents a method to quantify the changes in orbital elements after a GA: the Gravity Assist Mapping (GAM). The GAM model is based on the Gaussian Process method, a supervised machine-learning technique. It can be divided into two types: Gaussian Process Regression (GPR) for continuous outputs and Gaussian Process Classification (GPC) for discrete outputs. Both types are employed to solve a specific problem in this research: the GPR model is used for the quantification of flyby effects, and the GPC model is used for the identification of collision trajectories.

The development of GAM models is done in the regime of the CR3BP. In order to reduce the complexity, a first development is made for the planar situation. In particular, the first problem formulation considers the Sun-(Earth+Moon)-spacecraft system. It investigates the effect of a flyby that occurs anywhere above 300 km altitude at the Earth i.e., the secondary, not restricted to the SoI or the Hill sphere. In contrast, the Keplerian map, a semi-analytical method to describe dynamics in the CR3BP, focuses only on the cases outside the Hill sphere with a Jacobi constant around 3.0 and has poor accuracy for high three-body energies. Any GAM model is driven by such training samples; the model learns the effects of flyby's from training samples, which are generated by numerical propagation here. The influence of the number of training samples and their distribution on the quality of the prediction of post-flyby orbital states is analyzed. The results show that the GAM model is able to make predictions of the post-flyby state of an object with an arbitrary initial condition. It is demonstrated to be efficient and accurate when evaluated against the results of numerical integration. The proposed method is applicable to not only high-energy cases but also low-energy cases. The CPU time of a single prediction using GAM is only 1.16×10^{-6} s, whereas the KM method costs 0.064 s (desk-

top PC, Core i7 CPU and 8.00 GB RAM). For the Sun-(Earth+Moon)-spacecraft system, the mean absolute error of predicting the variation of the semi-major axis is 3.86×10^{-5} AU (a typical range of changes is -0.0115 AU to 0.0163 AU).

After successfully constructing the GAM model for the planar case, the model is extended to the fully spatial (3-dimensional) CR3BP. In terms of input space, the complexity of the problem is increased from three parameters to five parameters. A new mapping function for quantifying the flyby effects over one orbital revolution is defined. The GPR model is established by proper mean and covariance functions. To improve the efficiency of this method, a criterion in terms of the mean absolute error is proposed to determine the optimal size of the training dataset. Its robustness is investigated and discussed, to show the quality for practical usage. The influence of different input elements (covariance function, training dataset size, quasi-random numbers) on the flyby effects is studied. The accuracy and efficiency of the proposed model have been investigated for different energy levels, ranging from representative high- to low-energy cases. The corresponding value of the Jacobi constant, from 3.0012 to 0.5122, represents a wide energy range, underlying the broad pontential applications of the technique (unlike that of previous research, such as by Ross & Scheeres (2007) and Alessi & Sánchez (2016)). The GAM model shows significant improvements over an independent semi-analytical method (the KM developed by Alessi & Sánchez (2016)): the accuracy of predicting the variation of the semi-major axis is improved by a factor 3.3, whereas the efficiency goes up by a factor 1.27×10^4.

The above GAM models for the planar and fully spatial cases both employ the GPR, i.e., the output is a continuous function of flyby effects. The GPR makes a prediction for any initial condition it receives. However, it is not possible for the GPR to identify trajectories that lead to a collision with the secondary. Therefore, a classification tool is necessary to identify such collision cases. The GPC model is developed for this purpose, also based on the GP concept. When generating training samples using the numerical propagation of the CR3BP equations of motion, the data contain both collision trajectories and safe trajectories. At first, all training data are fed to the GPC model. After training, this GPC model is able to identify the impact trajectories given an arbitrary initial condition. The accuracy of detecting impact trajectories using GPC is better than 90%. The GPC model can be used stand-alone for collision detection, or combined with the GPR model. The distribution of collision cases predicted by the GPC model closely resembles the results obtained by the CR3BP propagation.

The GPR model is further improved by introducing the Jacobi constant as an extra input feature. The Jacobi constant represents the three-body energy in the CR3BP and provides valuable information about the dynamics of transfers. The performance of the models is analyzed in three systems: Sun-(Earth+Moon)-spacecraft, Jupiter-Callisto-spacecraft and Sun-Jupiter-spacecraft. With these cases, the value of the mass ratio μ increases successively by a factor of about 10. The results demonstrate that the evaluation of flyby effects can be efficiently and accurately made combining the GPC (i.e., classification) and GPR (i.e., regression) models. The proposed GAM approach is proven to be applicable to all studied CR3BP cases. For the regression part, more training samples are required for the CR3BP with bigger μ. The mean absolute error of predicting the change in semi-major axis is only 3.86×10^{-5} AU with a computational cost of 5.8×10^{-6} s per

sample (the Sun-(Earth+Moon)-spacecraft problem). Adding the Jacobi constant as an extra input element improves the accuracy of the prediction for all systems significantly (typically by 20%). The uncertainty associated with the predictions is also analyzed. It turns out that the outputs obtained by the CR3BP propagation are well included in the 95 % confidence intervals of the GPR predictions.

The GAM approach developed in this thesis can be used in the preliminary design of space missions. The high accuracy and efficiency make it a useful tool for quantifying flyby effects, especially if numerous initial conditions are to be evaluated.

SAMENVATTING

De gravity-assist (GA, jargon voor scheervluchten met een grote rol voor zwaartekracht-interactie) speelt een belangrijke rol bij ruimtevaart missies. Deze techniek werd voor het eerst toegepast bij Luna 3 in 1959. Een simpel model hiervoor kan worden beschreven door opeenvolgende kegelsneden. Dit simpele model splitst het probleem van de beweging onder invloed van de zwaartekracht van meerdere objecten op in een serie van dergelijke problemen met ieder slechts twee objecten. Een fundamentele aanname bij dit model is dat het traject van het ruimtevaartuig op ieder moment bepaald wordt door slechts één hemellichaam. Welk hemellichaam dit is, wordt bepaald door zijn invloedsfeer (in het Engels: 'sphere of influence'). Echter, het effect van een dergelijke GA is klein, wanneer het verschil in snelheid tussen het ruimtevaartuig en het hemellichaam ook klein is. Daarnaast lukt het ook niet om met dit model de bijzondere dynamica van het meerlichamenprobleem te beschrijven. Om deze reden is het drielichamenprobleem in het leven geroepen. Om het drielichamenprobleem te vereenvoudigen is het beperkt drielichamenprobleem (CR3BP) ontwikkeld. Hiermee kan het gedrag van de satelliet worden bestudeerd wanneer het krachten ondervindt van twee hemellichamen tegelijkertijd. Scheervluchten in het CR3BP worden vaak in kaart gebracht door gebruik te maken van een numerieke of semi-analytische aanpak (bijvoorbeeld de zgn. Flyby Map en de Keplerian Map). De combinatie van kunstmatige intelligentie en de eerdergenoemde aanpak vormt de basis voor dit proefschrift, dat focust op het onderzoeken van scheervluchten vanuit een machine-learning perspectief.

Dit proefschrift presenteert een methode om de veranderingen in de baanelementen na een GA te kwantificeren, genaamd Gravity Assist Mapping (GAM). De methode achter het GAM model is gebaseerd op Gaussische processen en is een zgn. geverifieerde machine-learning techniek. Deze techniek kan onderverdeeld worden in twee types: regressie van Gaussische processen (GPR), gebruikt voor continue output, en klassificatie van Gaussische processen (GPC) voor discrete output. In dit onderzoek worden beide types toegepast: de regressie wordt gebruikt voor de kwantificatie van flyby effecten en het klassificatie model wordt gebruikt voor het klassificeren van botsingstrajecten.

De ontwikkeling van de GAM modellen wordt gedaan voor het beperkte drielichamenprobleem (CR3BP). Om dit probleem verder te versimpelen begint de studie met een eenvoudige, 2-dimensionale situatie: het platte vlak. Deze situatie beschouwt een systeem bestaande uit een zon, de combinatie van aarde+maan, en een satelliet. We kijken naar de flyby's op een hoogte van meer dan 300 km boven het aardoppervlak. De flyby's langs de aarde+maan (de zgn. secondary) zijn niet beperkt tot de invloedsfeer of de Hill sphere. De Keplerian Map, een semi-analytische methode voor het CR3BP, focust zich daarentegen alleen op de gevallen buiten de Hill sphere met een Jacobi constante van ongeveer 3.0. Voor hogere waardes van de Jacobi constante is deze methode niet meer nauwkeurig. Een GAM model wordt gestuurd door trainingsinvoer; het model leert de dynamica van de flyby's aan de hand van de effecten in de aangedragen invoer die door

middel van numerieke propagatie tot stand zijn gekomen. We analyseren de gevoeligheid van het aantal gesimuleerde voorbeelden op de kwaliteit van de voorspelde post-flyby status. Uit de resultaten blijkt dat het GAM model nauwkeurige voorspellingen kan maken van de post-flyby status van een satelliet, onafhankelijk van de beginvoorwaarden. Aangetoond wordt dat het model efficiënt en accuraat is door het te vergelijken met resultaten verkregen door numerieke integratie. De voorgestelde methode is toepasbaar voor gevallen met een hoge energie, maar ook voor gevallen met lage energiewaardes. De rekentijd van een enkele voorspelling met GAM kost slechts 1.16×10^{-6} s, in tegenstelling tot methode van de Keplerian Map die 0.064 s kost (desktop PC met een Core i7 CPU en 8.00 GB RAM). Voor het systeem zon-(aarde plus maan)-satelliet is de gemiddelde waarde van de absolute fout in de voorspelling de halve-lange as 3.86×10^{-5} AU (met typische waardes tussen -0.0115 AU en 0.0163 AU).

Nadat het model is opgebouwd voor deze 2-dimensionale situatie breiden we het uit naar een volledig 3-dimensionaal CR3BP. Door dit te doen is het aantal vrije parameters in dit probleem verhoogd van drie naar vijf. Een nieuwe afbeeldingsfunctie is gedefinieerd om de flyby effecten na één omloop te kwantificeren. Het GPR model is tot stand gekomen door gebruik te maken van passende functies voor het middelen en het berekenen van de covariantie. Om de efficiëntie van deze methode te verbeteren is een criterium opgesteld voor de gemiddelde absolute waarde van de fout. Dit is gedaan om de optimale grootte van de training dataset te vinden. De robuustheid hiervan is onderzocht en de toepasbaarheid voor realistische scenario's is geanalyseerd. De gevoeligheid van de verschillende input elementen (de covariantie functie, de grootte van de training dataset, random getallen) op de flyby is onderzocht. De nauwkeurigheid en efficiëntie van het voorgestelde model is onderzocht voor verschillende energie niveaus, van gevallen met weinig energie tot gevallen met een hoge energie. De waarde van de Jacobi constante varieert tussen 3,0012 en 0,5122, een groot bereik, en kan daarmee op veel vlakken toegepast worden (in tegenstelling tot voorgaand onderzoek, zoals dat van Ross en Scheeres (2007) en van Alessi en Sánchez (2016)). Het GAM model laat een significante verbetering zien ten opzichte van de semi-analytische methode (de KM methode ontwikkeld door Alessi en Sánchez (2016)). De nauwkeurigheid en efficiëntie van het voorspellen van de variatie van de semi-lange as zijn verbeterd met een factor 3.3 en een factor 1.27×10^{4}, respectievelijk.

De bovengenoemde GAM modellen voor het eenvoudige planaire model en het 3D model maken beiden gebruik van de GPR methode, d.w.z. leveren continue output: de GPR methode maakt een voorspelling voor iedere beginwaarde die aangeboden wordt. Het is echter niet mogelijk voor de GPR methode om trajecten te klassificeren die leiden tot een botsing met het tweede hoofdlichaam. Om die reden is het noodzakelijk om een classificatie te maken. Hiervoor is de GPC methode ontwikkeld. Deze is net als de GPR methode gebaseerd op Gaussische processen. Trainingsinvoer wordt gegenereerd door middel van een numerieke propagatie van de bewegingsvergelijkingen voor het CR3BP. De data binnen het model bevatten vervolgens zowel botsingstrajecten als veilige trajecten. Als eerste wordt het GPC model gevoed met de training data. Hierna kan het GPC model trajecten klassificeren voor iedere willekeurige beginconditie. De gevallen die niet leiden tot een botsing worden doorgestuurd naar het GPR model. De nauwkeurigheid in het detecteren van deze trajecten met de GPC methode is beter dan 90%.

Het GPR model wordt verder verbeterd door de Jacobi constante als extra input toe te voegen. Deze constante vertegenwoordigt de energie in het CR3BP en zorgt voor waardevolle informatie over de dynamica van overgangsvluchten. De prestatie van de modellen is geanalyseerd in drie verschillende systemen bestaande uit: zon-(aarde+maan)-satelliet, Jupiter-Callisto-satelliet en zon-Jupiter-satelliet. De waarde van de massaverhouding μ neemt hierbij toe met een factor 10. De resultaten demonstreren dat de effecten van flyby's efficient en nauwkeurig geëvalueerd kunnen worden door de GPC (classificatie) methode te combineren met de GPR (regressie) methode. De voorgestelde GAM aanpak is aantoonbaar toe te passen op verschillende CR3BP systemen. Het GPC kan zelfstandig gebruikt worden voor het detecteren van botsingen, of gecombineerd worden met het GPR model. De verdeling van de gevallen met een botsing vertoont een grote gelijkenis met resultaten verkregen door numerieke CR3BP propagatie. Wanneer een CR3BP een grotere waarde voor μ heeft, zijn meer training samples nodig voor het regressie onderdeel. De gemiddelde waarde van de absolute fout in de verandering in de semi-lange as is slechts 3.86×10^{-5} AU met een rekentijd van 5.8×10^{-6} s per geval (bij het zon-(aarde+maan)-satelliet probleem). Het toevoegen van de Jacobi constante als een extra input verbetert de nauwkeurigheid van de voorspelling voor alle systemen significant (rond de 20%).

De GAM aanpak die is ontwikkeld in dit proefschrift kan gebruikt worden voor het ontwerpen van ruimtemissies. De grote nauwkeurigheid en efficientie van de methode maken de tool een zeer aantrekkelijk instrument om flyby effecten te kwantificeren, met name wanneer veel begincondities moeten worden geëvalueerd.

1

INTRODUCTION

IN this introductory chapter, first, the mechanism and application of a gravity assist (GA) in space missions are introduced. The patched-conics model for designing GA trajectories using a two-body framework is reviewed. The three-body problem (3BP), which is used by the current research, as well as the associated Jacobi's integral are elaborated upon. The numerical and semi-analytical tools that have been used to investigate the (effects of a) GA are summarised. Inspired by these techniques, this thesis aims to solve the problem by using machine learning, i.e., the Gaussian Process (GP) method. The basic idea of a GP method is introduced, with particular attention paid to its application in the field of celestial mechanics. Finally, the research questions are formulated and the outline of this thesis is presented.

1.1. GRAVITY ASSIST

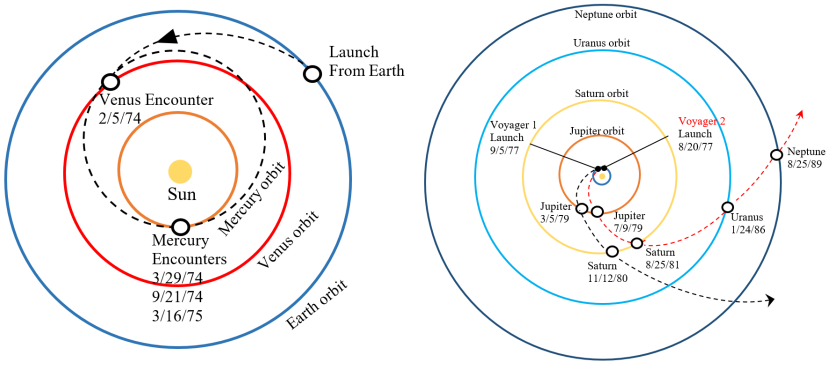
From the first spacecraft leaving Earth orbit (Luna 1, January 1959) to the recent three Mars missions (Mars 2020, Tianwen-1 and Hope in July 2020), a great number of space probes have been launched to explore the Solar System (Siddiqi 2018). These missions contribute to the assessment of the environment on other planets and moons, searching for evidence of extraterrestrial life, our understanding of the origin of the Solar System, etc. (Ness et al. 1974, Broadfoot et al. 1976, Glassmeier et al. 2007, Hassler et al. 2014). Typically, a mission targeted at the neighbour planets Mars or Venus employs a Hohmann transfer to reach them: the technique close to this kind of transfer is performed by the recent Mars missions. For a destination beyond these planets, the Hohmann transfer is not necessarily the first choice due to the large propellant requirement. In order to save propellant given a tight Δv budget, the GA technique has been commonly used by many space missions.

The GA, or flyby, is a mechanism to change the velocity of the spacecraft using the gravity of a celestial body (e.g., Beutler 2004). The GA body can be either a planet or a moon, for altering the orbit around the Sun or the target planet, respectively. In October 1959, a GA was first used by the Soviet mission Luna 3, photographing the far side of the

1

Moon (Biesbroek & Janin 2000). The first application of GA in interplanetary missions was Mariner 10 in November 1973 (Dunne 2011). The flyby at Venus changed the path of the spacecraft so that the perihelion was decreased effectively to meet Mercury, shown in Figure 1.1(a).

During the process of a GA, the sum of kinetic energy of the spacecraft and the GA body remains constant. The momentum variation gained by the spacecraft equals that lost by the GA body. The GA's can be categorised into two types in terms of the change in orbital energy: the first type increases the orbital energy by following the movement of the GA body; the second type decreases the orbital energy by flying in front of the GA body. A famous usage of the first type is the so-called 'Grand Tour', performed by Voyager 2 which visited Jupiter, Saturn, Uranus and Neptune. The combination of GA's made it possible to explore four planets in one mission given their perfect alignment, which occurs once every 175 years only (Flandro 1966). Also, although not part of the official mission plan, by means of GA's, the Voyager 1 and 2 became the first two spacecraft to leave the Solar System (Gurnett et al. 2013, Gurnett & Kurth 2019). Figure 1.1(b) shows the flyby trajectories of the two spacecraft. An example of the second type of GA is BepiColombo. The energy relative to the Sun is to be decreased using Earth, Venus and Mercury flybys (Benkhoff et al. 2010). The spacecraft is expected to insert in an orbit around Mercury in December 2025.



(a) GA trajectory of Mariner 10.

(b) GA trajectories of Voyager 1 and 2.

Figure 1.1: The trajectories of Mariner 10 and Voyager 1 and 2, redrawn based on the images from https://www.nasa.gov/feature/45-years-ago-mariner-10-first-to-explore-mercury https://www.jpl.nasa.gov/edu/news/2018/12/18/then-there-were-two-voyager-2-reaches-interstellar-space/

In the preliminary design of a GA mission, the patched-conics model is a useful tool to simplify calculations (e.g., Bate et al. 1971). The simplification is made by dividing a multi-body problem into several two-body problems. This model has a fundamental assumption: the trajectory of the spacecraft is driven by one celestial body only. When inside the Sphere of Influence (SoI) of a particular planet (or moon), it is only the gravity pull of that body that drives the dynamics. Otherwise, it is the Sun (or planet) only. With this, the patched-conics model is a succession of elementary two-body problem formulations. The impact of the GA is considered as instantaneous and each segment

is represented as an ideal Kepler orbit. These segments are patched together. Labunsky et al. (2017) introduced methods and models of trajectory design using multiple gravity assists based on the patched-conics model. Vasile & De Pascale (2006) proposed a simplified model for powered-flyby that is sufficiently accurate for preliminary design. The method preserved important classes of possible solutions and was verified for the Cassini Mission and other typical missions to Jupiter and Asteroid.

The patched-conics framework becomes less effective for small relative velocities (at infinity; v_∞) between the spacecraft and the GA body. The two-body formulation has a limitation in describing some dynamics in more detail, in particular for low-energy cases with phenomena like invariant manifolds. In the physical reality of a multi-body framework, the orbit of the spacecraft is continuously influenced by the smaller body. Take the Sun-Jupiter-spacecraft problem as an example: studies have shown that the gravitational attraction of Jupiter affects the trajectory of the spacecraft even in the exterior realm of its SoI (Greenberg et al. 1988, Gawlik et al. 2009). The special dynamics described by the three-body formulation has proven to be essential to design low-energy trajectories or to study the motion of asteroids between Mars and Jupiter (Beutler 2004, Campagnola et al. 2012, 2014). It is clearly worthwhile to investigate the dynamics of a flyby in the three-body framework.

1.2. CIRCULAR RESTRICTED THREE-BODY PROBLEM

After an analytically successful investigation of the two-body problem, scientists switched their interest to the n-body problem. Many attempts have been made to obtain closed-form solutions for the full dynamics in this n-body problem. It was found that analytical solutions are only available under certain very specific conditions. Scientists who have worked on this include Clairaut, Euler and Lagrange (Clairaut 1747, Euler 1772, Lagrange 1772). The three-body problem is a logical choice to start the investigation of the n-body problem, for which a general closed-form solution does not exist. Yet, the three-body researches contributed to many practical astrodynamics problems. For instance, the application to the Sun-Jupiter-asteroid situation shed light on the formation of main-belt asteroids. The Sun-Earth-Moon problem gives insight into the evolution of the Moon.

In order to reduce the degree of difficulty of the problem, Euler introduced the Restricted Three-Body Problem (R3BP). The R3BP assumes that one of the bodies is small and has negligible mass compared to the other two. This small body is under the influence of the gravitational attraction of the two massive bodies. To further simplify the R3BP, the two massive bodies are assumed to have circular orbits and rotate around their barycenter. Although the Circular Restricted Three-Body Problem (CR3BP) is simplified, it still contains some special properties of the 3BP, and is relevant in practice (Beutler 2004). In the Solar System, the assumption of circular orbits can be met because a large number of planets and moons (revolving around a specific planet) have very small eccentricity, which can be considered as nearly zero in the preliminary design.

By definition, the CR3BP describes the dynamics of three point masses: two massive bodies P_1 and P_2 and a third body P_3 with masses M_1, M_2 and M_3, respectively, which meet the condition $M_1 > M_2 \gg M_3$. P_1 and P_2 rotate around each other in circular orbits. Their orbital plane is typically taken as the plane of reference. In the rotating reference

frame, the origin is set at the barycenter of P_1 and P_2, and the X-axis is aligned with the direction to P_2. The primaries P_1 and P_2 rotate around the Z-axis in counterclockwise direction. The orthonormal reference frame is completed by the Y-axis. The equations of motion for the spacecraft are described as (e.g., Szebehely 1967)

$$\begin{cases} \ddot{x} - 2\dot{y} = x - \dfrac{(1-\mu)}{r_1^3}(x+\mu) - \dfrac{\mu}{r_2^3}(x-1+\mu) \\ \ddot{y} + 2\dot{x} = y - \dfrac{(1-\mu)}{r_1^3} - \dfrac{\mu}{r_2^3} y \\ \ddot{z} = -\dfrac{(1-\mu)}{r_1^3} z - \dfrac{\mu}{r_2^3} z \end{cases} \tag{1.1}$$

where x, y and z represent the normalized position coordinates in the barycentric Cartesian co-rotating reference frame. The normalization uses the distance $R_{P1,P2}$ between P_1 and P_2 for length, $(M_1 + M_2)$ for mass, and $\sqrt{R_{P1,P2}^3/(G(M_1+M_2))}$ for time. G is the universal gravitational parameter. μ is defined as the mass ratio which specifies a CR3BP system. r_1 and r_2 are the distances between P_3 and the primaries P_1 and P_2, respectively. These parameters are defined as

$$\begin{cases} \mu = M_2/(M_1+M_2) \\ r_1 = \sqrt{(x+\mu)^2 + y^2 + z^2} \\ r_2 = \sqrt{(x-1+\mu)^2 + y^2 + z^2} \end{cases} \tag{1.2}$$

μ is a key factor that defines the structure of the CR3BP. Three CR3BP systems are investigated in this dissertation. For the system of Sun-(Earth+Moon)-spacecraft, μ equals 3.036×10^{-6}. The Earth and Moon are considered as one point mass. μ of the Jupiter-Callisto-spacecraft and the Sun-Jupiter-spacecraft systems has a value of 5.667×10^{-5} and 9.537×10^{-4}, respectively. In order to investigate the motion of P_3, we can introduce the Jacobi's integral, which is time-independent. The value of Jacobi's integral is constant along the trajectory of P_3; it is the only known constant integral of the CR3BP. In a specific CR3BP system (i.e., with fixed μ), its value C_J is fully determined by the position and velocity of P_3.

In the rotating reference frame, the Jacobi constant is defined as (Poincaré 1899, Szebehely 1967):

$$C_J = x^2 + y^2 + \frac{2(1-\mu)}{r_1} + \frac{2\mu}{r_2} - (\dot{x}^2 + \dot{y}^2 + \dot{z}^2) + \mu(1-\mu). \tag{1.3}$$

The Jacobi constant reflects the energy level of the trajectory of the spacecraft. A high value of C_J represents a low energy level. Given an initial condition, the motion of P_3 is constrained by C_J. For some values of C_J, there are inaccessible regions with boundaries called zero-velocity surfaces. Given a mass ratio μ, the structure of the accessible regions, known as Hill's regions, can be categorised into five cases. Each category is obtained by using C_J associated with the Lagrange libration points.

Figure 1.2 shows the first four cases in the XY-plane using non-dimensional units. The fifth case is not presented here because the entire space is accessible to P_3. To arrive

at this illustration, a hypothetical value of 0.2 is taken for μ such that the inaccessible regions (blue area) are clearly noticeable. In this case, the values for the Jacobi constant of the Lagrange points are $C_{L1} = 3.9709$, $C_{L2} = 3.7124$, $C_{L3} = 3.3573$, $C_{L4} = C_{L5} = 3.0000$.

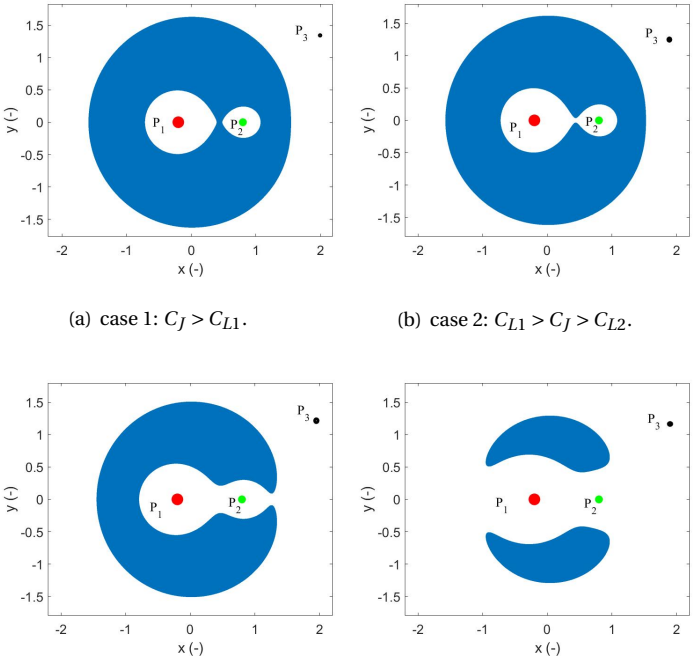
Case 1: The value of C_J is larger than C_{L1}. The accessible regions are divided into three parts: the vicinity of P_1, the vicinity of P_2 and the exterior region. If the initial location of P_3 is somewhere in the exterior region, it cannot access the neighbourhood of any of the two primaries. Also, it is not possible for a particle in the vicinity of one primary to reach the other one.

Case 2: $C_{L1} > C_J > C_{L2}$. When C_J decreases, the accessible regions around P_1 and P_2 become bigger. At the L_1 Lagrange point, the regions connect when $C_{L1} > C_J$. If P_3 is located in the vicinity of one of the primaries, it is possible to move to the vicinity of the other one. Transitions between the interior and exterior regions are not possible yet.

Case 3: $C_{L2} > C_J > C_{L3}$. If C_J is further decreased, the inaccessible region obviously shrinks. The interior and exterior Hill's regions are connected starting from the L_2 Lagrange point. If the initial location of P_3 is in the exterior Hill's region, it might travel into the neighbourhood of P_2 or even P_1, being trapped temporally before escaping or staying there for a long time.

Case 4: $C_{L3} > C_J > C_{L4/L5}$. The inaccessible region is split into two lobes and becomes smaller and smaller as C_J is being reduced.

If C_J is smaller than $C_{L4/L5}$, the symmetrical inaccessible regions vanish at the L_4 and L_5 Lagrange points, which turns out to be case 5. When talking about the initial condition of P_3 in this thesis, the initial location is somewhere in the exterior Hill's region, as illustrated in all individual plots in Figure 1.2.



(a) case 1: $C_J > C_{L1}$.

(b) case 2: $C_{L1} > C_J > C_{L2}$.

(c) case 3: $C_{L2} > C_J > C_{L3}$.

(d) case 4: $C_{L3} > C_J > C_{L4/L5}$.

Figure 1.2: Regions of possible motion for a particle P_3 given a Jacobi constant C_J in the rotating reference frame. Inaccessible regions are indicated in blue.

1.3. METHODS TO STUDY FLYBYS

There have been many attempts to study the third-body effects in the CR3BP regime. The previous research that inspired this study consists of semi-analytical and numerical methods, featuring the Keplerian Map (KM) and Flyby Map (FM), respectively (Ross & Scheeres 2007, Campagnola et al. 2012). Including this research, the typical works that will be mentioned in Chapters 2-4 are covered here in more detail.

1.3.1. SEMI-ANALYTICAL METHODS

Petrosky & Broucke (1987) and Chirikov & Vecheslavov (1989) first developed the KM based on the theory of nonlinear area-preserving maps. This technique is actually a mapping function used to update the energy variation of a particle. In terms of the Keplerian orbital elements, the energy is proportional to $1/a$. The mapping function is formulated using the phase angle between the particle and the secondary. This research shed light on the distribution and escape of particles in the CR3BP. The results obtained by the KM indicate that the motion of the particle in the CR3BP is sensitive to initial conditions. While applying the map to investigate the long-term motion of comets in nearly parabolic orbits, it was found that this motion can become chaotic. For instance, the

dynamics of comet Halley is proven to be chaotic on the long time-scale given the initial conditions taken from real data of 46 revolutions.

Using the KM, Ross & Scheeres (2007) investigated specifically the distant GA's in the planar CR3BP. Based on Picard's iteration, an energy kick function was built to approximate the variation of orbital elements over one revolution. The approach is applicable to near-Keplerian elliptical orbits of low energy levels, for a wide range of eccentricities. These orbits have periapse or apoapse touching the Hill sphere of the secondary. For the Jupiter-Callisto-spacecraft system, the evolution of the semi-major axis estimated by the KM shows a behavior that matches the results of a full CR3BP numerical propagation. In the phase space, the approach gives insight into the identification of transfer trajectories between orbits with different semi-major axes. It also allows for a fast approximation of possible orbits, which might yield ballistic transfers or escape trajectories. The accuracy of this method is found to be reasonably good only for very low three-body energy levels (Lantoine et al. 2011).

Grover & Ross (2009) applied the KM to design propellant-efficient trajectories in multiple three-body problems. The patched three-body systems are Jupiter-Europa-spacecraft and Jupiter-Ganymede-spacecraft. In order to obtain low-energy trajectories with a short time of flight, a controlled KM was developed by introducing a manoeuvre ΔV during the flyby. Jerg et al. (2009) used the KM to design optimal capture trajectories. The multi-flyby trajectory connects a large orbit relative to Jupiter and a capture orbit around Callisto. For an asteroid retrieval mission, the KM was used to update the orbital elements of a target asteroid encountered with the Earth (Sánchez et al. 2013). The attention focused on the Amor asteroids, assessing the retrieval opportunities of six candidates. When Peñagaricano Muñoa & Scheeres (2010) analysed two-point boundary value problems, they took the perturbation on the periapse into account for the construction of their KM. In addition, the map was improved to quantify the flyby effects in the fully spatial CR3BP.

Further on the spatial CR3BP, Alessi & Sánchez (2016) derived a formulation to estimate the variation of the Keplerian orbital elements due to a distant encounter. Based on perturbation theory, a kick map was developed which shows the variations of different elements as a function of the phasing angle. This angle is determined by the relative position between the secondary and particle at periapsis passage. The kick maps for the orbit of asteroid 2009 BD have almost identical behavior compared to the map generated by the CR3BP propagation. This method assumes a sufficiently small mass ratio, and may suffer from singularity problems for low eccentricity and inclination. A limitation is that the accuracy is very good for encounters taking place outside the Hill sphere. The obtained formulation is semi-analytical and not applicable to the cases of temporary capture (and subsequent escape).

Neves et al. (2018) analysed various approaches to model the third-body effect and tried to obtain a fully analytical solution. These approaches include: 1) the KM used by Alessi & Sánchez (2016); 2) the Periapsis-Apoapsis-Periapsis KM, replacing the independent variable true anomaly by the eccentric anomaly; 3) an Euler KM, employing the Euler method to numerically integrate the KM equations. They showed the accuracy of these models within their fields of application. An attempt was made to find the closed-form analytical solutions for the variation of the orbital elements of the massless

body under specific conditions in the CR3BP. For orbits of low eccentricity, an analytical approximation was formulated for the semi-major axis evolution.

The semi-analytical methods present concise formulations to approximate GA effects, but need numerical methods to generate the solutions. Generally, the computational effort is almost the same as that of a fully numerical approach.

1.3.2. NUMERICAL METHODS

In the CR3BP, Villac & Scheeres (2003) investigated the trajectories escaping the secondary, inspired by the Europa Orbiter mission. In order to simplify the analysis, a Poincaré map was used to reduce the dimension of the dynamics. The analysis concentrated on a set of values of the Jacobi constant. The map allows the characterization of the escape trajectories, which determines the bounds for initial conditions in terms of radius, longitude of periapsis and inclination. For a spacecraft initially located in a circular orbit around the secondary, the minimum total velocity increment ΔV required for a low-energy escape was obtained. Due to the symmetry properties of the system, the method can also be applied to the design of capture trajectories.

Villac & Scheeres (2004) analysed the planar Hill's problem using a numerical method. Different from the CR3BP, the Hill's problem is also a three-body problem, which assumes that the secondary and third bodies are both small with non-negligible masses. For the trajectory of two small bodies, the position space is partitioned into periapsis and apoapsis regions. This contributes to the qualitative analysis of the dynamics of the Hill's problem. The method was used to define regions of escape and capture for low-energy cases.

Campagnola & Russell (2010) developed a Tisserand-Poincaré (T-P) graph for the problem of endgame. The endgame focuses on the final phase of the trajectory design before the spacecraft enters the target orbit. The study was performed in the context of a multi-body problem, patching different CR3BP systems, for instance Jupiter as primary, and Europa and Ganymede as secondaries, respectively. The T-P graph is based on the Tisserand graph, which was developed in the two-body regime using a patched-conics model. For the CR3BP in this work, a Poincaré section was placed at the negative X-axis of the rotating reference frame. The advantage of using this method is that the Tisserand level sets are extended compared to that using a patched-conics model. From an energy point of view, the ballistic transfers between moons are demonstrated to be possible by the T-P graph. The applications include finding a transfer trajectory from Europa to Ganymede with about 1 km/s lower ΔV than the Hohmann transfer. It is worth noting that the T-P graph can not be used to estimate the flyby effect of a single encounter.

Campagnola et al. (2012) presented a FM inspired by the KM and T-P graph. The FM was developed in the framework of the planar CR3BP. The phase space of the spacecraft trajectory was parametrized using the semi-major axis a, the Tisserand parameter T, the true anomaly f and the longitude of the periapsis. An illustration of an FM is shown in Figure 1.3. The FM defines a mapping function between the states of the spacecraft before (subscript A) and after (subscript B) a flyby. The Poincaré section $\sum$ is selected following the work of (Campagnola & Russell 2010), with a particular definition of boundary conditions $R_2 > R_{2min}$ ensuring the validity of the Tisserand condition. R_{2min} is a specified minimum distance between the spacecraft and the secondary. The FM effectively

extended the applicability of the Tisserand graph over using the patched-conics model. It identified two types of flybys and showed that one type is more efficient than the other. In the Jovian system, a low-energy trajectory from Callisto to Europa saving 150 m/s ΔV is found.

The semi-analytical and numerical methods offer valuable insight into the effect of flybys in the three-body problem. Motivated by these methods, an attempt is made in this thesis to quantify these flyby effects from a machine-learning perspective. Rather than finding the analytical solutions for the variation induced by the flyby, the machine learning-based approach is expected to achieve three main goals: 1) applicable for a wide range of three-body energies; 2) a reasonable accuracy for preliminary design; 3) time-saving, i.e., faster than the semi-analytical and numerical methods.

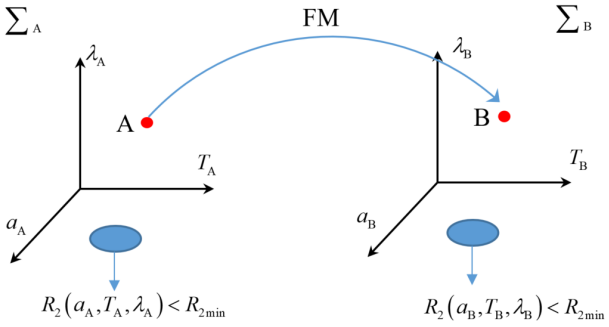


Figure 1.3: Schematics of the Flyby Map.
After (Campagnola et al. 2012).

1.4. MACHINE LEARNING BASED ON GAUSSIAN PROCESS

The Gaussian Process (GP) method is a supervised machine-learning method based on Bayesian inference (Rasmussen & Williams 2006). According to different practical problems, the method can be used for regression and classification, known as GPR and GPC, respectively. The GPR model was developed based on the method of kriging, which was first proposed by Krige (1951) for mine valuation in South Africa. Given samples from several boreholes, a prediction of the distribution of gold was obtained using GPR. In recent years, the Deep Neural Network (DNN) is very popular for its excellent performance on speech recognition, visual object recognition and object detection, etc. (LeCun et al. 2015). The 2018 Turing award went to three pioneers in DNN (Hinton & LeCun 2019). In order to train the internal parameters and achieve accurate predictions, a DNN model commonly needs a large amount of training data. Unlike the DNN, a GP method typically works well for a small dataset already, and provides information of uncertainty about the prediction. Rasmussen & Williams (2006) presents a systematic introduction to the GP method. The main concepts of a GP method are clearly covered here: it introduces the construction, model selection, training and prediction of GP models. It also analyses the difference between a GP method and other machine-learning tools such as the Support

Vector Machine.

The main concept of any GP model is the Gaussian process, which is defined as a collection of random variables. Any finite collection of these variables has a joint Gaussian distribution (Rasmussen & Williams 2006). When we are using a machine-learning tool to learn a model (e.g., a process in the real world), this model can be defined by a simple mapping function:

$$f(x) : x \mapsto y \tag{1.4}$$

The relationship between the input x and the output y is described by the mapping function $f(x)$. Depending on the type of output, a problem is defined as regression for continuous outputs or classification for categorical outputs. Taking weather forecast as an example, assuming x to be the date, the problem is regression if y = amount of precipitation, and classification if y = Sunny or Rainy. The property of a GP method allows it to solve both types of problems, which will be shown in Chapter 4.

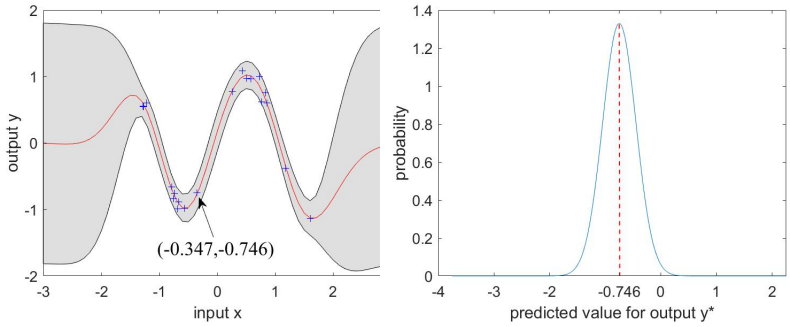
Figure 1.4 presents an example of using GPR to make predictions. In Figure 1.4(a), the blue 'plus' symbols represent hypothetical training data (x, y), which do not have a physical meaning. Assuming that (x, y) is empirical information observed from a real process $f(x)$, the red curve represents the GPR predictions after a process called training. See Chapter 2 for more details about this training process. Given a new test input x^*, the prediction $y*$ can be obtained from the corresponding value on the red curve. Making predictions for a test dataset using a machine-learning tool inherently involves some level of uncertainty. An assumption is made in the application of the GPR model, the uncertainty is in a form of Gaussian distribution containing standard deviation information. A specific advantage of the GPR approach is the fact that it is able to quantify this uncertainty, which is detailed in Chapter 4. It is obtained directly from the training process. The shaded area in Figure 1.4(a) denotes twice the standard deviation. Typically, dense observations in the input space lead to lower standard deviation, and the opposite is the case for regions with sparse training data. Therefore, the left and right input space have a higher standard deviation than the central part. Figure 1.4(b) presents the uncertainty of predicting a single sample (x^*, y^*)=(-0.347,-0.746). The Gaussian distribution describes the confidence of predicted outputs.

Figure 1.5 illustrates an application of the GPC model for the binary classification problem. The training samples are shown in points, each of which has two input parameters x_1 and x_2. The training outputs are labelled as 'positive' ('star' patterns) or 'negative' ('plus' patterns). Using the GPC model, the prediction over the whole input space is presented by the contour lines. The colour bar corresponding to the contour lines presents the GPC prediction in terms of the possibility of a sample being predicted as positive. The orange lines near the 'star' patterns indicate that a new sample taken from this area has a possibility of more than 50 % to be positive. When this colour changes to green or even blue, the prediction would be negative.

Based on the idea of GP, the model provides a practical and probabilistic approach to learning (Rasmussen & Williams 2006). This gives advantages to the interpretation of a prediction model. Also, the required amount of training data is small and the training procedure is simple. The GP method was applied to solve robot inverse dynamics problem for robot model-based control on a Barrett WAM robot arm (NguyenTuong et al. 2009). He & Siu (2011) addressed the problem of producing a high-resolution image

from a single low-resolution image based on GPR. THe GPR model is used to produce soft clustering of pixels based on their local structures.

It is of interest to use a GP method in the field of astrodynamics for solving some practical problems. Shang & Liu (2017) developed a GPR model to efficiently assess the accessibility of main-belt asteroids. The training dataset consists of 1500 training samples only, generated by globally optimal two-impulse or Mars gravity-assist transfers. The model costs only several minutes to estimate the total velocity increment ΔV for a probe to reach 600,000 asteroids. From a propellant point of view, about 4000 of them are selected as candidates for future asteroid exploration. Gao & Liao (2019) modeled the gravity field of small celestial bodies based on the GPR. The relative error of calculating the acceleration at a particular position is 1.27 % compared to the high-precision polyhedron modeling method. The efficiency is improved by a magnitude of 10^5. Bouwman et al. (2019) constructed a GPR model to replace the sequential quadratic programming for the low-thrust trajectory design. The model learns information from a training dataset obtained from shape-based methods and differential evolution. Given a condition such as departure time and time of flight, the model made accurate predictions on ΔV and propellant mass fraction. Moreover, a GPC model was built to identify infeasible trajectories for certain combinations of input parameters.



(a) Prediction using GPR. (b) Uncertainty of prediction.

Figure 1.4: Illustration of a prediction using Gaussian Process Regression.

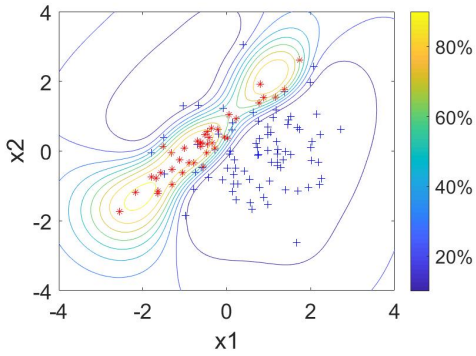


Figure 1.5: Illustration of a prediction using Gaussian Process Classification.

1.5. RESEARCH MOTIVATION

Although many studies on GA's have been done, either analytically or numerically, this thesis presents a new topic: quantification of GA effects in the three-body problem using a Gaussian Process method. Instead of attempting to solve the problem in an analytical formulation, we use a machine-learning tool to investigate the flyby effects. As introduced in the previous sections, the flyby is of significant importance for space missions. In the 3BP regime, understanding the dynamics of flyby's for a wide range of energies sheds light on the chaotic nature of the problem. The low-energy cases like ballistic capture and low-energy transfers between moons of a planet play an important role in propellant-saving exploration missions. For high energy levels, the analysis of third-body perturbations that result in chaotic transfers are beyond the scope of two-body regime and the patched-conics models (Petit & Henon 1986). Given an initial condition, an accurate and efficient way of quantifying the flyby effects is crucial to mission analysis, especially if a large number of initial conditions needs evaluation or a multi-flyby situation is involved. For instance, a collision probability analysis with respect to Europa has been performed for the Jupiter Icy Moons Explorer (Boutonnet et al. 2014). An attempt has been made to look for analytical solutions for the variations due to a flyby in the CR3BP, leading to semi-analytical formulations (Ross & Scheeres 2007, Alessi & Sánchez 2016, Neves et al. 2018). It is very interesting to investigate the flyby effects using a new machine-learning-based approach, the GP method.

The research questions are formulated as follows:

(RQ.a) Can a machine-learning method be used to quantify the GA effects in the three-body regime?

This is the main research question of the current thesis. A framework for modeling the variation of flyby effects needs to be defined. The CR3BP regime is employed to describe the motion of a particle with negligible mass under the influence of a flyby. In this regime, numerical methods have shown excellent performance but require expensive computational time, especially for numerous numbers of candidate initial con-

ditions. Semi-analytical methods capture the dynamics of flyby effects well. However, these methods require a computational time comparable to that of the numerical approach, and the accuracy is commonly good at low energy levels only (Ross & Scheeres 2007, Alessi & Sánchez 2016, Neves et al. 2018). The current study is inspired by these studies, particularly the KM and FM (Ross & Scheeres 2007, Campagnola et al. 2012). We are interested in the performance of the quantification of flyby effects by means of machine learning.

Machine learning is drawing much attention in recent years, showing strong predictive ability in various fields. The tool we select for solving the CR3BP issue is a probabilistic model. A GP method is a supervised learning technique, based on Bayesian principles. 'Supervised' indicates that we need to provide the model empirical information, i.e., training data, to learn a specific process (Mohri et al. 2018). The GP method uses the training data not only in the training procedure, but also in the prediction. In order to predict the flyby effects, the generation of training data and model construction play an important role, which will be investigated throughout this thesis.

(RQ.b) How can the Gaussian Process method be used to learn the effects of flyby's in a planar CR3BP framework?

For investigating the possibilities of modeling the flyby effects, the framework is first simplified in order to reduce the difficulty. A planar CR3BP regime is employed, reducing the mapping dimension from five to three. The GP method has one basic assumption: the points with closer inputs are likely to have similar outputs. Different sampling methods are compared to determine the best way of generating training inputs. For the training outputs, the numerical propagation of the CR3BP equations of motion is used to serve as the benchmark for the GP model. The GP model selection considers various covariance functions that look promising to capture the characteristics of flybys in the planar CR3BP. The performance of GP models will be analysed by comparing its accuracy and efficiency to a previous semi-analytical method, the Keplerian Map (Ross & Scheeres 2007).

(RQ.c) Can a Gaussian Process Regression model predict the flyby effects in a fully spatial CR3BP for a wide range of three-body energies, and if so, with what accuracy?

This research question is based on the research question RQ.b. Provided that the GP method can be applied to the planar CR3BP, some modifications are needed for generalisation to the spatial case. The boundary of the input space needs to be defined corresponding to the increased dimensionality. In order to obtain an informative training dataset, a larger data size is expected because of the larger dimension of the input space. It has to be determined which covariance function works best for the 3D scenario. The Sun-(Earth+Moon)-spacecraft system is considered to develop the proposed method.

(RQ.d) Can the performance of GP models be improved by using specific characteristics of the CR3BP? What classification model can be used to identify collision trajectories?

An advantage of the machine-learning tool is that its flexibility allows to incorporate new features as inputs. The mapping function of input-output is not necessarily fixed. The machine-learning approach has the potential to learn the underlying relationship of a mapping, though the inputs and outputs are not directly connected. The Jacobi constant is selected to be an additional input feature. The accuracy after adding this feature is compared to that of the original model. It is also necessary to assess the validity of GP models in different CR3BP systems. In addition to the Sun-(Earth+Moon)-spacecraft system, the Jupiter-Callisto-spacecraft and the Sun-Jupiter-spacecraft systems with a bigger mass ratio μ are investigated. The values of μ differs by a factor of about 10. Still, the GPR model cannot identify trajectories crashing into the secondary. To this end, a GPC model is developed for the identification of collision trajectories, complementary to the function of the GPR.

1.6. OUTLINE

All research questions are addressed in Chapters 2-4. Each chapter mainly focuses on one research question from RQ.b to RQ.d. The main research question RQ.a is addressed gradually.

Chapter 2 focuses on the development of Gravity Assist Mapping (GAM), a GPR model to quantify the flyby effects. First, it is presented how the GAM learns the features of flybys from training samples generated by planar CR3BP equations of motion. The covariance function, the key module of the GPR approach, is optimized according to the analysis of flyby dynamics. The number and distribution of training samples are selected by taking the Root-Mean-Square Error (RMSE) and the Mean-Absolute-Percentage Error (MAPE) of test samples as criterion. At last, the performance of the GAM technique is assessed. The computational effort is evaluated for different numbers of initial conditions.

Chapter 3 extends the model used in the previous chapter in order to quantify flyby effects in a full three-dimensional situation. The complexity of the CR3BP increases significantly when going from a planar model to a full spatial model. A new GPR-based GAM model is developed to solve this. The model aims at a wide range of energy levels. The energy of the considered cases ranges from values lower than that associated with the L1 libration point to values larger than that associated with L4/L5. Several thousands of training samples are required for the GPR model to learn the flyby effects . The prediction of output given an arbitrary input will be compared with a previous semi-analytical method and numerical integration methods.

Chapter 4 develops two GP models: a classification model for detecting collision trajectories, and a regression model for calculating flyby effects. The GPC model is trained in a sub-space of the input space for the GPR. The training outputs are labeled as 'impact' or 'no-impact', according to the numerical propagation result of the CR3BP. The collision-filtered training samples are forwarded to the GPR model. Based on the GPR model developed in Chapter 3, this chapter introduces the Jacobi constant as a supplementary feature for training. The performance of the GP models is assessed based on three systems: Sun-(Earth+Moon)-spacecraft, Jupiter-Callisto-spacecraft and Sun-Jupiter-spacecraft.

Chapter 5 concludes this thesis and makes a number of recommendations for future

work.

2

A Gravity Assist Mapping Based on Gaussian Process Regression

Abstract We develop a Gravity Assist Mapping to quantify the effects of a flyby in a two-dimensional circular restricted three-body situation based on Gaussian Process Regression (GPR). This work is inspired by the Keplerian Map and Flyby Map. The flyby is allowed to occur anywhere above 300 km altitude at the Earth in the system of Sun-(Earth+Moon)-spacecraft, whereas the Keplerian map is typically restricted to the cases outside the Hill sphere only. The performance of the GPR model and the influence of training samples (number and distribution) on the quality of the prediction of post-flyby orbital states are investigated. The information provided by this training set is used to optimize the hyper-parameters in the GPR model. The trained model can make predictions of the post-flyby state of an object with an arbitrary initial condition and is demonstrated to be efficient and accurate when evaluated against the results of numerical integration. The method can be attractive for space mission design.

2.1. Introduction

In the past decades, valuable insight about celestial bodies in our Solar System has been gained using spacecraft, such as Jupiter visited by Galileo (D'Amario et al. 1992). Provided the payload capacity of launch vehicles is sufficient, it is possible to fly a direct transfer to the above destination. However, it is not necessarily the first choice due to the required propellant mass. In order to reduce the propellant consumption, typically a gravity assist (GA) technique is used. For instance, a combination of flybys at Venus and Earth was performed by Galileo before arriving at the final target Jupiter. In the preliminary design, the patched conics approach is commonly used to investigate flybys,

This chapter has been published in Astronautical Sciences 68, pages:248–272(2021).

which splits a multi-body problem into a succession of two-body problems (Broucke 1988, Longuski & Williams 1991, Strange & Longuski 2002, de Almeida Prado & de Felipe 2007). This model has a fundamental assumption: the trajectory of the spacecraft is driven by one celestial body only. In addition to traditional GA's, distant encounters can be used to construct low-energy trajectories between the moons of giant planets (**?**Ross & Scheeres 2007, Whiffen 2003). However, the patched conics model cannot be used to study low-energy trajectories or describe the phenomena such as the invariant manifolds. The Circular Restricted Three-Body Problem (CR3BP) framework is typically used to conduct these studies (Koon et al. 2000, 2001, Howell et al. 2001).

Unlike the patched-conics model, the secondary keeps affecting the spacecraft in the CR3BP model. Distant encounters can be modelled in the CR3BP. However, there is no clear boundary condition for such a flyby because the motion of the spacecraft is always affected by the secondary. The Keplerian map is a useful tool to study the flyby effects in the CR3BP. This concept was first introduced by Petrosky and Broucke to study the chaotic behavior of comets near Jupiter (Petrosky & Broucke 1987). They studied perturbation theory in the CR3BP framework to investigate the energy change of a particle with negligible mass after each flyby. The Keplerian map was further improved by Shevchenko with quite elementary methods (Shevchenko 2011). Ross and Scheeres exploited the concept of the Keplerian map to investigate semi-analytically high-altitude flybys in the Planar Circular Restricted Three-Body Problem (PCR3BP) (Ross & Scheeres 2007). It can predict well the variation of Keplerian energy during a flyby by calculating quadratures of energy-kick functions. However, the assumption of constant orbital elements using Picard's method fails if the spacecraft is very close to the secondary. It focuses only on the cases of a Jacobi constant around 3.0 and has poor accuracy for high three-body energies. Later, the Keplerian map for the fully spatial case was developed for the initial-value problem, which allows for applications to low-energy spacecraft mission design (Peñagaricano Muñoa & Scheeres 2010). In order to model the variation of the orbital elements of a massless particle in the CR3BP, Alessi and Sánchez developed a type of Keplerian map, called Kick map, based on semi-analytical formulations in the spatial CR3BP (Alessi & Sánchez 2016). It is worth noting that the integration of this formulation is done numerically, which requires a computational effort similar to that of the propagation in the CR3BP. Moreover, this estimation is prone to higher error when the massless particle reaches a distance to the secondary smaller than the Hill radius (typically, about 0.01 AU in the Sun-(Earth+Moon)-spacecraft system). Motivated by the Keplerian map and the Tisserand-Poincaré graph, Campagnola et al. developed a flyby map to study flybys in the PCR3BP (Campagnola et al. 2012). The flyby map has a wider domain of applicability than the Keplerian map. It found two types of flybys and illustrated that the direct type exists at all energies and is more efficient than the retrograde type. The GPR-based Gravity Assist Mapping aims at developing an accurate and efficient method of quantifying the flyby effects. This method holds the advantage of a prominent accuracy at the cost of a low computational efficiency, because the state of the particle must not be integrated during the encounter.

Inspired by the Keplerian map and the flyby map, we develop a Gravity Assist Mapping (GAM) based on Gaussian Process Regression (GPR), a machine-learning technique. The method is used for quantifying flyby effects in the PCR3BP. It works for a

broad range of energy levels, i.e., beyond the domain of applicability the Keplerian map, and requires smaller computational cost.

In this research, the dynamical model of a PCR3BP is presented first in Section 2.2. In Section 2.3, it is described how the GPR model is to learn the features of flybys from training samples generated by PCR3BP equations of motion. The covariance function, the key module of the GPR model, is improved based on the analysis of flyby dynamics. The number and distribution of training samples are selected by taking the Root-Mean-Square Error (RMSE) and the Mean-Absolute-Percentage Error (MAPE) of test samples as criterion. In Section 2.4, the performance of the mapping technique is assessed. By training the model with the information provided by hundreds of training samples, the GPR-based approach can accurately capture the features of flybys. Finally, the conclusions are drawn in Section 2.5.

2.2. PLANAR CIRCULAR RESTRICTED THREE-BODY PROBLEM

In the preliminary design of a space mission, simplified dynamical models are commonly used to reduce the computational effort. In this research, the PCR3BP model is used for constructing GA trajectories.

The PCR3BP assumes a system with three bodies: a massless particle and two massive primaries P_1 and P_2 with mass M_1 and M_2, respectively. Generally, we refer to body P_2 with the smaller mass as the secondary. The motion of the particle is driven by the simultaneous gravitational pull of both primaries. The primaries are assumed to rotate around the barycenter of the system in circular orbits. The orbital plane of these two massive bodies is taken as the reference plane. Characteristic quantities are commonly defined such that a normalized model can be applied to any three-body system.

In the rotating reference frame, which has the barycenter as origin, the frame rotates at the same angular velocity as the primaries do around this barycenter. The equations of motion in non-dimensional units for the particle (spacecraft) are given as (Szebehely 1967)

$$\begin{cases} \ddot{x} - 2\dot{y} = x - \frac{(1-\mu)}{r_1^3}(x+\mu) - \frac{\mu}{r_2^3}(x-1+\mu) \\ \ddot{y} + 2\dot{x} = y - \frac{(1-\mu)}{r_1^3}y - \frac{\mu}{r_2^3}y \end{cases} \tag{2.1}$$

where $\mu = M_2/(M_1+M_2)$ represents the mass parameter, and r_1 and r_2 the normalized distances between the particle and the primary and secondary respectively. In this paper, the Sun-(Earth+Moon)-spacecraft system is used with a mass ratio of 3.036×10^{-6}. Obviously, Equation 2.1 is analytically unsolvable. Moreover, the trajectory solutions are very sensitive to the initial conditions. This leads to the difficulty of assessing flyby effects accurately. Given an initial condition, the flyby effects can be calculated using the numerical propagation of the CR3BP equations of motion.

2.3. GRAVITY ASSIST MAPPING

GPR is a regression method to estimate the relation between inputs and outputs in terms of a specific model. Krige first developed GPR for the estimation of the distribution of gold (Krige 1951). For astrodynamics purposes, GPR was adopted to efficiently assess

the accessibility of main-belt asteroids and found thousands of potential mission targets (Shang & Liu 2017). It has also been used for modelling the gravity field of small bodies or the optimization of low-thrust trajectories (Gao & Liao 2019, Bouwman et al. 2019). GPR is able to learn effectively the correlation between inputs and outputs of a process. A GPR-based method will be developed here to evaluate the post-flyby state of the spacecraft in a PCR3BP system.

2.3.1. GPR FUNDAMENTALS

As a supervised learning tool, GPR is used to predict the output at arbitrary input based on empirical information. Such information is obtained by observing the output of a group of independent input data in a specific input space. In this section, we will discuss the elementary aspects of GPR; the interested reader is referred to (Rasmussen & Williams 2006). In this work, both input and output are the Keplerian elements of the spacecraft state in an inertial reference frame. The origin is centered at the largest primary and all the bodies are contained in the X-Y plane. The connection between the state of the spacecraft before flyby and after flyby can be described by a mapping function:

$$F : [a_A, e_A, \omega_A] \mapsto [a_B, e_B, \omega_B] \tag{2.2}$$

where a is the semi-major axis, e the eccentricity and ω the argument of periapsis. Subscripts A and B denote before- and post-flyby, respectively. Each of the orbital elements represents a feature of the state vector. One set of input vector and corresponding output vector comprises a so-called sample. In the inertial frame, Figure 2.1 shows the osculating orbital elements of the spacecraft at the initial condition. The frame is centred at the primary and the X-axis points to the vernal equinox. Two assumptions are made for these initial conditions: the spacecraft starts from the apoapsis (i.e., $\theta = -180°$); the initial true anomaly of the secondary θ_{P2} is selected such that it arrives at the negative X-axis when the spacecraft (in terms of initial osculating orbit) reaches periapsis relative to the primary. The time span before the secondary reaches the negative X-axis is set to be half of the orbital period of the initial osculating orbit of the spacecraft. Therefore, the relationship of phasing between the spacecraft and the secondary is described by the argument of periapsis ω only. When $\omega = 180°$, the minimum distance between the particle and the secondary happens at the periapsis passage and a flyby effect can be expected (See Appendix A for more details). When ω is greater/smaller than 180°, the flyby occurs after/before the periapsis. The boundary of the input feature ω is defined in Section 2.3.3. When a flyby occurs, the relative phasing between the spacecraft and the secondary has a range from $-10°$ to $10°$.

The output will be predicted by GPR using the general function f

$$f : x \mapsto y \tag{2.3}$$

where $x = [a_A, e_A, \omega_A]$ is a vector, and y represents one of the variables of the post-flyby state $[a_B, e_B, \omega_B]$. Obviously, the dimension of the output vector determines the number of GPR models. In our case, the output vector consists of three elements, so three GPR models are required to build the Gravity Assist Mapping.

The method of Bayesian inference is used to predict the value of the output y with respect to the input x. This procedure works on a Gaussian process which is basically

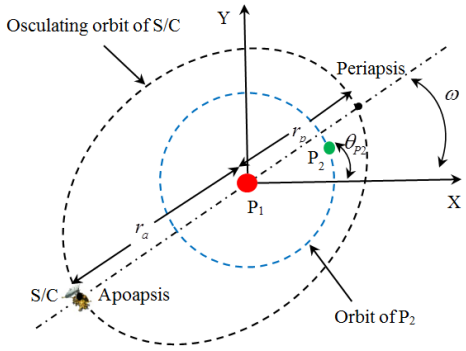


Figure 2.1: Osculating orbit of the spacecraft in the inertial frame centered at the primary.

an infinite collection of random variables. Selecting an arbitrary finite number of this collection can form a joint Gaussian distribution (Rasmussen & Williams 2006). For the GPR method, the random variable is taken as the value of the function $f(x)$. The Gaussian process based on the function $f(x)$ is specified by (Olds et al. 2007)

$$f(x) \sim \mathcal{GP}(m(x), k(x, x')) \tag{2.4}$$

where x' is an input vector in the input space. $k(x, x')$ represents the covariance function if $x \neq x'$ and the variance of x if $x = x'$. $m(x)$ is the mean function.

Commonly, the prior value for $m(x)$ is taken to be zero before obtaining more information about the mapping. After training the model, the posterior value will be inferred, which is not confined to be zero (Rasmussen & Williams 2006). Therefore, the module $k(x, x')$ becomes the only determinant of the process. The information about the Gaussian process in Equation 2.4 can be gained by observing flyby effects for known inputs. The observation of an output (one feature of the post-flyby state) for a given input (before-flyby state) is actually generated using the propagation in the PCR3BP model. The input and corresponding output compose a so-called training sample. Assuming we have N training samples, the training dataset D is defined as

$$D = \{(X, Y) | X = [x_1, x_2, \ldots, x_N], Y = [y_1, y_2, \ldots, y_N]\} \tag{2.5}$$

where X and Y are the training input matrix (the states before flyby) and the output vector (the selected feature of the post-flyby states), respectively. The output predicted by the GPR is assumed to differ from the ground truth by a value that follows a Gaussian distribution. Given the training inputs, the joint Gaussian distribution of outputs obtained from the GPR is:

$$Y \sim \mathcal{N}(0, K(X, X)) \tag{2.6}$$

where K is the covariance matrix consisting of N^2 covariance functions, written as

$$K(X,X) = \begin{pmatrix} k(x_1,x_1) & k(x_1,x_2) & \cdots & k(x_1,x_N) \\ k(x_2,x_1) & k(x_2,x_2) & \cdots & k(x_2,x_N) \\ \vdots & \vdots & \ddots & \vdots \\ k(x_N,x_1) & k(x_N,x_2) & \cdots & k(x_N,x_N) \end{pmatrix} \tag{2.7}$$

Assuming that y^* is the prediction for a new input vector x^*, then the training outputs Y and the predicted output y^* form a joint Gaussian distribution:

$$\begin{pmatrix} Y \\ y^* \end{pmatrix} \sim \mathcal{N}\left(\mathbf{0}, \begin{pmatrix} K(X,X) & K(X,x^*) \\ K(x^*,X) & k(x^*,x^*) \end{pmatrix}\right) \tag{2.8}$$

where $K(X,x^*)$ represents the vector of N covariance functions calculated at each entry of training inputs X and new input x^*. Clearly, because of symmetry, $K(x^*,X) = K(X,x^*)^T$.

By conditioning the joint Gaussian distribution in Equation 2.8 on the observations, the predicted distribution of output y^* has a Gaussian form and is completely specified by the mean $\mu(y^*)$ and the covariance $cov(y^*)$.

$$y^*|x^*,X,Y \sim \mathcal{N}(\mu(y^*), cov(y^*)) \tag{2.9}$$

The values of $\mu(y^*)$ and $cov(y^*)$ can be evaluated by

$$\mu(y^*) = K(X,x^*)^T K(X,X)^{-1} Y \tag{2.10}$$

and

$$cov(y^*) = k(x^*,x^*) - K(X,x^*)^T K(X,X)^{-1} K(X,x^*) \tag{2.11}$$

Generally, the expected value $\mu(y^*)$ can be taken as the predicted output y^*. Once a GPR model has been built, the prediction can be obtained by matrix multiplication of the $N \times 1$ vector $K(X,x^*)$, the $N \times N$ matrix $K(X,X)^{-1}$ and the $1 \times N$ vector Y. Equation 2.10 can be further simplified into:

$$y^* = K(X,x^*)^T Q_{train} \tag{2.12}$$

where $Q_{train} = K(X,X)^{-1} Y$ is precalculated. The prediction efficiency improves about N times after this simplification. When performing the evaluations for N^* new inputs, the outputs are obtained by

$$Y^* = K(X,X^*)^T Q_{train} \tag{2.13}$$

where $Y^* = [y_1^*, y_2^*, ..., y_{N^*}^*]$ is a vector of predicted outputs and $K(X,X^*)$ is an $N^* \times N$ matrix.

2.3.2. GPR COVARIANCE FUNCTION

The covariance function describes the relation between samples by evaluating a function in terms of inputs. It is the most basic and essential module for any GPR model. The covariance functions can have various mathematical expressions and parameters which

are called hyper-parameters. A perfect covariance function contributes to obtaining an exact representation of mapping between input and output. In addition, the generalization of a GPR model depends heavily on the choice of the covariance function. Unfortunately, a universal rule is not available for such a choice (Sun et al. 2018). In our research, it needs to capture the dynamics of flybys in the PCR3BP. Two criteria are considered to evaluate the performance of the covariance function. One criterion is to minimize the cost function:

$$f_{cost} = -\log p(y|x) \tag{2.14}$$

where the marginal likelihood $p(y|x)$ describes the probability of obtaining the true training output given a training input. This process is equivalent to maximize the following function (Rasmussen & Williams 2006):

$$\log p(y|x) = -\frac{1}{2} y^T K^{-1} y - \frac{1}{2}\log|K| - \frac{N}{2}\log 2\pi \tag{2.15}$$

Equation 2.15 is also the objective function to optimize the hyper-parameters using training samples. A larger $\log p(y|x)$ means that the possibility of observing true outputs is higher. Another criterion is to minimize the difference between the numerically propagated outputs of the PCR3BP and the predictions by the GPR. The differences are evaluated on test samples with inputs generated randomly in the input space.

The covariance function selected here initially is the Rational Quadratic Automatic Relevance Determination (RQARD) function (Duvenaud 2014). The classical RQ covariance function is stationary, i.e., it depends on the distances between sample points in Euclidean space only. The different influence of orbital elements on the flyby effect cannot be reflected by the RQ covariance function. To distinguish these effects, the Automatic Relevance Determination (ARD) distance measure is integrated into the RQ covariance function. Three specific weights l_a, l_e and l_ω (Equation 2.17 below) will be optimized based on the sensitivity of the flyby effects to the corresponding orbital elements. The optimized value of the weight is a reflection of the influence of that particular feature. The RQARD function is defined as:

$$k_{\text{RQARD}}(x, x') = \sigma^2 e^{(1+\frac{(x-x')^T Q(x-x')}{2\alpha})^{-\alpha}} \tag{2.16}$$

Here σ and α are the magnitude parameter and shape parameter of the output distribution, respectively. Q is a symmetric matrix containing weights

$$Q = \text{diag}(\frac{1}{l_a^2}, \frac{1}{l_e^2}, \frac{1}{l_\omega^2}) \tag{2.17}$$

where l_a, l_e and l_ω are weights for the elements a, e and ω respectively. Clearly, the RQARD formulation emphasizes the influence of near-by training samples, whereas samples further away have less influence.

Some covariance functions can interpret specific model characteristics, such as a cosine or periodicity. For this research, a new covariance function is built by taking the influence of the initial argument of periapsis ω_A on the variation of the semi-major axis a during a flyby into consideration. ω_A turns out to be the most influential parameter because it specifies the phasing between the massless particle and the secondary (Ross

& Scheeres 2007). Under the predefined boundary, Figure 2.2 shows a typical flyby effect of a_B for initial ω_A values between 174° and 186° at $[a_A, e_A] = [1.2591\text{AU}, 0.2]$ in the Sun+(Earth+Moon)-spacecraft system. The post-flyby a_B tends to have a sinusoidal curve in terms of ω_A. This is caused by the phase change of the flyby: the spacecraft can gain Keplerian energy if the secondary is ahead of it and vice versa. The Keplerian energy is computed for the osculating orbit with respect to the primary when the spacecraft moves far away from the secondary. For this near-Keplerian orbit, its Keplerian energy of the spacecraft with respect to the primary is proportional to the inverse of the semi-major axis (Ross & Scheeres 2007). The three-body energy is invariant during the flyby. In order to model this flyby effect, we incorporate a cosine function of ω_A into the GPR function. The combined covariance function (SUM) is expressed as

$$k_{\text{SUM}}(x, x') = \sigma^2 e^{(1+\frac{(x-x')^T Q(x-x')}{2\alpha})^{-\alpha}} + p^2 cos(\frac{\pi(\omega_A - \omega'_A)}{180h}) \tag{2.18}$$

where σ and α are hyper-parameters defined in Equation 2.16, and p and h are hyper-parameters denoting magnitude and phase transition, respectively. When ω'_A has a value close to ω_A, the last term in Equation 2.18 is approximately equal to its maximum p^2. This leads to a sample with ω'_A gaining more information from a sample with ω_A than from other samples.

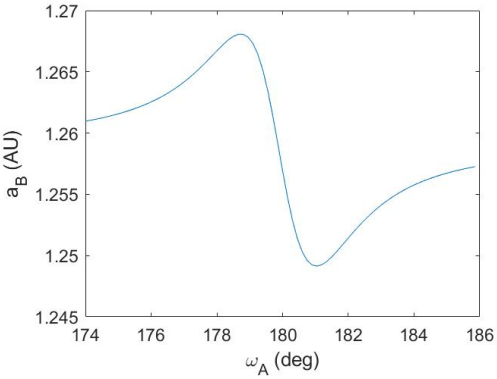


Figure 2.2: Variation of semi-major axis after a flyby. This result is obtained by numerical integration in the system of Sun-(Earth+Moon)-spacecraft.

2.3.3. TRAINING SAMPLES

As mentioned earlier, the GPR model obtains the empirical information by optimizing hyper-parameters using training samples. The training set, in particular their number and distribution, determines the performance, accuracy and efficiency of the GPR model. In principle, the accuracy of prediction improves with the information provided by an increasing number of training samples. However, a smaller training size will reduce the computational effort with respect to Equation 2.12. In order to balance accuracy and efficiency, the number of training samples is an essential variable which will be analysed in Section 2.4. Taking the Sun-(Earth+Moon)-spacecraft system as an example, the

training input vectors a_A, e_A, ω_A are randomly generated in the input space such that the initial radius of periapsis r_p, apoapsis r_a and ω_A meet the conditions

$$\begin{cases} r_p \in [1.00004464, 1.02], \text{AU} \\ r_a \in [1.01, 2.02], \text{AU} \\ \omega_A \in [170, 190], \text{deg} \end{cases} \tag{2.19}$$

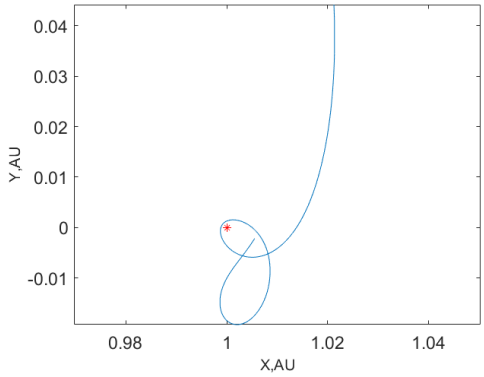


Figure 2.3: Example of a trajectory in synodic frame. The red star represents Earth. The Sun at $(-\mu, 0)$ is not shown in this figure.

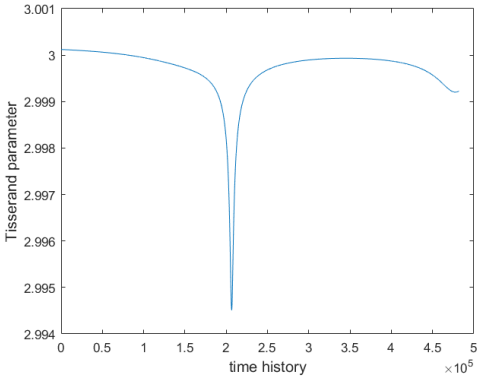


Figure 2.4: The evolution history of the Tisserand parameter for the trajectory of Figure 2.3. The X-axis shows the steps of the numerical integration for one orbital period.

The boundaries are selected to meet a number of requirements. First, the osculating orbit of the spacecraft has a Minimum Orbit Intersection Distance (MOID) with the secondary which allows a minimum of 300 km altitude at Earth taking the Galileo mission as reference (Anderson et al. 2008). Second, the spacecraft starts outside the Hill sphere

such that the status of the spacecraft with respect to the primary can be defined using the Keplerian elements. Third, ω_A is defined in order to observe an obvious flyby effect; beyond these boundaries effectively nothing happens (cf. Figure 2.2, the semi-major axis a_B after the flyby becomes closer to a_A in the left and right parts of the figure). After generating sets of $[r_p, r_a, \omega]$, the training inputs can be obtained by

$$\begin{cases} a_A = \frac{r_p + r_a}{2}, \text{AU} \\ e_A = \frac{r_a - r_p}{r_a + r_p} \\ \omega_A \in [170, 190], \text{deg} \end{cases} \tag{2.20}$$

Once the training inputs have been generated, the perfect post-flyby states are acquired by numerical propagation using Equation 2.1. The numerical propagation adopts adaptive step-size Runge-Kutta method with a maximum stepsize of 1×10^{-6} and relative and absolute tolerances of 1×10^{-12}, such that the quality of training samples is ensured. It is of importance to define a termination condition for the propagation: in principle one orbital period of the initial osculating orbit is defined as end epoch. The GPR-based Keplerian map is to assess the effect of a single flyby. However, the selected boundaries for input space do not guarantee a single flyby as shown in Figure 2.3. The initial condition of this particular trajectory is $[1.02\ \text{AU}, 0.007, 177.84°]$. Apparently, the spacecraft has a second close encounter with Earth at the final stage of the propagation. The second flyby results in an inconsistency of the training outputs: most are based on one flyby, some are based on two such events. There is no clear boundary for a flyby in the PCR3BP, which makes it difficult to remove second flybys by monitoring the position or velocity of the spacecraft. To overcome this difficulty, the Tisserand parameter is used as the index for the termination condition (Tisserand 1891):

$$T = \frac{1-\mu}{a} + 2\sqrt{a(1-e^2)} \tag{2.21}$$

Using this parameter as termination index is inspired by the value of the Tisserand parameter before and after a flyby when the spacecraft is far away from the secondary (Campagnola et al. 2012). Figure 2.4 illustrates the evolution of the Tisserand parameter for the trajectory in Figure 2.3. The Tisserand parameter drops dramatically when the first flyby happens, then it increases rapidly and begins to decrease again when the second close encounter is coming. To prevent the second flyby, the propagation is terminated when the Tisserand parameter is maximised after the first drop, at the top of the curve after the first flyby. In addition, the cases when an Earth impact takes place are removed for obvious reasons.

2.4. PERFORMANCE OF GPR-BASED GRAVITY ASSIST MAPPING

In order to evaluate the accuracy of the GPR-based Gravity Assist Mapping, a set of test samples has been generated whose inputs are randomly distributed over the input space. To quantify the quality of any combination of GPR model and number and

distribution of training samples, the Root-Mean-Square Error and the Mean-Absolute-Percentage Error are adopted:

$$\text{RMSE} = \sqrt{\frac{\sum_{i=1}^{n}(y^i_{\text{PCR3BP}} - y^i_{\text{GPR}})^2}{n}} \tag{2.22}$$

and

$$\text{MAPE} = \frac{100\%}{n}\sum_{i=1}^{n}\left|\frac{y^i_{\text{PCR3BP}} - y^i_{\text{GPR}}}{y^i_{\text{PCR3BP}} - x^i_{\text{PCR3BP}}}\right| \tag{2.23}$$

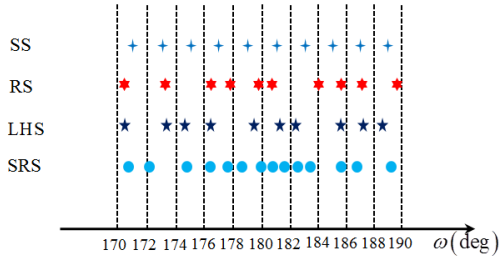


Figure 2.5: Illustration of training samples using different distributions.

In principle, an increment of the number of training samples can be expected to contribute to the reduction of errors because the GPR model is offered more empirical information. So the acceptable error level serves for choosing the size of the training dataset.

The procedure of building a GPR-based Gravity Assist Mapping is summarized in the following steps:

(1) Define the mass ratio for a specific three-body system. The Sun-(Earth+ Moon)-spacecraft system is taken as an example with a mass ratio of 3.036×10^{-6}. The mass ratio is the only key parameter in the PCR3BP, so it is easy to implement the GPR-based Gravity Assist Mapping for different three-body systems;

(2) Construct the covariance function. RQARD and SUM (Equations 2.16 and 2.18) are used and analyzed here;

(3) Generate training inputs. The input space defined in this work selects a broader boundary for flyby passage, instead of considering only the distant encounters outside the Hill sphere. Training set sampling uses four different distributions as shown in Figure 2.5: Systematic Sampling (SS), Random Sampling (RS), Latin Hypercube Sampling (LHS), and Stratified Random Sampling (SRS), further described in Section 2.4.1;

(4) Calculate training outputs using the numerical PCR3BP propagation. The termination conditions on time and the Tisserand parameter are employed to guarantee the consistency of the training outputs;

(5) Optimize the hyper-parameters based on the training samples, i.e., maximize the outcome of Equation 2.15. Conjugate gradient ascent is used to find optimal values of the hyper-parameters. This is used in a multi-start approach to avoid being trapped in local minima (more precisely: the gradient procedure is initiated at 10 different initial combinations of hyper-parameter values);

(6) Determine the optimal size of the set of training data. This is done by adding training samples gradually until a stable error, i.e., the difference between the outputs of the numerical PCR3BP and the GPR models on the test dataset, is observed.

(7) Predict the output given a new input. Based on the training samples, the selected covariance function and the corresponding optimized hyper-parameters, the prediction can be obtained.

2.4.1. ACCURACY

We generate a test set meeting the conditions of Equation 2.19 to assess the accuracy of the method. The test set consists of 500 (N^*) input vectors randomly distributed over the input space, and is kept fixed for all evaluations in a certain application. Their corresponding outputs are obtained by numerical propagation in the PCR3BP model and by the GPR model. To evaluate the performance for each model, both the MAPE and RMSE will be calculated using Equations 2.22 and 2.23. The Sun-(Earth+Moon)-spacecraft system is considered for all applications.

SYSTEMATIC SAMPLING

By assuming that the training data can represent the empirical information of the entire input space, we create a dataset by systematic sampling:

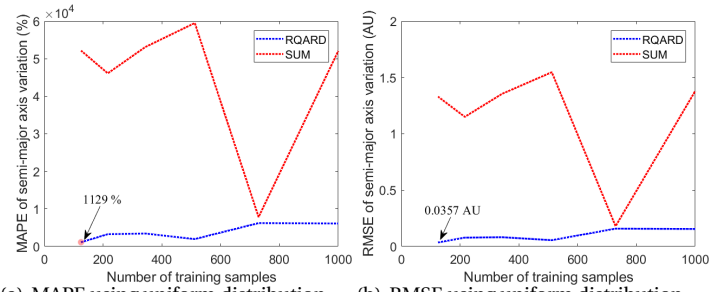
$$\begin{cases} r_p \in [1.00004464 : (1.02 - 1.00004464)/N_{division} : 1.02], \text{AU} \\ r_a \in [1.01 : (2.02 - 1.01)/N_{division} : 2.02], \text{AU} \\ \omega_A \in [170 : (190 - 170)/N_{division} : 190], \text{deg} \end{cases} \tag{2.24}$$

$N_{division} = 6, 7, 8, 9, 10, 11$ indicates to what extent each input feature is partitioned.

With increasing training size, Figure 2.6 shows MAPE and RMSE of the test set on the semi-major axis using the RQARD and SUM covariance functions, respectively. In this case, the RQARD function performs better than SUM, but MAPE is larger than 1000%. Obviously, the systematic sampling fails to offer useful information for the GPR model. This is because the GPR model uses Bayesian inference, a type of statistical inference, which strongly relies on the assumption of random selection Osborne (1942). If this assumption is not satisfied, as is the case with regular sampling, the basis for any GPR model drops out and the output is to be considered as unrealistic.

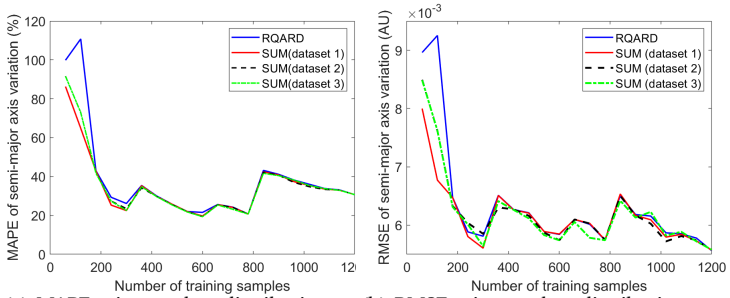
RANDOM SAMPLING

In this test, we distribute the training inputs randomly over the entire input space. MAPE and RMSE results are shown in Figure 2.7. The SUM function has a better performance than the RQARD function. For the case of 780 training samples, SUM obtains the lowest MAPE value of 19.7% and an RMSE of 5.8×10^{-3} AU. We generate another two sets of training samples using different random seed numbers and perform the predictions using the SUM function. The trends of MAPE and RMSE with respect to three training data sets show that the random seed number is not an influential parameter for the level of convergence.



(a) MAPE using uniform distribution. (b) RMSE using uniform distribution.

Figure 2.6: The error of test samples when the training dataset uses systematic sampling.



(a) MAPE using random distribution. (b) RMSE using random distribution.

Figure 2.7: The error of test samples when the training dataset has a random distribution.

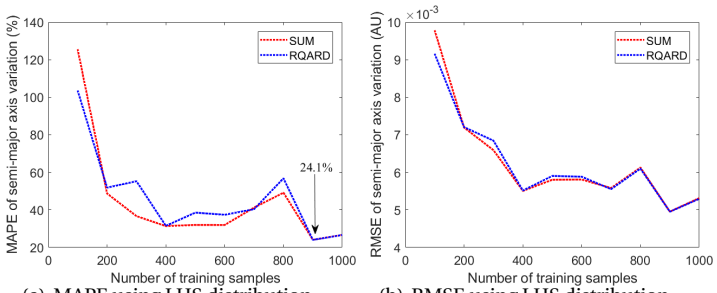
LATIN HYPERCUBE SAMPLING

LHS is a statistical method to generate random samples especially for multi-dimensional problems McKay et al. (1979). This method is generally used to avoid clustering of random samples in a specific subspace. Shown in Figure 2.8, the SUM function captures the model mechanics more quickly than RQARD. In addition, the SUM function has a lowest MAPE value of 24.1 % and an RMSE of 4.9×10^{-3} AU at a training data size of 900. The LHS has a poorer accuracy than random sampling with respect to MAPE.

STRATIFIED RANDOM SAMPLING

Figure 2.9 shows the number of MAPE values larger than 10% for the subspaces of ω_A using random sampling. Due to the stronger flyby effects around $\omega_A = 180°$ according to Figure 2.2, the error for these test points is worse. More training points can be added for the subspace of $\omega_A \in [175°, 185°]$ to gain more information over this subspace.

In order to improve the performance of prediction for the subspace of $\omega_A \in [175°, 185°]$, the sampling density of this range can be taken as twice as that of the remaining area: an SRS method. The training size is increased by 60 every time. For each 60 training samples, there are 10 samples within $[170°, 175°]$, 40 samples within $[175°, 185°]$ and 10 samples within $[185°, 190°]$. Figure 2.10 shows the corresponding RMSE results.



(a) MAPE using LHS distribution. (b) RMSE using LHS distribution.

Figure 2.8: The error of test samples when the training dataset has an LHS distribution.

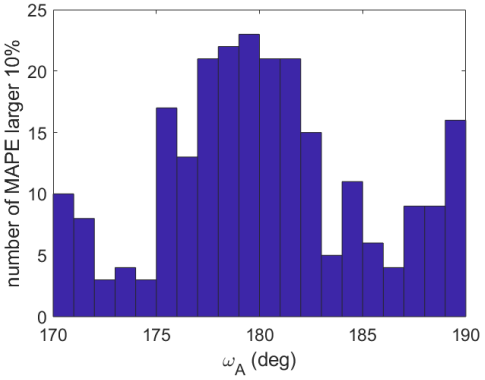
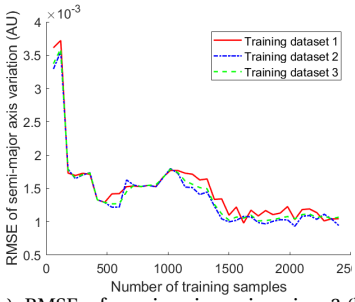


Figure 2.9: Number of MAPE larger than 10% based on random sampling.

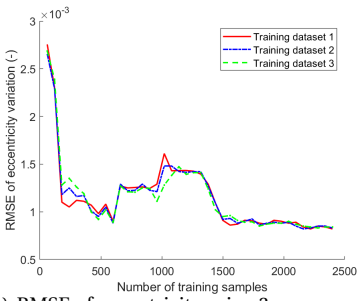
The performance becomes stable when using an increasing number of training samples. In Figures 2.10(a) and 2.10(b), the RMSE of these training datasets converges to the same level with a training data size larger than 1500. The prediction of ω needs more training samples than that of a and e. This indicates that the GPR function needs more information to learn the change of ω. We generate another two groups of training samples using the same SRS method with different random seed numbers. For the prediction of a, the standard deviation among these three groups is about 1.0×10^{-4} AU when a number of more than 1500 training samples are used. The random seed number for generating training dataset is a negligible influence factor for the performance of GPR models.

SUMMARY OF SAMPLING METHODS

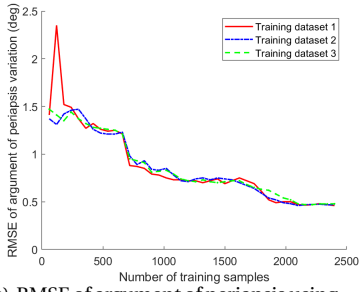
We used four types of sampling methods for the generation of the training samples for comparison. The prediction using RS performs better than that of SS and LHS. Based on RS, we developed an SRS specifically for this case ($h > 300$ km) due to the larger errors for the test samples in the subspace of $\omega_A \in [175°, 185°]$. The RMSE using SRS is reduced to less than 1/4 of that of RS. By using different training datasets, the training process of the GPR model is illustrated to be insensitive to the random seed numbers.



(a) RMSE of semi-major axis using 3 groups of training samples.



(b) RMSE of eccentricity using 3 groups of training samples.



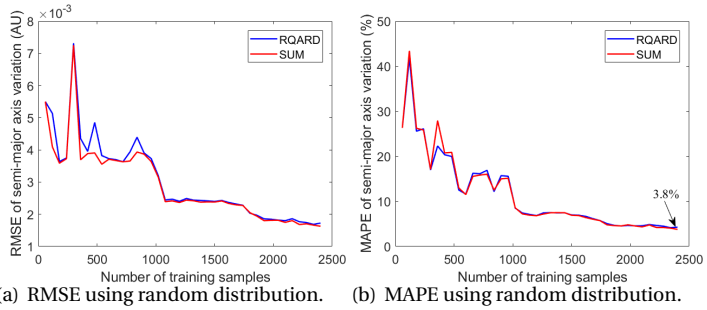
(c) RMSE of argument of periapsis using 3 groups of training samples.

Figure 2.10: The RMSE of test samples when the training dataset adopts a stratified random sampling. The SUM covariance function is used.

2.4.2. FLYBYS OUTSIDE THE SPHERE OF INFLUENCE OF THE SECONDARY

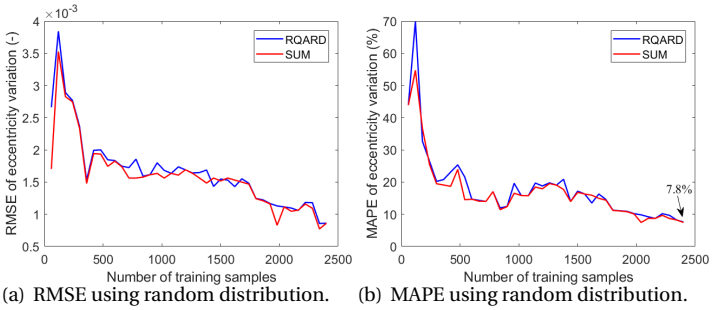
In this application, the periapsis of the initial osculating orbits of the spacecraft are chosen such that they are above the sphere of influence of the secondary (SoI): $r_p > 1.0062$ AU (i.e., $r_p > r_{SoI} + r_2$) in the system of Sun-(Earth+Moon)-spacecraft. Figure 2.11 shows the errors of the semi-major axis predictions after the flybys. After some fluctuations for training data sizes between 0 and 1000, the error becomes gradually smaller and stable until the data size reaches 2400. The RMSE is selected as the main criterion for optimizing hyper-parameters. Using the same set of hyper-parameters, MAPE could have a slightly different trend compared to that of RMSE. The performance of predicting eccentricity is presented in Figure 2.12. The RMSE of e decreases to a value of 1.0×10^{-3} using 2000 training samples. The maximum eccentricity e_A among the test cases is 0.33. In Figure 2.13, the RMSE of ω becomes lower than 0.4° with more than 1800 training samples. The convergence of optimizing hyper-parameters for the prediction of the output state is illustrated in Figure 2.14: the hyper-parameters l_a, l_e and l_ω are selected. When the training data size becomes larger than 1000, the GPR model converges gradually to the optimum. Although the optimization process converged at negative values for the hyper-parameters eventually, this has no consequences since the quadratic value of them is used (Equation 2.17). RS is used in this case because a significant difference

of error between different subspaces is not observed.



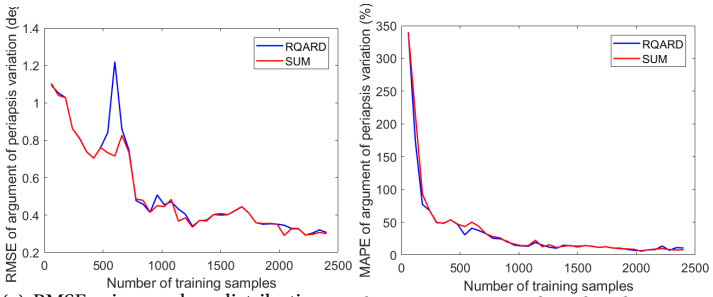
(a) RMSE using random distribution. (b) MAPE using random distribution.

Figure 2.11: The semi-major axis error of test samples when the training dataset has a random sampling. The periapsis of initial osculating orbit of the spacecraft is outside the sphere of influence of the secondary.



(a) RMSE using random distribution. (b) MAPE using random distribution.

Figure 2.12: The eccentricity error of test samples when the training dataset has a random sampling. The periapsis of initial osculating orbit of the spacecraft is outside the sphere of influence of the secondary.



(a) RMSE using random distribution. (b) MAPE using random distribution.

Figure 2.13: The ω error of test samples when the training dataset has a random sampling. The periapsis of initial osculating orbit of the spacecraft is outside the sphere of influence of the secondary. The best MAPE obtained is 5.7%.

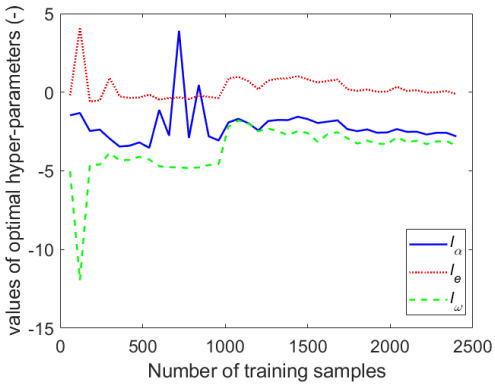


Figure 2.14: Evolution of optimal hyper-parameters. The flybys are outside the SoI of the secondary.

2.4.3. FLYBYS OUTSIDE THE HILL SPHERE OF THE SECONDARY

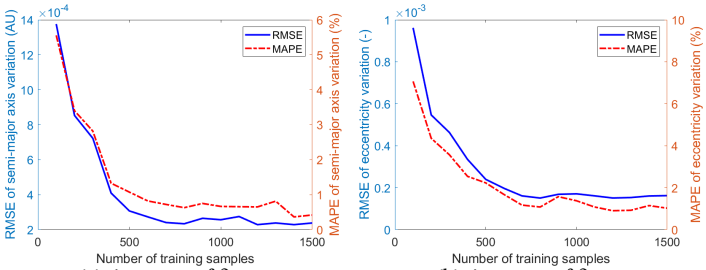
For the prediction of distant flyby effects, this application investigates the condition of the radius of periapsis being larger than 1.01 AU (i.e., $r_p > r_2 + r_{Hill}$). Figure 2.15 shows the accuracy of prediction with an increasing number of training samples. For three orbital elements, the curves flatten out gradually after using 1300 training samples. The accuracy is better than the case of SoI. MAPE of δa and δe is no more than 1%. The MAPE of $\delta\omega$ has been improved by a factor of two. It shows that the low-energy case is easier for GPR to learn. This is mainly because the flyby effects at low energies are less complex than the previous applications, e.g., the retrograde flybys disappear at such energies Campagnola et al. (2012). Therefore, less training samples are required to obtain a reliable model.

The comparison between the performance of our GPR-GAM and the Keplerian Map (KM) model of Ross et al. (Ross & Scheeres 2007) (fully reproduced here) is shown in Table 2.1. 1500 training samples are considered in our GPR model and the SUM covariance function is selected. The test set has 500 samples with inputs randomly distributed in the input space and outputs numerically calculated by the PCR3BP propagation. Both tests are performed in MATLAB@ 2017b using a laptop of Core i7 CPU and 8.00 GB RAM. The RMSE of predicting a using GPR-GAM is only 1/6 of that of the KM. In addition, the GPR-GAM improves the accuracy of predicting ω to a level of 4.72×10^{-3} rad. The KM fails to quantify the variation of ω accurately due to simply evaluating ω in terms of orbital period change. The CPU time of prediction spent on a single test sample is the same for both a and ω. KM is much slower in predicting a because of using the numerical integration but is faster in the prediction of ω.

The time efficiency of the GPR method is demonstrated for up to 300,000 test samples in Figure 2.16. When using 1500 training samples for the case outside the Hill sphere, the total prediction time increases linearly with the number of test samples. It costs 0.32 s for the evaluation of 300,000 samples. The prediction time per sample is always lower than 1.2×10^{-6} s. For the case outside the SoI in Figure 2.16 (b), the computational effort is bigger because of using 2400 training samples. It spends only 1.14 s for the same number

of test samples and the CPU time per sample is 3.8×10^{-6} s. When applying GPR to the case outside the Hill sphere, the generation of the training samples (0.28 s per sample) and training the GPR model (1530 s for 10 multi-starts) cost 1950 s in total. However, the GPR model is off-the-shelf once the training process is completed, and the values in Table 2.1 give an accurate estimation of the computation time needed to predict the flyby effect of a single input.

Table 2.2 summarizes the performance of GPR for various applications. The accuracy of the GPR model becomes better when the closest passage of the spacecraft moves further from the secondary. The radius of the Hill sphere is close to that of the SoI but the accuracy improves especially for a and e. The prediction of ω is the worst which can be further investigated.



(a) Accuracy of δa.

(b) Accuracy of δe.

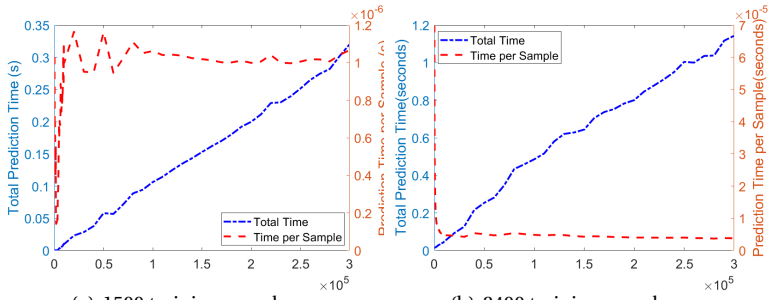
(c) Accuracy of $\delta\omega$.

Figure 2.15: The RMSE and MAPE of test samples when the training dataset adopts a stratified random sampling. The SUM covariance function is used.

Table 2.1: The performance of GPR-GAM compared to Ross & Scheeres (2007) (fully reproduced here).

	RMSE		Prediction CPU Time* [s]	
	a [AU]	ω [rad]	a	ω
GPR	2.38×10^{-4}	4.72×10^{-3}	1.16×10^{-6}	1.16×10^{-6}
Ross & Scheeres (2007)	1.42×10^{-3}	1.57	0.064	1.04×10^{-6}

* The computational time is obtained by taking the mean value of 1000 repeated experiments. The value is the CPU time spent on one test sample.



(a) 1500 training samples. (b) 2400 training samples.

Figure 2.16: Time efficiency of the GPR-GAM for one orbital element. Equation 2.13 is used for the prediction of multiple inputs. The cost of generating training samples is not included.

Table 2.2: Summary of the GPR performance for all applications (SUM covariance function).

Flyby distance	Distribution	MAPE [%] a	e	ω	RMSE a [AU]	e [-]	ω [rad]
>300 km altitude	RS	19.7	85.6	97.5	0.0056	0.0062	0.0511
	SRS	5.6	22.3	24.5	0.0009	0.0008	0.0082
>SoI	RS	3.8	7.8	5.7	0.0003	0.0007	0.0051
>Hill sphere	RS	0.4	1.0	3.1	0.0002	0.0002	0.0047

The number of training samples N for the three cases of flyby distance are 2400, 2400, 1500 respectively. The SRS is not applied to the second and third cases because the error in different parts of the input space is similar. The values of corresponding hyper-parameters are given in Appendix B.

2.5. CONCLUSIONS

A model of the Gravity Assist Mapping was proposed based on a machine-learning method called Gaussian Process Regression. It was trained by a dataset consisting of thousands of samples generated using numerical integration in the planar CR3BP framework. The generation of training data takes into account elliptical orbits of low, moderate and high eccentricity. We determined the size of training samples by choosing a stable RMSE on the test dataset. A covariance function (SUM) was developed combining the cosine term and the rational quadratic term with automatic relevance determination (RQARD), which captures well the dynamics of flybys. The GPR-based model can assess the flyby effect more efficiently given the initial condition of a particle compared to methods based on numerical integration. The model for various planar CR3BP systems can be built by simply changing the mass ratio parameter. Using a machine-learning method, this work investigated and predicted the flyby effect instead of numerical integration of the equations of motion in previous contributions. The domain of applicability is beyond that of the Keplerian map. The GPR-MAP requires to be trained to model the flyby effect of trajectories belonging to different domain.

The results demonstrate that the GPR-based Gravity Assist Mapping has a good performance on accuracy. Compared to previous work in literature, significant improvements have been made, in particular when using a combined covariance function and stratified random sampling. This can contribute to two main studies in astrodynamics. On the one hand, the characteristic of the GA in the CR3BP can be investigated by a great deal of accurate data produced by the GPR model, providing a deep understanding of the third-body effect. On the other hand, the GPR model can be considered to design multi-flyby missions, which takes the advantage of a high efficiency to update the post-flyby status. The developed GPR-GAM is still a prototype. It has some limitations, such as being unable to classify collision orbits in test samples because they are removed in the training samples. The post-flyby status near the neighbourhood of the secondary might be neglected using the Tisserand termination condition. Further improvements include developing a classification model to categorise collision orbits given an arbitrary initial condition and using a distance termination condition.

2.6. ACKNOWLEDGEMENT

The author Liu gratefully appreciates the China Scholarship Council (CSC) for its financial support.

2.7. APPENDIX A: MINIMUM DISTANCE BETWEEN SPACECRAFT AND SECONDARY

All variables are non-dimensionalized using $(M_1 + M_2)$ for the mass, the semi-major axis of the secondary a_{P2} for the length, and $\sqrt{a_{P2}^3/(G(M_1+M_2))}$ for the time. For the circular orbit of the secondary, the true anomaly θ_{P2} is set to be 0 at the position of the positive X-axis. The initial true anomaly of the secondary $\theta_{P2}^{initial}$ is selected such that it arrives at the negative X-axis when the spacecraft (in terms of initial osculating orbit) reaches periapsis. This is specified by:

$$\theta_{P2}^{initial} = \pi(1 - (a_A)^{\frac{3}{2}}) \tag{2.25}$$

where a_A is the semi-major axis of the initial osculating orbit of the spacecraft. The distance d between the spacecraft and the secondary is defined using the law of cosines:

$$d = \sqrt{r_1^2 + r_{P2}^2 - 2r_1 r_{P2} cos(\gamma)} \tag{2.26}$$

where r_1 represents the distance between the spacecraft and the primary, r_{P2} the distance between the secondary and the primary, γ denotes the angle contained between r_1 and r_{P2}. Since r_{P2} has a constant value of 1, Equation 2.26 is simplified to:

$$d = \sqrt{1 + r_1^2 - 2r_1 cos(\gamma)} = \sqrt{f(r_1, \gamma)} \tag{2.27}$$

Assuming a fixed value of γ, the partial derivative $\frac{\partial f(r_1,\gamma)}{\partial r_1}$ is:

$$\frac{\partial f(r_1, \gamma)}{\partial r_1} = 2r_1 - 2cos(\gamma) \tag{2.28}$$

The value of the right-hand side is always larger than zero given the initial conditions of this paper. The function $f(r_1, \gamma)$ is monotonically increasing. When r_1 equals the radius of periapsis r_p and $\gamma = 0$, d has the minimum value $| r_p - 1 |$. This condition corresponds to $\omega = \pi$.

2.8. APPENDIX B: VALUES OF HYPER-PARAMETERS

The optimized values of the hyper-parameters of the models in Table 2.2 are given below.

Table 2.3: The optimized hyper-parameters of GPR-GAM for the first case in Table 2.2.

Output	σ	α	l_a	l_e	l_ω	p	h
a	-1.2450	-1.2043	2.9608	0.7480	-4.2359	0.5279	-11.7555
e	-3.3324	-2.4650	4.8943	1.7674	-8.8055	-0.0180	-10.4190
ω	-2.1321	-2.1245	1.7252	1.2615	-3.2145	-0.2121	-7.2210

Table 2.4: The optimized hyper-parameters of GPR-GAM for the second case in Table 2.2.

Output	σ	α	l_a	l_e	l_ω	p	h
a	-2.0214	-2.5534	0.2000	-0.8179	3.4195	3.4419	-0.2592
e	-1.7113	-2.0229	0.1768	-0.7881	3.6202	2.8485	-0.2926
ω	-2.7010	-2.9755	0.5394	2.1231	1.4008	-0.3179	-1.1486

Table 2.5: The optimized hyper-parameters of GPR-GAM for the third case in Table 2.2.

Output	σ	α	l_a	l_e	l_ω	p	h
a	-1.9506	-3.3963	-3.3770	-0.3370	-3.4057	1.9674	-3.1555
e	-1.3524	-2.2365	0.2718	-4.2451	-1.9327	4.0959	-4.2971
ω	-1.8572	-2.6009	0.7222	3.4731	-1.1252	0.7433	0.8022

Table 2.6: The optimized hyper-parameters of GPR-GAM for the fourth case in Table 2.2.

Output	σ	α	l_a	l_e	l_ω	p	h
a	-1.8168	-2.4690	-0.2082	-1.5223	3.2384	0.9166	0.5527
e	-1.3950	-2.3792	-0.0306	-1.3639	2.2226	2.9447	-0.3096
ω	-1.4090	-2.9224	0.9743	4.7129	0.0638	-0.7502	0.2507

3

A Gravity Assist Mapping for the Circular Restricted Three-Body Problem using Gaussian Processes

Abstract Inspired by the Keplerian Map and the Flyby Map, a Gravity Assist Mapping using Gaussian Process Regression for the fully spatial Circular Restricted Three-Body Problem is developed. A mapping function for quantifying the flyby effects over one orbital period is defined. The Gaussian Process Regression model is established by proper mean and covariance functions. The model learns the dynamics of flyby's from training samples, which are generated by numerical propagation. To improve the efficiency of this method, a new criterion is proposed to determine the optimal size of the training dataset. We discuss its robustness to show the quality of practical usage. The influence of different input elements on the flyby effects is studied. The accuracy and efficiency of the proposed model have been investigated for different energy levels, ranging from representative high- to low-energy cases. It shows improvements over the Kick Map, an independent semi-analytical method available in literature. The accuracy and efficiency of predicting the variation of the semi-major axis are improved by factors of 3.3, and 1.27×10^4, respectively.

3.1. Introduction

For deep space missions, a flyby along a planet or a major moon is a common technique to save propellant. The state of the spacecraft is changed effectively due to the gravitational pull of these massive bodies. In the preliminary design of an interplanetary mission, the patched-conics model is typically used (Bate et al. 1971, Broucke 1988,

This chapter has been accepted in Advances in Space Research.

Longuski & Williams 1991, Strange & Longuski 2002). This model has a crucial assumption: the gravity of the flyby body (e.g., Mars) is considered if and only if the spacecraft comes into the sphere of influence of that body. The gravity effect of other objects (in particular, the Sun) can be ignored; the original problem is split up in a succession of elementary two-body problem formulations. The patched-conics model simplifies the preliminary mission design process. However, it has a limitation in describing the dynamics in more detail, in particular for low-energy cases with phenomena like invariant manifolds, which do exist in a three-body formulation. The special dynamics described by the three-body formulation have proven to be essential to design low-energy trajectories or study the movement of asteroids between Mars and Jupiter (Beutler 2004, Campagnola et al. 2012, 2014).

In order to reduce the degree of difficulty of a most general three-body formulation, the Circular Restricted Three-Body Problem (CR3BP) can be introduced (Szebehely 1967). Here, it is assumed that the mass of one of the three bodies can be neglected and the other massive bodies move in circular orbits about the barycenter of the system. Considering the Sun-Earth-spacecraft system, this formulation takes the gravitational attraction of the Sun and the Earth into account simultaneously. The domain of a flyby in the CR3BP is expanded beyond the sphere of influence of the Earth.

The Keplerian Map (KM) is a common method to study the flyby effects in the CR3BP. Based on perturbation theory, this method was initially developed to investigate the dynamics of comets around Jupiter (Petrosky & Broucke 1987, Chirikov & Vecheslavov 1989, Conley 1968, Malyshkin & Tremaine 1999, Zhou et al. 2000). Ross & Scheeres (2007) studied the KM in the system of a planar CR3BP, which models the flyby effect over one orbital period using an energy-kick function. It works effectively for an energy level of around 3.0 for the Jacobi constant. For a quantitative investigation in the spatial framework (CR3BP), a three-dimensional KM was developed to solve initial-value problems (Peñagaricano Muñoa & Scheeres 2010). The accuracy of this method is reasonably good for low three-body energy levels only. Alessi & Sánchez (2016) expanded this method and derived a formulation to estimate the variation of orbital elements due to the perturbation of a third body. It then may be used to study the effects of a flyby. The results are obtained by applying the classical Lagrange planetary equations and performing a first iteration of Picard's method over one orbital period. This formulation assumes a sufficiently small mass ratio and may suffer from singularity problems for low eccentricity and inclination. Neves et al. (2018) analysed various ways of modelling the third-body effect and tried to obtain a fully analytical solution. They showed the accuracy of these models within certain fields of application. It is well known that the 3BP is not solvable in an analytically closed form. Many attempts have been made to find closed analytical solutions for the variation of the orbital elements of the massless body under specific conditions in the CR3BP. The techniques mentioned above are semi-analytical and employ numerical methods to generate solutions. The Flyby Map developed by Campagnola et al. (2012) is also a typical approach to study the CR3BP, which is fully numerical and valid for a wide range of energy levels. It extended the functionality of the Tisserand graph over the applicability of the patched-conics model. According to the spacecraft velocity relative to the secondary, the flybys are categorised into two types: 'direct' and 'retrograde' flybys. In terms of the variation of semi-major axis, the direct flybys are shown

to be more efficient than the latter.

This paper is inspired by the works on the Keplerian Map and Flyby Map. Instead of solving the analytical forms of flyby effects in the CR3BP, we develop a Gravity Assist Mapping (GAM) based on Gaussian Process Regression (GPR) to quantify flyby effects in a full three-dimensional situation. Specifically, the flyby refers to an encounter of the massless body with the second-largest body. GPR is a supervised machine-learning method, which was developed by Krige (1951) for mine valuation. For a complete three-body problem, the technique of Deep Neural Network was used to predict the position of each body within a fixed duration (Breen 2019). The prediction of velocity was not investigated. Instead of using Cartesian coordinates, the GAM models use Keplerian orbital elements similar to the Keplerian Map. The size of the training data set for a DNN model requires around 10,000 samples. Moreover, the structure, the number of layers, the number of nodes, the activation function and the pooling function, etc. need to be determined. The model selection of the GPR method is simpler than that of the DNN. In addition, the GPR models work accurately at a lower level of training data size (Rasmussen & Williams 2006). In the field of astrodynamics, GPR has been used to evaluate the accessibility of asteroids, to model the gravity field of small bodies, and to design low-thrust trajectories (Shang & Liu 2017, Gao & Liao 2019, Bouwman et al. 2019). Liu et al. (2021) used GPR to evaluate flyby effects; their model was developed in a planar CR3BP and performs more efficiently and accurately than the work of Ross & Scheeres (2007). The complexity of the CR3BP increases significantly when going from a planar model to a full spatial model. In the current work, we develop a new GPR-based GAM to solve this. The Keplerian Map for approximating a particle's motion was initially developed for low three-body energies (Ross & Scheeres 2007). Here, we aim at developing an approach which is applicable for a wider range of energy levels. The energy of the considered cases ranges from values lower than that associated with the L1 libration point to values larger than that associated with L4/L5. Several thousands of training samples are required for the GPR model to learn the dynamics. The prediction of output given an arbitrary input will be compared with a previous semi-analytical method and numerical integration methods. The computational effort is expected to be further reduced particularly when a large group of initial conditions needs to be assessed.

3.2. CIRCULAR RESTRICTED THREE-BODY PROBLEM

The CR3BP describes the dynamics of three point masses: two massive bodies P_1 and P_2 and a third body P_3 with masses M_1, M_2 and M_3, respectively, which meet the condition $M_1 > M_2 \gg M_3$. Throughout this paper, flybys in the Sun-(Earth+Moon)-spacecraft system are investigated and the primary P_1, the secondary P_2 and massless particle P_3 are the Sun, the Earth+Moon and the spacecraft, respectively. In the rotating reference frame, the origin is set at the barycenter of P_1 and P_2, and the X-axis is aligned with the direction to P_2. The primaries P_1 and P_2 rotate around the Z-axis. The equations of

motion for the spacecraft are described as (Szebehely 1967):

$$\begin{cases} \ddot{x} - 2\dot{y} = x - \dfrac{(1-\mu)}{r_1^3}(x+\mu) - \dfrac{\mu}{r_2^3}(x-1+\mu) \\ \ddot{y} + 2\dot{x} = y - \dfrac{(1-\mu)}{r_1^3}y - \dfrac{\mu}{r_2^3}y \\ \ddot{z} = -\dfrac{(1-\mu)}{r_1^3}z - \dfrac{\mu}{r_2^3}z \end{cases} \tag{3.1}$$

where x, y and z represent the normalized position coordinates in the barycentric Cartesian reference frame, and

$$\begin{cases} \mu = M_2/(M_1+M_2) \\ r_1 = \sqrt{(x+\mu)^2 + y^2 + z^2} \\ r_2 = \sqrt{(x-(1-\mu))^2 + y^2 + z^2} \end{cases} \tag{3.2}$$

μ is defined as the mass parameter which differentiates between different implementations of the CR3BP, and is equal to 3.036×10^{-6} for the system of Sun-(Earth+Moon)-spacecraft. r_1 and r_2 are the distances between P_3 and the primaries P_1 and P_2, respectively.

A closed-form solution of Equation 3.1 does not exist; numerical integration methods are typically used to solve the problem. Some attempts were made to find analytical solutions for the variation of the orbital elements of P_3 (Ross & Scheeres 2007, Alessi & Sánchez 2016, Neves et al. 2018). Based on the concept of KM developed by Ross & Scheeres (2007), the current work studies a new approach to quantify the flyby effects for a wide range of Jacobi energy levels.

3.2.1. JACOBI CONSTANT

The CR3BP has an analytical integral of motion which is constant along the trajectory of the spacecraft. The value of this Jacobi constant C_J is fully determined by the position and velocity of the spacecraft. In the rotating reference frame, the Jacobi constant is defined as (Beutler 2004):

$$C_J = x^2 + y^2 + \frac{2(1-\mu)}{r_1} + \frac{2\mu}{r_2} - (\dot{x}^2 + \dot{y}^2 + \dot{z}^2) + \mu(1-\mu) \tag{3.3}$$

The Jacobi constant reflects the energy level of the trajectory of the spacecraft. A high value of C_J represents a low energy level. The region that is accessible for the spacecraft is related to the Jacobi constant for a given mass ratio. The Jacobi constant associated with the Lagrange libration points allows to characterize the accessible region for different cases. For $\mu = 3.036 \times 10^{-6}$, the values for the Jacobi constant of these points are $C_{L1} = 3.000898$, $C_{L2} = 3.000893$, $C_{L3} = 3.000003$, $C_{L4/L5} = 2.999997$. For some values of C_J, there are inaccessible regions which divide the accessible regions (i.e., Hill's regions) into different parts. The boundaries of these regions are zero-velocity surfaces. If the value of C_J is larger than C_{L1}, a particle remains in its initial region. When $C_{L1} > C_J > C_{L2}$,

it is possible for the spacecraft to move to the vicinity of the other primary if it started in the vicinity of either of the primaries. Transitions between the interior and exterior regions are not possible yet. This becomes possible when $C_J < C_{L2}$ (for details see Beutler (2004)). The current work focuses on the motion of a spacecraft which starts from the exterior region as shown in Figure 3.1.

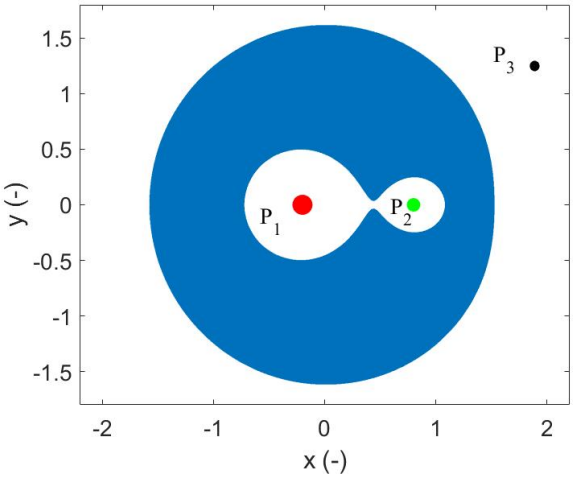


Figure 3.1: Hill's region when $C_{L1} > C_J > C_{L2}$.

3.3. GRAVITY ASSIST MAPPING

3.3.1. MAPPING FUNCTION

In this paper, we aim to develop a GPR model to quantify the flyby effects in the framework of a CR3BP, i.e., a GPR-based GAM (GPR-GAM). The mapping function of the GPR-GAM is described as:

$$\begin{aligned} &F_{\text{GPR-GAM}} : \boldsymbol{x} \mapsto \boldsymbol{y} \\ &\text{or } [a, e, i, \omega, \phi] \mapsto [\delta a, \delta e, \delta i, \delta\omega, \delta\Omega] \end{aligned} \tag{3.4}$$

where the vector $\boldsymbol{x}$ represents the orbital elements of the spacecraft before a flyby and $\boldsymbol{y}$ the variation of orbital elements due to that flyby, after one orbital period of the initial osculating orbit. a, e, i, ω, Ω are osculating semi-major axis, eccentricity, inclination, argument of pericenter and longitude of the ascending node of the spacecraft with respect to the primary. Particularly, the angular phasing angle $\phi = \tan^{-1}(\tan(\omega)\cos(i)) + \Omega$ shown in Figure 3.2 describes the relative position between the spacecraft and the secondary (Alessi & Sánchez 2016). When investigating the effects of a flyby, δa and δe have an explicit response to ϕ. The reference system will be detailed below.

3.3.2. CONFIGURATION OF FLYBY

In Equation 3.4, the input and output are defined in an inertial reference frame. Figure 3.2 shows the orbit of a spacecraft before a flyby in the Sun-(Earth+Moon)-spacecraft sys-

tem. The origin is located at the Sun, and the Earth revolves around it in the XY-plane, so that the reference plane is the ecliptic plane. Here, the input vector $\boldsymbol{x}$ represents the osculating orbital elements of the spacecraft at apoapsis. The angular phasing ϕ denotes the angle between the projection of the Sun-periapsis line of the spacecraft onto the XY-plane and the positive X-axis. The spacecraft starts from apoapsis with a true anomaly of $-\pi$, and the true anomaly of the Earth is initialized such as to reach the positive X-axis when the spacecraft has travelled half of its unperturbed orbital period.

The GPR is a supervised learning method which requires training samples consisting of input and output. In this paper, the boundary of the input space is defined such to obtain obvious flyby effects. In order to quantify the distance between the spacecraft and the secondary properly, we define a boundary based on the radius of periapsis r_p and the radius of apoapsis r_a, and then transform these into a and e.

The boundary is specified in Table 3.1. Based on this input space, the Jacobi constant ranges from 0.512176 to 3.001196, which covers energy levels from low to high (Ross & Scheeres 2007). The value of i only takes prograde orbits into account, which is the expected geometry for normal interplanetary missions. The boundary for ϕ is defined in order to observe an obvious flyby effect; the distance in phasing should not become too large. Figure 3.3 shows the histogram of the Jacobi constant of a group of 5000 randomly generated samples (in line with further calculations). In addition to this, we divide the range of ω into four quadrants:

$$\omega = \{\omega^{Q1}, \omega^{Q2}, \omega^{Q3}, \omega^{Q4} | (0, 90], (90, 180], (180, 270], (270, 360] (\text{deg})\}$$

Given the same a, e, i and ϕ, the output $\boldsymbol{y}$ is a periodic function in terms of ω:

$$\boldsymbol{y}(\omega | a, e, i, \phi) = \boldsymbol{y}(\omega + 180 | a, e, i, \phi), \omega \in [0, 180] \text{deg} \tag{3.5}$$

In order to achieve an accurate estimate of the flyby effects, two GPR models are built separately for ω^{Q1} and ω^{Q2}. Using the property of periodicity (Equation 3.5), these models can be simply applied to ω^{Q3} and ω^{Q4}. After generating training inputs, the training outputs are obtained through numerical propagation using Equation 3.1, as will be explained in Section 3.4.3.

Table 3.1: Boundary for the input space.

r_p [AU]	r_a [AU]	i [deg]	ω [deg]	ϕ [deg]
[1.00004464, 1.02]	[1.01, 3.03]	[0, 90]	[0, 360]	[-25, 25]

The minimum value of r_p corresponds to a minimum distance of 300 km between the spacecraft and the Earth surface when $\phi = 0$ deg, i.e., both of them arrive at the positive X-axis simultaneously.

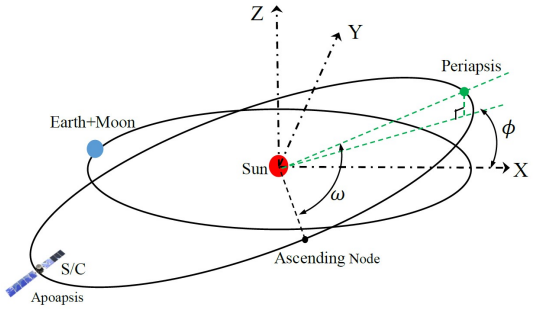


Figure 3.2: Schematics of a three-body system before a flyby. The orbit of the S/C is the initial osculating orbit centered at the Sun in the inertial reference frame; it starts at apoapsis.

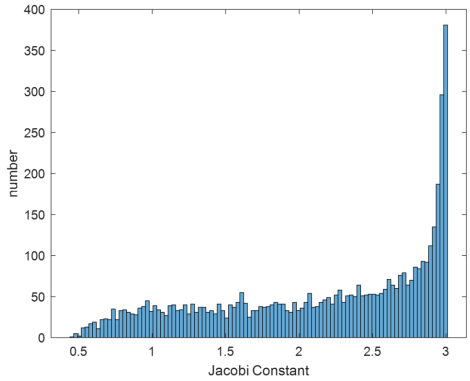


Figure 3.3: Histogram of the Jacobi constant for 5000 samples based on the input space defined in Table 3.1.

3.4. GAUSSIAN PROCESS REGRESSION MODEL

The development of a GPR-GAM considers two major steps: training and predicting. During the training part, the GPR model is built by digesting the empirical information from a certain process. In this paper, this process is the mapping of orbital elements before the flyby to the post-flyby changes of those elements. In the prediction part, the trained model estimates the output for an arbitrary input. We develop five GPR-GAM models with a single output; each model associates the five inputs into one of the outputs in Equation 3.4. This separation of constructing a GPR-GAM model simplifies the training process to a single-objective optimization problem. The mapping function of the basic GPR model is:

$$\begin{aligned} f_{\text{GPR-GAM}} : \mathbb{R}^5 &\rightarrow \mathbb{R} \\ \boldsymbol{x} &\mapsto y \end{aligned} \tag{3.6}$$

where y represents one of the outputs $[\delta a, \delta e, \delta i, \delta\omega, \delta\Omega]$ as in Equation 3.4.

The GPR-GAM model is developed in terms of the properties of a Gaussian Process (GP). A GP is a collection of random variables following their own Gaussian distribution.

A key property of any GP is that any finite number of these random variables forms a joint Gaussian distribution. Instead of simple variables, the GPR-GAM method takes the function $f(\boldsymbol{x})$ as that random variable. Therefore, the development of the GPR-GAM model works in function space. A GP over the function $f(\boldsymbol{x})$ is given as:

$$f(\boldsymbol{x}) \sim GP(m(\boldsymbol{x}), k(\boldsymbol{x}, \boldsymbol{x}')) \tag{3.7}$$

where $\boldsymbol{x}$ and $\boldsymbol{x}'$ are points in the input space; $m(\boldsymbol{x})$ and $k(\boldsymbol{x}, \boldsymbol{x}')$ are the mean function and the covariance function, respectively. Both functions will be explained in Sections 3.4.1 and 3.4.2, respectively. These functions specify the properties of a GP completely through their formulation and hyper-parameters. The hyper-parameters are actually the parameters of the mean function and the covariance function, and are not directly related to the actual data. . 'Hyper-' is used to emphasize that they are characteristics of a non-parametric model (Rasmussen & Williams 2006).

The formulation of $m(\boldsymbol{x})$ and $k(\boldsymbol{x}, \boldsymbol{x}')$ is selected by the authors. Each formulation has a set of corresponding hyper-parameters. The values of these hyper-parameters are initialized randomly. Then the learning procedure starts, which is actually a process of optimizing the hyper-parameters using the empirical information. This information comes from the training dataset D_{train}, defined by:

$$D_{\text{train}} = \{(\boldsymbol{X}, \boldsymbol{Y}) | \boldsymbol{X} = [\boldsymbol{x}_1, \boldsymbol{x}_2, \ldots, \boldsymbol{x}_N], \boldsymbol{Y} = [y_1, y_2, \ldots, y_N]\} \tag{3.8}$$

where $(\boldsymbol{x}_i, y_i)$ represents one training sample ($i = 1, 2, \ldots, N$), and N is the number of training samples. According to the properties of GP, the collection of training outputs $\boldsymbol{Y}$ can be written as a joint Gaussian distribution:

$$\boldsymbol{Y} \sim \mathcal{N}(\boldsymbol{m}(\boldsymbol{X}), \boldsymbol{K}(\boldsymbol{X}, \boldsymbol{X})) \tag{3.9}$$

where $\boldsymbol{m}(\boldsymbol{X})$ represents the mean function and will be elaborated in Section 3.4.1.

$$\boldsymbol{K}(\boldsymbol{X}, \boldsymbol{X}) = \begin{pmatrix} k(\boldsymbol{x}_1, \boldsymbol{x}_1) & k(\boldsymbol{x}_1, \boldsymbol{x}_2) & \cdots & k(\boldsymbol{x}_1, \boldsymbol{x}_N) \\ k(\boldsymbol{x}_2, \boldsymbol{x}_1) & k(\boldsymbol{x}_2, \boldsymbol{x}_2) & \cdots & k(\boldsymbol{x}_2, \boldsymbol{x}_N) \\ \vdots & \vdots & \ddots & \vdots \\ k(\boldsymbol{x}_N, \boldsymbol{x}_1) & k(\boldsymbol{x}_N, \boldsymbol{x}_2) & \cdots & k(\boldsymbol{x}_N, \boldsymbol{x}_N) \end{pmatrix} \tag{3.10}$$

The objective of the training process is to optimize the hyper-parameters such that the GPR-GAM model has the highest possibility to reproduce the ground-truth outputs. This is done by maximizing the marginal likelihood $\log p(\boldsymbol{Y}|\boldsymbol{X})$ using the following equation (Rasmussen & Williams 2006):

$$\log p(\boldsymbol{Y}|\boldsymbol{X}) = -\frac{1}{2}\boldsymbol{Y}^T \boldsymbol{K}^{-1} \boldsymbol{Y} - \frac{1}{2}\log|\boldsymbol{K}| - \frac{N}{2}\log 2\pi \tag{3.11}$$

In principle, the marginal likelihood represents the possibility of obtaining true outputs given the inputs and hyper-parameters. The hyper-parameters that yield a high marginal likelihood would give rise to good predictions. In order to predict the output

y^* for a new input $\boldsymbol{x}^*$, which are called test output and test input respectively, we construct the joint distribution of the training outputs and y^* as

$$\begin{pmatrix} \boldsymbol{Y} \\ y^* \end{pmatrix} \sim \mathcal{N}\left(\begin{pmatrix} \boldsymbol{m}(\boldsymbol{X}) \\ m(\boldsymbol{x}^*) \end{pmatrix}, \begin{pmatrix} \boldsymbol{K}(\boldsymbol{X},\boldsymbol{X}) & \boldsymbol{K}(\boldsymbol{X},\boldsymbol{x}^*) \\ \boldsymbol{K}(\boldsymbol{X},\boldsymbol{x}^*)^T & k(\boldsymbol{x}^*,\boldsymbol{x}^*) \end{pmatrix}\right) \tag{3.12}$$

where $\boldsymbol{K}(\boldsymbol{X},\boldsymbol{x}^*)$ represents the vector of the covariance functions evaluated at all pairs of training inputs $\boldsymbol{X}$ and new input $\boldsymbol{x}^*$.

Based on Bayesian inference, the predicted y^* given $\boldsymbol{x}^*$ has a Gaussian distribution

$$y^*|\boldsymbol{x}^*,\boldsymbol{X},\boldsymbol{Y} \sim \mathcal{N}(\mu(y^*), cov(y^*)) \tag{3.13}$$

where the value of $\mu(y^*)$ is obtained from

$$\mu(y^*) = m(\boldsymbol{x}^*) - K(\boldsymbol{X},\boldsymbol{x}^*)^T \boldsymbol{K}(\boldsymbol{X},\boldsymbol{X})^{-1}(\boldsymbol{Y} - \boldsymbol{m}(\boldsymbol{X})) \tag{3.14}$$

Typically, we take the value of $\mu(y^*)$ as the predicted output (Rasmussen & Williams 2006). In order to improve the efficiency of the prediction effort, we simplify Equation 3.14 by merging matrices (Liu et al. 2021):

$$\mu(y^*) = m(\boldsymbol{x}^*) - K(\boldsymbol{X},\boldsymbol{x}^*)^T \boldsymbol{Q}_{\text{train}} \tag{3.15}$$

where $\boldsymbol{Q}_{\text{train}} = \boldsymbol{K}(\boldsymbol{X},\boldsymbol{X})^{-1}(\boldsymbol{Y} - \boldsymbol{m}(\boldsymbol{X}))$ can be obtained immediately after the training process has been completed, with the obvious advantage that it needs to be calculated only once.

3.4.1. MEAN FUNCTION

When constructing GPR models for the planar CR3BP, a zero-mean function is sufficient for this simpler scenario (Liu et al. 2021). Due to the increased complexity of fully spatial CR3BP dynamics, compared to planar dynamics, we add a non-zero mean function in this work. The option of having a non-zero function offers the possibility to interpret the model and express the empirical information more correctly. The function employed here is a straightforward constant:

$$m(\boldsymbol{x}) = c \tag{3.16}$$

where c is one of the hyper-parameters that will be optimized while training.

3.4.2. COVARIANCE FUNCTION

The supervised machine learning generates predictions using the relationship between samples. The test output is predicted based on the correlation between its corresponding input and all the training inputs. In a GPR method, the covariance function is the most crucial module because it describes this similarity between different samples and plays a key role in interpolation. What remains is the specific formulation of this covariance function, and the number and distribution of training samples. In principle, there is no universal rule for choosing an appropriate covariance function. The choice is typically based on experience, the information about the characteristics of the relation between inputs and outputs provided by training samples, and the error over test samples. For the planar CR3BP, the authors selected a sum covariance function combining

a Rational Quadratic function with Automatic Relevance Determination (RQARD) and a cosine function to learn the flyby effects (Liu et al. 2021). In this new spatial framework, we test nine commonly-used covariance functions in addition to this sum function and elaborate on the three best options with the smallest error (Rasmussen & Williams 2006). Only the best three (based on first evaluations) are detailed here; the others will be mentioned briefly at the end of this section.

The first one is the Neural Network (NN) covariance function:

$$k_{\mathrm{NN}}(\boldsymbol{x},\boldsymbol{x'}) = s^2_{f,\mathrm{NN}} \cdot \sin^{-1}\left(\frac{\boldsymbol{x}\boldsymbol{P}\boldsymbol{x'}^{\mathrm{T}}}{\sqrt{(1+\boldsymbol{x}\boldsymbol{P}\boldsymbol{x'}^{\mathrm{T}})(1+\boldsymbol{x'}\boldsymbol{P}\boldsymbol{x}^{\mathrm{T}})}}\right) \tag{3.17}$$

where $s^2_{f,\mathrm{NN}}$ is the signal variance. The matrix $\boldsymbol{P} = \eta \times I_5$ is a matrix of η times I_5, which represents an identity matrix of size five. Both $s^2_{f,\mathrm{NN}}$ and η are hyper-parameters to be optimized.

The second one is the Squared Exponential covariance function with Automatic Relevance Determination (SEARD):

$$k_{\mathrm{SEARD}}(\boldsymbol{x},\boldsymbol{x'}) = s^2_{f,\mathrm{SEARD}} \cdot \exp\left(\frac{-(\boldsymbol{x}-\boldsymbol{x'})\cdot\boldsymbol{Q}\cdot(\boldsymbol{x}-\boldsymbol{x'})^{\mathrm{T}}}{2}\right) \tag{3.18}$$

where $s^2_{f,\mathrm{SEARD}}$ is the signal variance for this function.

The third one is the aforementioned RQARD:

$$k_{\mathrm{RQARD}}(\boldsymbol{x},\boldsymbol{x'}) = s^2_{f,\mathrm{RQARD}}\left(1+\frac{(\boldsymbol{x}-\boldsymbol{x'})\boldsymbol{Q}(\boldsymbol{x}-\boldsymbol{x'})^{T}}{2\alpha}\right)^{-\alpha} \tag{3.19}$$

Here $s^2_{f,\mathrm{RQARD}}$ and α are the signal variance and shape parameter, respectively.

The latter two covariance functions include an ARD term $\boldsymbol{Q}$, which is a symmetric matrix containing weights

$$\boldsymbol{Q} = \mathrm{diag}\left(\frac{1}{l_a^2}, \frac{1}{l_e^2}, \frac{1}{l_i^2}, \frac{1}{l_\omega^2}, \frac{1}{l_\phi^2}\right) \tag{3.20}$$

where l_a, l_e, l_i, l_ω and l_ϕ, which are called characteristic length-scale hyper-parameters, are weights for the elements a, e, i, ω and ϕ, respectively. This ARD term controls the similarity between samples. Therefore, these five input elements have a different influence on predicting the output of a particular test sample.

In order to show the improvement of using ARD, the performances of another two covariance functions Squared Exponential (SE) and Rational Quadratic (RQ) are also presented in Section 3.5.1. For these two functions without the ARD term, the terms $\boldsymbol{Q}$ in Equations 3.18 and 3.19 have a simplified form $\frac{1}{l^2} \times I_5$.

The other four covariance functions that were initially considered are Piecewise Polynomial, Matérn, and these two formulations using ARD (Rasmussen & Williams 2006). As mentioned earlier, their performance was such that they were excluded from further analysis. It is particularly noteworthy that the sum covariance function is also excluded, which has a good performance for the planar CR3BP studied by the authors previously (Liu et al. 2021). It indicates that the characteristics of the mapping function have changed considerably from the planar CR3BP to the spatial one. A different covariance function is required for the new mapping function, i.e., Equation 3.6.

3.4.3. TRAINING SAMPLES

Training samples are necessary for the supervised learning method to provide empirical information about a system. In this paper, a training sample consists of a training input of five orbital elements and an output of the variation of a specific element due to a flyby (Equation 3.6). For the GPR-GAM model, the inputs are generated randomly using the uniform distribution in the input space. The range of values of each dimension is different (Table 3.1) which would let the large-value input element dominate the results. To avoid this, the inputs are normalized to $[0,1]$.

Given a training input, the final state of the spacecraft is obtained by numerically propagating the motion as described by Equation 3.1 over one unperturbed orbital period of the initial osculating orbit. The propagation employs an RK4 integration in MATLAB@ 2018b (The Mathworks Inc. 2018) using a relative error tolerance of 1.0×10^{-9} and an absolute error tolerance of 1.0×10^{-12}, leading to uncertainties of 0.15 km (i.e., 1.0×10^{-7} % on the scale of the problem) after one orbital period of integration. Starting from the exterior Hill's region, it is possible for the spacecraft to be temporally trapped by the secondary. The training output, i.e., the variation of Keplerian orbital elements, is calculated by subtracting that input from the final osculating orbital elements of the spacecraft.

The training process then uses a conjugate gradient method to look for the optimal values of hyper-parameters maximizing the outcome of Equation 3.11 (Rasmussen & Williams 2006). Based on the theory of Bayesian inference, the GPR-GAM model has a high possibility of producing results close to the training outputs with optimal hyper-parameters. Training samples are also used by the GPR-GAM model when predicting the output for a new input $\boldsymbol{x}^*$, as in the matrix $\boldsymbol{X}$ denoted in Equation 3.14.

3.4.4. ACCURACY EVALUATION AND TRAINING DATA SIZE

To evaluate the performance of a GPR-GAM model, a set of test samples is generated. Following the same way of generating training samples, a group of 500 test samples T are generated and used. This set is kept constant for all further evaluations. Exact test outputs y_{CR3BP}'s (still, the effects δa etc. of the flyby) obtained by the numerical CR3BP propagation are employed as benchmark and compared with results $y_{\text{GPR-GAM}}$'s obtained by GPR-GAM. The criterion to quantify their quality, we use the Mean Absolute Error (MAE), which is defined as:

$$\text{MAE}(N) = \frac{1}{N^*} \sum_{i=1}^{N^*} |y_{\text{CR3BP},i} - y_{\text{GPR-GAM},i}| \tag{3.21}$$

This equation shows the MAE of N^* test samples when using N training samples.

Generally, a larger training dataset can be expected to decrease the MAE since more information is provided for the GPR-GAM model to capture the characteristics of the problem at hand. However, as shown in Equation 3.14, more training samples will increase the burden of computation. It remains a problem to identify the optimal size of the training dataset due to the trade-off between accuracy and efficiency. Bouwman et al. (2019) defines a threshold ϵ_{MAE} for the fluctuation of the MAE. Increasing the number of training samples is stopped when ΔMAE becomes smaller than that particular threshold. For GPR-GAM, we improve this method so that it works for different types of

output elements. When estimating the MAE of test data T with an increasing number of training samples, the training data size N starts from 100 and is increased with 100 each time.

A set of MAE's over a certain range of training data size is defined as:

$$E(N) = [\text{MAE}(N), \text{MAE}(N+n), ..., \text{MAE}(N+9n)] \tag{3.22}$$

where $n = 100$. The fluctuation of MAE over this range is given as:

$$\Delta\text{MAE}(N) = \max(E(N)) - \min(E(N)) \tag{3.23}$$

When the number of training samples for the GPR-GAM model meets the condition, i.e., ΔMAE is smaller than 5 % of the lowest MAE over the last 10 iterations:

$$\frac{\Delta\text{MAE}(N)}{\min(E(N))} < 0.05, \tag{3.24}$$

The prediction converges at the point k_s:

$$k_s = \underset{k=1,...,9}{\operatorname{argmin}} E(N) \tag{3.25}$$

The value N_s is selected as the final training data size:

$$N_s = N + n \times k_s \tag{3.26}$$

This criterion can be applied to every output element and ensures that an accurate GPR-GAM model with minimum training samples is obtained.

3.5. PERFORMANCE OF GPR-BASED GRAVITY ASSIST MAPPING

3.5.1. DIFFERENT COVARIANCE FUNCTIONS

The performance of GPR-GAM is shown in Figure 3.4 for an increasing number of training samples, from 100 to 5000. Figure 3.4 presents the MAE of test samples using five different covariance functions. The covariance functions of SE and RQ with the ARD strategy are shown in addition to those without ARD. For every output element, RQARD performs better than RQ. Also, SEARD obtains more accurate results than SE except for predicting $\delta\omega$. The results demonstrate the positive effect of ARD. For predicting an output, the five input features play different roles in Equation 3.6. The accuracy is improved by emphasizing the effects of some particular inputs using ARD. The RQARD function has the best accuracy with regards to all outputs. The NN function comes second except for quantifying $\delta\Omega$.

Using RQARD, the MAE tends to become stable when the training data size is larger than 3000. To better quantify this by applying the criterion of Section 3.4.4, the minimum stable training data size for the five outputs to converge are [4700,3500,4300,4200,4500]. Due to the complexity of the spatial CR3BP, significantly more training samples are required than for the planar case (Liu et al. 2021), which should not come as a surprise.

Predicting eccentricity is the fastest to become stable. When increasing the number of training samples from 100 to 1000, the MAE of δa and δi decrease sharply. However the results tend to be really stable using more than 4000 training samples.

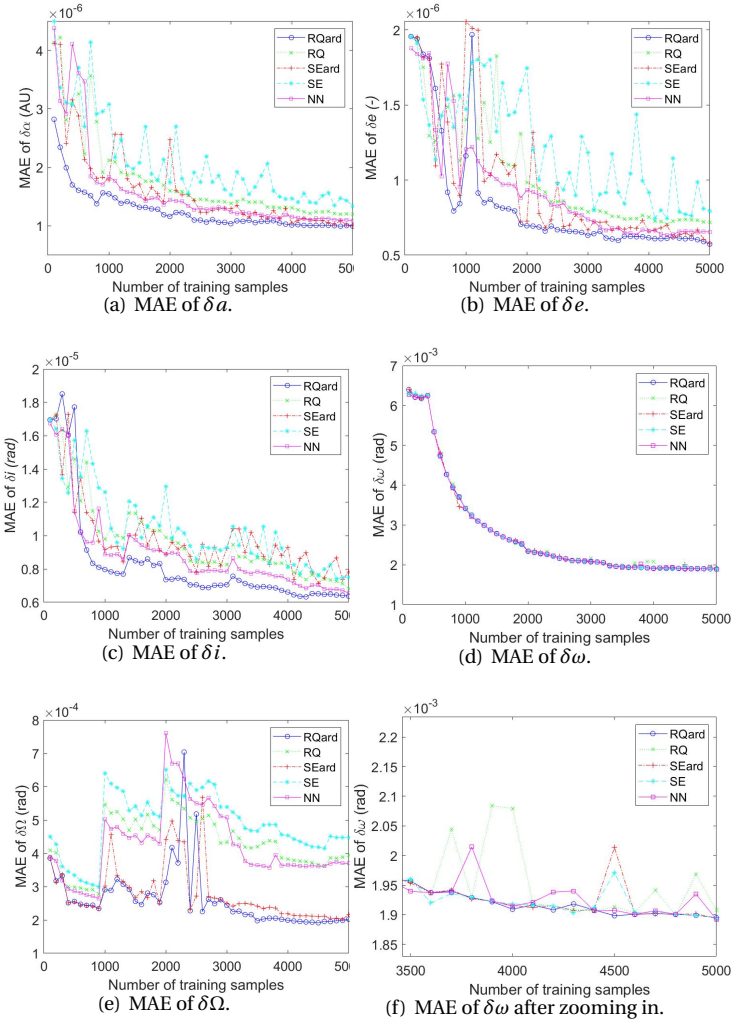
Inspecting Figures 3.4(c)-3.4(f), in particular the zoomed-in plot in Figure 3.4(f), the conclusion can be drawn that $\delta\omega$ is the most difficult element to predict, both in terms of level of convergence and in level of agreement between various covariance functions. Predicting $\delta\Omega$ is the second-most difficult thing to do (at a level of 1.0×10^{-5} rad), whereas δi is the most accurate one.

The full set of hyper-parameters of the mean and covariance functions are presented in Table 3.2. According to Equation 3.20, the influence of an input element on predicting a particular output is inversely proportional to the absolute value of the corresponding length-scale. For a set of five length-scales in predicting a specific output, a lower value of length-scale represents a higher influence of the corresponding input feature because all the inputs are normalized before training. For example, the eccentricity plays an important role in predicting δa, $\delta\omega$ and $\delta\Omega$. The predictions of δe and δi most rely on i and ω, respectively. The second-most influential input element is i ranking first in the prediction of δe, and second in that of δa, δi and $\delta\omega$. This illustrates the importance of inclination in the flyby effects in a spatial CR3BP. The signal variance parameter $s^2_{f,\text{RQARD}}$ represents the amplitude of the RQARD covariance function. The value of $s^2_{f,\text{RQARD}}$ for ω is small, which means that the similarity between two samples is small even though their inputs are close to each other. The shape parameter α controls how the similarity changes when a sample moves away from the other sample. The similarity is highest when two samples coincide. A larger α indicates that the similarity decreases fast when two samples are separating. The prediction of $\delta\omega$ has a small $s^2_{f,\text{RQARD}}$ and large α, which means an accurate prediction of $\delta\omega$ relies on training samples very close to the test samples.

Table 3.2: The values of length-scales, signal variance and shape parameters using the RQARD covariance function.

	δa	δe	δi	$\delta\omega$	$\delta\Omega$
l_a^2	14.54_5	17.56_5	8.79_5	1.14_4	13.39_4
l_e^2	0.17_1	0.28_2	1.01_3	0.02_1	0.07_1
l_i^2	0.21_2	0.03_1	0.38_2	0.57_2	7.17_3
l_ω^2	0.66_3	4.06_4	0.04_1	2.25_3	26.15_5
l_ϕ^2	1.07_4	0.48_3	1.28_4	6.90_5	3.79_2
$s^2_{f,\text{RQARD}}$	8.29	15.76	19.98	0.068	0.518
α	0.15	3.22	1.23	3.37	1.21
c	0.002	−0.001	0.001	0.03	0.0004

* The subscript of the length-scale hyper-parameters stands for the order of influence of an input element on predicting a particular output using the GPR model.



(a) MAE of δa. (b) MAE of δe.

(c) MAE of δi. (d) MAE of $\delta\omega$.

(e) MAE of $\delta\Omega$. (f) MAE of $\delta\omega$ after zooming in.

Figure 3.4: The MAE of test samples when using five different covariance functions.

3.5.2. COMPARISON WITH SEMI-ANALYTICAL METHOD

For the same group of test samples, we compare the GPR-GAM model to the Semi-analytical Keplerian Map (SKM), which is called the Kick Map, developed by Alessi & Sánchez (2016). Since the Kick Map is known for working well outside the Hill sphere of the perturbing body, an additional comparison is performed for the case of $r_2 > 0.01$ AU. The performance of a Planar-GPR-GAM (P-GPR-GAM) model developed by the authors for the planar CR3BP is also presented (Liu et al. 2021). The P-GPR-GAM studies flyby effects in a three-parameter input space of a, e and ω.

In Table 3.3, the MAE and CPU time of these methods are presented. Note that the

SKM statistics are based on an implementation of this technique on the first author's PC (Core i7 CPU and 8.00 GB RAM), and not on the numbers reported in Alessi & Sánchez (2016) itself; in this way platform-specific effects are eliminated. In the case of $r_2 > 6678$ km, the GPR-GAM has a better accuracy than the SKM for all outputs. The advantage is clear with regards to δa, δe and δi. In particular, the MAE of δa by SKM is about 3.3 times that of using GPR-GAM. The efficiency of predicting δa is improved by a factor of 1.27×10^4. 7.5×10^{-2} s is the total time of SKM predicting five outputs of a single test sample. However, predicting these five outputs using GPR-GAM, which spends a total of 2.0×10^{-5} s, is still much faster than SKM. GPR-GAM has seen an improvement on the accuracy and efficiency over SKM. Note that the time in this table shows the prediction component only. The training time needed for a GPR-GAM model using 4000 training samples is 32 minutes on the first author's PC. The advantage of GPR-GAM is that it has to learn only once and can be easily applied to a next prediction. This is certainly based on the assumption that the input space in interest does not change.

In the case of $r_2 > 0.01$ AU, the accuracy of both SKM and GRP-GAM is improved due to simpler dynamical properties. The overall accuracy of SKM approximations increases by about a factor of two. With regards to δa, δi and $\delta\Omega$, GPR-GAM still has better performance over SKM. SKM is more accurate for the prediction of $\delta\omega$. When the minimum distance between the spacecraft and the Earth is larger than the radius of the Hill sphere, the difference of MAE between these two methods decreases. In the planar case, P-GPR-GAM uses only 1500 training samples to reach a stable prediction. In addition to a lower dimension of the input space, using less training samples benefits the efficiency drastically.

Figure 3.5 shows the difference of predictions between the GPR-GAM and the numerical integration method for the first 100 test samples. For most of the predictions, the GPR-GAM outputs almost overlap with those of the numerical integration. The maximum error is 3.3×10^{-4} AU (49,369 km) for the No.87 test sample, where an obvious separation is observed. However, this magnitude of error does not occur often. The minimum error within these 100 samples is only 4.7×10^{-8} (7 km) AU. By converting the Keplerian orbital elements into the Cartesian coordinates in the ecliptic coordinate system centered at the Sun, the MAE of Cartesian distances between the GPR-GAM outputs and those of the numerical integration is calculated by

$$\mathrm{MAE}_{\mathrm{Car}} = \frac{1}{100}\sum_{i=1}^{100} |\mathrm{orb2Car}(\boldsymbol{y}_{\mathrm{CR3BP},i}) - \mathrm{orb2Car}(\boldsymbol{y}_{\mathrm{GPR\text{-}GAM},i})| \tag{3.27}$$

where $\boldsymbol{y}$ consists of all the Keplerian orbital elements of the spacecraft. orb2Car is a function converting $\boldsymbol{y}$ into Cartesian coordinates, and only the position components are used. When taking into account the error of predicting every Keplerian orbital element, $\mathrm{MAE}_{\mathrm{Car}}$ for these 100 test samples is 37,063 km.

3.5.3. ROBUSTNESS WHEN CHANGING TRAINING DATASET

To study the influence of the training samples on the accuracy, we generate another two groups of training samples in the same input space. These groups use different seed numbers to generate different (quasi) random samples. For all cases, the RQARD covariance function is selected to build the GPR-GAM model. For the same set of test samples,

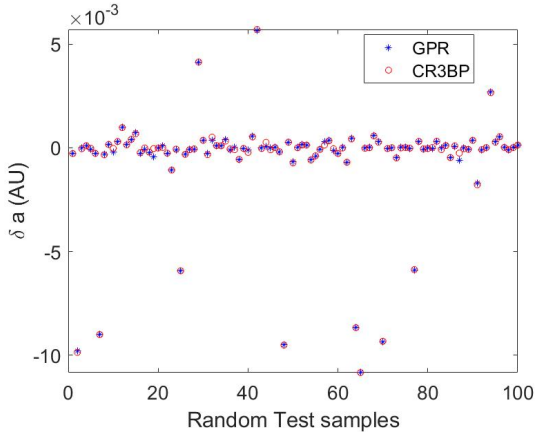


Figure 3.5: The difference of predictions between the GPR-GAM and the numerical integration for 100 random test samples.

Figure 3.6 shows the comparison of the GPR-GAM models trained by these three groups of training samples. The MAE drops sharply when going from 100 to 1000 samples and there is an obvious difference between datasets when using less than 3000 training samples. A steady decline begins at 3000 until the three curves converge using more than 4000 training samples. An eventual close agreement between three groups ensures the robustness of prediction using GPR-GAM.

The prediction of δe converges and stabilizes fastest, which supports the observation of using the least number of training samples in Section 3.5.1. In Figure 3.6(c), for inclination, the results of the three datasets converge slowly after going through a gap of 0.5×10^{-5} rad. A minimum dataset of 4000 training samples is required to obtain a stable performance of δi. In Figure 3.6(d), the behavior of $\delta\omega$ using three datasets has an insignificant difference compared to the other output elements, but the zoomed-in plot (Figure 3.4(f)) shows a more clear difference. Predicting $\delta\Omega$ is also difficult for GPR-GAM with regards to the significant fluctuation in Figure 3.6(e). Using different groups of training samples, the fluctuation can be observed between 1000 and 3000 training samples. $\delta\Omega$ has relatively larger values due to a flyby than those of δi and $\delta\omega$, which is a cause of this fluctuation. Increasing the number of training samples is necessary to stabilize the MAE.

The results are summarised in Table 3.4. To quantify the consistency of the MAE outcome, the Percentage of Deviation (PD) is calculated by subtracting the lowest value from the highest value, and then dividing the result by the highest value. Quantifying $\delta\omega$ with different training datasets has the lowest PD. The accuracy of evaluating $\delta\Omega$ is affected more seriously than the other orbital elements. However, the absolute difference between different groups is small and the convergence is good as shown in Figure 3.6(e).

Table 3.3: The performance of GPR-GAM compared to that of Alessi & Sánchez (2016) (fully reproduced here). The computational time is obtained by taking the mean value of 1000 repeated experiments. The value is the CPU time spent on one test sample. The CPU time of numerical integration of the equations of motion of CR3BP is listed in the last row.

MAE		δa [AU]	δe [-]	δi [rad]	$\delta\omega$ [rad]	$\delta\Omega$ [rad]
$r_2 > 6678$ km	GPR-GAM	5.7×10^{-5}	3.4×10^{-5}	6.3×10^{-6}	1.9×10^{-3}	1.9×10^{-4}
	SKM	1.9×10^{-4}	7.6×10^{-5}	1.1×10^{-5}	2.2×10^{-3}	2.1×10^{-4}
$r_2 > 0.01$ AU	GPR-GAM	3.3×10^{-5}	2.1×10^{-5}	3.8×10^{-6}	1.3×10^{-3}	0.8×10^{-4}
	SKM	7.3×10^{-5}	2.1×10^{-5}	5.9×10^{-6}	1.2×10^{-3}	1.0×10^{-4}
Time* [s]	GPR-GAM	5.9×10^{-6}	1.5×10^{-6}	3.8×10^{-6}	3.6×10^{-6}	5.2×10^{-6}
	SKM	7.5×10^{-2}	7.5×10^{-2}	7.5×10^{-2}	7.5×10^{-2}	7.5×10^{-2}
	P-GRP-GAM	1.2×10^{-6}	1.2×10^{-6}	-	1.2×10^{-6}	-
	CR3BP	5.9×10^{-1}	5.9×10^{-1}	5.9×10^{-1}	5.9×10^{-1}	5.9×10^{-1}

Table 3.4: The performance of GPR-GAM over three different training datasets using the same test dataset. The values of MAE and PD are shown.

	δa [AU]	δe [-]	δi [rad]	$\delta\omega$ [rad]	$\delta\Omega$ [rad]
Group 1	5.74×10^{-5}	3.42×10^{-5}	6.32×10^{-6}	1.89×10^{-3}	1.91×10^{-4}
Group 2	5.56×10^{-5}	3.49×10^{-5}	6.34×10^{-6}	1.90×10^{-3}	1.96×10^{-4}
Group 3	5.64×10^{-5}	3.56×10^{-5}	6.68×10^{-6}	1.89×10^{-3}	2.02×10^{-4}
PD	3.14 %	3.93 %	5.38 %	0.53 %	5.44 %

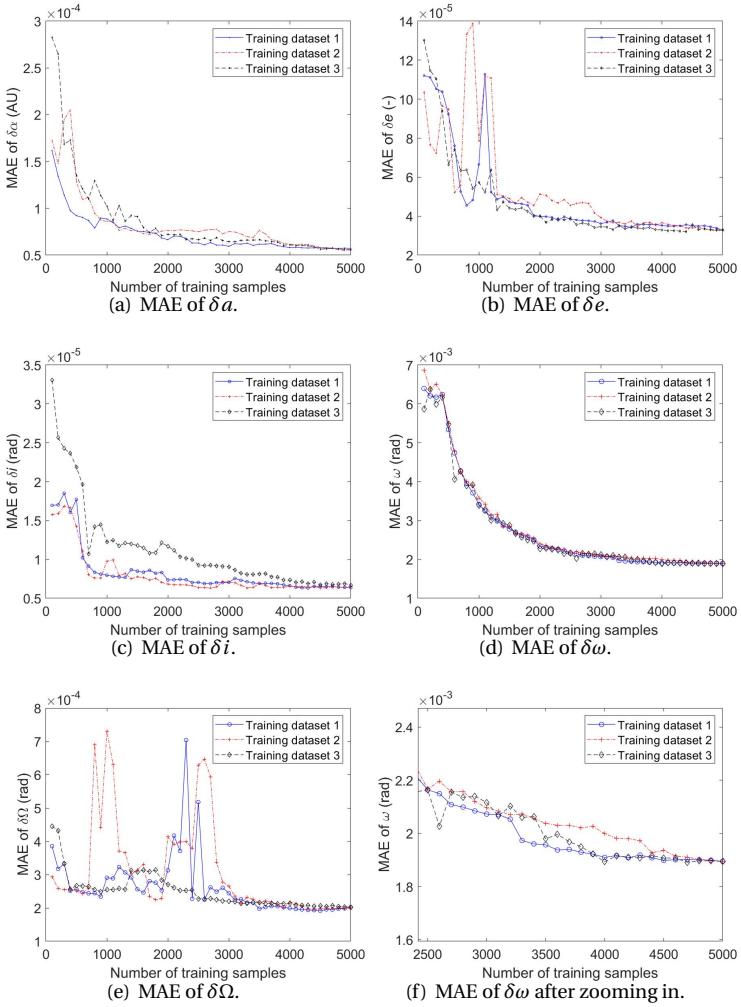


Figure 3.6: The MAE of test samples when using three different training datasets.

3.5.4. ROBUSTNESS WHEN CHANGING TEST DATASET

To evaluate the robustness of GPR-GAM when predicting different test datasets, we generate another two groups of test samples using different random seed numbers. Akin to the group used before, each group has 500 samples. The same training data set and the RQARD covariance function are used for building the GPR-GAM model. By applying the criterion of Section 3.4.4 to the results in Figure 3.4, the training data size is set at [4700, 3500, 4300, 4200, 4500] for each orbital element. The performance of these three groups is presented in Table 3.5, where the first line obviously refers to results already presented in Tables 3.3 and 3.4. The PD is calculated for these parameters. These PDs are smaller than those in Table 3.4, except for $\delta\omega$. Gradually, the GPR-GAM model has a better ro-

bustness when changing test datasets than that when changing training datasets. The results for δa have the smallest PD which is only 0.69%. This suggests that predicting δa using GPR-GAM has the best generalization (Rasmussen & Williams 2006). For every element, the overall PD's between groups is lower than 3.0%. The results demonstrate that the GPR-GAM model has a stable performance when predicting the flyby effects for a new given initial condition.

Table 3.5: The performance of GPR-GAM over three different test datasets using the same training samples. The values of MAE and PD are shown.

	δa [AU]	δe [-]	δi [rad]	$\delta\omega$ [rad]	$\delta\Omega$ [rad]
Group 1	5.74×10^{-5}	3.42×10^{-5}	6.32×10^{-6}	1.89×10^{-3}	1.91×10^{-4}
Group 2	5.78×10^{-5}	3.40×10^{-5}	6.39×10^{-6}	1.84×10^{-3}	1.89×10^{-4}
Group 3	5.75×10^{-5}	3.38×10^{-5}	6.24×10^{-6}	1.85×10^{-3}	1.93×10^{-4}
PD	0.69 %	1.17 %	2.35 %	2.65 %	2.07 %

3.5.5. LOW-ENERGY CASES

The computations so far sample the input space in a uniform way, irrespective of the energy level. One could argue that the high-energy part of the domain does not really need a three-body formulation, and that the low-energy part of the domain is the more challenging. In order to analyze the specific application of the GPR-GAM model to low-energy transfers, we generate a group of 500 test samples with $C_J > C_{L3}$. According to the initial condition defined by the input space, the spacecraft starts from somewhere in the exterior Hill's region. This means that the following situation when the Jacobi constant is smaller than that of C_{L1} is not considered: the spacecraft is located initially in the vicinity of either the Sun or the Earth and travels to the neighbourhood of the other. However, it does allow the cases of the spacecraft starting from the exterior realm and moving into the neighbourhood of the Earth when $C_{L2} > C_J > C_{L3}$. The predictions are obtained by using the training dataset 1 and the corresponding hyper-parameters shown in Table 3.2, so for the model that is based on the entire energy range. The MAE for the low-energy part of the domain are shown in Table 3.6.

Table 3.6: The performance of GPR-GAM over the test samples with Jacobi constant $C_J > C_{L3}$.

	δa [AU]	δe [-]	δi [rad]	$\delta\omega$ [rad]	$\delta\Omega$ [rad]
low-energy domain	3.71×10^{-5}	4.28×10^{-5}	8.08×10^{-6}	2.56×10^{-3}	2.31×10^{-4}
full domain	5.7×10^{-5}	3.4×10^{-5}	6.3×10^{-6}	1.9×10^{-3}	1.9×10^{-4}

The MAE's remain at the same level as those of the entire Jacobi energy cases in Table 3.4, with a slight degradation of about $20-30\%$ (except for δa). This is mainly because of the more complex variation of orbital elements due to the higher chaotic dynamics, particularly when $C_{L2} > C_J > C_{L3}$. Using the CR3BP propagation, the variation of the

semi-major axis for 500 test samples in the low-energy regime is shown in Figure 3.7. Although the majority of δa has an absolute value of about 1.0×10^{-3} AU, a significant percentage of them are larger than 0.01 AU. The initial conditions with smaller phasing angle would generally result in larger variation of semi-major axis. The largest change among these is -0.0771 AU. In the rotating reference frame, the trajectory of this case ($C_J = 3.00057$) is shown in Figure 3.8. The osculating Keplerian elements of its initial condition $[a, e, i, \omega, \phi, \theta]$ are [1.028AU,0.008,0.086°,0.932°,-0.524°,-180°].

The value of C_J meets the condition $C_{L2} > C_J > C_{L3}$. The interior and exterior Hill's region are connected which allows a spacecraft motion between these two realms. Over one unperturbed orbital period computed from the initial condition, the spacecraft travels from the exterior Hill's region into the neighbourhood of the Earth temporarily before moving out. The closest distance between the spacecraft and the center of the Earth is 7223 km. The variation of the Keplerian elements and the predictions obtained by GPR-GAM are shown in Table 3.7. The predictions are well approximated especially for δa and δe. The prediction of δi, $\delta\omega$ and $\delta\Omega$ yield poorer accuracy compared to the MAE in Table 3.6,but the relative error is small compared to the large variation of these elements. It is worth noting that the value of the semi-major axis of the final condition is 0.951 AU, which is beyond the predefined input space in the current work. Extending the input space to the initial conditions of starting from the interior Hill's region is of interest for future work.

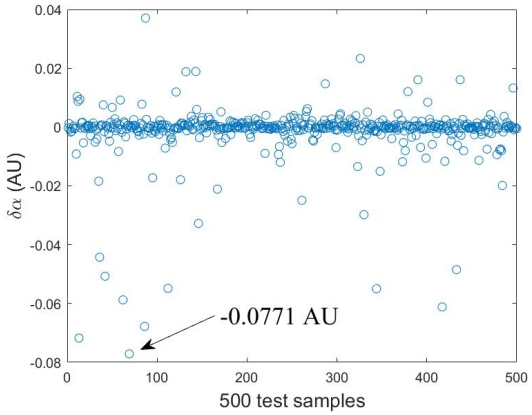


Figure 3.7: The variation of the semi-major axis for 500 test samples in the low-energy regime.

Table 3.7: The prediction of the test sample ($C_J = 3.00057$).

	δa [AU]	δe [-]	δi [deg]	$\delta\omega$ [deg]	$\delta\Omega$ [deg]
Numerical integration	-0.0771	0.0470	20.1637	-20.2493	9.0656
GPR-GAM	-0.0773	0.0464	20.1127	-20.4192	8.9913
PD	0.26%	1.28 %	0.25 %	0.85 %	0.82 %

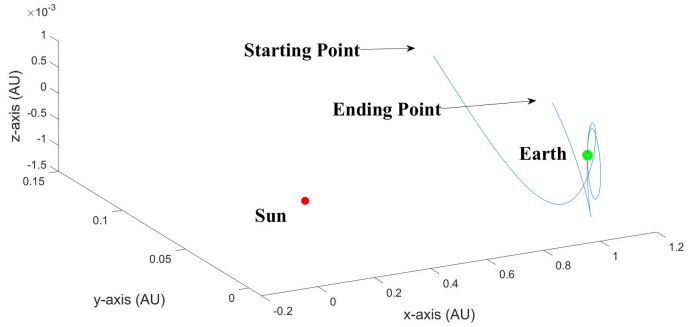


Figure 3.8: Trajectory of a test sample ($C_J = 3.00057$). In order to show the trajectory properly in three dimensions, the scales of three axes are not equal.

3.6. CONCLUSIONS

A Gravity Assist Mapping was developed based on Gaussian Process Regression for evaluating flyby effects in the fully spatial CR3BP. A new criterion was proposed to select the number of training samples for every orbital element. Due to the increasing complexity of a fully spatial CR3BP compared to that of the planar one, a larger training dataset is required. Compared to a previous semi-analytical method, GPR-GAM has achieved a better accuracy. The CPU time for prediction is a factor 10^3 faster than that of this semi-analytical method. The domain of applicability is extended beyond that of the Keplerian Map.

According to the MAE, the RQARD covariance function performs best. Based on the optimized values of the length-scale, the influence of different input elements on flyby effects was discussed. The stable robustness of GPR-GAM was illustrated by changing the training and test datasets. It indicates that quantifying δa has the best generalization property. The prediction of $\delta\omega$ is more difficult than that of the other output elements. Further attention needs to be paid to the various ranges of output. The results show the ability of the GPR-GAM model to predict the dynamics of the CR3BP system with relatively high accuracy compared to SKM. The GPR-GAM model has been shown to be a versatile tool that can be applied for a wide range of energy levels. The quality of predictions in the low-energy domain is almost comparable to that in the high-energy part.

The input space considers only the initial conditions of the spacecraft starting from the exterior Hill's region, and the time of flight is limited to one orbital period of the initial osculating orbit. The periapsis passage of the initial osculating orbit is located in the same region. The phenomenon of the spacecraft temporarily being captured by the secondary is observed for this input space, which is not amenable to a patched-conics technique. However, future work will consider extension of the input space to include more complex situations. The angular phasing and the longitude of the ascending node are used in the input and the output vectors, respectively. The flexibility of the GPR technique allows the mapping function to have different formulations. In order to explore the

applications of GPR-GAM, four subjects are of interest for further investigation: a) using an alternative set of orbital elements; b) employing the orbital elements in the rotating reference frame; c) extending the time of flight to multiple orbital periods, or using a Poincaré section as boundary conditions, such that more complex dynamics is involved and more CPU time could be saved on predicting the status of the spacecraft; d) establishing GPR-GAM for different CR3BP systems considering the various goals of space missions, e.g., SMART-1 (Moon) or exploration missions to Jupiter or Saturn (Sánchez et al. 2015).

3.7. ACKNOWLEDGEMENT

The author Liu gratefully acknowledges the China Scholarship Council (CSC) for its financial support.

4

A Machine-Learning Design Tool for Quantifying Gravity Assist Effects in the Circular Restricted Three-Body Problem

Abstract A method to quantify the changes in orbital elements after a gravity assist, the Gravity Assist Mapping, is developed for the Circular Restricted Three-Body Problem. The mapping function is inspired by the Keplerian Map and the Flyby Map. Instead of solving the mapping analytically or numerically, the model is built using a Gaussian Process-based machine-learning technique. The first main element is the development of a classification model to identify impact trajectories given arbitrary initial conditions. The second main element is a regression model to predict the flyby effects after completion of one orbital revolution. The Jacobi constant is used to show the broad energy range where the proposed method can be applied, and is also employed as an extra parameter for a more accurate modeling of the flyby dynamics. The performance of the models is analyzed in three systems with increasing mass ratio between the primary and secondary body in order to study the impact of this mass ratio: Sun-(Earth+Moon)-spacecraft, Jupiter-Callisto-spacecraft, and Sun-Jupiter-spacecraft. The results demonstrate that the evaluation of flyby effects can be efficiently and accurately made combining the classification and regression models. The reliability of detecting impact trajectories is better than 90%. The mean absolute error of predicting the change in semi-major axis is only 3.86×10^{-5} AU (a typical range of changes is -0.0115 AU to 0.0163 AU), with a computational cost of 5.8×10^{-6} s per sample, which is several million times faster than

This chapter is to be submitted to Advances in Space Research (2021).

solving the equations of motion by numerical integration.

4.1. INTRODUCTION

The gravity assist, or flyby, is a commonly used technique in space missions. Here, the flyby at one celestial body (e.g., Jupiter) is typically used to change the velocity of a spacecraft with respect to a primary body (e.g., the Sun). When designing such a flyby trajectory, the patched-conics model is widely adopted for a first-order analysis (Broucke 1988, Longuski & Williams 1991, Strange & Longuski 2002, de Almeida Prado & de Felipe 2007). This model is simple, and only considers the gravitational attraction of a single celestial body at a time. Reaching beyond the limitations of the patched-conics model, the mechanisms of distant flybys or invariant manifolds are of significant importance in celestial mechanics, and only exist in the framework of Circular Restricted Three-Body Problem (CR3BP) (Szebehely 1967, Koon et al. 2000, Beutler 2004, Campagnola et al. 2012). As an example, it can be shown that the gravitational attraction of Jupiter also affects the trajectory of the spacecraft even beyond its Sphere of Influence (SoI) (Greenberg et al. 1988, Gawlik et al. 2009).

The CR3BP is a simplified model for studying the three-body problem. The motion of the spacecraft is always affected by the primary and secondary body simultaneously, which rotate around their barycenter. The behavior of flybys in the CR3BP can be a complex issue because of the potentially chaotic dynamics.

The Keplerian Map (KM) developed by Ross et al. (2007) approximates the flyby effects in the planar CR3BP and gives insight into the chaotic zone in the phase space (Ross & Scheeres 2007). The changes in Keplerian orbital elements over one orbital period are estimated using perturbation theory and Picard's method. Based on this KM, Peñagaricano Muñoa et al. (2010) developed an approach for the fully spatial CR3BP (Peñagaricano Muñoa & Scheeres 2010). Also for the spatial framework, Alessi et al. (2016) developed a kick map to evaluate the flyby effects by applying the Picard iteration to the Lagrange planetary equations (Alessi & Sánchez 2016). The above methods are semi-analytical and obtain integration solutions by quadratures or numerical methods. The situations of transition between interior and exterior Hill's regions and temporary capture around the secondary are not taken into account. It is also not possible to identify initial conditions that lead to an impact on the secondary.

Inspired by the KM and the Tisserand-Poincaré graph, Campagnola et al. (2012) developed a Flyby Map (FM) to extend the applicability of the Tisserand graph to low energy levels (Campagnola et al. 2012). For the planar CR3BP, it reveals that direct flybys are more efficient than retrograde flybys. This method is as accurate as the numerical integration of the CR3BP equations of motion, and has an equivalent computational cost.

The aim of this work is to develop an alternative approach, Gravity Assist Mapping (GAM), to quantify the flyby effects in the fully spatial CR3BP. The approach consists of two models built up using a machine-learning technique based on Gaussian Processes (GP) (Rasmussen & Williams 2006). In the past decade, the Deep Neural Network technique has become very popular for its excellent performance on speech recognition, visual object recognition and object detection, etc. (LeCun et al. 2015). Unlike neural network methods, the GP method requires a relatively smaller training dataset, and has a simple model selection procedure (Rasmussen & Williams 2006). The prediction is

based on one assumption: close inputs are likely to have similar outputs.

The first developed model is the Gaussian Process Classification (GPC), to be used for identifying impact trajectories given initial conditions. The second model is the Gaussian Process Regression (GPR) for predicting the variation of the Keplerian orbital elements after a flyby. The GP approach is a supervised learning method developed by Krige for making predictions based on empirical information (Krige 1951). In the astrodynamics framework, it has been successfully used for assessing the accessibility of asteroids, optimizing low-thrust trajectories, and building gravity models (Shang & Liu 2017, Bouwman et al. 2019, Gao & Liao 2019).

This paper is organized as follows: Section 4.2 defines the framework of the CR3BP. The equations of motion are used for the generation of samples for the GP models. The mapping functions for classification and regression are given in Section 4.3. The construction and training process of GAM models are also presented here, as well as the generation of training datasets and criteria for accuracy measurement. Section 4.4 shows the performance of the GP models in three applications: the Sun-(Earth+Moon)-spacecraft, the Jupiter-Callisto-spacecraft and the Sun-Jupiter-spacecraft systems. Section 4.5 addresses the possibility of making quality assessments using GP models. Section 4.6 summarizes the results, and discusses the advantages and limitations of GAM. Recommendations on the usage of GAM and how to improve its performance in future work are also made.

4.2. CIRCULAR RESTRICTED THREE-BODY PROBLEM

The CR3BP describes the motion of three point masses. It assumes two massive bodies P_1 and P_2 with masses M_1 and M_2, respectively. P_1 and P_2 are known as the primary and secondary, respectively, which meet the condition $M_1 > M_2$. P_1 and P_2 rotate around their barycenter O_B and have a constant distance a_{P2}. The third body P_3 has a negligible mass. The motion of P_3 is affected by the other two bodies simultaneously, which is specified by the equations of motion in the rotating reference frame (Beutler 2004):

$$\begin{cases} \ddot{x} - 2\dot{y} = x - \frac{(1-\mu)}{r_1^3}(x+\mu) - \frac{\mu}{r_2^3}(x-1+\mu) \\ \ddot{y} + 2\dot{x} = y - \frac{(1-\mu)}{r_1^3}y - \frac{\mu}{r_2^3}y \\ \ddot{z} = -\frac{(1-\mu)}{r_1^3}z - \frac{\mu}{r_2^3}z \end{cases} \tag{4.1}$$

where $\mu = M_2/(M_1+M_2)$ is the mass parameter. $r_1 = \sqrt{(x+\mu)^2+y^2+z^2}$ and $r_2 = \sqrt{(x-1+\mu)^2+y^2+z^2}$ represent the distance of the spacecraft to P_1 and P_2, respectively. The variables are normalized using (M_1+M_2) for mass, a_{P2} for length, and $\sqrt{a_{P2}^3/(G(M_1+M_2))}$ for time. G is the universal gravitational parameter. In this paper, $\mu = 3.036\times10^{-6}$ for the Sun-(Earth+Moon)-spacecraft system, $\mu = 5.668\times10^{-5}$ for the Jupiter-Callisto-spacecraft system, and $\mu = 9.537\times10^{-4}$ for the Sun-Jupiter-spacecraft system. The CR3BP has one integral, the Jacobi constant C_J, written as (Beutler 2004):

$$C_J = x^2 + y^2 + \frac{2(1-\mu)}{r_1} + \frac{2\mu}{r_2} - (\dot{x}^2+\dot{y}^2+\dot{z}^2) + \mu(1-\mu) \tag{4.2}$$

C_J is the only known variable that has a constant value in the CR3BP. It is determined by the initial condition of the spacecraft and is a measure for its three-body energy: a larger C_J represents a lower amount of specific energy. In this work, the initial condition is such that the vehicle is outside the Hill sphere. The center of the Hill sphere is at the secondary, and its radius is defined as (Szebehely 1967):

$$r_H = \sqrt[3]{\frac{\mu}{3(1-\mu)}} \tag{4.3}$$

4.3. Gravity Assist Mapping

4.3.1. Mapping Function

In this work, the GAM describes the variation of the state of the spacecraft using the Keplerian orbital elements with respect to the primary body. In an inertial reference frame, the initial osculating orbit of the spacecraft is specified as depicted in Figure 4.1. The origin is centered at the primary and the reference plane is taken as the orbital plane of the secondary. The positive X-axis is in the direction of the vernal equinox. The initial condition S_A and the final condition S_B of the spacecraft before and after a flyby are defined as:

$$S_A = \{(a,e,i,\omega,\Omega,\theta) \mid r_2(a,e,i,\omega,\Omega,\theta) > 2r_H, \theta = -\pi\} \tag{4.4}$$

$$S_B = \{(a,e,i,\omega,\Omega,\theta) \mid r_2(a,e,i,\omega,\Omega,\theta) > 2r_H, \theta = \pi\} \tag{4.5}$$

where a represents the semi-major axis, e eccentricity, i inclination, ω arguement of periapsis, Ω right ascension of ascending node, θ true anomaly.

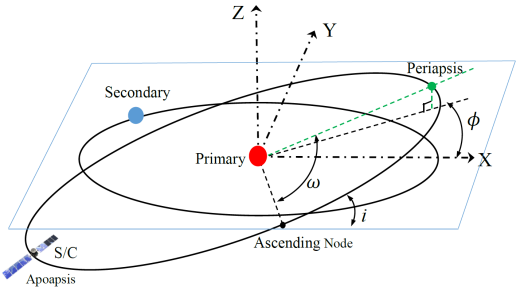


Figure 4.1: The initial osculating orbit of the spacecraft centered at the primary in an inertial reference frame.

The spacecraft starts from the apoapsis with a distance to the secondary larger than two times r_H in the exterior of the Hill sphere. After about one orbital revolution, the final condition represents the apoapsis when the spacecraft is at least $2r_H$ away again from the secondary, such that the spacecraft is not in the vicinity of the secondary even if it could be temporarily captured. In order to represent the relative position between the spacecraft and the secondary, a phasing angle ϕ is defined specifically for the initial condition (Alessi & Sánchez 2016):

$$\phi = \tan^{-1}(\tan(\omega)\cos(i)) + \Omega \tag{4.6}$$

Here, ϕ represents the phasing angle between the positive X-axis and the projection of the periapsis line of the spacecraft orbit onto the reference plane at the moment of periapsis passage. The initial true anomaly θ_{P2} of the secondary is defined from the orbit of the spacecraft:

$$\theta_{P2} = -\pi\sqrt{a^3} \tag{4.7}$$

This ensures that the secondary reaches the positive X-axis when the spacecraft has traveled half of the orbital revolution. In particular, the spacecraft arrives at the primary-secondary line if $\phi = 0$.

The GAM method aims at building mapping functions F_{GAM} to predict output y corresponding to a set of inputs $\boldsymbol{x}$:

$$F_{\text{GAM}}(\boldsymbol{x}) : \boldsymbol{x} \mapsto y \tag{4.8}$$

The specific elements for $\boldsymbol{x}$ and y can be defined differently for different objectives. For impact analysis, the function of Gaussian Process Classification-based Gravity Assist Mapping (GPC-GAM) is defined as:

$$\begin{aligned} &F_{\text{GPC-GAM}} : \boldsymbol{x} \mapsto y \\ &[a, e, i, \omega, \phi] \mapsto [C] \end{aligned} \tag{4.9}$$

where $\boldsymbol{x}$ represents the initial Keplerian elements of the spacecraft, before the flyby. C is a class label denoting impact trajectories:

$$C = \begin{cases} +1 & impact \\ -1 & no\ impact \end{cases} \tag{4.10}$$

A trajectory is classified as 'impact' if the spacecraft gets close to the secondary, i.e., within a distance of 300 km above its surface or less (taking the second flyby of the Galileo mission performed at Earth as reference (Anderson et al. 2008)). Otherwise, we clearly are in a flyby situation. A well-trained GPC-GAM can tell whether a given input leads to a collision, before the process to quantify the flyby effect is even started. The model can also be used stand-alone for impact classification, given a large number of initial conditions. Previously, van der Weg & Vasile (2014) introduced the concept of survivability and event maps in the coupled circular restricted three-body problem involving Sun, Earth and Moon. The method was used to obtain suitable conditions such that the spacecraft crashes on, or is weakly captured by the Moon.

The mapping function of Gaussian Process Regression-based Gravity Assist Mapping (GPR-GAM) is defined as:

$$\begin{aligned} &F_{\text{GPR-GAM}} : \boldsymbol{x} \mapsto y \\ &[a, e, i, \omega, \phi] \mapsto \delta a, \delta e, \delta i, \delta\omega, or\ \delta\Omega \end{aligned} \tag{4.11}$$

where the δ represents the change of a Kepler element due to the gravity assist. Obviously, the input vector is identical to that of the GPC-GAM. y is the variation of one of the Keplerian elements after a flyby. The flexibility of a GP method allows it to use different features as input. Incorporating new features could potentially help the GPR model to better learn the dynamics of a flyby. To this end, we develop an alternative model

based on GPR-GAM by adding the Jacobi constant as the sixth input feature, although $\boldsymbol{x}$ already provides all information to uniquely identify a specific trajectory. This new model is defined as:

$$\begin{aligned} F_{\text{GPR-GAM-J}} &: \boldsymbol{x} \mapsto y \\ [a, e, i, \omega, \phi, C_J] &\mapsto \delta a, \delta e, \delta i, \delta \omega, or\ \delta\Omega \end{aligned} \tag{4.12}$$

C_J is calculated with the initial conditions using Equation 4.2. Even when applying this model to a multi-revolution situation, C_J needs no updating because of its invariant property (Alessi & Sánchez 2016).

4.3.2. GAUSSIAN PROCESS MODELS

The prediction models built are based on the theory of the Gaussian Process (Rasmussen & Williams 2006). A GP is a collection of random variables, each of which follows a Gaussian distribution. The random variables refer to the value of the function $F_{\text{GAM}}(\boldsymbol{x})$ in Equation 4.8. The GP is fully defined by the mean function $m(\boldsymbol{x})$ and the covariance function $k(\boldsymbol{x}, \boldsymbol{x}')$ (Rasmussen & Williams 2006):

$$f(\boldsymbol{x}) \sim GP(m(\boldsymbol{x}), k(\boldsymbol{x}, \boldsymbol{x}')) \tag{4.13}$$

Note that $m(\boldsymbol{x})$ and $k(\boldsymbol{x}, \boldsymbol{x}')$ can have different values, given the expressions for different input elements. The nature of the GP models learning the mapping functions, i.e., Equations 4.9, 4.11 and 4.12, is actually a process of training the parameters according to observed examples. In our study, a constant mean function and the Rational Quadratic function with Automatic Relevance Determination (RQARD) are selected (Liu et al. 2021), which are defined respectively as (Rasmussen & Williams 2006):

$$m(\boldsymbol{x}) = c \tag{4.14}$$

and

$$k_{\text{RQARD}}(\boldsymbol{x}, \boldsymbol{x}') = s^2_{\text{RQARD}}\left(1 + \frac{(\boldsymbol{x} - \boldsymbol{x}')\boldsymbol{Q}(\boldsymbol{x} - \boldsymbol{x}')^T}{2\alpha}\right)^{-\alpha} \tag{4.15}$$

The parameters in the GP method are called hyper-parameters. c, s^2_{RQARD} and α are a constant, signal variance and shape hyper-parameters, respectively. $\boldsymbol{Q}$ is a symmetric matrix containing length-scale hyper-parameters:

$$\boldsymbol{Q} = \text{diag}\left(\frac{1}{l_1^2}, \frac{1}{l_2^2}, ..., \frac{1}{l_d^2}\right) \tag{4.16}$$

where d is the number of features in the input vector $\boldsymbol{x}$.

Training the hyper-parameters requires empirical information provided by observed examples. In supervised machine learning, the examples are defined as training samples. Each sample $(\boldsymbol{x}, y)$ consists of a training input $\boldsymbol{x}$ and a training output y. The collection of training samples D_{train} is defined by:

$$D_{\text{train}} = \{(\boldsymbol{X}, \boldsymbol{Y}) | \boldsymbol{X} = [\boldsymbol{x}_1, \boldsymbol{x}_2, \ldots, \boldsymbol{x}_N], \boldsymbol{Y} = [y_1, y_2, \ldots, y_N]\} \tag{4.17}$$

where N is the size of the training dataset. For a specific problem, the training inputs are randomly generated in a predefined input space. The training outputs are obtained

from integrating the CR3BP equations of motion using Equation 4.1, and are considered as the ground-truth observations. The generation of a training dataset will be elaborated upon in the next section.

The basic property of a GP is that any finite number of variables in a GP has a joint Gaussian distribution. Given the training inputs, the joint Gaussian distribution of outputs obtained from a GP is:

$$\boldsymbol{f} \sim \mathcal{N}(\boldsymbol{m}(\boldsymbol{X}), \boldsymbol{K}(\boldsymbol{X}, \boldsymbol{X})) \tag{4.18}$$

where $\boldsymbol{m}(\boldsymbol{X})$ is an N-dimensional vector of constant elements c. $\boldsymbol{K}(\boldsymbol{X}, \boldsymbol{X})$ is an $N \times N$ matrix:

$$\boldsymbol{K}(\boldsymbol{X}, \boldsymbol{X}) = \begin{pmatrix} k(\boldsymbol{x}_1, \boldsymbol{x}_1) & k(\boldsymbol{x}_1, \boldsymbol{x}_2) & \cdots & k(\boldsymbol{x}_1, \boldsymbol{x}_N) \\ k(\boldsymbol{x}_2, \boldsymbol{x}_1) & k(\boldsymbol{x}_2, \boldsymbol{x}_2) & \cdots & k(\boldsymbol{x}_2, \boldsymbol{x}_N) \\ \vdots & \vdots & \ddots & \vdots \\ k(\boldsymbol{x}_N, \boldsymbol{x}_1) & k(\boldsymbol{x}_N, \boldsymbol{x}_2) & \cdots & k(\boldsymbol{x}_N, \boldsymbol{x}_N) \end{pmatrix} \tag{4.19}$$

where $k(\boldsymbol{x}_i, \boldsymbol{x}_j)$ is the RQARD covariance function given in Equation 4.15.

The training process is aimed to have $\boldsymbol{f}$ approximate $\boldsymbol{Y}$ by optimizing hyperparameters, which is accomplished by maximizing the log marginal likelihood (Rasmussen & Williams 2006):

$$\log p(\boldsymbol{Y}|\boldsymbol{X}) = -\frac{1}{2}\boldsymbol{Y}^T \boldsymbol{K}^{-1} \boldsymbol{Y} - \frac{1}{2}\log|\boldsymbol{K}| - \frac{N}{2}\log 2\pi \tag{4.20}$$

Here, $p(\boldsymbol{Y}|\boldsymbol{X})$ specifies the probability of reproducing $\boldsymbol{Y}$ using the GP model. The conjugate gradient algorithm is used for optimization (Rasmussen & Williams 2006). When the training procedure has been completed, the prediction of output f^* for an arbitrary input $\boldsymbol{x}^*$ can be made. According to the property of GP, $\boldsymbol{Y}$ and f^* also follow a joint Gaussian distribution described by:

$$\begin{pmatrix} \boldsymbol{Y} \\ f^* \end{pmatrix} \sim \mathcal{N}\left(\begin{pmatrix} \boldsymbol{m}(\boldsymbol{X}) \\ m(\boldsymbol{x}^*) \end{pmatrix}, \begin{pmatrix} \boldsymbol{K}(\boldsymbol{X}, \boldsymbol{X}) & \boldsymbol{K}(\boldsymbol{X}, \boldsymbol{x}^*) \\ \boldsymbol{K}(\boldsymbol{X}, \boldsymbol{x}^*)^T & k(\boldsymbol{x}^*, \boldsymbol{x}^*) \end{pmatrix} \right) \tag{4.21}$$

where $\boldsymbol{K}(\boldsymbol{X}, \boldsymbol{x}^*)$ is a vector of covariance functions:

$$\boldsymbol{K}(\boldsymbol{X}, \boldsymbol{x}^*) = \begin{pmatrix} k(\boldsymbol{x}_1, \boldsymbol{x}^*) \\ k(\boldsymbol{x}_2, \boldsymbol{x}^*) \\ \vdots \\ k(\boldsymbol{x}_N, \boldsymbol{x}^*) \end{pmatrix} \tag{4.22}$$

By conditioning the joint Gaussian distribution on $\boldsymbol{Y}$, the distribution of the prediction f^* is

$$f^*|\boldsymbol{x}^*, \boldsymbol{X}, \boldsymbol{Y} \sim \mathcal{N}(\mu(f^*), cov(f^*)) \tag{4.23}$$

where

$$\mu(f^*) = m(\boldsymbol{x}^*) - K(\boldsymbol{X}, \boldsymbol{x}^*)^T \boldsymbol{K}(\boldsymbol{X}, \boldsymbol{X})^{-1} (\boldsymbol{Y} - \boldsymbol{m}(\boldsymbol{X})) \tag{4.24}$$

$$cov(f^*) = k(\boldsymbol{x}^*, \boldsymbol{x}^*) - \boldsymbol{K}(\boldsymbol{X}, \boldsymbol{x}^*)^T \boldsymbol{K}(\boldsymbol{X}, \boldsymbol{X}) \boldsymbol{K}(\boldsymbol{X}, \boldsymbol{x}^*) \tag{4.25}$$

The Gaussian distribution of the prediction f^* has a mean value of $\mu(f^*)$. We generally take $\mu(f^*)$ as the predicted output y^* for the input $\boldsymbol{x}^*$ because the probability of

obtaining $\mu(f^*)$ is highest. The covariance $cov(f^*)$ describes the uncertainty measurement of the result. For the GPC-GAM model, the prediction y^* is typically a value within the range $[-1,+1]$. $y^*>0$ is classified as impact trajectory and vice versa (Equations 4.9, 4.10).

4.3.3. GENERATION OF TRAINING DATASET

In order to generate a training dataset, the boundary conditions of the input space need to be defined. The boundary conditions are defined in terms of a specific scenario concerning the selected CR3BP model, energy level, etc. The selected boundaries will be presented in the next section for different applications. In a predefined domain, the inputs $\boldsymbol{x}$ are randomly generated following a uniform distribution. Equation 4.4 specifies the initial conditions of the spacecraft. The training output is calculated by numerical integration of the equations of motion in Equation 4.1 over one orbital revolution of the initial osculating orbit. The final condition in Equation 4.5 ensures that the spacecraft moves away from the secondary, especially when the spacecraft has been temporally (a duration of no more than 10 orbital periods according to the initial condition) trapped in the vicinity of the secondary. The integrator uses an adaptive step-size Runge-Kutta 7(8) method with a maximum step size of 1×10^{-6}, relative and absolute tolerances of 1×10^{-12}, such that the accuracy of numerical integration is good enough to be used as a ground-truth benchmark.

For the GPC-GAM model, the training outputs are labelled according to the collision situation during propagation. The values +1 and -1 are assigned to impact and safe trajectories, respectively (Equation 4.10). The impact criterion of 300 km above the secondary corresponds to a distance of 6678 km to the center of Earth for the Sun-(Earth+Moon)-spacecraft system, and 2710 km to the center of Callisto for the Jupiter-Callisto-spacecraft system. It is 76,541 km to the center of Jupiter for the Sun-Jupiter-spacecraft system, corresponding to a distance of 5000 km larger than the equatorial radius of Jupiter, considering its thick atmosphere. For the GPR-GAM(-J) model, the training outputs are the changes in the Keplerian elements corresponding to the initial and final state.

In addition to the training dataset, a test dataset D_{test} is generated based on the same procedure. D_{test} is used for independently evaluating the accuracy of the GP models by comparing the predictions to the ground truth.

4.3.4. ACCURACY ASSESSMENT

Two criteria are used to evaluate the performance of the GP models. The Mean Absolute Percentage Error (MAPE) is adopted for the GPC-GAM model:

$$\epsilon_{\text{MAPE}}(N)=\frac{100\%}{N^*}\sum_{i=1}^{N^*}\frac{|C_{\text{GPC-GAM},i}-C_{\text{CR3BP},i}|}{2} \tag{4.26}$$

where N^* is the size of the test dataset, $C_{\text{CR3BP},i}$ represents the ground-truth output obtained using CR3BP propagation by numerical integration, and $C_{\text{GPC-GAM},i}$ is the prediction of the GP model. The denominator 2 is used for the mispredicted sample when $|C_{\text{GPC-GAM},i}-C_{\text{CR3BP},i}|=2$. The value of $\epsilon_{\text{MAPE}}(N)$ is expected to decrease with increasing number of training samples N.

The Mean Absolute Error (MAE) is used as the error measure for the GPR-GAM(-J) model, and is defined as:

$$\epsilon_{\text{MAE}}(N) = \frac{1}{N^*} \sum_{i=1}^{N^*} |y_{\text{GPR-GAM(-J)},i} - y_{\text{CR3BP},i}| \tag{4.27}$$

The accuracy of GP models is expected to benefit from a larger training dataset because more empirical information is used. However, a larger dataset leads to a lower efficiency according to Equation 4.24. In order to balance accuracy and efficiency, the optimal size of the training dataset is decided upon by increasing the number of training samples with steps n until a certain condition is met (Liu et al. 2021). First, a set of error evaluations corresponding to various sizes of the training dataset is defined as:

$$\xi(N) = [\epsilon(N), \epsilon(N+n), ..., \epsilon(N+9n)] \tag{4.28}$$

where $n = 100$. The difference between the maximum and minimum of $\xi(N)$ is defined as the convergence of $\xi(N)$:

$$\Delta\xi(N) = \max(\xi(N)) - \min(\xi(N)) \tag{4.29}$$

For the GPC and GPR models, the optimal size of the training dataset is determined by different conditions, reflecting the differences in character of the two outcomes:

$$N_{\text{GPC}} = \{\text{argmin}(\xi_{\text{MAPE}}(N)) \mid \Delta\xi_{\text{MAPE}}(N) < 5\%\} \tag{4.30}$$

$$N_{\text{GPR}} = \{\text{argmin}(\xi_{\text{MAE}}(N)) \mid \frac{\Delta\xi_{\text{MAE}}(N)}{\min(\xi_{\text{MAE}}(N))} < 5\%\} \tag{4.31}$$

4.4. PERFORMANCE OF GRAVITY ASSIST MAPPING

4.4.1. SUN-(EARTH+MOON)-SPACECRAFT

The input space of the flyby trajectories in the Sun-(Earth+Moon)-spacecraft system is shown in Table 4.1. For each model, the inputs of training samples are randomly generated in the given domains following a uniform distribution. The minimal value of r_p corresponds to a distance of 300 km above the surface of the Earth. The GPR-GAM model covers an energy level from low to high. The GPC-GAM model has a smaller range of energy, which is focused on low-energy conditions only. This is because very few impact trajectories are observed using the full input space of the GPR-GAM, e.g., only three for 20,000 samples. Clearly, such a number of impact samples is not sufficient to train a classification model. After a grid search, we found that most impact samples are located in this sub-space (boundary for GPC-GAM in Table 4.1) of the GPR domain, i.e., 1073 in 20,000 samples. Given a new input $\boldsymbol{x}^*$, the GPC-GAM is used to filter impact samples first, then the GPR-GAM is used to quantify flyby effects over the broader domain.

CLASSIFICATION MODEL

Here, the GPC-GAM model is built for the Sun-(Earth+Moon)-spacecraft problem. The distribution of training samples in the input space is random. The initial condition of

Table 4.1: The input space for the Sun-(Earth+Moon)-spacecraft system. The boundaries are defined such to obtain obvious flyby effects following (Liu et al. 2021).

	GPC-GAM	GPR-GAM
r_p [AU]	[1.000045, 1.02]	[1.000045, 1.02]
r_a [AU]	[1.02, 1.2]	[1.02, 3.0]
i [deg]	[0, 1]	[0, 90]
ω [deg]	[0, 1]	[0, 90]
ϕ [deg]	[-1, 1]	[-25, 25]
C_J [-]	[2.9980, 3.0012]	[0.5107, 3.0012]

true anomaly is $-\pi$, and the final condition is π. An extra boundary condition (Equations 4.4 and 4.5) is introduced so that the spacecraft is at least $2r_H$ away from the Earth. The training outputs are generated by numerically propagating the CR3BP equations of motion. As defined in Equation 4.10, a trajectory is labeled as 'impact' if it enters the sphere of 300 km above Earth's surface during the propagation. The size of the training dataset is increased gradually by 100 training samples. Due to the limited number of impact situations, 10 impact samples are added to every 100 training samples. The test dataset has 500 impact and 500 no-impact cases, following the same way of obtaining the training dataset. In Figure 4.2, the Percentage Accuracy (PA) is shown, which is calculated as $(1-\xi_{\text{MAPE}})$. The accuracy of the impact and no-impact test samples are presented individually. The PA of both curves converges to a level of about 90.0% using at least 3000 training samples. When using 4300 training samples, the best results are 90.8% for the impact case and 90.0% for the no-impact case, respectively. At the start point of the curve, the PA of the impact case is 0.0%, and that of the no-impact case is 100.0%. This is because the number of impact samples is very small, leading to a systematic error. The GPC-GAM model lacks sufficient empirical information, and classifies all test samples as no-impact. Table 4.2 shows an error matrix. The PA of the impact case refers to the true positive rate (TPR), which equals 454/500 = 90.8%. For the no-impact case, the PA is the true negative rate (TNR), i.e., 450/500 = 90.0%. A high TPR means that the model can recognize most of the impact trajectories. This guarantees that less impact cases will be forwarded to the next stage, the GPR-GAM. The TNR determines how many no-impact trajectories are recognized by the classification model.

For the specific input space of the GPC-GAM model, Figure 4.3(a) shows the difference between the impact and no-impact trajectories (both obtained by numerical integration) in terms of a and e. For the samples with a larger than 1.04 AU, almost all impact trajectories have a large eccentricity. These correspond to r_p close to a value of 1.0 AU,

i.e., a closer distance to the Earth during flyby. The Jacobi constant C_J is shown in Figure 4.3(b). Impact samples exist in a range from 2.998 to 3.000893. The lowest value of C_J for the impact and no-impact cases are close (2.998). The largest value of $C_J = 3.000893$ equals that of the L_2 Lagrange point. The Hill's region around the Earth is disconnected from the exterior Hill's region when $C_J > 3.000893$. The spacecraft will not impact the Earth if it was to start outside the Hill's region.

For the test dataset, the actual distributions of collision cases and the ones predicted by the GPC-GAM model are shown in Figure 4.4. The predicted distribution closely resembles the results obtained by the CR3BP propagation, which confirms the validity of the GPC-GAM model.

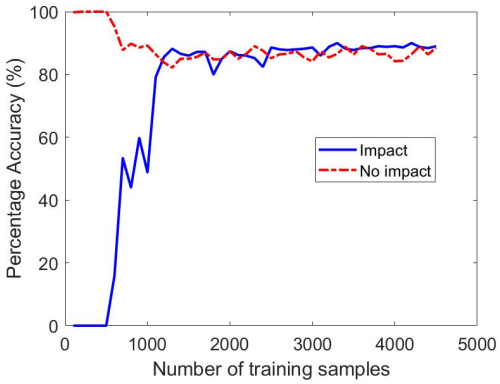


Figure 4.2: Percentage accuracy of the GPC-GAM model for the Sun-(Earth+Moon)-spacecraft system.

Table 4.2: Error matrix of GPC-GAM for the Sun-(Earth+Moon)-spacecraft system.

Truth \ Prediction	Impact	No-impact
Impact	454	46
No-impact	50	450

REGRESSION MODEL

The selected boundary for the input space is defined in Table 4.1. The training inputs are randomly generated in this input space. After eliminating the impact samples using the GPC-GAM model, the training outputs are obtained through numerical integration using Equation 4.1. The GPR-GAM model is built using gradually increasing training samples, from 100 to 5000. As a verification, 500 random training samples are selected to compare the training outputs predicted by the GPR-GAM model with those obtained from

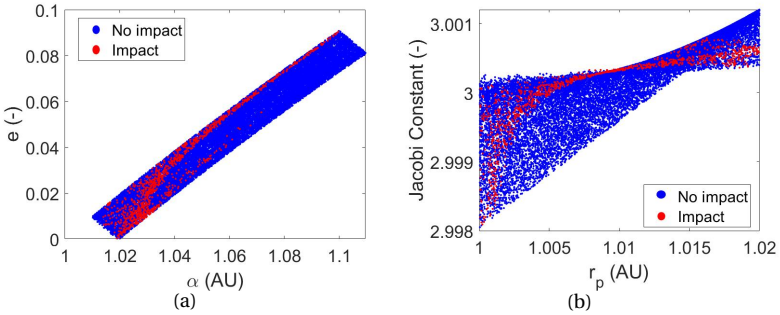


Figure 4.3: The inputs of the impact and no-impact trajectories (Sun-(Earth+Moon)-spacecraft system).

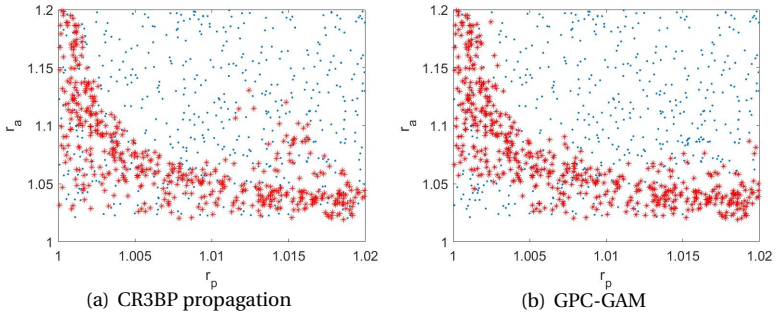


Figure 4.4: Actual and predicted collision cases (Sun-(Earth+Moon)-spacecraft system). Red = impact, blue = no-impact.

the CR3BP propagation. MAE of these results are shown in the bottom of Table 4.3 . Figure 4.5 shows this comparison in terms of δa when the GPR-GAM is trained using 5000 samples. The predicted outputs coincide well with the training outputs, which proves the correctness of the developed model. For each element, the MAE between the results of the GPR-GAM and the CR3BP propagation is shown in Table 4.3. The MAE of predicting δa is only at the magnitude level of 1.0×10^{-6} AU. Using Equation 4.25, the variance of the GAM-GPR prediction is obtained. For 5000 training samples, the predicted standard deviation σ has a range of $[2.62 \times 10^{-4}, 3.50 \times 10^{-4}]$AU. The error bars corresponding to the 95% confidence interval (1.96σ) are plotted in Figure 4.5. It shows that the outputs of CR3BP propagation are within the 95% confidence interval of the GPR predictions.

In order to further assess the accuracy of the GPR-GAM model, an independent (i.e., not part of the training dataset) group of 500 samples is selected as test dataset. For each Keplerian element, three groups of training samples are employed which are generated using different random seed numbers. In addition to the GPR-GAM model, a regression model introducing the Jacobi constant as the sixth input feature is also developed, i.e., the GPR-GAM-J model. The MAE of the flyby effects predicted by both the GPR-GAM and the GPR-GAM-J models is shown in Figure 4.6. Clearly, using a small number of training

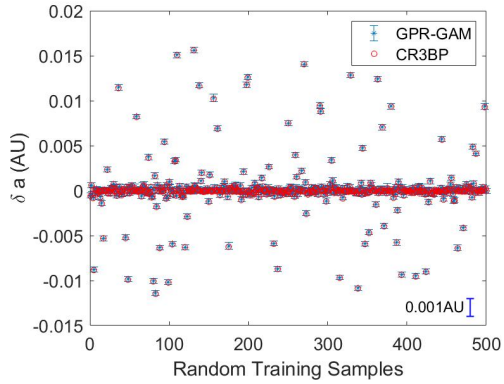


Figure 4.5: Comparison between the training outputs obtained by the GPR-GAM and by the CR3BP propagation (Sun-(Earth+Moon)-spacecraft system).

samples causes an unstable performance: there are fluctuations at the beginning, and the performance differences between the three training datasets is obvious. However, increasing the number of training samples improves both accuracy and consistency. The curves of the three datasets converge when using more than 3000 training samples. This indicates that the GP models start to learn the mechanism of flyby effects and are able to give accurate predictions. When the number of training samples is approaching 5000, the curves of different datasets converge to within 5%, for all elements. The performance of the GPR-GAM model is not influenced by the seed number of the training dataset. Adding the information of the Jacobi constant improves the performance of the model significantly, especially for ω and Ω. Clearly, the three-body energy is a useful feature for quantifying the flyby effects. For the same group of training datasets, the trend of using C_J is similar to that of the results without C_J. This is because the sequence of adding training samples is identical between the GPR-GAM and GPR-GAM-J models, and every batch of 100 new samples provides the same empirical information (the seed number is identical). The MAE of the GAM models is summarised in Table 4.3. The MAE of δa is improved 32% by adding C_J. The accuracy of estimating δa and δe is at the magnitude level of 1.0×10^{-5}. Quantifying $\delta\omega$ has the worst performance, but still an improvement of about 70% when adding C_J information. The prediction of $\delta\Omega$ takes the most advantage of using C_J, improving the accuracy by 75%.

4.4.2. JUPITER-CALLISTO-SPACECRAFT

In order to evaluate the general performance of the GAM models, i.e., in various CR3BP systems with different values for μ, the Jupiter-Callisto-spacecraft scenario ($\mu = 5.668 \times 10^{-5}$) is also selected. The gravity assist is a commonly used technique for designing endgame strategies (Campagnola & Kawakatsu 2012). The outer Galilean moon Callisto is chosen as flyby body in this section because gravity assists by Callisto can be used to save propellant when a spacecraft enters the Jovian system (**?**). The length is normalized using the semi-major axis of the orbit of Callisto.

Table 4.3: The performance of GPR-GAM using the three training datasets (Sun-(Earth+Moon)-spacecraft system). The same pair of training and test groups using the Jacobi constant is presented in the second part.

Group	δa [AU]	δe [-]	δi [rad]	$\delta\omega$ [rad]	$\delta\Omega$ [rad]
1	5.76×10^{-5}	3.46×10^{-5}	6.28×10^{-6}	1.83×10^{-3}	1.86×10^{-4}
2	5.79×10^{-5}	3.42×10^{-5}	6.32×10^{-6}	1.81×10^{-3}	1.88×10^{-4}
3	5.77×10^{-5}	3.39×10^{-5}	6.26×10^{-6}	1.82×10^{-3}	1.91×10^{-4}
$1(J)$	3.86×10^{-5}	2.72×10^{-5}	2.78×10^{-6}	5.49×10^{-4}	4.62×10^{-5}
$2(J)$	3.88×10^{-5}	2.75×10^{-5}	2.81×10^{-6}	5.55×10^{-4}	4.65×10^{-5}
$3(J)$	3.90×10^{-5}	2.71×10^{-5}	2.75×10^{-6}	5.47×10^{-4}	4.68×10^{-5}
			Verification on training samples		
1	1.37×10^{-6}	1.11×10^{-6}	1.54×10^{-7}	1.03×10^{-4}	9.75×10^{-6}

Table 4.4: The input space for the Jupiter-Callisto-spacecraft system.

	GPC-GAM	GPR-GAM
r_p [-]	[1.001439, 1.06]	[1.001439, 1.06]
r_a [-]	[1.08, 1.5]	[1.08, 3.0]
i [deg]	[0, 1]	[0, 90]
ω [deg]	[0, 1]	[0, 90]
ϕ [deg]	[-1, 1]	[-25, 25]
C_J [-]	[2.9912, 3.0099]	[0.5068, 3.0017]

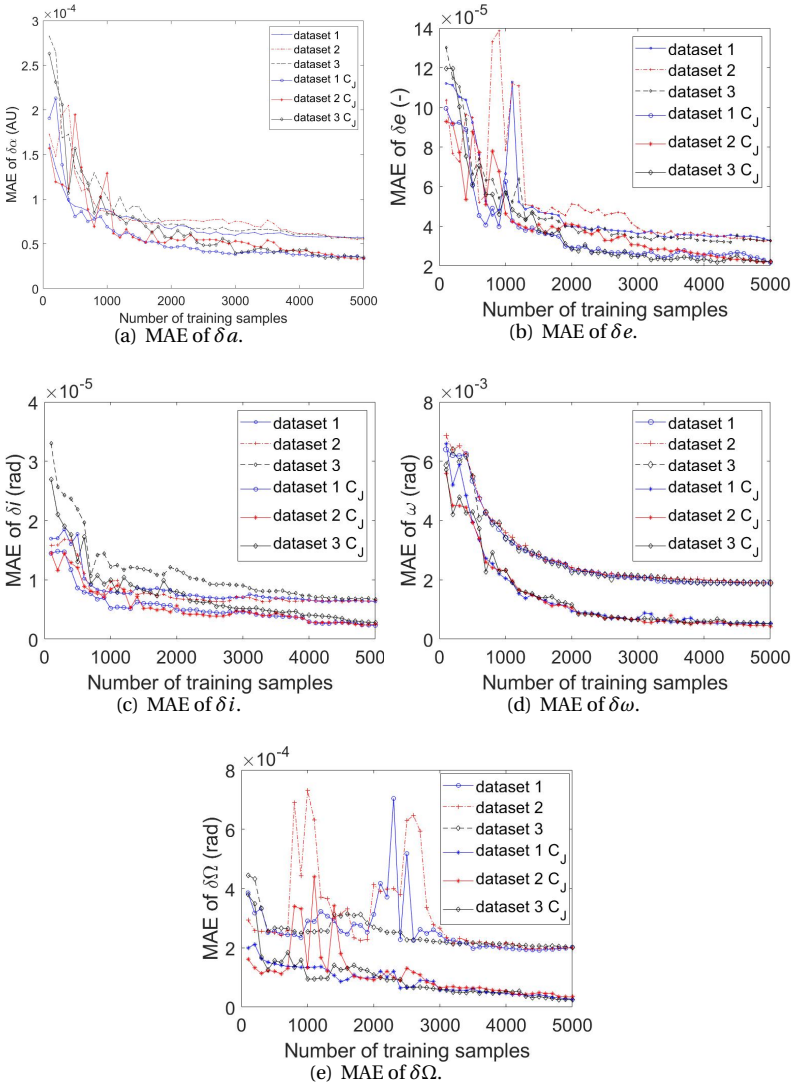


Figure 4.6: The MAE of test samples when using three different training datasets for the Sun-(Earth+Moon)-spacecraft system, each with and without the usage of the Jacobi constant.

CLASSIFICATION MODEL

In Table 4.4, the input space for GPC-GAM is specified to cover an energy range from slightly smaller than that of the L_4/L_5 Lagrange point to a value larger than that of the L_1 Lagrange point. The value of 1.001439 for pericenter distance corresponds to 300 km above the surface of Callisto. For generating the training dataset and test dataset the same strategy is adopted (as shown in Section 4.4.1). Due to the limited number of

impact trajectories, each 100 training samples uses 90 no-impact and 10 impact trajectories. The test dataset has 500 impact and 500 no-impact trajectories. The accuracy of predicting impact trajectories is shown in Figure 4.7. The performance becomes stable using more than 2000 training samples. Increasing the size of the training dataset improves the accuracy steadily. The best PA is 90.6% for impact samples and 90.8% for no-impact samples, respectively. Table 4.5 shows that the TPR is lower than that of Sun-(Earth+Moon) scenario, but the TNR gets better. A smaller number of no-impact trajectories will be falsely eliminated. For the integrated trajectories of the CR3BP, the patterns of input features are shown in Figure 4.8. The impact samples appear in an area similar to that in Figure 4.3. A major difference is shown in the left plot, where more impact trajectories occur with moderate values of e and a larger than 1.15. These samples have a low three-body energy as shown in the green rectangle in the right plot. They are attracted by Callisto during the flyby even though the r_p according to the initial conditions are larger than the radius of the Hill's sphere (0.0268). Figure 4.9 shows the comparison between the results of the GPC-GAM model and the CR3BP propagation. As before, the accuracy of the GPC-GAM model is indicated by the resemblance between the plots. The model has a misclassification for the inputs of $r_p = [1.025, 1.04] \cap r_a = [1.3, 1.4]$, corresponding to the samples in the green rectangle in Figure 4.8(b).

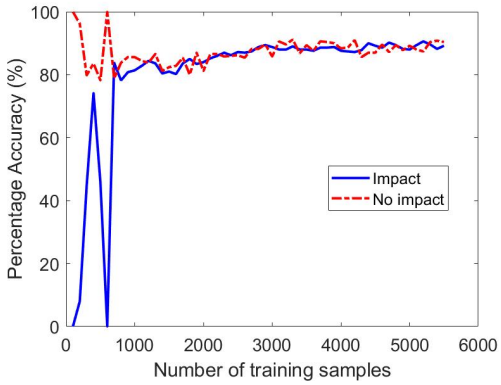


Figure 4.7: Percentage accuracy of the GPC-GAM model for the Jupiter-Callisto-spacecraft system.

Table 4.5: Error matrix of GPC-GAM for the Jupiter-Callisto-spacecraft system.

Truth \ Prediction	Impact	No-impact
Impact	453	47
No-impact	46	454

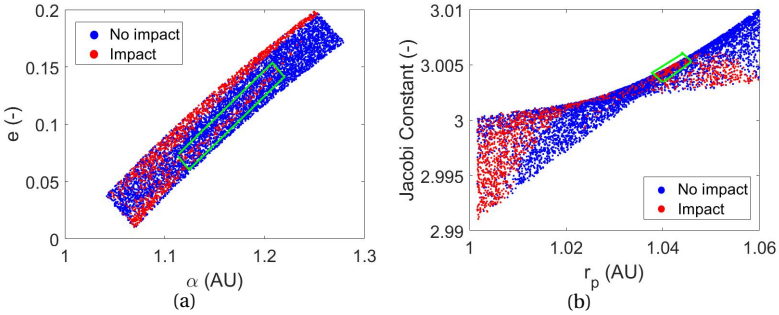


Figure 4.8: The inputs of the impact and no-impact trajectories for the Jupiter-Callisto-spacecraft system.

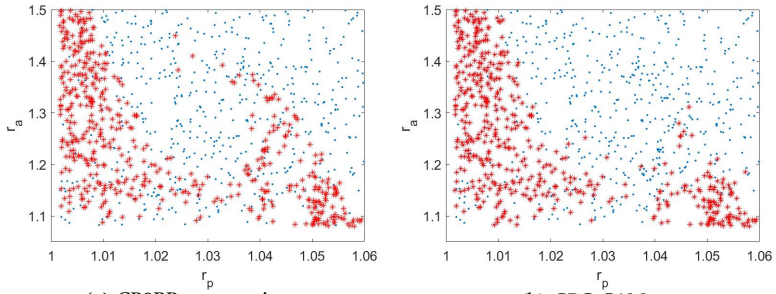


Figure 4.9: Actual and predicted collision cases (Jupiter-Callisto-spacecraft system). Red = impact, blue = no-impact.

REGRESSION MODEL

Here, the GPR-GAM models are built using different groups of training samples (different random seed numbers). For the same group of training samples, a GPR-GAM-J model is developed to evaluate the performance after adding the Jacobi constant as an extra input feature. The test dataset consists of 500 samples randomly generated in the input space presented in Table 4.4. The convergence of the MAE requires more training data than the previous scenario. In particular, the MAE of predicting $\delta\omega$ using 5500 samples improves by 0.3×10^{-3} rad than using less than 5000 samples. Using different random seed numbers has little influence on the convergence. In Figures 4.10(a) and 4.10(b), the effects of adding C_J are difficult to observe but the improvements are significant as presented in Table 4.6. The accuracy of predicting δa and δe is improved by 46.4% and 24.1%, respectively. The effect is most significant for the predictions of $\delta\Omega$, achieving an improvement of 82.7%. Predictions of δi have the best results in terms of MAE, which is only 9.15×10^{-6} rad. Like in the Sun-(Earth+Moon)-spacecraft scenario, predicting $\delta\omega$ is the most difficult. The main reason is that the variations of ω due to a flyby are relatively larger than those of i and Ω. This occurs for initial conditions with small values of eccentricity, especially for the cases $e < 0.1$.

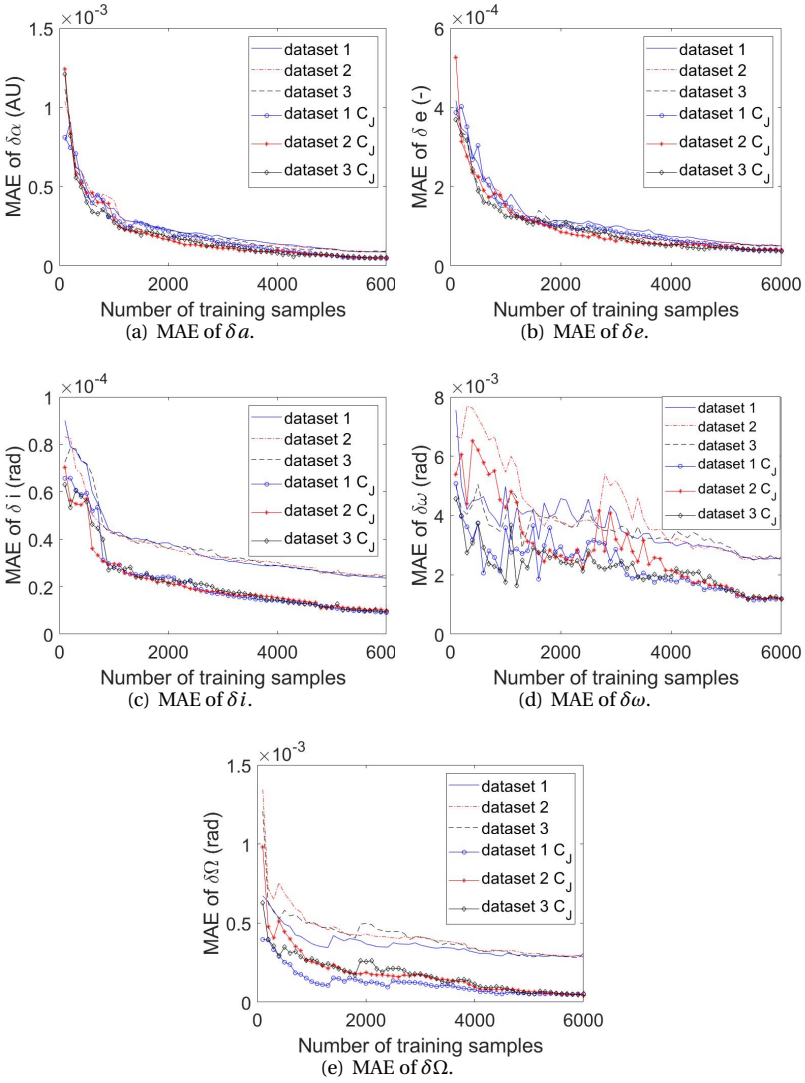


Figure 4.10: The MAE of test samples when using three different training datasets for the Jupiter-Callisto-spacecraft system, with and without usage of the Jacobi constant.

4.4.3. SUN-JUPITER-SPACECRAFT

The Sun-Jupiter-spacecraft system is selected as the third case to evaluate the performance of GAM models for the CR3BP system, with μ (9.537×10^{-4}) larger than the two previous cases. In this system, the values of the Jacobi constant corresponding to the L_1 ($C_J = 3.0384$) and L_2 ($C_J = 3.0397$) Lagrange points also become larger. The input space for the GPR-GAM model covers a broad range of C_J with values from 0.4367 to 3.0967 as shown in Table 4.7. The GPC-GAM has a smaller range such that more cases of crashing

Table 4.6: The performance of GPR-GAM models using three training datasets for the Jupiter-Callisto-spacecraft system. The same pair of training and test groups using the Jacobi constant is presented in the second part.

Group	δa [-]	δe [-]	δi [rad]	$\delta\omega$ [rad]	$\delta\Omega$ [rad]
1	8.90×10^{-5}	5.14×10^{-5}	2.43×10^{-5}	2.51×10^{-3}	2.88×10^{-4}
2	9.05×10^{-5}	5.22×10^{-5}	2.48×10^{-5}	2.54×10^{-3}	2.82×10^{-4}
3	9.01×10^{-5}	5.15×10^{-5}	2.45×10^{-5}	2.49×10^{-3}	2.84×10^{-4}
1(J)	4.77×10^{-5}	3.90×10^{-5}	9.15×10^{-6}	1.17×10^{-3}	4.97×10^{-5}
2(J)	4.83×10^{-5}	3.91×10^{-5}	9.61×10^{-6}	1.16×10^{-3}	5.11×10^{-5}
3(J)	4.93×10^{-5}	3.84×10^{-5}	9.55×10^{-6}	1.19×10^{-3}	5.05×10^{-5}

into Jupiter are included. It focuses on the low-energy level.

CLASSIFICATION MODEL

Since Jupiter is a gas giant, a trajectory would be classified as impact if the closest distance between the spacecraft and the center of Jupiter is smaller than 76,541 km. The size of the training dataset is gradually increased from 100 to 5000. The test dataset has 500 impact and 500 no-impact cases. The percentage accuracy for impact predictions is shown in Figure 4.11. When using more than 3000 training samples, the PA of both cases starts to converge to a level larger than 92.0%. Using 3300 training samples, the PA of impact and no-impact cases are 96.2% and 92.0%, respectively. The accuracy is better than that of Sun-(Earth+Moon)-spacecraft and Jupiter-Callisto-spacecraft. The distribution of impact cases in terms of r_p, r_a and C_J is shown in Figure 4.12. There are two major situations of collision: the first one is a group of cases shown in red at the top of Figure 4.12(a) corresponding to the points with $C_J < 3.0$ in Figure 4.12(b); the second situation shown in the middle of Figure 4.12(a) has larger r_p (initial condition) as well as larger C_J. In Figure 4.12(b), C_J has a narrow range of values when r_p has a value of about 1.06, because it is a projection of the twisted curve (Figure 4.12(c)) onto the $r_p - C_J$ plane. In the 3-dimensional figure, C_J increases as r_a decreases when $r_p < 1.06$, which is in contrast to the situation when $r_p > 1.06$. This also applies to the cases of Sun-(Earth+Moon)-spacecraft and Jupiter-Callisto-spacecraft. In Figure 4.13, the features of collision cases between the CR3BP propagation and the GPC-GAM model are in agreement. 482 out of 500 impact trajectories are identified correctly. Some low-energy collision cases are not identified by the model, which is mainly due to the insufficient number of similar training samples.

REGRESSION MODEL

For the Sun-Jupiter-spacecraft case, the GPR-GAM and GPR-GAM-J models are developed following strategies identical to the previous two cases. The MAE of the test dataset (500 samples) with respect to different orbital elements is shown in Figure 4.14. For each model, three datasets are generated using different random seed numbers. The convergence of the three curves of MAE starts when using more than 5000 training samples.

Table 4.7: The input space for the Sun-Jupiter-spacecraft system.

	GPC-GAM	GPR-GAM
r_p [-]	[1.00009, 1.2]	[1.00009, 1.2]
r_a [-]	[1.15, 1.5]	[1.15, 3.0]
i [deg]	[0, 1]	[0, 90]
ω [deg]	[0, 1]	[0, 90]
ϕ [deg]	[-1, 1]	[-25, 25]
C_J [-]	[2.9894, 3.0481]	[0.4367, 3.0967]

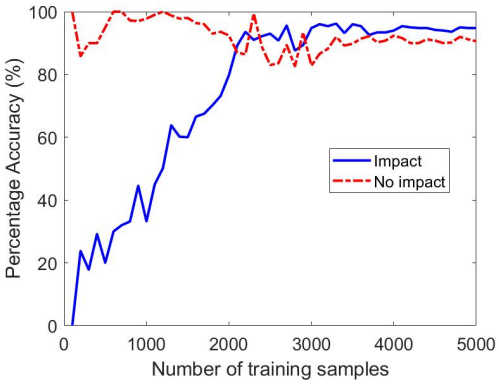


Figure 4.11: Percentage accuracy of the GPC-GAM model for the Sun-Jupiter-spacecraft system.

For the prediction of δe, δi and $\delta\omega$, the models need at least 6000 training samples to become stable. This requires a larger size of the training dataset than for the Jupiter-Callisto-spacecraft system. In Table 4.8, the improvement of accuracy is clearly seen when introducing the Jacobi constant. In particular, the prediction of δi and $\delta\Omega$ is improved by 58% and 43%, respectively. Compared to the results in Tables 4.3 and 4.6, the MAE increases as the mass ratio μ increases.

4.4.4. TIME EFFICIENCY OF GAM MODELS

The computational time of the GAM models consists of two parts: training and prediction. The training process includes the generation of training samples and the GAM model optimization. All simulations are completed using MATLAB® 2018b on the first

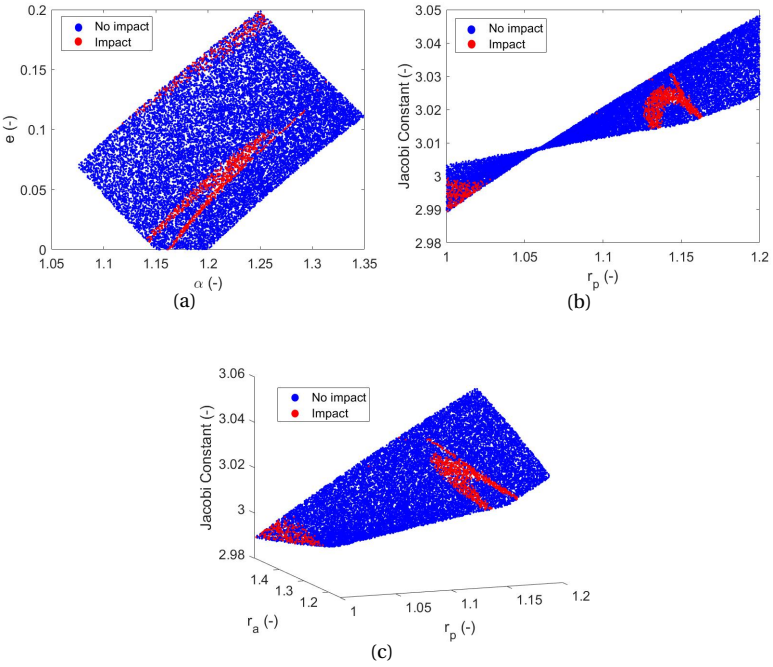
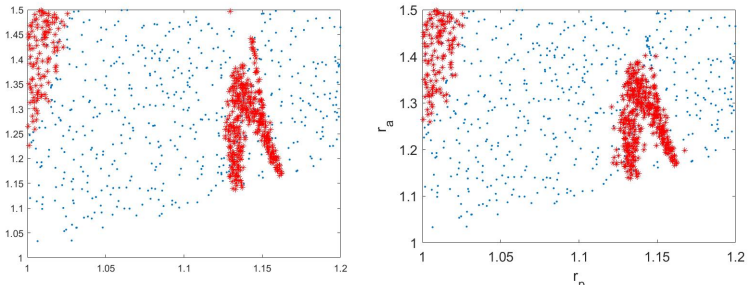


Figure 4.12: The inputs of the impact and no-impact trajectories for the Sun-Jupiter-spacecraft system.



(a) CR3BP propagation (b) GPC-GAM

Figure 4.13: Actual and predicted collision cases (Sun-Jupiter-spacecraft). Red = impact, blue = no-impact.

author's PC (Core i7 CPU and 8.00 GB RAM) (The Mathworks Inc. 2018). Given a training input, the output is obtained by a numerical CR3BP propagation which takes about 0.59 s per sample. Table 4.9 presents the CPU time of training and prediction. Each column corresponds to a specific case in Sections 4.4.1-4.4.3. Training a GAM model, i.e., optimizing hyper-parameters, costs about 53 minutes (CPU) when using 4500 training samples. A multi-start approach is used to avoid being trapped in local minima. The

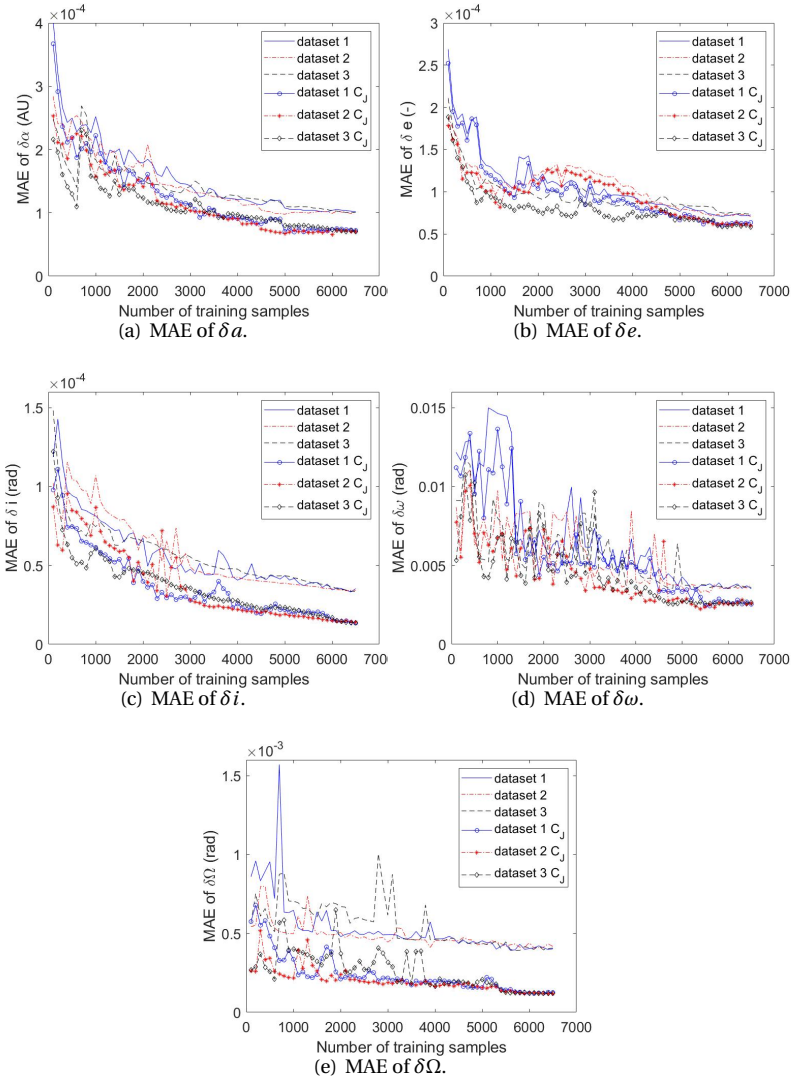


Figure 4.14: The MAE of test samples when using three different training datasets for the Sun-Jupiter-spacecraft system, with and without usage of the Jacobi constant.

conjugate gradient algorithm is randomly initiated at 10 different initial combinations of hyper-parameter values. As indicated by Equation 4.24, the prediction time spent on one test sample depends on two major factors, i.e., the number of training samples and the size of the input vector. Compared to the case of 4500 samples, using an extra 1000 training samples adds more than 30% CPU time to the prediction. The GPR-GAM-J has six input features, and takes an extra 10% CPU time compared with GPR(C)-GAM. Note that the GAM model developed in this work is a supervised machine-learning tool.

Table 4.8: The performance of GPR-GAM models using three training datasets for the Sun-Jupiter-spacecraft system. The same pair of training and test groups using the Jacobi constant is presented in the second part.

Group	δa [-]	δe [-]	δi [rad]	$\delta\omega$ [rad]	$\delta\Omega$ [rad]
1	1.02×10^{-4}	7.08×10^{-5}	3.32×10^{-5}	3.46×10^{-3}	4.01×10^{-4}
2	9.91×10^{-5}	7.11×10^{-5}	3.39×10^{-5}	3.28×10^{-3}	4.07×10^{-4}
3	1.01×10^{-4}	7.30×10^{-5}	3.35×10^{-5}	3.54×10^{-3}	3.92×10^{-4}
1(J)	7.16×10^{-5}	6.05×10^{-5}	1.34×10^{-5}	2.43×10^{-3}	1.20×10^{-4}
2(J)	6.93×10^{-5}	6.03×10^{-5}	1.41×10^{-5}	2.34×10^{-3}	1.18×10^{-4}
3(J)	7.08×10^{-5}	5.92×10^{-5}	1.38×10^{-5}	2.52×10^{-3}	1.20×10^{-4}

Though the training process costs much computational effort, the model can be significantly time-saving when the flyby effects for millions of initial conditions need to be quantified in the search for optimal flybys.For a single sample, the GPR-GAM prediction is more than 5 million times faster than numerical integration.

Table 4.9: Computational time of GAM models using different number of training samples. The computational time is obtained by taking the mean value of 1000 repeated experiments. The value of prediction is the CPU time spent on one test sample.

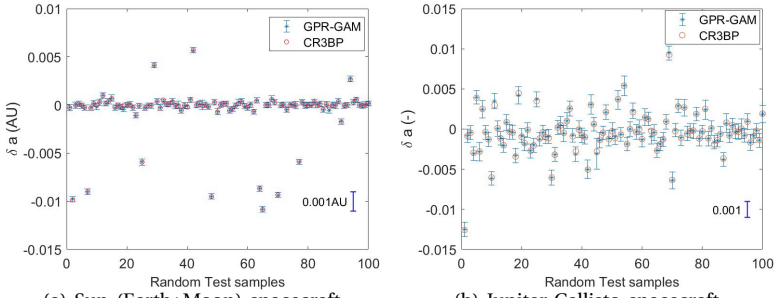
	4500 samples	5500 samples	6000 samples
Training [min]	53.0	66.3	75.7
GPR(C)-GAM [s]	3.9×10^{-6}	5.2×10^{-6}	6.2×10^{-6}
GPR-GAM-J [s]	4.2×10^{-6}	5.8×10^{-6}	6.6×10^{-6}
numerical [s]	0.59	0.59	0.59

4.5. QUALITY ASSESSMENT

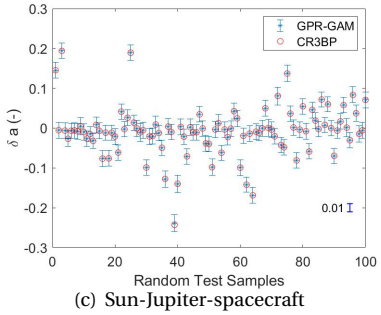
When predicting the output of a single test sample, the result can be obtained by the GAM model using Equations 4.24 and 4.25. In previous sections, the expectation obtained by Equation 4.24 is taken as the predicted output. The GP method has the ability to provide formal uncertainty estimates using the variance σ^2 calculated by Equation 4.25. Figure 4.15 shows the error bars associated with the GAM-GPR predictions of test samples for each CR3BP system. In order to show the error bars more clearly, 100 out of 500 samples are randomly selected. The middle point of the error bar is the predicted output $y_{\text{GPR-GAM}}$ using Equation 4.24. The length of the error bar represents the 95% confidence interval, i.e., $[y_{\text{GPR-GAM}} - 1.96\sigma, y_{\text{GPR-GAM}} + 1.96\sigma]$. For the Sun-(Earth+Moon)-spacecraft system in Figure 4.15(a), the standard deviation σ ranges from 2.65×10^{-4} to 4.83×10^{-4} AU. The ranges of σ for the Jupiter-Callisto-spacecraft and the Sun-Jupiter-spacecraft systems become bigger, which are from 4.21×10^{-4} to 9.86×10^{-4} and from 9.72×10^{-3} to 1.20×10^{-2}, respectively. Table 4.10 presents σ of five orbital elements.

Clearly, the uncertainty of the GAM-GPR preditions increases when the mass ratio μ is bigger. With a larger influence of the secondary, the absolute effect of a flyby can be expected to become larger, but also GP models' inherent uncertainty. This result is consistent with the increasing values of the MAE in Tables 4.3, 4.6 and 4.8. The variance provides useful information about the uncertainty of predictions. For all systems, the physical truth, i.e., the outputs of CR3BP propagation by numerical integration y_{CR3BP}, are well included in the 95% confidence interval of the GAM-GPR predictions.

In order to assess the performance of these predictions, 95% credibility interval is calculated for each system. Table 4.11 presents the upper bound of this uncertainty interval, out of 500 test samples. such that 95% of the absolute errors between $y_{\text{GPR-GAM}}$ and y_{CR3BP} lie in such interval. From Bayesian analysis point of view, the value represents the probability of the error for predicting the output of an arbitrary sample smaller than the upper boundary. In contrast, the confidence interval shown in Figure 4.15 explains the predictive uncertainty of the GP method with a focus on the procedure. If the process of making predictions is repeated enough times, 95% of the intervals should contain the true value.



(a) Sun-(Earth+Moon)-spacecraft (b) Jupiter-Callisto-spacecraft



(c) Sun-Jupiter-spacecraft

Figure 4.15: The uncertainty as predicted by the GPR-GAM models. Samples of the flyby effects on the semi-major axis, for all three systems, based on the GPR-GAM model (including 95% confidence intervals) and ground truth.

Table 4.10: The range of standard deviation σ associated with the orbital elements for different CR3BP systems.

	Sun-(Earth+Moon)-S/C	Jupiter-Callisto-S/C	Sun-Jupiter-S/C
δa [-]	$[2.65,4.83]\times10^{-4}$	$[4.21,9.86]\times10^{-4}$	$[9.72,12.04]\times10^{-3}$
δe [-]	$[1.30,2.57]\times10^{-4}$	$[2.36,5.60]\times10^{-4}$	$[6.30,8.59]\times10^{-4}$
δi [rad]	$[1.99,2.74]\times10^{-5}$	$[3.55,7.40]\times10^{-5}$	$[5.05,9.17]\times10^{-5}$
$\delta\omega$ [rad]	$[3.92,7.08]\times10^{-3}$	$[5.35,8.97]\times10^{-3}$	$[7.69,12.38]\times10^{-3}$
$\delta\Omega$ [rad]	$[4.71,9.52]\times10^{-4}$	$[6.31,11.20]\times10^{-4}$	$[9.18,14.60]\times10^{-4}$

Table 4.11: The upper bound of 95% uncertainty interval associated with the orbital elements for different CR3BP systems.

	Sun-(Earth+Moon)-S/C	Jupiter-Callisto-S/C	Sun-Jupiter-S/C
δa [-]	$[2.05]\times10^{-4}$	$[2.85]\times10^{-4}$	$[2.92]\times10^{-4}$
δe [-]	$[7.63]\times10^{-5}$	$[9.48]\times10^{-5}$	$[12.30]\times10^{-4}$
δi [rad]	$[2.21]\times10^{-5}$	$[6.49]\times10^{-5}$	$[8.11]\times10^{-5}$
$\delta\omega$ [rad]	$[4.28]\times10^{-3}$	$[5.17]\times10^{-3}$	$[10.23]\times10^{-3}$
$\delta\Omega$ [rad]	$[4.06]\times10^{-4}$	$[7.01]\times10^{-4}$	$[9.78]\times10^{-4}$

4.6. CONCLUSIONS

A design tool, Gravity Assist Mapping, for quantifying the gravity assist effects is developed in this work. This tool is based upon the Gaussian Process method, a supervised machine-learning model. The applications in three CR3BP systems are considered: Sun-(Earth+Moon)-spacecraft, Jupiter-Callisto-spacecraft and Sun-Jupiter-spacecraft. The values of the mass ratio μ differ by a factor of about 300. For each system, a classification model (GPC-GAM) and a regression model (GPR-GAM) are developed for different purposes.

The GPC-GAM model is used for detecting impact trajectories given arbitrary initial conditions. This model can be used in combination with the regression model for eliminating impact samples, or can be used stand-alone for mission analysis of impact avoidance. The values of TPR and TNR being better than 90% allows it to recognize and eliminate impact samples effectively. The main contribution of the GPC-GAM model is that it can be used to identify qualitatively the safe region in the input space. The TNR would identify most of the collision cases reducing the need have to propagate all the initial conditions. However, the current accuracy is not high enough for a detailed mission design. The input space is much smaller than that of the regression case due to the limited number of impact occurrences. The GPC-GAM model needs sufficient empirical information to learn the characteristics of impact trajectories. How to omit impact samples more effectively, especially for some low-energy situations, is an interesting topic for further study.

The results of the GPR-GAM model demonstrate that the method can be used in CR3BP systems with a wide range of mass ratio parameters μ. Introducing the Jacobi constant as an additional input feature improves the accuracy by more than 20%, and for some parameters even by 70%. Representing the three-body energy, the Jacobi constant is proven to be an important factor for the GAM to learn gravity assist effects. The MAE for predicting δa is only 3.86×10^{-5} AU for the Sun-(Earth+Moon)-spacecraft scenario. Predicting $\delta\omega$ and $\delta\Omega$ is more difficult than that of the variations in other Keplerian elements. More training samples are required for the CR3BP with bigger μ. Compared to a semi-analytical method, e.g., the Keplerian map, this method can be applied to a wider range of three-body energies, from 0.501 to 3.002. The GAM models have the ability to assess the uncertainty of predictions. The outputs obtained by the CR3BP propagation are well within the 95% confidence interval of the GAM predictions. The GAM model has an advantage of time efficiency over numerical methods, such as the ones relying on a numerical integration using CR3BP equations of motion or the flyby map.

4.7. ACKNOWLEDGEMENT

The author Liu gratefully acknowledges the China Scholarship Council (CSC) for its financial support.

5

CONCLUSIONS AND RECOMMENDATIONS

This thesis mainly investigated the dynamics of gravity assists in the Circular Restricted Three-Body Problem. In order to quantify the flyby effects, the Gravity Assist Mapping technique was developed as a design tool. The models of Gravity Assist Mapping were constructed based on the idea of the Gaussian process method, a machine-learning technique. First, quantifying the flyby effects in the planar CR3BP was solved using GPR. The results demonstrated the availability of GAM models in a simplified three-body regime, the planar Sun-(Earth+Moon)-spacecraft system (Chapter 2). Second, the proposed models were applied to the fully spatial CR3BP (Chapter 3). The dimension of phase space was increased by two. The GAM models were correspondingly developed further, to accommodate the increased complexity of the problem. The flybys under different three-body energy levels were discussed in terms of the Jacobi constant. At last, a classification model (GPC) was developed for the identification of trajectories impacting the secondary (Chapter 4). For the GPR, the Jacobi constant was incorporated into the model to improve the accuracy of prediction. Both GPC and GPR models were applied to the Sun-(Earth+Moon)-spacecraft, Jupiter-Callisto-spacecraft and Sun-Jupiter-spacecraft systems.

The research questions presented in Chapter 1 were addressed in Chapters 2-4. The conclusions on them are given in this chapter. Recommendations and possible future work will be summarised in full.

5.1. CONCLUSIONS

(RQ.a) Can a machine-learning method be used to quantify the GA effects in the three-body regime?

This question was gradually addressed through three steps. First, the Circular Restricted Three-Body Problem is selected as the regime to investigate the flyby effects. Chapter 2 started to solve the question in the planar CR3BP. The results demonstrated

the GP method to be a competent approach to quantify the flyby effects. Second, the GAM models were modified and applied to the spatial CR3BP in Chapter 3. The method was proved to be effective for the 3-D case, but needed more training data to converge. In Chapter 4, the last step focused on improving the performance of quantification and classifying impact trajectories. For the GPR, the improvement on accuracy was achieved by introducing the Jacobi constant into the training and prediction process. The identification of impact trajectories was performed by the GPC model. This model also used the concept of Gaussian process but worked with outputs consisting of categories labeled as impact and no-impact. When compared to a semi-analytical method, the GAM models made more accurate and more efficient predictions. This work makes two major contributions to astrodynamics. On the one hand, the characteristics of GA in the CR3BP can be investigated by a great deal of accurate data produced by the GPR model, providing a deep understanding of the third-body effect. On the other hand, the GPR model can be considered to design multi-flyby missions involving an optimization problem, and offers the advantage of a high efficiency to update the post-flyby status.

More conclusions about the proposed method are given below, to address the following questions in full detail.

(RQ.b) How can the Gaussian Process method be used to learn the dynamics of flyby's in a planar CR3BP framework?

Chapter 2 answered this question. The GAM model was constructed based on the GPR approach. In order to learn the dynamics of a flyby in the planar CR3BP framework, GAM was trained by a dataset consisting of thousands of samples. The training inputs were generated in a predefined phase space using various sampling methods. The integration was used to calculate training outputs through equations of motion in the planar CR3BP, i.e., Sun-(Earth+Moon)-spacecraft in this chapter. The training dataset serves as the ground truth for the machine-learning approach. The generation of the training dataset takes into account elliptical orbits of low, moderate and high eccentricity. A covariance function (SUM) was developed combining the cosine term and the rational quadratic term with automatic relevance determination (RQARD), which captures well the dynamics of flybys. The optimal number of training samples was determined by choosing a stable value of RMSE on the test dataset, which was generated randomly in the input space.

After training, the GPR-based model can assess the flyby effect more efficiently given the initial condition of a particle, compared to methods based on numerical integration. The CPU time of a single prediction is only 1.16×10^{-6} s. In terms of three-body energy, the domain of applicability is beyond that of the Keplerian map. The results demonstrate that the GPR-based Gravity Assist Mapping has a good performance on accuracy, achieving an RMSE of 2.38×10^{-4} AU for predicting semi-major axis variation. Compared to the Keplerian Map, significant improvements have been made, in particular when using a combined covariance function and stratified random sampling.

(RQ.c) Can a Gaussian Process Regression model predict the flyby effects in a fully spatial CR3BP for a wide range of three-body energies, and if so, with what accuracy?

For the fully spatial CR3BP, a GAM model was developed in Chapter 3. Due to the

increased complexity of a fully spatial CR3BP compared to that of the planar one, a larger training dataset is required. This modification was made based on the basic assumption of any GP method: close inputs in the input space have close outputs. The increased dimensionality needs more training samples to guarantee the required sampling density that is required for accurate output predictions. A criterion using MAE was proposed to select the minimum number of training samples for every orbital element.

In terms of accuracy, the covariance function RQARD performs best. Based on the optimized values of the length-scale, the influence of different input elements on flyby effects was discussed. The robustness of GPR-GAM was illustrated by changing the training and test datasets. It indicates that quantifying δa has the best generalization property. The prediction of $\delta\omega$ is more difficult than that of the other output elements. The results show the ability of the GPR-GAM model to predict the flyby dynamics of the CR3BP system with relatively high accuracy compared to the semi-analytical Kick Map (Alessi & Sánchez 2016). The GPR-GAM model has been shown to be a versatile tool that can be applied for a wide range of energy levels. The quality of predictions in the low-energy domain is almost comparable to that in the high-energy part. The input space considers only the initial conditions of the spacecraft starting from the exterior Hill's region. The phenomenon of the spacecraft temporarily being captured by the secondary is observed, which is not amenable to a patched-conics technique. The CPU time for prediction is a factor 10^3 faster than that of the Kick Map. The MAE values for predicting δa and δi are 5.7×10^{-5} AU and 6.3×10^{-6} rad, respectively. Quantifying $\delta\omega$ has a larger MAE of 1.9×10^{-3} rad.

(RQ.d) Can the performance of GP models be improved by using specific characteristics of the CR3BP? What classification model can be used to identify collision trajectories?

Chapter 4 addressed these two questions. The performance of a supervised learning tool can be improved mainly by two aspects: the training dataset and the model construction. Chapter 3 has adapted the model according to the characteristics of the spatial CR3BP. This chapter created a more informative training dataset of quantifying flyby effects. A GPR-GAM-J model was developed by adding Jacobi constant information. Introducing the Jacobi constant as an additional input feature improves the overall accuracy by more than 20 %. The Jacobi constant representing the three-body energy is proved to be an important factor for the GAM to learn gravity assist effects. The error for predicting δa is only 3.86×10^{-5} AU for the Sun-(Earth+Moon)-spacecraft scenario. Compared to a semi-analytical method, e.g., the Keplerian map, this method can be applied to a wider range of three-body energies, from 0.501 to 3.002. The results in different CR3BP systems demonstrate that the model can be used in scenarios with a different mass ratio parameter μ. More training samples are required for the CR3BP with bigger μ. The GPR-GAM model is able to measure the uncertainty associated with the predictions. The results show that 95 % confidence region of the GAM predictions well include the outputs obtained by the CR3BP propagation.

It was found that trajectories impacting the secondary exist for the input space defined in this thesis, especially for low energy levels. The GPR-GAM model cannot identify these trajectories given initial conditions. To this end, a GPC-GAM model

was developed for classifying impact and no-impact cases. This model can be used in combination with the regression model for eliminating impact samples, or used stand-alone for mission analysis of impact avoidance. The accuracy of true positive and true negative rates, both better than 90 %, allows it to identify and eliminate impact samples effectively. The GAM model has a great advantage of time efficiency over numerical methods, such as the numerical propagation using CR3BP equations of motion or the flyby map. Since the GP method uses training data when predicting, the CPU time of prediction is proportional to the size of the training dataset. When using 5500 training samples, GPR-GAM-J costs 5.8×10^{-6} s to predict a single test sample. The GP models are easy to be developed. For various CR3BP systems, it can be built by simply changing the mass ratio parameter.

5.2. RECOMMENDATIONS

There are still many open questions related to the proposed method. Here, the recommendations for future work are pointed out.

The boundary of the current input space is limited. The GP models learn the dynamics of flybys through a training dataset, which is generated in a predefined input space. The predictions are only valid for this input space. Although a model can be built for any new area in the input space, the construction of a new, extra model requires extra effort. The boundary selected in this thesis is supposed to be sufficiently broad because it contains energy levels from low to high. Also: it is realistic for learning flyby effects, an initial state that is too far off is paintless. However, other areas in the input space could be interesting for specific missions. There two potential solutions: 1) Define an input space broad enough to meet the requirements of almost every mission. Then, the boundary for each orbital element has to be extended. In order to preserve the sampling density, the size of the training dataset needs to be increased correspondingly. This would weaken the efficiency advantage of the GP models; 2) Separate the input space into a number of subspaces and develop a GAM model for each subspace. The training data and optimized hyper-parameters of each model are saved in a library. This approach is based on the off-the-shelf concept.

When increasing the number of training samples, the error of predictions converge after a given number of samples and at a non-zero value. This implies a systematic error in the modelling that additional training data cannot resolve. The specification of the model selection can help to refine the predictions of the model, e.g., using covariance functions that could better interpret the problem in this work. In Chapters 2 and 3, various functions have been used to find a structure suitable for predicting the flyby effect. Methods such as spectral mixture kernels and mixtures of Gaussian processes can be used to reduce systematic modelling issues in future work (Tresp 2001, Parra & Tobar 2017).

In the current research, the initial position of the spacecraft is located in the exterior Hill's region. For the initial osculating orbit, the periapsis passage occurs when the phasing angle equals zero. The flybys only considered the cases of the semi-major axis larger than 1 AU. Two improvements are necessary for future work: 1) For the situation of the semi-major axis smaller than 1 AU, the cases of the initial condition that the spacecraft

does start at the apoapsis is to be considered; 2) GAM models for the situations of the semi-major axis smaller than 1 AU and flybys happening at the apoapsis passage are to be developed.

Adding the Jacobi constant into inputs demonstrated the flexibility of the GPR technique to allow the mapping function to have different formulations. In order to explore the applications of GPR-GAM, it is of interest to use an alternative set of orbital elements. One option is the features used by (Campagnola et al. 2012), which is not available to the current semi-analytical methods. A Poincaré section like being used in Chapter 4 should be set, because the Tisserand parameter is one of the features. Another option is to replace the inertial reference frame with the rotating reference frame, and employ the coordinates of position and velocity as inputs and outputs. The Poincaré section also works for this approach to decrease the dimension space. Another improvement that can be made is to extend the time of flight to multiple orbital periods, such that more complex dynamics is involved. Once a GP model is trained, the prediction spends exactly the same amount of CPU time as that for the case with only one orbital period. So, more CPU time could be saved on predicting the status of the spacecraft. How to train GP models to learn the more complex behavior is a new challenge obviously. In Chapter 3, the GP method is proved to work for the case of a test sample temporarily being captured. It would be a topic for future work to see how the GP method is applicable to different events, e.g., direct, retrograde flybys, temporary captured orbits, collision orbits.

For the classification model, the accuracy of about 90 % is not sufficient for detailed mission design. This is mainly due to the limited number of impact samples. The GPC-GAM model is proved to be able to identify the region that would lead to collision trajectories. The accuracy of prediction can be improved by generating the training samples within such region to increase the number of impact samples. Regarding the range of input space, it is much smaller than that of the regression case due to the limited number of impact occurrences obtained from a wider range. The GPC-GAM model needs sufficient empirical information to learn the characteristics of impact trajectories. Although the impact trajectories have been detected effectively, some cases outside the predefined input space could be overlooked. For some part of the input space, the possibility of impact is extremely low leading to few impact samples. The number of these samples is not sufficient for adequately training a GPC model. To generate a large number of samples randomly will definitely solve the problem, but the efficiency advantage of machine learning is undermined. In order to omit impact samples more effectively, how to apply the GPC model to a larger input space is recommended for future study.

After successfully applying GP models to the CR3BP, a logical candidate for subsequent investigations is the elliptic R3BP (ER3BP). The envisioned applications include the Sun-Mercury-spacecraft system. The eccentricity of Mercury's orbit (0.207) is not negligible. To investigate the flybys with respect to Mercury using a GP method, the ER3BP framework is preferable. Apparently, the structures like Hill's sphere and the motion of the small particle in the ER3BP are different from that of the CR3BP. The machine-learning technique is expected to contribute to exploring dynamics in this system.

REFERENCES

Alessi, E. & Sánchez, J. (2016), 'Semi-Analytical Approach for Distant Encounters in the Spatial Circular Restricted Three-Body Problem', *Journal of Guidance, Control and Dynamics* **39**(2), 351–359.

Anderson, J., Campbell, J., Ekelund, J., Ellis, J. & Jordan, J. (2008), 'Anomalous orbital-energy changes observed during spacecraft flybys of Earth', *Physical Review Letters* **100**(9), 091–102.

Bate, R. R., Mueller, D. D. & White, J. E. (1971), 'Fundamentals of astrodynamics', *Courier Corporation* pp. 3–419.

Benkhoff, J., Van Casteren, J., Hayakawa, H., Fujimoto, M., Laakso, H., Novara, M., Ferri, P., Middleton, H. & Ziethe, R. (2010), 'BepiColombo—comprehensive exploration of Mercury: Mission overview and science goals', *Planetary and Space Science* **58**(1-2), 2–20.

Beutler, G. (2004), 'Methods of Celestial Mechanics: Volume I: Physical, Mathematical, and Numerical Principles', *Springer Science Business Media* pp. 152–174.

Biesbroek, R. & Janin, G. (2000), 'Ways to the Moon', *ESA bulletin* **68**(103), 92–99.

Boutonnet, A., Schoenmaekers, J., Martens, W. & Yamaguchi, T. (2014), 'Mission analysis update for the Jupiter Icy Moon Explorer (JUICE)', *AAS/AIAA Space Flight Mechanics Meeting, American Astronautical Society* p. 370.

Bouwman, L., Liu, Y. & Cowan, K. (2019), 'Gaussian Process models for preliminary low-thrust trajectory optimization', *Astrodynamics Specialist Conference, Portland, ME* (873).

Breen, P. G., F. C. B. T. (2019), 'Newton vs the machine: solving the chaotic three-body problem using deep neural networks', *ArXiv* **000**, 1–6.

Broadfoot, A. L., Shemansky, D. E. & Kumar, S. (1976), 'Mariner 10: Mercury atmosphere', *Geophysical Research Letters* **3**, 10.

Broucke, R. (1988), 'The Celestial Mechanics of Gravity Assist', *AIAA/AAS Astrodynamics Conference, Minneapolis, Minnesota* .

Campagnola, S., Boutonnet, A., Schoenmaekers, J., Grebow, D. J., Petropoulos, A. E. & Russell, R. P. (2014), 'Tisserand-leveraging transfers', *Journal of Guidance, Control, and Dynamics* **37**(2), 1205–1222.

Campagnola, S. & Kawakatsu, Y. (2012), 'Three-Dimensional Resonant Hopping Strategies and the Jupiter Magnetospheric Orbiter', *Journal of Guidance, Control, and Dynamics* **35**(1), 340–344.

Campagnola, S. & Russell, R. (2010), 'Endgame problem part 2: multibody technique and the Tisserand-Poincaré graph', *Journal of Guidance, Control, and Dynamics* **33**(2), 476–486.

Campagnola, S., Skerritt, P. & Russell, R. (2012), 'Flybys in the planar, circular, restricted, three-body problem', *Celestial Mechanics and Dynamical Astronomy* **113**(3), 343–368.

Chirikov, R. V. & Vecheslavov, V. V. (1989), 'Chaotic dynamics of comet Halley', *Astronomy and Astrophysics* **221**, 146–154.

Clairaut, A. (1747), 'Du Systéme du Monde dans les Principes de la Gravitation Universelle', *Memoires de l'Acaddmie royale des Sciences de Paris* pp. 329–364.

Conley, C. C. (1968), 'Low energy transit orbits in the restricted three-body problems', *SIAM Journal on Applied Mathematics* **16**(4), 732–746.

D'Amario, L. A., Bright, L. E. & Wolf, A. A. (1992), 'Galileo trajectory design', *Space science reviews* **60**(1-4), 23–78.

de Almeida Prado, A. F. B. & de Felipe, G. (2007), 'An analytical study of the powered swing-by to perform orbital maneuvers', *Advances in Space Research* **40**(1), 102–112.

Dunne, J. A. (2011), 'Mariner 10 Venus encounter', *Science* **183**(4131), 1289–1291.

Duvenaud, D. (2014), 'Automatic model construction with Gaussian processes', *Doctoral dissertation, University of Cambridge* .

Euler, L. (1772), 'Theoria motuum lunae nova methodo pertractata', *Academiae imperialis scientiarvm* **2**(22), 1–411.

Flandro, G. A. (1966), 'Fast reconnaissance missions to the outer solar system utilizing energy derived from the gravitational field of Jupiter', *Astronautica Acta* **12**(4), 329–337.

Gao, A. & Liao, W. (2019), 'Efficient gravity field modeling method for small bodies based on Gaussian process regression', *Acta Astronautica* **157**, 73–91.

Gawlik, E. S., Marsden, J. E., Campagnola, S. & Moore, A. (2009), 'Invariant manifolds, discrete mechanics, and trajectory design for a mission to Titan', *Advances in the Astronautical Sciences* **134**(3), 1887–1904.

Glassmeier, K., Boehnhardt, H., Koschny, D., Kührt, E. & Richter, I. (2007), 'The Rosetta mission: Flying towards the origin of the Solar System', *Space Science Review* **128**, 1–21.

Greenberg, R., Carusi, A. & Valsecchi., G. B. (1988), 'Outcomes of planetary close encounters: a systematic comparison of methodologies', *Icarus* **75**(1), 1–29.

Grover, P. & Ross, S. (2009), 'Designing trajectories in a planet-moon environment using the controlled keplerian map', *Journal of Guidance Control and Dynamics* **32**(2), 437–444.

Gurnett, D. A. & Kurth, W. S. (2019), 'Plasma densities near and beyond the heliopause from the Voyager 1 and 2 plasma wave instruments', *Nature Astronomy* **3**(11), 1024–1028.

Gurnett, D. A., Kurth, W. S., Burlaga, L. F. & Ness, N. F. (2013), 'In situ observations of interstellar plasma with Voyager 1', *Science* **341**(6153), 1489–1492.

Hassler, D. M., Zeitlin, C., Wimmer-Schweingruber, R. F., Ehresmann, B., Rafkin, S., Eigenbrode, J. L., Burmeister, S., Brinza, D. E., Weigle, G., Böttcher, S., Böhm, E., Burmeister, S., Guo, J., Köhler, J., Martin, C., Reitz, G., Cucinotta, F. A., Kim, M., Grinspoon, D., Bullock, M. A., Posner, A., Gómez-Elvira, J., Vasavada, A., Grotzinger, J. P. & MSL Science Team (2014), 'Mars' surface radiation environment measured with the Mars Science Laboratory's Curiosity rover', *Science* **343**, 6169.

He, H. & Siu, W. C. (2011), 'Single image super-resolution using Gaussian process regression', *CVPR 2011* pp. 449–456.

Hinton, Y. B. & LeCun, Y. (2019), 'A prize for discoveries past, present and future', *Nature* **568**, 526–531.

Howell, K., Marchand, B. & Lo, M. (2001), 'Temporary satellite capture of short-period Jupiter family comets from the perspective of dynamical systems', *Journal of Astronautical Sciences* **49(4)**, 539—-557.

Jerg, S., Junge, O. & Ross, S. (2009), 'Optimal capture trajectories using multiple gravity assists', *Communications in Nonlinear Science and Numerical Simulation* **14**(12), 437–444.

Koon, W., Lo, M., Marsden, J. & Ross, S. (2000), 'Heteroclinic connections between periodic orbits and resonance transitions in celestial mechanics', *Chaos: An Interdisciplinary Journal of Nonlinear Science* **10**, 427–469.

Koon, W., Lo, M., Marsden, J. & Ross, S. (2001), 'Low energy transfer to the moon', *Celestial Mechanics and Dynamical Astronomy* **81**, 63–73.

Krige, D. (1951), 'A statistical approach to some basic mine valuation problems on the Witwatersrand', *Journal of the Southern African Institute of Mining and Metallurgy* **52**(6), 119–139.

Labunsky, A. V., Papkov, O. V. & Sukhanov, K. G. (2017), 'Multiple gravity assist interplanetary trajectories.', *Routledge* pp. 8–31.

Lagrange, J. L. (1772), 'Essai sur le probleme des trois corps', *Prix de l'académie royale des Sciences de paris* **9**(292).

Lantoine, G., Russell, R. & Campagnola, S. (2011), 'Optimization of low-energy resonant hopping transfers between planetary moons', *Acta Astronautica* **68**(7-8), 1361–1378.

LeCun, Y., Bengio, Y. & Hinton, G. (2015), 'Deep learning', *Nature* **521**(7553), 436–444.

Liu, Y., Noomen, R. & Visser, P. (2021), 'A Gravity Assist Mapping Based on Gaussian Process Regression', *The Journal of the Astronautical Sciences, accepted for publication* .

Longuski, J. M. & Williams, S. N. (1991), 'Automated design of gravity-assist trajectories to mars and the outer planets', *Celestial Mechanics and Dynamical Astronomy* **52**(3), 207–220.

Malyshkin, L. & Tremaine, S. (1999), 'The Keplerian map for the planar restricted three-body problem as a model of comet evolution', *Icarus* **141**(2), 341–353.

McKay, M. D., Beckman, R. J. & Conover, W. J. (1979), 'Comparison of three methods for selecting values of input variables in the analysis of output from a computer code', *Technometrics* **21**(2), 239–245.

Mohri, M., Rostamizadeh, A. & Talwalkar, A. (2018), 'Foundations of machine learning', *MIT press* pp. 6–8.

Ness, N. F., Behannon, K. W., Lepping, R. P., Whang, Y. C. & Schatten, K. H. (1974), 'Magnetic field observations near Mercury: Preliminary results from Mariner 10', *Science* **185**, 4146.

Neves, R., Sánchez, J. P., Colombo, C. & Alessi, E. M. (2018), 'Analytical and Semi-Analytical Approaches to the Third-Body Perturbation in Nearly Co-Orbital Regimes', *69th International Astronautical Congress, Bremen, Germany* pp. 1–10.

NguyenTuong, D., Seeger, M. & Peters, J. (2009), 'Model learning with local gaussian process regression.', *Advanced Robotics* **23**(15), 2015–2034.

Olds, A., Kluever, C. & Cupples, M. (2007), 'Interplanetary Mission Design Using Differential Evolution', *Journal of Spacecraft and Rockets* **44**(5), 1060–1070.

Osborne, J. G. (1942), 'Sampling errors of systematic and random surveys of cover-type areas', *Journal of the American Statistical Association* **37**(218), 256–264.

Parra, G. & Tobar, F. (2017), 'Spectral mixture kernels for multi-output gaussian processes.', *31st Conference on Neural Information Processing Systems. Long Beach, CA, USA* pp. 1–10.

Peñagaricano Muñoa, O. & Scheeres, D. J. (2010), 'A Perturbation Theory', *Acta Astronautica* **67**(2), 27–37.

Petit, T. & Henon, M. (1986), 'Satellite encounters', *Icarus* **555**, 536–555.

Petrosky, T. & Broucke, R. (1987), 'Area-preserving mappings and deterministic chaos for nearly parabolic motions', *Celestial Mechanics and Dynamical Astronomy* **42**(1), 53–79.

Poincaré, H. (1899), ' Les méthodes nouvelles de la mécanique céleste', *Gauthier-Villars* **3**.

Rasmussen, C. & Williams, C. (2006), 'Gaussian processes for machine learning', *Massachusetts Institute of Technology Press,* pp. 7–31, 79–128.

Ross, S. & Scheeres, D. (2007), 'Multiple gravity assists, capture, and escape in the restricted three-body problem', *IAM Journal on Applied Dynamical Systems* **6**(3), 576–596.

Sánchez, J. P., Alessi, E. M., Yárnoz, G. D. & McInnes, C. (2013), 'Earth resonant gravity assists for asteroid retrieval missions', *64th International Astronautical Congress* pp. IAC–13.

Sánchez, J. P., Colombo, C. & Alessi, E. (2015), 'Semi-analytical perturbative approaches to third body resonant trajectories', *Proceedings of the 66th International Astronautical Congress, International Astronautical Federation (IAC), Jerusalem, Israel* pp. IAC—15.

Shang, H. & Liu, Y. (2017), 'Assessing Accessibility of Main-Belt Asteroids Based on Gaussian Process Regression', *Journal of Guidance, Control, and Dynamics* **40**(5), 1144–1154.

Shevchenko, I. (2011), 'The Kepler map in the three-body problem', *New Astronomy* **16**(2), 94–99.

Siddiqi, A. A. (2018), 'Beyond earth: A chronicle of deep space exploration, 1958-2016', *National Aeronautics and Space Administration, Washington DC* pp. 5–6.

Strange, N. & Longuski, J. (2002), 'Graphical method for gravity-assist trajectory design', *Journal of Spacecraft and Rockets* **39**(1), 9–16.

Sun, S., Zhang, G., Wang, C., Zeng, W., Li, J. & Grosse, R. (2018), ' Differentiable Compositional Kernel Learning for Gaussian Processes', *Proceedings of the 35th International Conference on Machine Learning* **80**, 4828–4837.

Szebehely, V. (1967), 'Theory of Orbits', *Academic Press, New York* .

The Mathworks Inc. (2018), 'MATLAB version 9.5.0.944444'.

Tisserand, F. (1891), 'Traité de mécanique céleste', *Gauthier-Villars* .

Tresp, V. (2001), 'Mixtures of gaussian processes.', *Advances in neural information processing systems* pp. 654–660.

van der Weg, W. & Vasile, M. (2014), 'Sun–earth l1 and l2 to moon transfers exploiting natural dynamics.', *Celestial Mechanics and Dynamical Astronomy* **140**, 287–308.

Vasile, M. & De Pascale, P. (2006), 'Preliminary design of multiple gravity-assist trajectories.', *Journal of Spacecraft and Rockets* **43**(4), 794–805.

Villac, B. & Scheeres, D. (2003), 'Escaping trajectories in the Hill three body problem and application', *Journal of Guidance Control and Dynamics* **26**(2), 224–232.

Villac, B. & Scheeres, D. (2004), 'On the concept of periapsis in Hill's problem', *Celestial Mechanics and Dynamical Astronomy* **90**, 165–178.

Whiffen, G. J. (2003), 'An investigation of a Jupiter Galilean moon orbiter trajectory', *AAS/AIAA Astrodynamics Specialist Conference, Big Sky, MN* pp. 03–544.

Zhou, J. L., Sun, Y. S., Zheng, J. Q. & Valtonen, M. J. (2000), 'The transfer of comets from near-parabolic to short-period orbits: map approach', *Astronomy and Astrophysics* **364**, 887–893.

ACKNOWLEDGEMENTS

It was one of my crucial choices to pursue a PhD at Delft University of Technology, though I was doubting about it a few times. I would like to owe the completion of this dissertation to everyone who supported and supports me. Everyone who is reading the words here has made my life beautiful.

I am eternally grateful for the support from my promotor, Prof. Pieter Visser. Pieter offered me this valuable opportunity to work for the department Astrodynamics and Space Missions. It was a treasurable experience to work with the colleagues in the department and learn about many topics. I can imagine the amount of workload for a professor, but Pieter always valued my progress and did his best to give useful feedback within one week. The support on the conferences and summer school is much appreciated. Pieter has made a very positive influence on my research by his integrity, humility and considerateness.

I would like to express my appreciation to my daily supervisor Ron Noomen. His support on my topic is obvious. The suggestions and feedback include not only the overall idea but also grammar and spelling. His carefulness polished our work from scratch to golden deliveries. His encouragement was consistent no matter what kind of obstacles I was facing. His words contained warm and positive power, which is most precious in modern society. As a married man (one year before me), his view of life and marriage motivates me.

Our department offered diverse and warm team. Every colleague played an important role in watering the giant tree, and offering shade for students. Thanks Relly for your assistance on my work and the social events you have organized. The PhD life in the Netherlands would be messy without your help. Thanks Kevin for co-supervising master students with me. I learnt from you about teaching, critical thinking and Dutch-American jokes. Thanks Dominic for instructing me to use TUDAT, which made an essential part of my research. Thank you, Bart, Bernhard, Bert, Christian, Daphne, Eelco, Elisabetta, Ernst, Erwin, Francesco, Imke, Jeannette, José, Leonid, Marc, Sebastiaan, Stéphanie, Wim, Wouter, Vidhya, Yanli. I appreciate these colleagues' insight on the research. The talks with you inspired me on my topic and life. Special thanks goes to the colleagues on the 8th floor, Gill, Jian Guo, Debby, Mariëlle, Angelo and Prem.

We have so many PhDs and postdocs from different countries and cultures. The time spent with you makes an unforgettable memory of my life. Haiyang, Svenja, Tim, Günther, Jacco, Teresa, Bas, Marc, Jesse, Hermes, Tatiana, Dora, Marie, Löic, Marcel, Minghe, Zixuan, Linyu, Marsil, Daduí, Johan, Daniel, Fiona, Mario, Victor, Ruipeng. Thank you for every enjoyable moment with you. Spetial thanks to Jesse for translating the summary of thesis from English to Dutch. Thanks to my office mates, who shared ideas, coffee and candies. I appreciate Xinyuan about his worldview, and the knowledge on the craft beer. Thanks Gourav for making me believe the relationship between the Chinese and Indian people could be so friendly. Thank you for the best curry I have ever had.

Thanks Kartik for sharing your idea on commercial space, which opens a new world of capital markets for me. Joining the PhD board of the AE faculty gave me the opportunity to learn the educational and administrative structure of Dutch universities. Thanks Dirk, Bram, Bruce, Tiago, Ines, Vis, Stefano, Nancy for organizing scientific and social events together.

I am extremely lucky to have made many friends in Delft. Even though we are pursuing different careers all over the world, I do believe we would meet sometime and grab a beer as we used to do. The crazy guys from 'white building' makes me aware of the power of laugh. Thank you, Yan Teng, Jing Wan, Yan Song, Anton, Rui Li, Kaiyi Zhu, Sitong Luo. Zhenwu Wang, I appreciate your accompany of doing exercise, by which you lost a lot of weight, you are welcome. Zongchen Li and Bing Huang, thank you for every talk we had and the support from each other. Thanks Jian Zhang for the dinner and BBQ, moments of doing exercise and taking courses. Thank you, Jin Chang and Yun Wan, another couple of North Chinese+South Chinese shows the success of this kind of combination. Special thanks to the 'five guys' group: Brother Li, you are such a warm guy; Mingxue Zheng, wish you, your husband and your son a wonderful life; Hongjuan Wu, wish you an academic success; Pan Zhang, wish you a girlfriend/wife who shares the same view of life. Thanks to my friends, Ling Jia, Yunlong Guo, Haopeng Wang, Yaya Liu, Yujie Zhao, Na Chen, Yande Jiang.

Special thanks to the ACSSNL team. Qingyu Meng, Ping Luo, Xiaoxiao Cheng, Wei Bai, Tao Sun, my sincere appreciation for your support to us. Ding Ding, thank you for inviting me to the team and sharing your thoughtful ideas and providing your comprehensive suggestions. Jiakun Gong, Han Bao, Yi Yu, Zheng Wu, Yu Chen, Miaozhen Huang, Chenglong Deng, Jing Wang, Qiyao Hu, Geng Chen, Aiyi Li, Yingying Geng, Quanxing Wan, Qizhi Ren, Tianrun Yang, Yifan Chen, I enjoyed organizing every event with you. The members in my small team, Bin Liang, Shaoya Ren and Lingyu Lu, thank you for your assistance to make remarkable things done.

It is not possible for me to succeed the PhD without the support of my family. The phone call at weekends is like a string connecting a kite and the land. My parents would always deliver good news and positive stuff to me. I am glad to inherit their optimistic and integral personality. My grandma does not talk too much but I know how much she loves me. I can feel my grandpa standing behind me and supporting me. I am a single child, but my uncles, aunts, and cousins make a warm and supporting family for me. I would like to thank my parents in law and the other members of my wife. Your trust and hospitality turns Hunan province into my second hometown.

I would like to give special thanks to my wife, Jin Liu. My wife has a nick name for me, Xingxing, which is interpreted as little star in Chinese. But I do believe that you are the brightest star in my life, shedding light upon my path. The full support from you contributes to achieving my goals. I would love to build life with you, and hold your hands till the end of my life.

Yuxin Liu
June, 2021
Delft, The Netherlands

CURRICULUM VITAE

Yuxin LIU

25-04-1990 Born in Harbin, China

EDUCATION

2009–2013 Beijing Institute of Technology
Bachelor in Engineering
Detection, Guidance and Control Technology

2013–2016 Beijing Institute of Technology
Master in Engineering
Aeronautical and Astronautical Science Technology

2016–2020 Delft University of Technology - Ph.D student
Astrodynamics and Space Missions

Thesis: Quantification of flyby effects in the three-body problem using the Gaussian process method

Promotor: Prof. dr. ir. P.N.A.M. Visser

Supervisor: ir. R. Noomen

LIST OF PUBLICATIONS

Journal Articles

1. Y. Liu, R. Noomen, P. Visser, *A Gravity Assist Mapping Based on Gaussian Process Regression*, The Journal of the Astronautical Sciences (2021), Vol.68:248-272.

2. Y. Liu, R. Noomen, P. Visser, *A Gaussian Process Regression Based Gravity Assist Mapping for the Circular Restricted Three-Body Problem*, Advances in Space Research (2021), Vol.68(6):2488-2500..

3. Y. Liu, R. Noomen, P. Visser, *A Machine-Learning Design Tool for Quantifying the Gravity Assist Effects in the Circular Restricted Three-Body Problem*, to be submitted to Advances in Space Research.

4. H. Shang, Y. Liu, *Evaluating Accessibility of Main-belt Asteroids Using Gaussian Process Regression*, Journal of Guidance, Control, and Dynamics (2017), Vol. 40 (5): 1144-1154.

Conference Papers

1. L. Bouwman, Y. Liu, K. Cowan, *Gaussian Process models for preliminary low-thrust trajectory optimization*, Astrodynamics Specialist Conference, Portland, ME ,171(873), pp. 3687-3706, 2019.

2. P. Pérez, Y. Liu, K. Cowan, *Global optimization of low-thrust interplanetary trajectories using a machine learning surrogate*, Astrodynamics Specialist Conference, Lake Tahoe, 20-663, 2020.