UNIVERSITY OF SOUTHAMPTON

Faculty of Engineering and Physical Sciences
School of Engineering

Pushing the Envelope of Exoplanet Evolution Modelling

by

Peter Bartram

MSc, BEng

ORCiD: 0000-0002-4062-1165

A thesis for the degree of
Doctor of Philosophy

August 2021

Abstract

Pushing the Envelope of Exoplanet Evolution Modelling

by Peter Bartram

Propelled by the discovery of the first exoplanet thirty years ago, the scientific community has rallied and made tremendous strides towards a full understand of the formation and evolution of planetary systems. During this period, over 4,300 confirmed exoplanets have been detected, and the resulting dataset has driven a revolution by allowing for new formation theories to be proposed and tested. Despite these advances, there is still much about these processes that remains unknown. Numerical n-body simulations of planetary systems are now commonly used to push these frontiers. When performing these investigations, the numerical integration process still poses a very specific set of challenges and therefore demands the continued development of state-of-the-art tools.

There are three key novel components to this thesis. Firstly, I perform a detailed analysis of numerical integrators built around multistep collocation methods to quantify their numerical performance over a wide subset of their possible configuration space. Highly favourable performance is observed when specific configurations are applied to globally stiff problems.

Secondly, I present my new tool, the Terrestrial Exoplanet Simulator (TES), a novel n-body integration code for the accurate and rapid propagation of planetary systems in the presence of close encounters. TES builds upon the classic Encke method and integrates only the perturbations to Keplerian trajectories to reduce both the error and runtime of simulations. A suite of numerical improvements is presented that together make TES optimal in terms of growth of energy error. Lower runtimes are found in the majority of test problems when compared to direct integration using other leading tools.

Finally, using TES, I perform a large simulation campaign to further understand the stability of compact three-planet systems. This work addresses a key limitation in the majority of stability studies by using TES to integrate precisely up to the first collision of planets. Integrations span up to a billion orbits to explore a wide parameter space of initial conditions in both the co-planar and inclined cases. I calculate the probability of collision over time and determine the probability of collision between specific pairs of planets. I find systems that persist for over 10^8 orbits after an orbital crossing and show how the post-crossing survival time of systems depends upon the initial orbital separation, mutual inclination, planetary radius, and the closest encounter.

Contents

List of Figures

List of Tables

Declaration of Authorship

I declare that this thesis and the work presented in it is my own and has been generated by me as the result of my own original research.

I confirm that:

1. This work was done wholly or mainly while in candidature for a research degree at this University;
2. Where any part of this thesis has previously been submitted for a degree or any other qualification at this University or any other institution, this has been clearly stated;
3. Where I have consulted the published work of others, this is always clearly attributed;
4. Where I have quoted from the work of others, the source is always given. With the exception of such quotations, this thesis is entirely my own work;
5. I have acknowledged all main sources of help;
6. Where the thesis is based on work done by myself jointly with others, I have made clear exactly what was done by others and what I have contributed myself;
7. Parts of this work have been published as:

   Peter Bartram and Alexander Wittig. Terrestrial Exoplanet Simulator (TES): an error optimal planetary systems integrator that permits close encounters. *Monthly Notices of the Royal Astronomical Society*, 504(1):678–691, 2021.

   Peter Bartram, Alexander Wittig, Jack J Lissauer, Sacha Gavino, and Hodei Urrutxua. Orbital stability of compact three-planet systems, II : Post-instability impact behaviour. *In print, Monthly Notices of the Royal Astronomical Society*, 2021

Signed: Date:

Acknowledgements

Above all others, I would like to thank my supervisors, Alex Wittig and Hodei Urrutxua, for their teachings and support over the past four years. I feel very lucky to have met and had the chance to work with both of them, and I only hope that I can pass along the belief, guidance, and patience that they have offered me.

I would also like to thank Jack Lissauer and Sacha Gavino for the opportunity of our collaboration, as I learnt a huge amount from our conversations.

I owe a great deal to the Next Generation Computational Modelling (NGCM) Centre for Doctoral Training (CDT) for their excellent syllabus, continued professional development, and general activities. Naturally, this thanks also extends to the Engineering and Physical Sciences Research Council (EPSRC) for funding the NGCM (EP/L015382/1).

Last but certainly not least, I would like to thank all of my fellow students and office mates for the shared experiences and camaraderie over the years.

To my family.

Chapter 1

Fundamentals of planetary formation, evolution, and simulation

This initial chapter is a literature review detailing the current state-of-the-art in several subfields of exoplanet science and modelling. Firstly, it serves as an introduction to the processes by which planets form and evolve. This literature review spans many decades from the times when only analytical estimates of formation processes were available, right through to results obtained with more modern numerical simulations. The focus of this thesis, and therefore also this chapter, is on the formation of rocky planets similar to our own, but a brief discussion of the formation and evolution of gas giants such as Jupiter is included for completeness. Secondly, this chapter also provides an introduction to the challenges involved in modelling the formation and long-term evolution of rocky planetary systems. Finally, an overview of some of the leading modelling tools in the field are introduced.

1.1 Motivation and methods in the study of planet formation

There are few questions captivate the curious mind as much as that of how did we come to live on this pale blue dot? A full understanding of the origins and architecture of even our own solar system remains elusive even to this day, but an full understanding of how planetary systems come to be has wide reaching implications. If, as we now know, exoplanets are common in the universe, then what system architecture

has allowed for life to flourish here on Earth, and are there other similar planetary systems that could potentially also harbour life?

Immanuel Kant was well known for his writings about the possibility of extraterrestrial life. He was also responsible for one of the earliest published scientific works, based upon the observations of Thomas Wright, postulating that stars and planetary systems form from collapsed nebula (Kant, 1755), a proposition now known as the nebula hypothesis. Laplace (1835) later devoted a small section of one of his works to the nebula hypothesis; in it, he argued that over time nebulae collapse to form stars, and that a disk of material left around that star will eventually separate into rings, finally leading to the formation of planets. Laplace's predecessors, such as Newton, had observed the curious properties of our solar system, such as the prograde orbital nature of all of the planets or the fact that all planets orbit on almost the same plane, and arrived at a requirement for an omnipotent creator. When presented with the same observations and Kant's work on the nebula hypothesis, Laplace concluded that there was no need for this creator, at least for this period in cosmic history, and that the peculiarities could be explained through the prehistory of the solar system. This was the first notable consideration of the implications of the prehistory of planetary systems, a central topic in this thesis. As will be seen in the remainder of this chapter, piecing together of this prehistory is not straight-forward. However, there are many features that can be observed that place practical constraints on certain aspects of the formation process, and when these are combined it becomes possible to build up a more accurate time line. Progress towards the formation process was slow for many years after Laplace, but by the time that Safronov (1972) published his treatise on terrestrial planet formation, the theories were at least qualitatively developed for the inner planets. As computing power was still in its infancy at this time, his works were all analytical and, as such, made wide reaching assumptions. It is only with modern increases in computational power that high precision numerical simulation of the formation process has become possible, providing unparalleled glimpses of nature in the prehistory of our solar system and exoplanet systems. The ability of numerical simulations to test formation theory has led to otherwise unlikely revelations, such as a possible explanation for the low mass of Mars (Walsh et al., 2012). Numerical simulations are required, in part, because of the impracticalities, both temporal and technological, in observing the formation of terrestrial planets. An exception to this rule that can likely offer a rare glimpse into the formation process are the so-called "extreme debris disks". These are young protoplanetary disks that emit strongly in the infrared spectrum due to the stellar irradiation of dust created in giant impacts between bodies (Watt et al., 2021; Su et al., 2019).

Progress on planetary formation until the late 20^{th} century was throttled due to the technical limitations of astronomy. Until the early 1990s we were only aware of nine planets, Pluto not yet having lost this accolade. While it was always considered a possibility that there would be planetary systems similar to our own orbiting other stars, the technology was not there to detect them. Without these observations, all of our formation theories had to be tested against the architecture of our solar system, therefore limiting the confidence in our models. In 1992, Polish and Canadian astronomers Wolszczan and Frail (1992) published their seminal work confirming the detection of a planetary system around the pulsar PSR1257+12. Three years later, Mayor and Queloz (1995) published their detection of a Jupiter-mass planet orbiting a solar-type star. These discoveries ushered in the age of exoplanet observation and as a result Mayor and Queloz won the 2019 Nobel Prize in Physics. Fast forward two and a half decades from those first discoveries and the number of confirmed exoplanets detected now stands at 4383, largely due to the efforts of the Kepler space observatory (NASA, 2018; Petigura et al., 2013) launched in 2009. Figure 1.1 shows the distribution of observed exoplanet masses against their semi-major axis. This data clearly shows the presence of a large number of Jupiter-like planets but also the existence of so-called "super Earths". Where the term super Earth is used to refer to a planet that is primarily rocky, with a mass larger than that of Earth by up to a maximum of roughly an order of magnitude.

The Kepler space telescope is now retired, but its success and that of other similar missions has renewed interest in the study of exoplanets and planet formation in general. Enthusiasm is clear worldwide, e.g. both ESA and NASA have a new generation of "planet searchers" either in orbit or under development. Despite the existence of more exotic techniques, e.g. gravitational microlensing, the majority of exoplanet data comes from two observation techniques: the transit method and radial velocity search. The transit method observes the incoming light from a star and looks for the characteristic dip in the light flux indicating the presence of a planet between the observer and the targeted star. Through these measurements, the radius of the exoplanet can readily be determined. Additionally, by making repeated observations of the same planet over multiple orbits, the orbital period can be ascertained. Moreover, if a mass estimate of the host star is known, then this also allows for the semi-major axis of the planet's orbit to be estimated. The transit method, however, offers no knowledge of the mass of an observed planet and instead a radial velocity search must be used. Radial velocity searches measure the Doppler shift in the emitted light of a host star caused by the star and exoplanet orbiting a common barycentre. Through this measurement it becomes possible to provide a lower limit to the mass of the planet

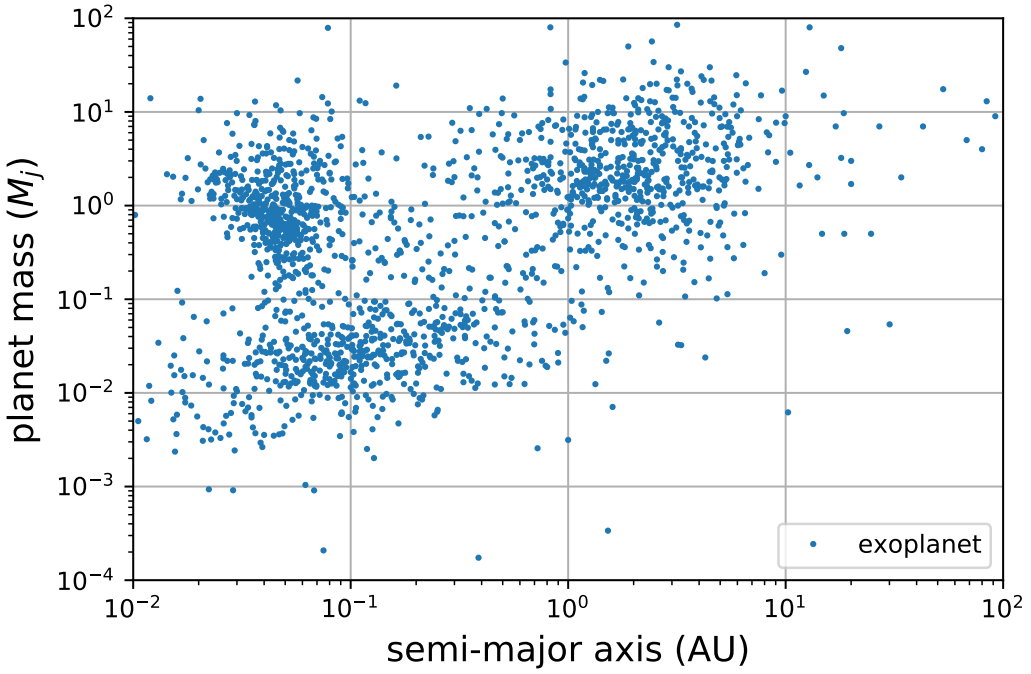


Figure 1.1: The distribution of observed exoplanet masses, in units of Jupiter mass (M_J), against their semi-major axis in astronomical units.

observed. A combination of observations with the transit method and a radial velocity search is particularly powerful as it allows for both the mass and radius of the planets to be combined to derive a bulk density, which then offers clues as to the composition of the planet.

The prominent exoplanet hunting satellite mission presently is the Transiting Exoplanet Survey Satellite (TESS) developed by NASA. TESS is a transiting observatory and builds upon the original work of the Kepler mission. It is currently surveying 200,000 star systems every two minutes searching for changes in their light curves. It is expected that TESS will find roughly 15,000 exoplanets over its mission duration (Barclay et al., 2018), roughly a four-fold increase to the currently known number. On the other side of the Atlantic Ocean, the European Space Agency launched the Characterising Exoplanets Satellite (CHEOPS) mission in 2020. CHEOPS is also a transiting observatory but with a different overall aim to the TESS mission. Instead of detecting a large number of new exoplanets, CHEOPS is designed to provide precise follow-up measurements of systems already detected via a radial velocity search. In this sense, the TESS and CHEOPS missions synergize well with one another as well as both complementing ground-based observatories.

Looking to the future, there are four major space-based observatories that are planned to launch between 2021 and 2029; namely, the James Webb Space Telescope (JWST),

Nancy Grace Roman Space Telescope, Planetary Transits and Oscillations of Stars (PLATO), and the Atmospheric Remote-Sensing Infrared Exoplanet Large-Survey (ARIEL). Each of these missions will further improve the exoplanet database. This richer dataset will allow for a more precise comparison with the outputs of simulations, thereby allowing for models to be validated in ways that have never been possible before. This is incredibly exciting because, as we will see, the current exoplanet database has already revolutionised our understanding of planetary formation. Additionally, it has created a unique opportunity at this time for this thesis to contribute to the understanding of exoplanet systems through the development and application of novel numerical techniques to the formation and evolution process.

1.2 The phases of planetary formation

The initial conditions for planet formation are a freshly born star and a surrounding accretion disk (Meyer et al., 2006). An accretion disk forms around a star because the gas and metal making up the proto-stellar material has too much angular momentum for it to fall directly onto the surface of the star and be captured. In order for the material to be accreted onto the star, its angular momentum must therefore be dissipated. Fortuitously, this is a very slow process and it therefore becomes possible for planets to form within the disk, thereby leading to quasi-stable systems capable of supporting life.

The term “planet” can occasionally be a source of controversy among astronomers, e.g. what designation should be assigned to Pluto? In this work, the term planet refers to a large body orbiting at least one star. Additionally, this object must be massive enough that it collapses to a spheroid under its own gravity and clears its orbital neighbourhood of any other large bodies. Any major object less massive than this instead receives the designation of a dwarf planet. Finally, an upper mass bound is needed to delineate planets from binary stars. Therefore, planets cannot obtain any substantial fraction of their luminosity from nuclear fusion, thereby fixing the upper limit to their mass at the deuterium burning threshold, approximately 2.4687×10^{28} kg for solar composition objects (Armitage, 2009).

In the planetary formation process, there are three overlapping phases of evolution that are still broad enough to warrant distinction from one another, these are:

1. The evolution of the protoplanetary disk.

2. The formation and evolution of rocky worlds.

3. The formation and evolution of gas giants.

The evolution of the protoplanetary disk dictates the initial conditions in both the terrestrial and gas giant planet formation processes. Furthermore, the evolution of gas giants at least elicits a gravitational effect on the terrestrial formation process, and it is likely that, as we shall see, planetary migration means that gas giants can remove material from the terrestrial formation region, thereby having an even more profound effect on the formation of rocky worlds. All this is to say, despite my choice of categorisation, each of these processes occurs in the wider context of all others. A second reason for these categories is that each requires different modelling techniques to be able to understand their evolution. For example, the study of protoplanetary disks will oftentimes require the use of a radiative transfer model (Akimkin et al., 2013) to understand the heating of the gas in the disk. In contrast, the study of terrestrial planet formation is generally accessible to n-body methods alone (Ida and Makino, 1993; Kokubo and Ida, 1996, 1998; Kokubo, 2000; Ohtsuki et al., 2002; Kokubo and Ida, 2002; Leinhardt and Richardson, 2005; Kokubo et al., 2006; Bartram et al., 2021) as the majority of the evolutionary process takes place after the gas present in the disk has dissipated. Finally, studying gas giants will often necessitate the use of hydrodynamic simulations to understand the temperatures, pressures and flows of gas surrounding a planetary core (Machida et al., 2010; Szulágyi et al., 2016). The work in this thesis makes use exclusively of n-body methods as the work is focused on understanding the evolution of planets within the terrestrial formation region. The upcoming discussion on planetary formation therefore also focuses on the terrestrial formation and evolution processes.

1.3 The protoplanetary disk

As discussed, modelling of protoplanetary, or circumstellar, disks is beyond the scope of this thesis. However, some relevant aspects of disks are discussed here, such as composition and density, so that the rest of the formation process can be discussed in context.

1.3.1 Planetesimal formation

The term planetesimal is used to describe a body in a protoplanetary disk that is sufficiently massive to gravitationally attract other planetesimals in a disk yet still small enough that it does not have a dominant influence on the overall dynamics of the disk. An order of magnitude lower bound for the size of a planetesimal is 1 km. Generally, the upper limit to what is considered a planetesimal is approximately 1000 km, where, as will be seen later, the bodies are then referred to as oligarchs.

Protoplanetary disks initially evolve through angular momentum transport mechanisms allowing the accretion of material onto the star. Eventually, these mechanisms cause the opacity of the disk to drop sufficiently that heating via UV radiation from the star creates a high enough pressure difference to excite the remaining gas out of the system. During this period, planetesimals must form such that other formation processes can then begin. The exact process by which planetesimals initially grow to be greater than one meter in diameter is a long standing mystery with formation theories (Blum, 2018) known as the meter size barrier. Consequently, the formation of the first planetesimals is the subject of ongoing research (Morbidelli and Raymond, 2016), with Grishin et al. (2019) even suggesting that they could be captured from interstellar space.

1.3.2 The snowline

A key feature of any protoplanetary disk is known as the "snowline". The snowline is crucial to our understanding of planetary formation as it delineates the terrestrial, i.e. rocky, planet formation region from the gas and ice giant formation region.

In a circumstellar disk, the pressure is such that the critical temperature for water to exist as a vapour is approximately 120 K. In turn, this implies two distinct regions in the disk: one where ice is available for accretion into planets and one where it is not; the radius at which this transition occurs is termed the snowline. In our own solar system, the snowline is thought to have been located at approximately 2.7 AU[1] but the location depends on, e.g. the energy output of the star, and it is therefore variable for other systems. Interestingly, the location of the snowline in our solar system implies that water was not available for accretion when the Earth initially formed, meaning that it must have formed dry with the oceans being obtained later on through collisions with other wet bodies (Morbidelli et al., 2000).

[1] I have chosen to use capital letters for the astronomical unit symbol throughout this work.

1.3.3 The minimum mass solar nebula (MMSN)

To determine the mechanisms of formation it is necessary to infer the mass of the material present to begin with. It is only possible to calculate a lower bound for this value, which is termed the minimum mass solar nebula (Weidenschilling, 1977). Hayashi (1981) provided a commonly quoted value for the surface density profile of the material in a disk with respect to the orbital radius, r, such that

$$\Sigma_{gas}(r) = \quad 1.7 \times 10^3 \, r^{-3/2} \ \text{g cm}^{-2},$$
$$\Sigma_{solid}(r) = \begin{cases} 7.1 \, r^{-3/2} \ \text{g cm}^{-2} & r < 2.7 \text{ AU}, \\ 30 \, r^{-3/2} \ \text{g cm}^{-2} & r > 2.7 \text{ AU}, \end{cases}$$

where Σ_{gas} and Σ_{solid} are the surface densities of the gas and solid material within the disk. These estimates are based upon observations of our solar system. These estimates show that the disk is primarily composed of gas with very little solid material present at all by comparison. The effects of the snowline can also be seen, with a large increase in solid material present for accretion being estimated beyond 2.7 AU due to the presence of ice. Kokubo and Ida (2002) found that numerical n-body simulations that use these values of initial disk density can indeed form planetary systems similar to those observed, with a leading surface density coefficient for solid material within 2.7 AU of $10 \, \text{g cm}^{-2}$ generating the closest analogues of our solar system.

1.3.4 Processes affecting planetesimal velocity dispersion

Planets form slowly over time through the accretion of planetesimals into oligarchs into protoplanets into planets. A prerequisite for accretion is the possibility of collisions between planetesimals. In a dynamically cold disk composed of low mass bodies, the chance of collisions is lower due to the circular nature of their orbits and the weak gravitational interactions between them. Therefore, for any growth stage beyond planetesimal formation to begin, it is necessary for dynamic heating of the disk to occur. There are four processes that have the effect of either exciting or dampening the planetesimal velocities, and these therefore control the likelihood of impact and accretion. These processes are:

1. Viscous stirring.

2. Dynamical friction.

3. Gas drag.

4. Inelastic collisions.

In a protoplanetary disk composed of equal low mass bodies, as is expected in the early stages of evolution, viscous stirring is the only excitation process present. Viscous stirring is the cumulative result of a series of weak gravitational encounters between planetesimals that will convert orbital energy into velocity dispersion in the form of increased eccentricities and inclinations, henceforth termed random velocities. It was shown by Ohtsuki et al. (2002) that the effects of viscous stirring work to increase the mean eccentricity, $\langle e \rangle$, and mean inclination, $\langle i \rangle$, with a ratio of $\langle e \rangle \approx 2 \langle i \rangle$, in a time period that means it will effectively cause dynamical heating of the disk before the predicted formation of any large bodies. An order of magnitude analytical estimate (Armitage, 2009, p. 169) of the timescale for heating to plateau is 6×10^3 yr.

Dynamic friction is caused by nature's tendency to equipartition kinetic energy evenly amongst bodies through gravitational scattering. When a disk is composed of a bi-modal distribution of masses m and M, with associated random velocities of σ_m and σ_M this partitioning of kinetic energy takes the form

$$\frac{1}{2} m \sigma_m^2 \approx \frac{1}{2} M \sigma_M^2.$$

This means that the more massive particles, M, will be dampened to lower random velocities, whereas less massive particles will be excited to higher random velocities. Ohtsuki et al. (2002) show that although both mass distributions will be heated by viscous stirring, the random velocities of the more massive particles will systematically remain below those of the other distribution. Therefore, a mass dependency has been introduced to the random velocities. It is common in simulations to group planetesimals together into a larger mass to save on computation; however, this mass dependency means that to study the early stages of disk evolution, it is favourable to maximise the particle number to more closely represent the initial distribution of masses.

The presence of gas in the early protoplanetary disk means that planetesimals will experience the effects of drag as they orbit. Due to the planetesimal size, this drag will be in the Stokes regime and can be readily estimated for a given disk density and population of planetesimals. Armitage (2009, p. 170) provides an analytical estimate that shows the timescale over which drag will affect planetesimal random velocities

is 1×10^6 years, thereby making it appear to be a negligible contribution to their velocity dispersion. However, the presence of gas will work to keep particles smaller than planetesimals on circular coplanar orbits, which through the equipartitioning of energy due to dynamic friction will cause a dampening effect on the planetesimals themselves. This does not, however, seem to change the final mass distribution of protoplanets attained through simulation when compared to gas free models (Kokubo, 2000).

Inelastic collisions between particles will cause dissipation of random velocities. However, the point mass treatment used in n-body simulations does not allow for these effects to be considered. In this work, simulations are strictly terminated upon first impact of planets thereby circumventing this potential problem. Elsewhere, however, n-body simulations incorporating a fragmentation model have been applied to understand the subsequent fragmentation of bodies as a result of inelastic collisions (Leinhardt and Richardson, 2005).

1.4 Terrestrial planet formation

Understanding the formation and evolution of terrestrial planets is especially exciting as it works towards explaining the history of our home planet. From an astronomical perspective, there are three key phases to terrestrial planet formation, which are:

1. Run-away growth.
2. Oligarchic growth.
3. Final assembly.

1.4.1 Run-away growth

Run-away growth is the first regime in time to be entered. It is characterised by the absence of any individual bodies that are massive enough to dominate the dynamics of the disk. All planetesimals experience a rapid increase in mass during this phase; however, some grow more rapidly than others and become oligarchs that then dominate the dynamics of the disk. The mass growth rate of a planetesimal M with radius

R accreting field planetesimals of mass m is given by (Raymond and Cossou, 2014)

$$\frac{\mathrm{d}M}{\mathrm{d}t} \approx n_m \pi R^2 \left(1 + \frac{V_{esc}^2}{V_{rel}^2}\right) V_{rel}\, m \tag{1.1}$$

where n_m is the number density of the smaller field planetesimals to be accreted, V_{esc} is the escape velocity from the surface of the primary mass M and $V_{rel} = \sqrt{V_M^2 + V_m^2}$. The left hand term within the parenthesis is the change in mass caused by the physical cross sectional area of the planetesimal and the right hand term is the increased cross sectional area due to gravitational focusing (Kokubo and Ida, 1996). Gravitational focusing is effective during this stage of growth as the velocity dispersion of field planetesimals is kept low due to the effects of gas drag within the disc, meaning that $V_{rel} < V_{esc}$. Under these conditions Eq. (1.1) reduces to

$$\frac{1}{M}\frac{\mathrm{d}M}{\mathrm{d}t} \propto \Sigma_{solid} M^{1/3} v^{-2}, \tag{1.2}$$

where Σ_{solid} and v are the surface density and velocity dispersion of the field planetesimals, respectively. Finally, it can be assumed that in the early stages of planetary formation, the growth of large planetesimals has little effect on the properties of the disk, meaning that Σ_{solid} and v can both be assumed to be constant, resulting in the final rate of change of planetesimal mass

$$\frac{1}{M}\frac{\mathrm{d}M}{\mathrm{d}t} \propto M^{1/3}.$$

Thus, it can be seen that the growth rate of planetesimals has a positive mass dependence. Given this result and two objects with masses M_1 and M_2 the rate of change of the ratio of their masses is

$$\frac{\mathrm{d}}{\mathrm{d}t}\frac{M_1}{M_2} = \frac{M_1}{M_2}\left(\frac{1}{M_1}\frac{\mathrm{d}M_1}{\mathrm{d}t} - \frac{1}{M_2}\frac{\mathrm{d}M_2}{\mathrm{d}t}\right) \propto \frac{M_1}{M_2}\left(M_1^{1/3} - M_2^{1/3}\right).$$

This positive mass dependence therefore causes more massive planetesimals to grow even more rapidly in comparison to their less massive contemporaries leading to the emergence of a small number of oligarchs. Run-away growth is an inherently self-limiting process: as planetesimals become more massive they remove mass from the background population of field planetesimals and therefore reduce the surface density of the disk in Eq. (1.2) (Kokubo and Ida, 1998). Run-away growth has been observed in a number of seminal n-body simulations where both gaseous and gas-free systems

have been considered (Kokubo and Ida, 1996; Kokubo, 2000). These experiments found that run-away growth is clearly visible in timescales as short as 2×10^5 years.

1.4.2 Oligarchic growth

The second formation stage in time is the oligarchic growth phase. Oligarchs are bodies that are massive enough to dominate the dynamics of the disk as a whole. Oligarchs do not have a precise definition, and the size and mass cut-off technically depends upon the mass of a specific disk. However, for simplicity, the term oligarch generally is taken to refer to bodies larger than 1000 km in diameter that are smaller than roughly the size of the Moon, i.e. approximately 3000 km in diameter. Bodies larger than this are referred to as protoplanets. Oligarchic growth begins once the disk surface density, Σ_{solid}, has been sufficiently depleted by the run-away growth phase (Kokubo and Ida, 1998). Additionally, Ida and Makino (1993) showed that the run-away growth phase increases the velocity dispersion, v, in Eq. (1.2) such that at the start of the oligarchic growth phase $v \propto M^{(1/3)}$ thereby resulting in a mass growth rate of oligarchs of

$$\frac{1}{M}\frac{\mathrm{d}M}{\mathrm{d}t} \propto M^{-1/3}.$$

This negative mass dependence causes the switch to orderly growth and from then on neighbouring oligarchs grow at roughly the same rate. During this growth phase, orbital repulsion maintains a nearly constant separation between oligarchs. In short, if two oligarchs approach each other too closely, then they will become excited to a higher eccentricity and inclination, and over time dynamical friction will then re-circularise their orbits at a higher semi-major axis, thereby maintaining the separation distance between them (Kokubo and Ida, 1995). Oligarchic growth has also been observed in n-body simulations (Kokubo and Ida, 1998; Kokubo, 2000; Kokubo and Ida, 2002) where integrations of 4000 bodies over a period of 5×10^5 years did indeed find the slow down of runaway growth resulting in two large almost equal mass protoplanets containing 41% of the initial mass of the disk. The separation distance of the two protoplanets was approximately 5 Hill radii, where 1 Hill radii is the distance from a planet at which its gravitational effects will dominate those of the star, although during the process oligarchs did collide if multiple close approaches were made before their respective orbits could be re-circularised by the dynamical friction.

As the disk field planetesimal population becomes further depleted and oligarchs increase in mass, they will eventually have consumed the majority of the material within their orbital Hill sphere. At this point, the oligarchic growth phase is concluded and

the disk transitions into the final assembly phase. Additionally, the mass at which this occurs is known as the isolation mass. Analytical estimates exist (Lissauer, 1993) for the expected isolation mass with respect to the initial surface density Σ_{solid} at a given semi-major axis. Armitage (2009, p. 165) states that the isolation mass for a body at 1 AU under the assumption of $\Sigma_{solid} = 10$ g cm^2 to be approximately $0.07M_{\oplus}$, i.e. bodies at 1 AU will grow to be roughly the mass of the moon before clearing their Hill radius of planetesimals. In contrast, the same calculations found that the isolation mass for Jupiter's core is closer to 9 $M_{\oplus}$ which, as will be seen, is important for the formation of gaseous planets.

This then concludes the early growth phase of the planetary formation process. It is theorised that this process is rapid, taking on the order of 0.01 - 1 Myr to complete and resulting in 100 - 1000 protoplanets within the terrestrial planet region with masses, in disks similar to our own, approximately that of the Moon or Mercury.

1.4.3 Final assembly phase

The final assembly phase of terrestrial planet formation is perhaps the most fascinating and describes the period of time from the formation of protoplanets right up to the point that the terrestrial planets have fully formed. This phase is the period in time least accessible to statistical approaches to its understanding because the reduced number of bodies present means that few assumptions about the dynamics can be made. Therefore, the acquisition of knowledge about this phase lies predominantly in the domain of n-body simulation. Due to the stochastic nature of these models, the results about this phase are statistical in nature but can also qualitatively explain a some of the less obvious macroscopic features within our solar system (Walsh et al., 2012; Raymond et al., 2009; Tsiganis et al., 2005; Gomes et al., 2005; Chambers, 2004).

The beginning of the final assembly phase is not precisely defined but is said to begin when roughly half of the mass of the disk is contained within protoplanets (a.k.a. planetary embryos) and the other half remains in planetesimals / oligarchs (Kokubo, 2000). Simulations of this phase (Raymond et al., 2006; O'Brien et al., 2006) begin with a small number of protoplanets (< 100) and approximately 1000 oligarchs; it is believed that this number is enough to ensure damping due to dynamical friction is present. Both Raymond and O'Brien found that the final assembly phase takes on the order of 10^8 years which stands in stark contrast to the much shorter runaway and oligarchic growth phases (approximately 10^5 years).

1.5 Gas giant formation

Despite the focus on the terrestrial planets later on in this thesis, the formation of gas giants also helps sculpt the terrestrial formation region (Batygin and Laughlin, 2015) and as such a very brief discussion is included here for completeness. There are two competing models for the formation of gaseous planets of a nature similar to Jupiter and Saturn. The first is the core accretion model and the second is the disk instability model; only the core accretion model is discussed here.

1.5.1 Core accretion model

In the core accretion model, giant planet cores form in the outer solar system through the previously discussed effects of gravitational focusing but also via pebble accretion as a result of planetesimal interaction with the sub-Keplerian velocity gas disk (Chambers, 2021, 2014; Lambrechts and Johansen, 2012). In either case, these cores form externally to the snowline and therefore consist of both rock and ice. A gas envelope cannot be maintained around a planet core if the speed of sound in the surrounding disk is higher than the gravitational escape velocity at the surface of the core. Analytical estimates indicate that it is unlikely the isolation masses required to maintain this envelope can be reached within the snowline, and therefore indicates that giant planets likely favour forming external to this location where the majority of the core mass can be made up of water ice.

Recently, however, proposals have been made whereby one could envisage a much higher surface density present than the minimum mass extra-solar nebula, allowing for formation to occur much closer to the star (Batygin et al., 2016a). In either case, once this critical mass is reached the capture of gas onto the star is rapid, with estimates of a few million years (Helled et al., 2014), well within the lifetime of the gas within the protoplanetary disk. This implies that the gas giants would have been present to influence the formation of the terrestrial planets, and this is reflected in the initial conditions of recent models (Raymond et al., 2006, 2009; Raymond and Cossou, 2014). These classical models begin the final assembly process with protoplanets and oligarchs present in the terrestrial planet region and with fully formed gas giants present in the outer solar system. It has been shown that the specific orbital parameters of these giants strongly affects the macroscopic properties of the final systems formed (Raymond et al., 2006; O'Brien et al., 2006). In extreme cases, dynamical

instabilities in the giant planet's orbits can actually completely remove all other material from the solar system (Raymond and Cossou, 2014) putting a dramatic end to the formation process.

1.5.2 Migration

Although gas giants are thought to favour forming external to the snowline, this does not mean that they have to remain there. A combination of observational evidence and simulation data strongly suggest that gas giant orbits migrate through interaction with the gas disk.

In order for migration to occur, a torque must be applied. It is theorised that this torque is due to interaction between the planet and the gas in the disk. A planet in an axisymmetric disk experiences no torque and it is therefore a requirement for migration that the disk is altered in some way that breaks this symmetry, be this through gravitational interaction with the planet itself or otherwise. A discussion about the mechanisms through which migration occurs is beyond the scope of this work, but work in this direction has been extensive and is ongoing (Goldreich, 1980; Tanaka and Ward, 2004; Jimenez and Masset, 2017). There are two types of migration, dubbed Type I and Type II. Type I migration is theorised to occur for low mass planets when the interaction between the planet and the disk is not strong enough to open up an annular gap in the gas of the disk. Ward (1997) showed that Type I migration can only realistically cause an inwards migration. In contrast, Type II migration is theorised to occur for higher mass planets where the planet-disk interaction is strong enough to open up an annular gap, this can potentially cause both positive and negative torques to be applied causing migration inwards or outwards (Ward, 1997). Application of migration to Jupiter and Saturn in simulations of our solar system has achieved success in replicating some macroscopic features of the own solar system and has led to models such as "the grand tack" (Walsh et al., 2012) that provides a possible explanation for the low mass of Mars. Planetary migration is also of interest for explaining the discovery of "Super Earths" and "Hot Jupiters" by the Kepler mission (Raymond and Cossou, 2014).

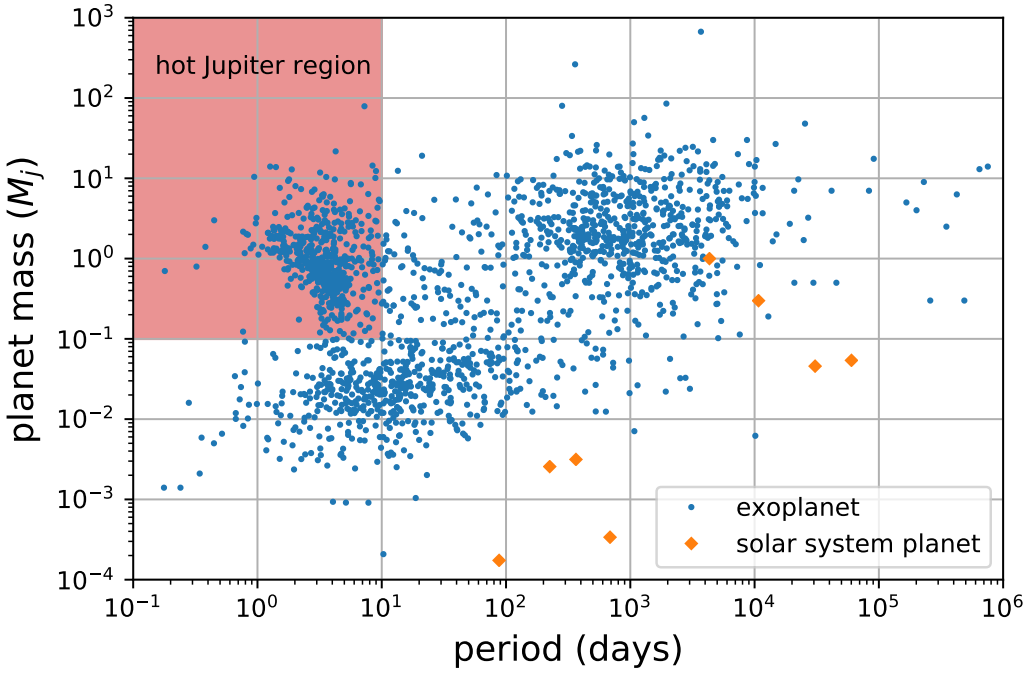


Figure 1.2: Planetary mass against orbital period for characterised planet systems. Exoplanets are shown in blue and solar systems planets are shown as an orange diamond. The hot Jupiter region is marked in red.

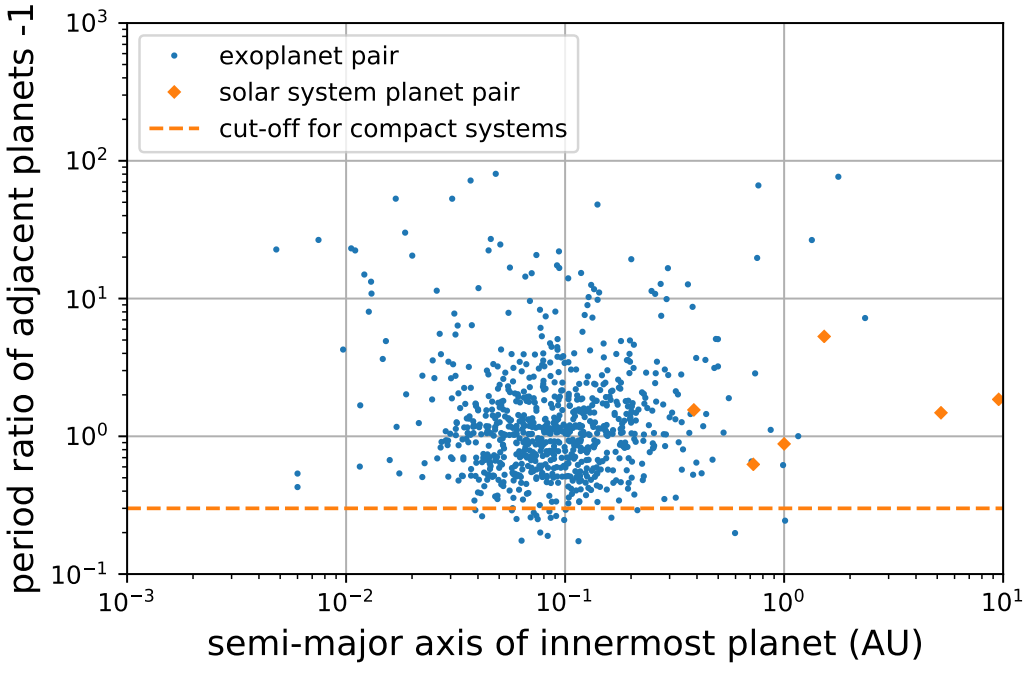


Figure 1.3: Period ratio of adjacent planets minus one, to enable the log scale, against the semi-major axis of the innermost planet. Exoplanets are shown in blue and solar systems planets are shown as an orange diamond. The dashed orange line indicates the cut-off for systems to be considered compact.

1.6 Exoplanet demographics

One of the most exciting aspects of the study of exoplanets is the recent increase in observational data thanks to the Kepler and TESS missions as well as a large number of ground-based observatories. As of 1^{st} April 2021, there have been 4,375 confirmed exoplanet discoveries which have together yielded a large enough dataset to allow for statistical studies to be performed on, e.g. their orbital parameters, radii and masses; thus, the sub-field of exoplanet demographics was born.

Methods of detection such as the transit method and radial velocity search are not without their biases, and both methods favour the detection of larger/more massive planets with a short orbital period. Despite this, a collection of statistically significant, interesting, and unexpected features are present within the dataset. In particular, "hot Jupiters", and "compact systems" are especially exciting as each of these features challenges conventional formation theories.

1.6.1 Hot Jupiters

Hot Jupiters are defined as short period planets ($<$ 10 days) with a mass roughly that of Jupiter ($>$ 0.1 M_j) (Wang et al., 2015). Their existence highlighted an inconsistency between previous formation theories formulated around observation of our own solar system and the more general case of exoplanet systems. As discussed, in our solar system gas giants are only found external to the snowline; however, the discovery of hot Jupiters means that this observation does not hold in the general case. Figure 1.2 shows the mass of detected planets against their orbital period for all known exoplanets with available data. Exoplanets are shown in blue and the planets of our solar system are shown as an orange diamond. The location of the data points for our solar system being situated apart from those for exoplanet systems is a result of the previously discussed observational biases. The area marked in red indicates the hot Jupiter region of parameter space, within which are many hot Jupiters. Despite the number found, the expected occurrence rate of hot Jupiters is less than 1% (Wang et al., 2015).

The challenge in explaining hot Jupiters lies in the fact that there is not enough mass present, in general, in the protoplanet feeding zone internal to the snowline for a suitable mass planetary core (10 $M_\oplus$) to form that can therefore sustain a gaseous envelope (Dawson and Johnson, 2018). The dampening effect of protoplanetary disk gas upon eccentricities precludes highly eccentric orbits allowing for the accretion

of mass from space external to the snowline. Thus, in-situ formation is unlikely to have occurred unless the minimum mass extra-solar nebula (MMEN) is much more massive that the MMSN in these systems (Bailey and Batygin, 2018; Batygin et al., 2016b). The leading hypothesis is that the core of these planets formed externally to the snowline where the isolation mass is high enough for a gas giant to form, and that these planets then migrated inwards through interaction with the protoplanetary disk (Naoz et al., 2011). In either case, the detection of this family of planets is one of the key findings of exoplanet demographics to date.

1.6.2 Compact planetary systems

A second enigma of exoplanet demographics is the existence of compact systems. A compact system is one where the period ratio of two adjacent planets in a given exoplanet system is very close to one, typically defined as < 1.3. As a comparison, the closest period ratio of adjacent planets in our own solar system is that of Venus and Earth with a value of 1.6; our solar system is non-compact. In contrast, the largest period ratio present in our solar system is that of Mars and Jupiter with a value of 6.3. Compact systems are of interest because the close orbital spacing of these systems means that the dynamics are dominated by strong orbital resonances rather than secular dynamics (Wisdom, 1980; Quillen, 2011; Obertas et al., 2017; Hadden and Lithwick, 2018; Petit et al., 2020). Numerical evidence suggests (Smith and Lissauer, 2009; Rice et al., 2018; Tamayo et al., 2016, 2020a) that this will cause them to become unstable over relatively short astronomical timescales (typically $< 10^9$ dynamic periods) making their presence in the exoplanet dataset the subject of active research (Lissauer and Gavino, 2021). Figure 1.3 shows the period ratio of adjacent planets, adjusted by one to enable the logarithmic scale, against the semi-major axis of the innermost planet in a given pair of planets. Data are included for all known exoplanet systems with available parameters. Planet pairs found in exoplanet systems are shown in blue, whereas pairs from our own solar system are shown as an orange diamond. The dashed orange line indicates the cut-off for a planet pair to be considered compact. There are twenty-three such compact planet pairs underneath the requisite period ratio. The study of compact planetary systems is a key focus of this thesis and forms the basis of Chapter 5, where the behaviour of these systems is studied right up until a collision of planets.

1.7 Planetary dynamics as a gravitational n-body problem

The discussion thus far has focused around planet formation theory and exoplanet demographics. The remainder of this chapter instead considers the requirement to accurately simulate planetary dynamics. To this end, the discussion will centre around the architecture of a gravitational n-body simulation, the computational challenges involved, and the current state-of-the-art in this field.

The term gravitational n-body simulation, henceforth n-body simulation, has been used so far without a formal definition. An n-body simulation is a simulation that describes the dynamics of a group of n particles under the influence of gravitational forces over a specified period of time. Therefore, as a modelling technique, it is directly applicable to the study of planetary formation and evolution, especially once the gas in the protoplanetary disk has dissipated and the dynamics are exclusively gravitational. Despite over 300 years of effort, the n-body problem only has known analytical solutions in the simplest of cases, e.g. the two-body case, and therefore requires approximate numerical solutions to be obtained instead. Given a set of initial conditions composed of the position and velocity of each n particles, an approximation to the state of the system at a later point in time can be obtained through numerical integration. Ergo, the accuracy of any approximations to planetary dynamics obtained in this manner depends on the accuracy of the integrators used.

1.7.1 Equations of motion

Throughout this work, the effects of gravity are assumed to be purely Newtonian as the relativistic effects are understood to be small in this context. Later on, this thesis makes heavy use of Hamiltonian dynamics and as such the gravitational n-body equations of motion are introduced here using the same formalism. The Hamiltonian function, or simply the Hamiltonian for short, for n particles of mass m_i in Cartesian coordinates with position vectors $\mathbf{q}_i$ specified in an inertial reference frame and with conjugate momenta $\mathbf{p}_i$ where $i = 1, ..., n$ is

$$H(\mathbf{p}, \mathbf{q}) = \sum_{i=1}^{n} \frac{\mathbf{p}_i \cdot \mathbf{p}_i}{2\, m_i} - \sum_{i=1}^{n} \sum_{j>i}^{n} \frac{G m_i m_j}{|\mathbf{q}_i - \mathbf{q}_j|}, \tag{1.3}$$

where G is the gravitational constant. Throughout this work, the symbol $\cdot$ is used to signify the dot product. Hamilton's equations allow for the time derivatives of

the position and momentum of a single particle to be obtained from the Hamiltonian directly, these are given by

$$\begin{aligned}\frac{\mathrm{d}\mathbf{q}_i}{\mathrm{d}t} &= \frac{\partial H}{\partial \mathbf{p}_i},\\ \frac{\mathrm{d}\mathbf{p}_i}{\mathrm{d}t} &= -\frac{\partial H}{\partial \mathbf{q}_i}.\end{aligned}$$

Ergo, using a notation such that $\mathbf{q}_{ij} = \mathbf{q}_j - \mathbf{q}_i$ the equations of motion are

$$\begin{aligned}\frac{\mathrm{d}\mathbf{q}_i}{\mathrm{d}t} &= \frac{\mathbf{p}_i}{m_i},\\ \frac{\mathrm{d}\mathbf{p}_i}{\mathrm{d}t} &= \sum_{\substack{j=1\\ i\neq j}}^{n} \frac{G m_i m_j}{|\mathbf{q}_{ij}|^3}\mathbf{q}_{ij}.\end{aligned} \tag{1.4}$$

In the purely gravitational case, the equations of motion are therefore conservative as the change in momentum only depends upon the relative location of the particles in space. However, the final equations can easily be expanded to include an arbitrary perturbation to the change in momenta, thereby increasing the scope of addressable problems, e.g. to include the effects of gas drag owing to the gas in an early-stage protoplanetary disk. Introducing non-conservative forces, however, removes several properties that, as we will explore in Chapter 2, are highly favourable to modelling the long-term evolution of planetary systems.

1.7.2 Integrals of motion

The absence of analytical solutions to the n-body problem not only necessitates the use of numerical integration to obtain approximations to the dynamics, but it also means that there are no known solutions that can be used to determine the precision of the approximations themselves. One potential solution to this problem is the creation of a high-precision reference solution that uses a more precise floating-point representation in its generation than would typically be used in practice. In Chapter 4, I use a quadruple-precision floating-point representation for this purpose, which has a maximum precision 16 orders of magnitude above that typically used in scientific computing. The consequence of performing calculations that make use of this extra precision is an increased computational cost and solutions of this nature are therefore resource intensive to generate. Furthermore, an additional limitation is that they are only valid for a single set of initial conditions. Worse still, the e-folding time, i.e. the time taken for the state of a system to diverge from nearby states by a factor of e, of

planetary systems coupled with the stochasticity of n-body simulations due to finite precision and the choice of time step during integrations means that solutions are only useful for a relatively short interval of time after the initial conditions. It is therefore typical to turn to integrals of motion as a proxy for the precision of a given simulation. The n-body problem has four integrals of motion, of which three are vector invariants and one is a scalar quantity.

Firstly, the Hamiltonian, Eq. (1.3), which is equivalent to the energy of a system, is a conserved quantity. Secondly, the angular momentum, $\mathbf{J}$, of a system is conserved and is defined as

$$\mathbf{J} = \sum_{i=1}^{n} \mathbf{q}_i \times \mathbf{p}_i.$$

Finally, an additional three constants of motion can be obtained by considering the symmetry of the forces in Eq. (1.4) to state that

$$\sum_{i=1}^{n} m_i \ddot{\mathbf{r}}_i = 0.$$

Analytically integrating this equation shows that the position of the centre of mass $\mathbf{c}$ moves with constant velocity, thus $\dot{\mathbf{c}} =$ constant and $\ddot{\mathbf{c}} = 0$. This makes a total of ten conserved scalar quantities that can be measured at any point in time within a simulation to help ascertain the numerical performance. In practice, energy and angular momentum are most frequently used although even in this case the conservation is often dominated by the behaviour of a few relatively massive bodies.

1.8 Computational challenges in planetary dynamics

The n-body equations of motion, Eq. (1.4), contain two problematic subtleties that must be addressed to effectively simulate the formation of planetary systems.

1.8.1 Calculating the gravitational forces between bodies

The first of these subtleties is that the computational cost in evaluating the total force experiences by bodies scales as $O(n^2)$, which limits the size of systems that can be simulated to modest particle numbers. In turn, this limits the point in the formation process that simulations can be initialised due to the particle number required

increasing the further back in time one wishes to study. In addition to this temporal limitation, it also means that the effects of dynamical friction cannot be fully explored. This is because mass within a planetary system must be non-physically grouped together for computational efficiency which is postulated to be hiding some macroscopic features of planetary evolution (O'Brien et al., 2006).

There are several existing methods for reducing the $O(n^2)$ scaling. For instance, the Ahmed-Cohen method introduces a neighbourhood around each particle and assumes that the dominant force contribution will come from this neighbourhood. It therefore becomes possible to integrate the distant force contributions on a larger time step thereby increasing efficiency. This method has been shown to reduce the computation cost to scale as $O(n^{1.6})$ (Makino and Aarseth, 1992). The seminal method, however, was contributed by Barnes and Hut (1986). They used a spatial decomposition that allows for the separations to be calculated in $O(n \log n)$ time. This algorithm is prolific and has allowed for much larger simulations to be possible across a number of fields. The applicability of either of these algorithms is determined by the coefficient multiplying their scaling factor, as this will determine the particle number at which the algorithms become more efficient than the direct approach. Additionally, each of these methods approximates the force in one way or another and as such impacts the precision of simulations. Given both of these factors, combined with the relatively low particle number used throughout the remainder of this thesis, I have not chosen to use any approximations to gravity and instead chose to accept the $O(n^2)$ computation cost.

1.8.2 Close encounters

The second subtlety is that of close encounters which have two negative effects. First, during a close encounter the dynamic timescale of the encounter becomes very different from the orbital timescale, thereby imposing additional requirements on the numerical integration process in order to resolve both long and short timescales precisely. In this case, the equations of motion are said to have become "stiff". Secondly, in the limit of the close approach, $\lim_{r_{ij} \to 0}$, the equations of motion become singular and, as such, cannot be worked with. In practice, the latter effect is not typically a problem as particles are generally assumed to merge before this happens, alternatively a form of softened potential could be used. Still, the additional constraints placed upon the choice of integrator are incredibly wide reaching, as will be shown in Chapter 2. The study of compact exoplanet systems in Chapter 5 requires that

the orbital propagation method developed in Chapter 4 can gracefully handle close encounters all the way to the collision of planets.

In addition to the use of a "stiffly stable" integrator, regularisation is another potential avenue for gracefully handling close encounters. Regularisation involves the reformulation of the equations of motion such that the equations become non-singular and enjoy numerically superior behaviour during a close encounter. In the co-planar case, a complex conformal mapping, known as the Levi-Civita transform (Levi-Civita, 1920), onto a parametric plane can eliminate the singularity present in a two body problem and enhance the approach behaviour. Moreover, in the three dimensional case, the Kustaanheimo and Stiefel (1965) (KS) transform achieves the same purpose, although this requires mapping the current state onto the surface of a 3-sphere embedded in $\mathbb{R}^4$ space and then propagating the system forward in this higher dimensional space. The KS transform has been further developed to handle encounters between three bodies (Aarseth and Zare, 1974), and there are many global regularisation techniques based around it for n bodies (Bettis and Szebehely, 1971; Heggie, 1974; Mikkola, 1985; Mikkola and Aarseth, 1989). While these techniques are highly effective (Amato et al., 2017) and moderately efficient, they have not found application in this work as preference is given to a careful choice of the numerical integrator instead.

1.8.3 Accurate long-term integrations

Solar-mass stars have a lifetime of approximately ten billion years, and the story of planetary formation, stability, and evolution thus spans the same period. As such, n-body based studies require accuracy over these timescales. Integrations spanning ten billion dynamical periods are prohibitively expensive (Lissauer and Gavino, 2021) but integrations spanning a billion dynamical periods are now commonplace. Assuming an exact force calculation, the precision of a numerical integrator over these timescales will determine the overall precision of the n-body simulation. This imposes a very tight requirement on the conservation of integrals of motion to anyone wishing to simulate the formation process in detail (Saha and Tremaine, 1992). Failure to conserve the angular momentum, for example, can result in increases in the orbital eccentricity, inclination, and semi-major axis of the planets and therefore invalidate the results of a simulation. Therefore, any inaccuracies present in the numerical integration process, however slight, have the potential to build up over the timescale being studied and corrupt the approximations to the dynamics obtained. This is such an important topic that Chapter 2 is entirely dedicated to understanding the options

available, and the remainder of this section is therefore meant only as a very brief introduction. There are currently two main approaches to ensuring the conservation of invariants over these timescales:

1. Symplectic integration.
2. Non-symplectic (traditional) integrators that follow Brouwer's law.

Symplectic integrators, such as the seminal Wisdom-Holman (WH) map (Wisdom and Holman, 1991), make use of Hamiltonian theory to split the system Hamiltonian into distinct parts, such that the associated equations of motion for each part can be integrated analytically in isolation. The splitting process introduces a small error in the description of the system meaning that these schemes solve exactly for a system that is slightly perturbed from the original. This leads to no long-term error growth of conserved quantities and only a linear error growth in angle variables, such as mean anomaly, as opposed to the quadratic growth present with more traditional integration schemes (Kinoshita et al., 1990). One of the downsides of the symplectic schemes is that their symplectic nature is broken if the step size is changed; ergo, all particles must be integrated on a fixed step size, ad infinitum. In terms of efficiency, this would be a major performance bottleneck were it not for the fact that symplectic schemes can make an order of magnitude fewer evaluations of the force function per orbit than traditional schemes, provided only moderate precision is required. However, the inability for symplectic schemes to vary their step size does pose a problem for resolving the dynamics of close encounters. Special adaptations to symplectic schemes can be made that allow for so-called "hybrid" schemes to integrate close encounters using a traditional integrator without breaking the symplecticity of the overall scheme. In the case of the hybrid MERCURY and GENGA (Grimm and Stadel, 2014) schemes this means switching to a non-symplectic Bulirsch-Stoer scheme, whereas another scheme known as QYMSYM (Moore and Quillen, 2011) chose to use a Hermite scheme instead. In either case, this means that close encounters need to be detected and handled, consequently making hybrid symplectic schemes much more computationally complex than they would otherwise be and creating a performance bottle neck that has resulted in a MERCURY runtime of up to 16 months for final assembly simulations with large particle numbers (Raymond et al., 2006).

Increases in computational power mean that it has now become possible to use traditional integration schemes in this problem domain. As a comparison, for simulations of the outer solar system planets over a period of a billion Jupiter orbits, I have found that a particular symplectic scheme has a runtime of approximately ten hours,

whereas a highly efficient traditional scheme, running on the same hardware, takes approximately five days. Non-symplectic integrations of this nature are therefore right on the cusp of what is considered a reasonable length computation. The benefit of these schemes is that it is possible to achieve a degree of precision far higher than that achieved with a symplectic scheme. However, in order for this benefit to be realised, it is a requirement that the scheme follows Brouwer's law (Brouwer, 1937) which is the best possible error growth rate for a given floating-point representation. Currently, very few schemes achieve this error growth rate, but IAS15 (Rein and Spiegel, 2015) in the REBOUND (Rein and Liu, 2012) package is a notable example. Schemes that do not follow Brouwer's law have also been applied but the conservation levels are many orders of magnitude worse (Sharp and Newman, 2016).

1.9 High performance computing (HPC)

Previous discussions have highlighted that computational power is a key constraint on the possibility of more detailed and precise formation models, both through the calculation of the forces involved and through the integration process. Aside from using more efficient integration and force calculation algorithms, another possibility is the use of high performance computing (HPC) techniques, whereby aspects of a program can be run in parallel across multiple computing units to reduce the overall runtime. There are currently three main options available, each with their own specific intended application area. They are: MPI (Kowalik, 1996), OpenMP (Chandra et al., 2001), and Graphical Processing Units (GPUs) through either CUDA (Patterson, 2010) or OpenCL (Scarpino, 2012). MPI and OpenMP are both tools designed to enable utilisation of multiple processing units via distributed computing. The former works by having multiple copies of a program running with a communications channel between them. This can be any protocol but is typically PCI on super computers (EPCC, 2017). Having a communication channel means that the number of computing units that can be connected is not limited, allowing theoretically for infinite scaling. The cost of this flexibility, however, is that the speed of this link will often become the performance bottle neck, limiting the overall system performance. In contrast, OpenMP only operates across computing units that share a common memory address space, typically in the region of 24 cores (EPCC, 2017). This means that memory access and therefore communication time are limited only by the RAM access time of the system. Ergo, for small programs requiring limited scaling of processor numbers, OpenMP will generally outperform MPI, but the hardware available puts a hard limit on the final scaling

of the system size. Both MPI and OpenMP have found applications in n-body simulations (Aarseth, 1999; Rein and Liu, 2012). MPI from a spatial decomposition point of view, particularly in conjunction with the Barnes and Hut algorithm, and OpenMP for generally decreasing the runtime of large computational loops. In Chapter 5, MPI is used extensively to distribute Monte-Carlo style integrations across over a thousand cores on the Iridis supercomputer. The third, and most modern option, is the use of GPUs. While originally designed for rendering graphics in real time, they have found applications in scientific computing, specifically the n-body problem (Miki and Umemura, 2017) with NVIDIA finding that they can achieve a timed performance increase of 100 times over using a single threaded processor (NVIDIA, 2008) when applied to the force calculation of a system where $n = 1000$. The application of GPUs to planetary dynamics has had some successes (Grimm and Stadel, 2014); however, efficiently implementing complex integration schemes on GPUs poses particular difficulties due to the amount of instruction branching required. A promising new family of integrators that do not suffer from this problem are the embedded operator splitting (EOS) methods (Rein, 2020) which may offer a more efficient means of using GPUs to solve planetary dynamics problems in the near future.

Table 1.1: Comparison of the macroscopic features of integrators in state-of-the-art planetary dynamics n-body integration packages. Green and red indicate that a feature is present or absent, respectively.

Integration package:	MERCURY			REBOUND			GENGA	EnckeHH	TES
Integrator name:	RADAU	MVS	HYBRID	WHFAST	MERCURIUS	IAS15	GENGA	EnckeHH	TES
Overall precision									
High									
Moderate									
Integrator type									
Symplectic									
Traditional									
Close encounter resolution									
Hybrid symplectic scheme									
Adaptive step-size integrator									
Regularisation									
RHS optimisations									
Analytical Keplerian motion									
Encke based method									
High performance computing									
GPU									
OpenMP									
MPI									

1.10 State-of-the-art planetary dynamics modelling tools

There are myriad integration packages available for the n-body practitioner wishing to model planetary dynamics. There are far too many to discuss them all as new tools are frequently developed to address modelling a particular niche in the formation process. As a result, this section focuses only on key packages that have seen mass adoption or are particularly relevant to the work in this thesis.

The first package of note is NBODY6, which is the sixth iteration of the NBODY package developed over 40 years at the University of Cambridge (Aarseth, 1999). NBODY is particularly remarkable as it led the way in regularisation (Aarseth and Zare, 1974; Mikkola and Aarseth, 1993) and numerical integration for many years. Despite being a formidable tool, NBODY is aimed at galactic scale simulations and, as such, lacks the long-term precision features required for solar system dynamics. Therefore, NBODY is no longer widely used by the planetary dynamics community and is not discussed any further here.

The de facto standard tool for modelling solar system dynamics is called MERCURY (Morbidelli et al., 2000; Tsiganis et al., 2005; O'Brien et al., 2006; Raymond and Cossou, 2014) and was created by Chambers (1999). MERCURY is written in Fortran and contains multiple integrators available through a common interface. Firstly, an implementation of the WH mapping is included for modelling systems where bodies are not expected to experience close encounters. However, at the heart of the package is the hybrid symplectic scheme which finds utility through the addition of an elegant solution to handle close encounters without breaking the symplecticity of the WH mapping. In addition to the symplectic methods, MERCURY contains several non-symplectic schemes and one of particular note is the RADAU scheme of Everhart (Everhart, 1974, 1985) which is discussed in Chapter 2. MERCURY is intended to run on a single workstation and therefore does not possess any HPC aspects to allow for a greater particle number to be achieved.

A modern addition to the list of integration packages is REBOUND (Rein and Liu, 2012). REBOUND is a framework for enabling easy use of multiple state-of-the-art integrators. The package is primarily written in C99 but also boast a simple python interface to allow for ease of use. REBOUND contains several integrators that are similar in nature to those of MERCURY, however, each of the implementations has been refined to great effect enabling much more precise solutions to be obtained than the with the initial versions. Firstly, the MVS scheme has been improved to contain no bias in round-off error and is called WHFAST. Secondly, the hybrid scheme has been

updated to use a more precise non-symplectic routine to handle close encounters and is known as MERCURIUS (Rein et al., 2019b), so called in appreciation of the original MERCURY integrator. Finally, Everhart's RADAU scheme included in MERCURY has been further refined and forms the basis of the error optimal scheme IAS15 in REBOUND. A combination of the quality of integrators and ease of use means that REBOUND is now incredibly widely adopted.

A less widely known but still very powerful tool is known as GENGA (Grimm and Stadel, 2014). The integrator in the GENGA package is very similar to the hybrid scheme within MERCURY; however, it has been modified such that it can run efficiently on any NVIDIA GPU. In comparison to MERCURY, GENGA achieves a slightly more consistent conservation of energy while boasting runtimes 40 times faster for systems containing large particle numbers. As a result, GENGA has enabled Nice model simulations to be run with 2048 particles, which was a two-fold increase of any prior simulation.

The last tool that will be discussed is that of Hernandez and Holman (2020) and is known as EnckeHH. This tool makes use of a branch of the REBOUND code base and utilises the IAS15 integrator. The contribution of EnckeHH is a modification to the equations of motion to enable the dominant Keplerian dynamics of the central body to be taken into account analytically. EnckeHH greatly improves the performance of IAS15 in the case of fixed step sizes such that it follows Brouwer's law. Many of the concepts in EnckeHH are also used by the scheme developed in Chapter 4 despite being developed independently at the same time.

One of the key contributions of this thesis is the development of a novel tool for application to the long-term study of planetary dynamics in the presence of close encounters. Importantly, this tool is able to match the precision levels achieved by the best existing tools but with a reduced computational cost. This enables larger ensembles of simulations to be run for a given available computational resource, or, alternatively, allows for a given set of simulations to be performed with a reduced environmental impact through reduction of CO_2 emissions. The tool developed is known as the Terrestrial Exoplanet Simulator (TES) (Bartram and Wittig, 2021) and is discussed at length in its own chapter, Chapter 4. It is important at this stage to understand the novelty of this tool in the context of all others previously discussed, and to do this a comparison of the macroscopic features of each integration routine is presented in Table 1.1. The final column contains the features of TES, where it can be seen that TES is the only high precision, traditional, adaptive step-size integrator that uses the dominant Keplerian dynamics to form an Encke method. TES is able

to resolve close encounters between Earth-mass planets at close to machine precision and is capable of runtimes up to 20% faster than IAS15 for low mass systems.

Chapter 2

On numerical integration for n-body simulations

The equations of motion governing the accelerations in a system of particles acting under mutual self-gravitation are well known. Therefore, given a set of initial conditions, the n-body problem, and thus, to a good approximation, solar system dynamics, is reduced to the solution of a system of ordinary differential equations. In all but the most trivial of cases, the n-body problem has no known solution and instead one must resort to using numerical integration techniques to obtain an approximate solution. Clearly, the accuracy of this solution is of paramount importance to the practitioner wishing to capture the remarkable complexity of n-body physics.

Numerical integration is a large field with a plethora of options available. The scope of problems across many diverse fields requiring accurate numerical approximations to the solution of ODEs ensures that research is continuously evolving. In general, there are two broad families of numerical integration techniques: multistage and multistep methods. The distinction between these two families is the way in which information about the ODE is obtained.

Multistage methods, which include the classic Runge-Kutta schemes, work by obtaining information about the derivative within the current step, i.e. no solution points from previous steps are used in determining the next solution value. Multistage methods are convenient as they are self-starting, i.e. they do not require information about past solution points to begin an integration. In contrast, multistep methods work by using past solution data points to reduce the number of evaluations of the derivative required within a single step oftentimes leading to highly efficient schemes. However, this performance comes at a cost: multistep methods are not self-starting and

extension to variable step size requires careful backward interpolation between past solution points.

It should be noted that these two categories are incredibly general and there is a substantial richness of schemes within both of these groups. Indeed, as we shall see in Chapter 3, schemes also exist that are capable of combining information about the system of equations being integrated from both previous solution points and the derivative within the current step.

It is notoriously difficult to define how good a particular integration scheme is; however, one key metric that is intricately linked to the performance of all integration schemes is the order of convergence of the scheme. In general applications, high-order schemes are used when a high degree of precision is required in the solution. They are used such that the per-step integration error can be minimised efficiently. However, when modelling the dynamics of planetary systems, the way that errors accumulate over time is equally as important as the error associated with a single integration step. As such, high-order methods alone are often not sufficient when modelling exoplanet systems. Additionally, as we will also see, there are other properties of the planetary n-body problem that can be leveraged to create more interesting and exotic integration schemes. For example, the symplectic geometry of the phase space, the time-symmetric nature of the problem, or the dominant contribution to the dynamics of the central body.

In addition to these favourable symmetries and geometries for creating new schemes, the planetary n-body problem also comes with three additional requirements that must be addressed by any aspiring integrator, which are:

1. Ensuring that solutions obtained remain accurate over the timescales required, in solar system formation and stability studies typically 10^9 dynamical periods.
2. Ensuring that integrators can precisely model close encounters between objects.
3. Ensuring that such long-duration simulations can be completed within the available computing time.

Only through addressing all three of these problems can one arrive at a scheme at all suitable for general use within this problem domain.

This chapter is a broad introduction to the most commonly used and/or cutting edge numerical integration methods that meet these requirements. Section 2.1 provides a general overview as to the sources of error when performing numerical integration.

Section 2.2 provides an introduction to the symplectic methods commonly used in this field. Section 2.3 focuses on non-symplectic integration methods and highlights a few key schemes that are commonly used. Additionally, some other more exotic techniques are explored here. Finally, Section 2.4 brings all of the schemes together to identify the performance envelope for each scheme in the broader context of all others in a series of numerical experiments.

2.1 Sources of numerical error

To understand the performance of a particular integrator, it is a requirement to first identify the sources of numerical error as many schemes are capable of operating in multiple error growth modes. The following discussion is applicable to both symplectic and traditional integration schemes, although a broader discussion of the sources of numerical error for the former case can be found in Section 2.2.

2.1.1 Machine precision

Computers aspiring to perform calculations of a scientific nature require a means of representing numbers. Integer arithmetic is used in some cases where only integer values are needed as this arithmetic has the advantage of operations being rapid on a computer. One drawback of this choice is that in order to represent the real numbers it is necessary to store them as the ratio of two integers and perform operations accordingly, which increases the memory requirements and computational costs of calculations. In contrast to the low precision of integer arithmetic, ball arithmetic can be used to not only store a real number but to also provide an upper limit to the error associated with that number due to finite precision calculations; note that ball arithmetic does not address modelling errors.

The commonly used compromise between these two extremes of performance and precision are the floating-point numbers which are used throughout the remainder of this work. IEEE754 (IEEE, 2019) is the standard defining the representation of these variables and the rules of their arithmetic, including rounding. There are three commonly used word lengths for floating-point variables: single, double and quadruple precision, and these have a word length of 32, 64, and 128 bits respectively. The larger the word length, the more significant figures can be represented, and it is therefore important to choose the correct size. In this work, double precision is predominantly

used as single precision is not accurate enough and quadruple precision has too much computational overhead due to the hardware available in most computers. In double precision, 52 out of 64 bits make up the mantissa and allow for a range of around sixteen significant decimal figures. Given this range, the minimum relative error in any calculation is $\approx 1.1 \times 10^{-16}$. I call this limit upon the relative precision $\epsilon_{machine}$.

2.1.2 Truncation error

Truncation error, alternatively referred to as integrator or discritisation error, is the error introduced when an infinite process is instead represented by a finite one. In numerical integration schemes, this substitution often occurs when an infinite series expansion is truncated such that it only includes a finite number of terms. Truncation error is notoriously difficult to quantify as it depends upon many scheme-specific parameters, e.g. the order of the scheme and the convergence criteria used in implicit solvers will both contribute.

Step size control algorithms use integrator specific techniques to provide an estimate of the truncation error incurred within a given integration step. In the subsequent integration step, the error estimate is used to select a step size that will maintain a near constant per step truncation error at a previously specified tolerance level. When using traditional integration schemes, the truncation error growth rate is $\propto t$ in the integrals of motion which translates to a growth rate $\propto t^2$ in positional variables (Sharp, 2006). In practice though, application specific integrators, e.g. symplectic or time-symmetric schemes, are often used that enable an upper limit to be place on the truncation error. These types of scheme are explored later in this chapter. Furthermore, it is possible to select a suitable step size such that the magnitude of the truncation error is below $\epsilon_{machine}$, and in this case round-off error can be made to be the dominant error source. I term this error contribution $\epsilon_{truncation}$.

2.1.3 Round-off error

The finite precision resulting in $\epsilon_{machine}$ also means that round-off error is introduced when arithmetic operations are performed on floating-point variables. This error source affects all programs that use floating-point arithmetic and is not limited only to integrators. When an arithmetic operation is performed that cannot be expressed exactly by the current floating-point representation, IEEE754 ensures that the result

of the calculation will be rounded up or down to the nearest value that can be. Consequently, the round-off error per floating point operation is $< \frac{1}{2}\epsilon_{machine}$. Therefore, the round-off error per operation is tiny in magnitude. However, these errors can rapidly accumulate over time because of the large number of floating-point operations that comprise even a single integration step. Especially when one considers the more than 10^{10} integration steps in typical solar system simulations.

As a simplified example, let us consider the summation of m elements, each identified by a subscript i, of a random vector, $\mathbf{X}$, such that $X_i \in [0, 1)$ sampled from a continuous uniform distribution. The result of this summation is $\phi = \sum_{i=0}^{m} X_i$. For simplicity, assume that $\mathbf{X}$ is sorted in ascending order, failure to do this can result in an additional source of lost precision when elements of different magnitudes are summed. In this case, round-off error is the only error source and can therefore be considered in isolation. The error growth when summing to obtain ϕ depends upon each realisation of $\mathbf{X}$ and is therefore best considered as an average of a number of realisations. Figure 2.1 shows how round-off errors grow as more arithmetic operations are performed, i.e. as the summation to obtain ϕ is performed. To determine the round-off error, a separate summation is performed using quadruple-precision floating point which is then compared to the double-precision summation. One hundred individual realisations are shown in blue and each can be seen to follow a random walk. The RMS of all realisations considered is shown in orange. Finally, a linear model fitted to the log of the RMS and the operations performed is shown in red. The slope of this model is $1/2$ which shows that the error is growing like a random walk and is therefore the best possible growth in round-off error. Additionally, this indicates that there is no bias present in the summation process, as is expected due to the random nature of $\mathbf{X}$.

A numerical integration can be viewed as a series of additions. For example, when performing an update to the position x at step k the new state is obtained as $x_{k+1} = x_k + \Delta x_k$ where Δx_k is an update obtained through a numerical integrator step. Therefore, if one assumes that Δx_k is exact to the available precision, i.e. does not contain any truncation error, then when this process is repeated over m steps a numerical integration simply becomes a series of m additions. Ergo, the best possible error growth attainable when performing a numerical integration is also $\propto m^{1/2}$.

Brouwer (1937) showed that when integrating over time, t, to obtain approximations to the solution of celestial mechanics problems the best possible relative energy error growth is also $\propto m^{1/2}$ whereas for the orbital longitude it is $\propto m^{3/2}$ owing to the mean longitude being the result of an additional integral. Together these two statements are referred to as Brouwer's law and are the absolute limit in precision over

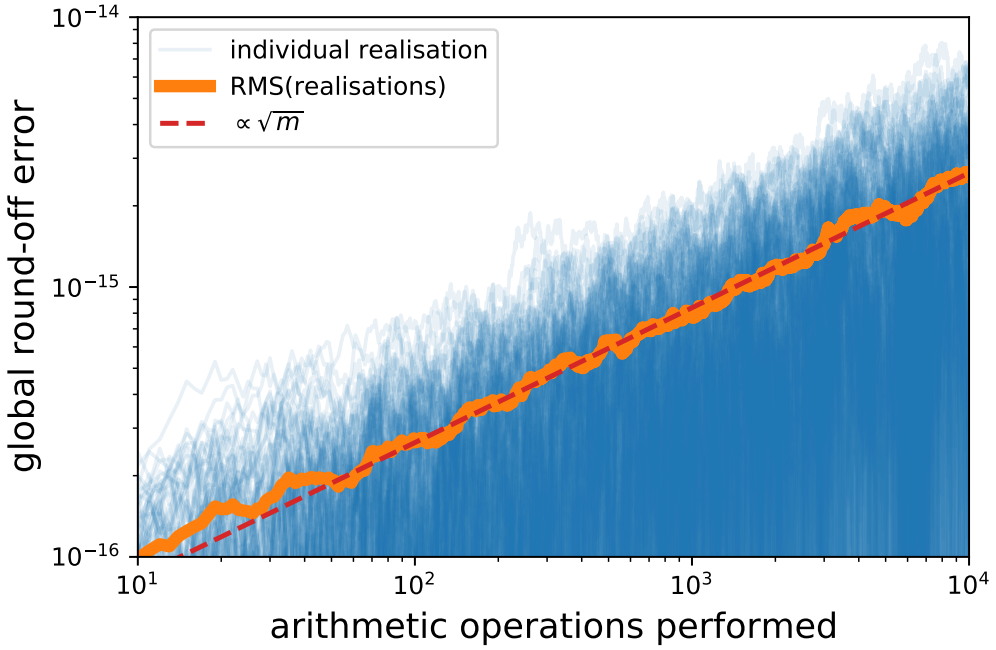


Figure 2.1: The accumulation of round-off error when performing a summation over the elements of the random vector $\mathbf{X}$ against the number of arithmetic operations performed, m. One hundred individual realisations of the random vector $\mathbf{X}$ are shown in blue. The RMS of all realisations is shown in orange. Finally, a linear model fitted to the RMS is shown in red.

time that can be obtained for a given size floating point representation. There are a small number of schemes that have been designed that follow Brouwer's law, e.g. Rein and Spiegel (2015) and Grazier et al. (2005) have both developed traditional integration schemes that do. I call this error contribution ϵ_{round}.

2.1.4 Kahan summation

Kahan summation (Kahan, 1965), also referred to as compensated summation, is a technique used to improve the precision obtained when summing over a sequence of floating-point numbers. Compensated summation allows for the error made during the summation of two floating-point numbers to be obtained and kept as an additional floating-point variable, known as the compensation variable. This error is then subtracted from future additions. Repeated application of this process allows for round-off errors to be greatly suppressed. The listing below shows a python implementation of the compensated summation algorithm applied to the random variable $\mathbf{X}$ from Section 2.1.3.

An additional use of compensated summation that is particularly applicable in this work is the summation of numbers of different magnitudes. Once again, consider the update of the positions at the end of a numerical integration step, $x_{k+1} = x_k + \Delta x_k$. Commonly, Δx_k is two orders of magnitude smaller than x_k, and yet the resulting summation must be represented within the approximately sixteen significant decimal digits available in x_k. Ergo, the two least significant digits in Δx_k are simply lost. In this case, compensated summation can maintain a record of this lost precision that can then be subtracted from subsequent position updates in later integration steps. Rein and Spiegel (2015) found a reduction of two orders of magnitude in the size of the random walk in relative energy error when applying compensated summation to their integrator updates.

```
def kahan_summation(Xi, phi, comp):
    y = Xi - comp
    t = phi + y
    comp = (t - phi) - y
    phi = t
    return phi, comp

phi = 0
comp = 0
for Xi in X:
    phi, comp = kahan_summation(Xi, phi, comp)
```

Listing 2.1: Python implementation of the compensated summation algorithm.

2.1.5 Bias error

In addition to round-off errors that follow Brouwer's law, there is also the possibility of biased round-off errors. Biased round-off errors occur when there is a systemic source of errors that are not symmetrically distributed with a mean of zero. It is possible that almost any combination of floating point operations within a numerical integration step could result in biased round-off error contributions; however, key examples of sources of bias error are intrinsic functions, e.g. the trigonometric functions, or the use of insufficiently precise integration coefficients. Biased round-off

errors can cause a linear growth in energy over time. That is to say, even if the truncation error is suppressed below floating-point minimum and compensated summation is used, there is still the possibility of linear error growth over time as a result of the round-off errors becoming biased. Linear bias error is the norm in a large number of integration schemes (Bulirsch and Stoer, 1966; Sharp, 2006). Worse still, it is possible that schemes appear to be unbiased over short integration periods but over longer integration periods the accumulation of round-off errors becomes biased and leads to transition away from Brouwer's law and into a linear energy growth regime. Therefore, when testing an integrator for bias error it is essential to perform integrations of a length that are representative of the timescales expected in the final application domain. I call contributions from this source ϵ_{bias}.

2.1.6 Total numerical integration error

At this stage, all sources of numerical error associated with an integration scheme have been identified. Therefore, the total numerical error in terms of relative energy error is just the sum of each of these sources, hence

$$\epsilon_{total} = \epsilon_{machine} + \epsilon_{round} + \epsilon_{truncation} + \epsilon_{bias}. \tag{2.1}$$

However, it is important to note the contribution of the dynamics themselves to the final system state error. The often chaotic dynamics of the n-body problem mean that small errors made in, e.g., position can, over time, result in solutions that exponentially diverge from the physical reality while also maintaining an excellent energy conservation. Therefore, exact trajectories of the n-body problem can only be trusted to be accurate over short timescales (Hernandez et al., 2021).

2.2 Symplectic integration

One class of numerical integration scheme that has been used extensively in the simulation of solar system dynamics is that of symplectic schemes. Symplectic integration algorithms (SIAs) are widely used in celestial mechanics for two key reasons:

1. they exhibit a bounded maximum truncation error,
2. they are highly efficient as they account for the dominant dynamics of the central body.

The upper bound on truncation error makes these schemes excellent candidates for the long-term integration of planetary systems. Oftentimes, when using an SIA the relative energy error of a system is the same in the first and final years of a billion-year integration. SIAs possess these advantageous properties because they are designed to preserve the symplectic structure of a Hamiltonian phase space, i.e. they preserve the Poincaré integral invariants. In turn, this has favourable properties for the conservation of other, more common, invariants such as energy and angular momentum. The symplectic structure exploited requires that the time evolution of a system is governed by a Hamiltonian, which dictates that these techniques can only be applied to model conservative systems[1]. Fortuitously, this includes the gravitational n-body problem as SIAs (Wisdom and Holman, 1991; Chambers, 1999; Moore and Quillen, 2011; Grimm and Stadel, 2014; Rein and Tamayo, 2015) have enabled a much deeper understanding of solar system dynamics (Clement et al., 2021; Esteves et al., 2020; Raymond and Cossou, 2014; Marois et al., 2010; Laskar and Gastineau, 2009; Chatterjee et al., 2008; Jurić and Tremaine, 2008) than would have likely been possible due to computational constraints.

Next, I introduce some well-known concepts related to the formal definition of symplecticity and what it means for a routine to be symplectic. This introduction is by no means exhaustive and exists to allow for the symplectic scheme widely used in solar system dynamics to be discussed in context.

2.2.1 Background and theory

Given a generalised position vector $\mathbf{q}(t) \in \mathbb{R}^d$ and conjugate momentum vector $\mathbf{p}(t) \in \mathbb{R}^d$, the state of a Hamiltonian system is $\mathbf{z}(t) = (\mathbf{q}, \mathbf{p})^T \in \mathbb{R}^{2d}$ where $d = 3n$ and represents the three physical dimensions of space for each of n bodies with mass m_i. The Hamiltonian describing the gravitational n-body problem is then

$$H(\mathbf{q}, \mathbf{p}) = \sum_{i=1}^{n} \frac{\mathbf{p}_i \cdot \mathbf{p}_i}{2m_i} - \sum_{i=1}^{n} \sum_{i>j}^{n} \frac{G m_i m_j}{|\mathbf{q}_i - \mathbf{q}_j|}. \tag{2.2}$$

Hamilton's equations can be applied to arrive at the equations of motion for a given system, namely

[1] Recently, Tamayo et al. (2020b) showed why it is actually possible to model weakly perturbed dissipative systems with SIAs, but this is not applicable to the work herein.

$$\frac{\mathrm{d}\mathbf{q}_i}{\mathrm{d}t} = \frac{\partial H}{\partial \mathbf{p}_i}, \; \frac{\mathrm{d}\mathbf{p}_i}{\mathrm{d}t} = -\frac{\partial H}{\partial \mathbf{q}_i}. \tag{2.3}$$

In a more compact notation,

$$\frac{\mathrm{d}}{\mathrm{d}t}\mathbf{z} = J\,\nabla_z H(\mathbf{z}), \tag{2.4}$$

where J is the "canonical structure matrix" of dimensions $2d \times 2d$ and

$$J \equiv \begin{bmatrix} 0 & +I_d \\ -I_d & 0 \end{bmatrix}$$

with I_d the $d \times d$ identity matrix.

The natural way to discuss symplectic integration is through flow maps. Given a known state $\mathbf{z}_0 = \mathbf{z}(t_0)$ then the solution $\mathbf{z}(t)$ of the system at time t will be written as $\mathbf{z}(t;\ \mathbf{z}_0)$. This notation is used to distinguish that the flow map is general for all possible states within the phase space but that a particular solution depends on the particular set of initial conditions. The flow map of the system can now be defined as

$$\boldsymbol{\phi}_t : \mathbb{R}^{2d} \mapsto \mathbb{R}^{2d},$$

such that

$$\boldsymbol{\phi}_t(\mathbf{z}_0) = \mathbf{z}(t;\ \mathbf{z}_0),$$

where the subscript t also indicates time. The flow map is a physical property of all Hamiltonian systems and a precise approximation of this mapping is what all numerical integration algorithms are trying to achieve; however, as we will see later, this becomes more explicit with SIAs. Distinct from the flow map itself is the flow map approximation, $\boldsymbol{\psi}_t$, i.e. an integration routine, such that

$$\boldsymbol{\psi}_t(\mathbf{z}_0) \approx \boldsymbol{\phi}_t(\mathbf{z}_0).$$

Two particular forms of the flow map approximation are discussed in the next section. Finally, a smooth map $\boldsymbol{\psi}$ is called a symplectic map with respect to the canonical structure matrix if its Jacobian, $\nabla\boldsymbol{\psi}$, satisfies

$$\nabla\boldsymbol{\psi}^T\, J^{-1}\, \nabla\boldsymbol{\psi} = J^{-1} \tag{2.5}$$

for all points in phase space (Leimkuhler and Reich, 2005). Therefore, the creation of a symplectic integration routine requires only that the flow map approximation be

selected such that Eq. (2.5) is satisfied up to a given order (Stuchi, 2002, p. 78) and that any individual terms within the flow map can be obtained either analytically or numerically.

2.2.2 Application to solar system dynamics

The time derivative of any variable on a Hamiltonian phase space is given by the Poisson bracket. The time derivative of the system in Eq. (2.4) is

$$\frac{\mathrm{d}\mathbf{z}}{\mathrm{d}t} = \sum_{i=1}^{n}\left(\frac{\partial \mathbf{z}}{\partial \mathbf{q}_i}\frac{\mathrm{d}\mathbf{q}_i}{\mathrm{d}t} + \frac{\partial \mathbf{z}}{\partial \mathbf{p}_i}\frac{\mathrm{d}\mathbf{p}_i}{\mathrm{d}t}\right),$$

which when combined with Eq. (2.3) can be written as

$$\frac{\mathrm{d}\mathbf{z}}{\mathrm{d}t} = \sum_{i=1}^{n}\left(\frac{\partial \mathbf{z}}{\partial \mathbf{q}_i}\frac{\partial H}{\partial \mathbf{p}_i} - \frac{\partial \mathbf{z}}{\partial \mathbf{p}_i}\frac{\partial H}{\partial \mathbf{q}_i}\right) = \{\mathbf{z}, H\},$$

where $\{\square, \square\}$ is the Poisson bracket. The Poisson bracket can also be used as an operator and is often written containing a single parameter in this case, e.g. $\{\,, H\}\,\mathbf{z} \equiv \{\mathbf{z}, H\}$.

As we will see, the trick of symplectic integrators is that they split the Hamiltonian into two or more parts such that, e.g. $H = H_A + H_B$. Defining two operators, A and B, such that $A = \{\,, H_A\}$ and $B = \{\,, H_B\}$ means that the formal solution to the equations of motion, i.e. the flow map, described by H after a single time step, h, are

$$\boldsymbol{\phi}_{t_0+h}(\mathbf{z}_0) = \mathrm{e}^{h(A+B)}\,\mathbf{z}(t_0;\ \mathbf{z}_0),$$

where A and B are evaluated at start of the interval h. Writing the formal solution to the Hamiltonian evolution in this manner does nothing to help obtain a given solution. However, presenting the flow map in this manner is favourable to deriving symplectic schemes and to understanding the errors associated with them. It is important that the splitting is chosen such that the evolution under H_A and H_B can be obtained easily in isolation from one another. In this case, the evolution owing to each operator can be applied one after the other to create a first-order approximation to the flow map, hence

$$\boldsymbol{\phi}_{t_0+h}(\mathbf{z}_0) = \mathrm{e}^{hA} \circ \mathrm{e}^{hB}\,\mathbf{z}(t_0;\ \mathbf{z}_0) + O(h) = \boldsymbol{\psi}_{t_0+h}(\mathbf{z}_0) + O(h) \tag{2.6}$$

where $\circ$ represents composition. To understand physically what this splitting procedure represents and understand the introduction of the error term, $O(h)$, it is elucidating to consider a concrete example of the splitting procedure. In particular, if we consider the n-body Hamiltonian in Eq. (2.2) then a kinetic/potential energy splitting would be such that

$$H_A = \sum_{i=1}^{n} \frac{\mathbf{p}_i \cdot \mathbf{p}_i}{2m_i}, \quad H_B = -\sum_{i=1}^{n} \sum_{i>j}^{n} \frac{G m_i m_j}{|\mathbf{q}_i - \mathbf{q}_j|}.$$

When the evolution under each of these Hamiltonians is performed independently, the updated positions, are given by $\mathrm{e}^{hA}\mathbf{q}(t_0;\ \mathbf{z}_0)$. Therefore, for a single body, denoted by subscript i, after a single step the analytical solution to the evolution under H_A is given by

$$\mathbf{q}_i(t_1) = \mathbf{q}_i + \frac{\mathbf{p}_i}{m_i} h,$$

where all terms to the right of the equals sign are taken at $t = 0$ and $t_1 = t_0 + h$. Likewise, the updated momenta are given by $\mathrm{e}^{hB}\mathbf{p}(t_0;\ \mathbf{z}_0)$, and the analytical solution to the evolution under H_B is therefore given by

$$\mathbf{p}_i(t_1) = \mathbf{p}_i - h \sum_{\substack{j=1 \\ j \neq i}}^{n} \frac{G m_i m_j}{|\mathbf{q}_i - \mathbf{q}_j|^3} \left(\mathbf{q}_i - \mathbf{q}_j\right).$$

In this case, evolving the system under H_A independently only updates the positions, $\mathbf{q}$, whilst the momenta, $\mathbf{p}$, remain constant. Likewise, evolving the system under H_B independently only updates the momenta while the positions remain constant. Therefore, the flow map approximation described by Eq. (2.6) represents a linear update to the momenta for a single step of size h followed by a similar linear update to the positions. Clearly, this evolution is not physical, but as the step size approaches zero, the linear approximations will more closely resemble the true evolution. This particular mapping is widely known as the leapfrog integrator.

Both e^{hA} and e^{hB} are symplectic mappings and evolving either the positions or momenta via them ensures that the Poincaré integral invariants are conserved for the respective Hamiltonian system, H_A or H_B. If each mapping is symplectic, then the composition of the pair of mappings must also be. Therefore, the overall composition must also conserve the Poincaré integral invariants (Tamayo et al., 2020b). It is this feature of SIAs based around symplectic composition that gives rise to the excellent long-term conservation properties typically associated with integration schemes based upon it.

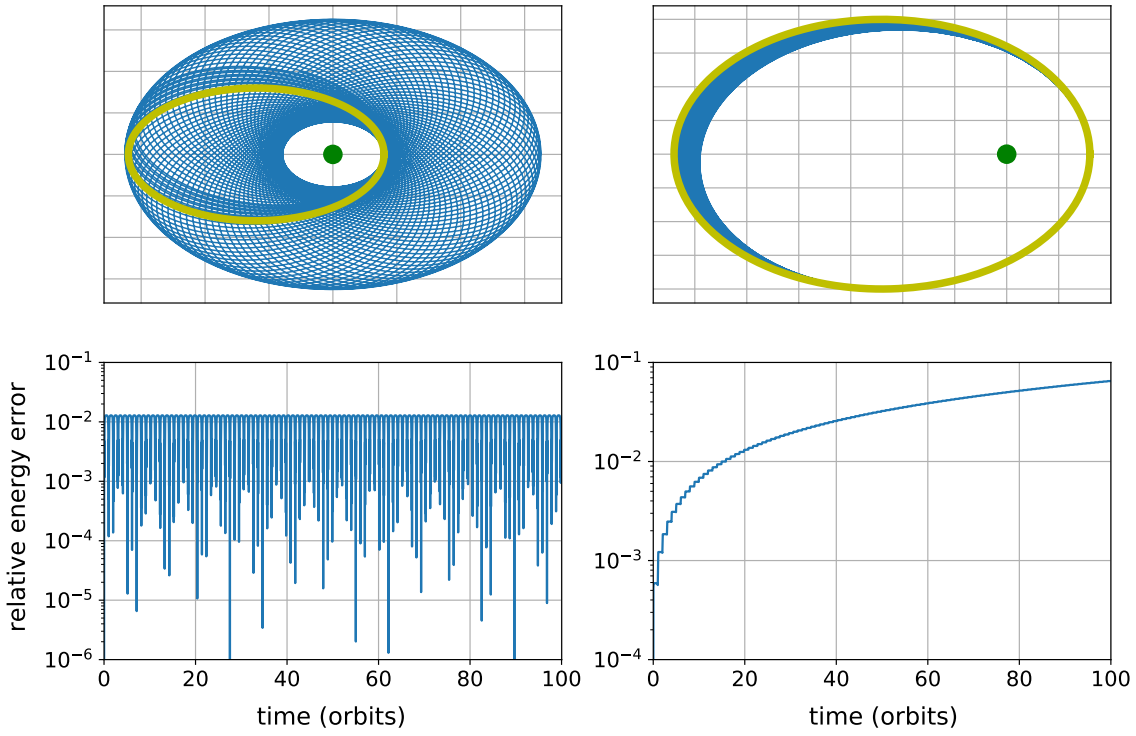


Figure 2.2: Comparison of the behaviour of the symplectic leapfrog scheme, in the left column, against the non-symplectic RK4 scheme, in the right column, when integrating a two-body problem with an eccentricity of 0.4. The top panels show how the orbit changes over a period of 100 orbits. Here, the bold yellow line shows the true trajectory and the blue lines show the integrated trajectory. The bottom panels show the relative energy error for the two schemes over the same timescale.

Operator splitting schemes evidently evolve a Hamiltonian system; however, what is less evident is what Hamiltonian system a given operator splitting method is actually evolving. To understand this, and the $O(h)$ error term in Eq. (2.6) we need to understand in what way $\mathrm{e}^{hA} \circ \mathrm{e}^{hB}$ differs from $\mathrm{e}^{h(A+B)}$, i.e. the way in which the flow map approximation, $\boldsymbol{\psi}$, differs from the flow map, $\boldsymbol{\phi}$. When using Lie derivative operators, equivalent to our Poisson brackets in this case, the Baker-Campbell-Hausdorff (BCH) identity (Steinberg, 2008) can be used to combine the exponents of non-commuting operators into a single exponent. Physically, it is understandable that evolution via e^{hA} and e^{hB} are non-commutative: performing an independent linear update to the positions and then the momenta is not the same as performing the update to the momenta first followed by the positions. Therefore, the BCH identity can be used to obtain a series expansion describing the true Hamiltonian of the system being integrated, which is commonly referred to as the shadow Hamiltonian (Rein et al., 2019a). In the case of the kinetic/potential splitting above, the shadow Hamiltonian, $\tilde{H}$, is given by (Saha and Tremaine, 1992)

$$\tilde{H} = H + \frac{h}{2}\{H_A, H_B\} + O(h^2). \tag{2.7}$$

There are several important features to note here. Firstly, the leading error term is due to the Poisson bracket of H_A with H_B which is only zero when the two Hamiltonians are equal. Secondly, the leading error term is of order $O(h)$, and decreasing the step size will therefore cause the shadow Hamiltonian to more closely approximate the true Hamiltonian. The final feature is less obvious and also highly important in motivating the rest of the work in this thesis: the presence of the step size in the shadow Hamiltonian introduces a dependency of the system being integrated on the step size used. Put another way, changing the step size during an integration will actually change the physical system being modelled and result in a shift in the energy of the system with each change. Over time, these shifts can break the symplecticity of the scheme (Chambers, 1999), and this therefore imposes the limitation that symplectic integrators are forced, with few exceptions (Duncan et al., 1998), to use a fixed step size. The inability to vary the step size from step to step during an integration is a key drawback of the symplectic schemes. Without this ability, they are unable to ensure that the acceleration is smooth within a given time step in highly dynamic regions of phase space, such as during close encounters, resulting in an inability for them to properly resolve the dynamics at the step sizes typically used.

Figure 2.2 shows a comparison between the symplectic leapfrog scheme, described above, against a very widely used non-symplectic explicit fourth-order Runge-Kutta scheme (RK4) (Press et al., 2007) for the two-body problem with an eccentricity of 0.4 where both schemes take seventy steps per orbit. In this plot, the benefits of symplectic integrators can be seen: the energy for the symplectic scheme varies by a large amount but a clear upper limit exists. Excluding errors due to floating-point precision or close encounters, this upper limit will not increase for the full duration of typical solar system dynamics integrations. In contrast, despite having a per-step energy violation below that of the leapfrog scheme, the RK4 integrator suffers from an accumulation of truncation errors that over approximately thirty orbits causes the relative energy error to surpass that of the leapfrog scheme. While the precision in this example is very low, it is this concept that makes symplectic integrators highly favourable for long-term integration. The orbital plot panel for the RK4 scheme shows that the change in energy has resulted in a change in the semi-major axis of the orbit and in a small precession. The same panel for the leapfrog shows what the variation in energy represents for the symplectic scheme: instead of a change in semi-major axis, the orbit has precessed through a full 360°. However, as the upper limit to the energy error is bounded, the semi-major axis of the orbit remains largely unchanged. The orbit followed by the leapfrog in this plot is exactly that described by the shadow Hamiltonian in Eq. (2.7).

I will end my discussion of general n-body symplectic schemes by saying that higher order operator splitting methods exist, with fourth order being particularly common (Kinoshita et al., 1990; Forest and Ruth, 1990; Yoshida, 1991), but these require more evaluations of the force per time step and are therefore more computationally expensive. As a result, these are not generally used for solar system dynamics, with the second-order Wisdom-Holman mapping, discussed next, being favoured instead.

2.2.3 Wisdom-Holman mapping

Wisdom and Holman (1991) developed a symplectic mapping for the planetary n-body problem, which is commonly referred to simply as the WH map. The WH map built upon the earlier work of Wisdom (1982) but generalised the method to a much broader region of phase space, i.e. no longer limited to studying a particular resonance or eccentricity. It allows for moderate precision integrations to be performed with no secular error growth in conserved quantities with only twenty evaluations of the force per orbit, over an order of magnitude fewer than a traditional integrator. These performance improvements allowed for the first billion year integration of the outer solar system to be performed and helped confirm the findings of Sussman and Wisdom (1988) that the motion of Pluto is chaotic. Given the highly desirable properties of this mapping, many widely used integration packages are based upon, or at least incorporate, a version of it. Such packages include: SYMBA (Levison and Duncan, 1994), MERCURY (Chambers, 1999), GENGA (Grimm and Stadel, 2014) and REBOUND (Rein and Tamayo, 2015).

In this description, I will follow the analysis by Chambers (1999) which differs from the original method for deriving the WH map. The basic premise is that instead of the Hamiltonian being split into kinetic and potential energy parts, it is split such that the dominant Keplerian motion about the central body can be evolved analytically. To enable this, so-called democratic-heliocentric coordinates are used which express positions relative to the central body and momenta relative to the barycentre. Duncan et al. (1998) provide a generating function to transform between the Hamiltonian in standard coordinates, e.g. Eq. (2.2), and that in democratic heliocentric coordinates. See Section 4.2.2 for a definition of the coordinate transformation, where the terms $\hat{\mathbf{q}}$, $\hat{\mathbf{p}}$ and $\hat{H}(\hat{\mathbf{z}})$ there are equivalent to $\mathbf{q}$, $\mathbf{p}$ and H in this section. In democratic-heliocentric coordinates, the Hamiltonian is comprised of four parts:

1. H_{kep}, the unperturbed Keplerian motion;

2. H_{star}, thus called because $-\sum_{i=1}^{n} \mathbf{p}_i$ is the barycentric momentum of the star (Duncan et al., 1998);

3. H_{pert}, the gravitational interactions between bodies;

4. H_{com}, the movement of the centre of mass;

such that $H = H_{kep} + H_{star} + H_{pert} + H_{com}$. Without loss of generality, I assume a fixed centre of mass and therefore ignore the contributions from the H_{com} part of the Hamiltonian. The particular form of the remaining terms are

$$\begin{aligned} H_{star} &= \frac{1}{2m_0}\left|\sum_{i=1}^{n}\mathbf{p}_i\right|^2, \\ H_{kep} &= \sum_{i=1}^{n}\left(\frac{\mathbf{p}_i\cdot\mathbf{p}_i}{2\,m_i} - \frac{G\,m_i\,m_0}{|\mathbf{q}_i|}\right), \\ H_{pert} &= -\sum_{i=1}^{n-1}\sum_{j=i+1}^{n}\frac{G\,m_i\,m_j}{|\mathbf{q}_i-\mathbf{q}_j|}, \end{aligned} \tag{2.8}$$

where the index zero refers to the central body, and there are n secondary bodies. In this case, the equations of motion resulting from each term in the Hamiltonian, H, can be independently integrated analytically: H_{star} and H_{pert} simply by a linear update and H_{kep} via solving Kepler's equation and applying the f and g functions (Danby, 1992, p. 162).

In keeping with the previous notation, I define three new operators A, B and C, such that $A = \{,H_{kep}\}$, $B = \{,H_{pert}\}$, and $C = \{,H_{star}\}$. Using these operators, the WH mapping is given by the composition

$$\boldsymbol{\psi}_{t_0+h}(\mathbf{z}_0) = \mathrm{e}^{\frac{h}{2}B}\circ\mathrm{e}^{\frac{h}{2}C}\circ\mathrm{e}^{hA}\circ\mathrm{e}^{\frac{h}{2}C}\circ\mathrm{e}^{\frac{h}{2}B}\,\mathbf{z}(t_0;\,\mathbf{z}_0) \tag{2.9}$$

where the error due to the Hamiltonian splitting is such that (Chambers, 1999)

$$\boldsymbol{\phi}_{t_0+h}(\mathbf{z}_0) = \boldsymbol{\psi}_{t_0+h}(\mathbf{z}_0) + O(\epsilon h^2) \tag{2.10}$$

where, provided that orbits remain well separated, $\epsilon = \sum_{i=1}^{n} m_i/m_0$. Therefore, by using this splitting the WH map is a second order symplectic mapping with an additional property that the magnitude of the leading error term depends upon the system mass ratio, ϵ, i.e. the ratio of the mass of the planets to that of the star; in our solar system

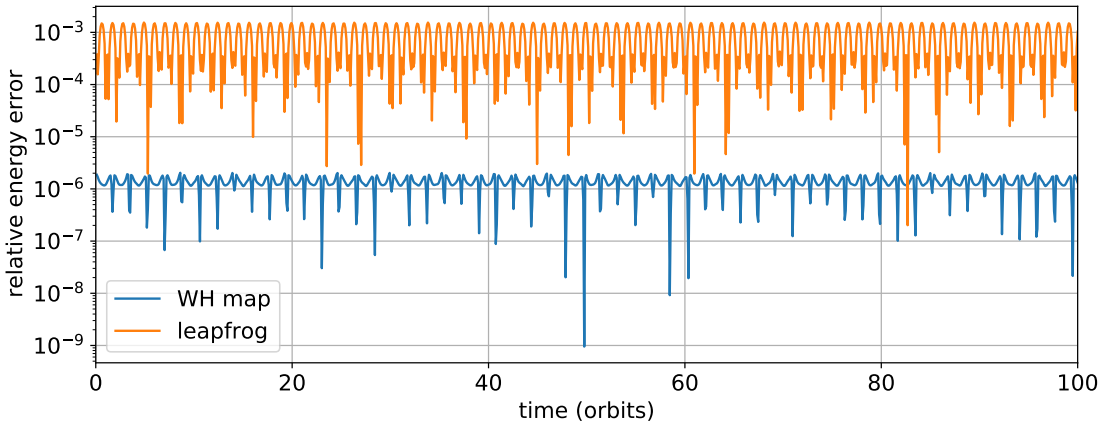


Figure 2.3: Comparison of the energy conservation of the WH map against that of the leapfrog scheme for a simulation of the outer planets of the solar system. Each integrator uses only twenty steps, and therefore twenty evaluations of the force, per Jupiter orbit.

$\epsilon \approx 10^{-3}$. This reduction in the size of the leading error term is therefore highly substantial and is also central to the performance of the scheme. The final result of this is a much lower energy error than if a simple kinetic/potential energy splitting method were used. Figure 2.3 highlight this for a simulation of the outer planets of our solar system over a hundred orbits when both integrators take twenty steps per orbit. The energy for both schemes contains bounded high frequency oscillations that are characteristic of symplectic schemes, but the relative energy error is smaller in the case of the WH map by approximately ϵ. The reduction in error, ϵ, is present because, generally, H_{kep} is much larger than either H_{pert} or H_{star}. During close encounters between planets, however, H_{pert} can become large relative to H_{kep} and, under these circumstances, the reduction in error due to ϵ is lost. Likewise, close encounters between a planet and the central body can cause an increase in the magnitude of H_{star} which also leads to a reduction in ϵ; the use of this coordinate system for modelling highly eccentric ($e > 0.99$) orbits must therefore be considered carefully. The combination of these factors with the inability to vary the step size while maintaining symplecticity means that in its original form the WH map cannot be used to model the dynamics of systems where close encounters are expected between bodies.

In his seminal tool, MERCURY, Chambers (1999) introduced the concept of a hybrid symplectic integrator to overcome this limitation, and in doing so created a scheme that is both symplectic and can also conserve energy during close encounters. To do

this, he redefined H_{kep} and H_{pert} in Eq. (2.8) as

$$H_{kep} = \sum_{i=1}^{n}\left(\frac{|\mathbf{p}_i|^2}{2\,m_i} - \frac{G\,m_i\,m_0}{|\mathbf{q}_i|}\right) - \sum_{i=1}^{n}\sum_{j=i+1}^{n}\frac{G\,m_i\,m_j}{|\mathbf{q}_i-\mathbf{q}_j|}\big[1-K(\mathbf{q}_i-\mathbf{q}_j)\big],$$
$$H_{pert} = -\sum_{i=1}^{n}\sum_{j=i+1}^{n}\frac{G\,m_i\,m_j}{|\mathbf{q}_i-\mathbf{q}_j|}\big[K(\mathbf{q}_i-\mathbf{q}_j)\big],$$

where K is an arbitrary scalar function of the separation between two bodies and is referred to as the Hamiltonian switching function. Research into switching functions is ongoing (Hernandez, 2019; Rein et al., 2019b) but the basic premise is that during a close encounter the switching function will take the large problematic terms in H_{pert} and move them into H_{kep}. This ensures that the favourable ratio of the two terms is maintained such that ϵ is still present in the leading error term in Eq.(2.10). It also ensures that the shadow Hamiltonian itself remains unchanged, i.e. this method does not require the step size used in the symplectic composition to change, and therefore that symplecticity is unbroken. Through this method the WH mapping can be extended to handle close encounters; however, the evolution due to H_{kep} can no longer be obtained analytically. Instead, it is now necessary to numerically integrate the evolution due to H_{kep} in order to obtain e^{hA} in Eq. (2.9). This integration is performed with a traditional integration scheme, such as the Bulirsch-Stoer scheme in MERCURY or IAS15 in the REBOUND implementation, MERCURIUS (Rein et al., 2019b). It is therefore less computationally efficient to use this method in the presence of many close encounters. The effectiveness of this technique for handling close encounters is discussed in Section 2.4.

As I will discuss later, it is this concept of using the known Keplerian motion present in planetary systems to reduce numerical errors that motivated me to look for ways to incorporate this knowledge into a non-symplectic integration framework.

2.3 Non-symplectic integration

I now turn my attention to the broad category of non-symplectic integration for planetary systems. Non-symplectic integration in this case simply means that the integrator was not specifically designed to preserve the symplectic structure of the phase space. This is an important distinction to make as it is entirely possible that non-symplectic schemes can preserve this property despite their design being wholly unaware of the

underlying symplectic structure of the phase spaces they are integrating. As mentioned previously, there are a large diversity of non-symplectic numerical integration techniques and many of them can be used to model general n-body dynamics. Fewer of them are capable of reaching the precision required for modelling solar system dynamics, and fewer still are capable of obtaining these solutions for a favourable computational cost. Owing to these facts, there are only a handful of schemes that have stood the test of time and are widely used by the exoplanet dynamics community. In particular, Everhart's RADAU and the Bulirsch-Stoer scheme find widespread usage while other avenues of research, such as time symmetric schemes or block time step schemes, are ongoing. Each of these options are considered in this section.

2.3.1 Everhart's Radau scheme

A non-symplectic scheme of particular note is Everhart's RADAU Everhart (1974, 1985). RADAU is an implicit multistage method that is mathematically equivalent to a Runge-Kutta scheme. However, Everhart showed that it is possible to solve the implicit set of equations efficiently using multiple iterations of a predictor-corrector scheme instead of using a more traditional method such as matrix inversion Hairer and Wanner (1991)[p. 128]. In doing this, he eliminated the costly need to calculate a Jacobian matrix and created a scheme that forty-five years later is still a particularly efficient and accurate choice for integrating astrophysical problems.

RADAU is so named because it makes use of Radau spacing (Radau, 1880) in the internal quadrature used. The use of Radau spacing in this manner allows for the order of convergence of the scheme to be increased from s or below, where s is the number of stages used, to an order of $2s-1$; this makes the algorithm particularly effective for creating high order methods where the -1 term becomes less relevant. RADAU can be used with any reasonable value of s to create various order schemes; however, 9^{th} and 15^{th} order are particularly common (Everhart, 1985; Rein and Spiegel, 2015). As such, the following discussion focuses on a 9^{th} order implementation.

As RADAU is a multistage method, it is particularly straightforward to implement a step size control algorithm and this is one of the key features that enables it to be applied to a wide range of problems such as sun-grazing comets or near-earth object dynamics (Amato et al., 2017). Despite being a highly effective algorithm in its own right, Rein and Spiegel (2015) further improved the RADAU algorithm to create a modernised implementation operating at 15^{th} order called IAS15. IAS15 incorporates three main adaptations to the original RADAU scheme. Firstly, a new step size

control algorithm is implemented. This algorithm is non-dimensional and is therefore more portable to a variety of problems. Secondly, IAS15 incorporates a new convergence criterion for the predictor/corrector scheme that enables a dynamic number of iterations to ensure the implicit solver has fully converged. This is in contrast to the original RADAU scheme whereby a fixed number of two iterations per step were used. Finally, IAS15 provides a floating-point aware algorithm that makes use of compensated summation to ensure that round-off errors accumulate as slowly as possible and that the overall scheme follows Brouwer's law for over a billion orbital periods. In the discussion below, I include the additional IAS15 algorithms but adapted to operate with a 9^{th} order quadrature.

I wish to simultaneously solve $3n$ coupled equations of the form

$$\ddot{q} = F(\mathbf{q}, t),$$

one for each directional component of n bodies where $\mathbf{q}(t) \in \mathbb{R}^{3n}$ is the position vector. To do this, RADAU expands the acceleration, $\ddot{q}$, in time, t, into a truncated series so that

$$\ddot{q} \approx \ddot{q}_0 + a_0 t + a_1 t^2 + a_2 t^3 + a_3 t^4. \tag{2.11}$$

It is prudent to rewrite this expansion in a way that is independent of the size of a particular integration step, $\mathrm{d}t$. For this, Everhart introduced $h = {}^{t}/_{\mathrm{d}t}$ and $b_i = a_i\, \mathrm{d}t^{i+1}$ such that

$$\ddot{q} \approx \ddot{q}_0 + b_0 h + b_1 h^2 + b_2 h^3 + b_3 h^4 \tag{2.12}$$

where $\mathrm{d}t$ is the size of an integration step, and $\ddot{q}_0$ is the acceleration at the start of the step. The coefficients b_i are unknown and the integration algorithm iterates to converge to their values. To do this, it is necessary to perform evaluations of $\ddot{q}$ at s locations either at the start or within an integration step. The locations are chosen according to the Radau quadrature formulas, and the locations in time at which these evaluations are performed are denoted by h_i where $i \in 0...s$. The acceleration can also be written as

$$\ddot{q} \approx \ddot{q}_0 + g_0 h + g_1 h(h-h_1) + g_2 h(h-h_1)(h-h_2) + g_3 h(h-h_1)(h-h_2)(h-h_3) \tag{2.13}$$

where, using the abbreviations $r_{ij} = 1/(h_i - h_j)$ and $r_{i1} = 1/h_i$,

$$\begin{aligned} g_0 &= (\ddot{q}_1 - \ddot{q}_0)\, r_{10}, \\ g_1 &= ((\ddot{q}_2 - \ddot{q}_0)\, r_{20} - g_1) r_{21}, \\ g_2 &= (((\ddot{q}_3 - \ddot{q}_0)\, r_{30} - g_1) r_{31} - g_2)\, r_{32}, \\ g_3 &= ((((\ddot{q}_4 - \ddot{q}_0)\, r_{40} - g_1) r_{41} - g_2)\, r_{42} - g_3) r_{43}. \end{aligned} \tag{2.14}$$

Note that in this form the values of g_i only depend upon the force evaluations at the locations h_k where $k \leq i$. Additionally, Eqs. (2.12) and (2.13) can be used with a set of recurrence relationships (Everhart, 1985) to convert between the b and g coefficients.

Once the b coefficients are known, the expansion in Eq.(2.12) can be integrated analytically with respect to h, either once or twice, to obtain an approximation to the position or velocity at any point in time during a given step. Therefore, predictions for the position and velocity are given by

$$q(h) = q_0 + h\,\mathrm{d}t\,\dot{q}_0 + h^2 \mathrm{d}t^2 \left[\frac{\ddot{q}}{2} + \frac{b_0}{6}h + \frac{b_1}{12}h^2 + \frac{b_2}{20}h^3 + \frac{b_3}{30}h^4 \right] \tag{2.15}$$

and

$$\dot{q}(h) = \dot{q}_0 + h\,\mathrm{d}t \left[\ddot{q} + \frac{b_0}{2}h + \frac{b_1}{3}h^2 + \frac{b_2}{4}h^3 + \frac{b_3}{5}h^4 \right]. \tag{2.16}$$

Additionally, at the end of a step, when $t = \mathrm{d}t$ and $h = 1$, the equations simplify and the position and velocity are

$$q_1 = q_0 + \mathrm{d}t\,\dot{q}_0 + \mathrm{d}t^2 \left(\frac{\ddot{q}}{2} + \frac{b_0}{6} + \frac{b_1}{12} + \frac{b_2}{20} + \frac{b_3}{30} \right) \tag{2.17}$$

and

$$\dot{q}_1 = \dot{q}_0 + \mathrm{d}t \left(\ddot{q} + \frac{b_0}{2} + \frac{b_1}{3} + \frac{b_2}{4} + \frac{b_3}{5} \right) \tag{2.18}$$

where q_1 and $\dot{q}_1$ are the position and velocity at the end of the step. These equations are all implicit as the values of the b's are unknown at the start of a step. To obtain the value of the b's the following algorithm is followed:

1. Predict the value of the position and velocity at a given substep, h_k, via Eqs. (2.15) and (2.16).
2. Evaluate the force using this position and velocity.
3. Update the values of g_i via Eq. (2.14).
4. Update the corresponding values of b_i.

5. Repeat for all substeps.

6. Repeat until the values of b_i have converged.

Once the b's are known with sufficient accuracy, then Eqs. (2.17) and (2.18) can be used to obtain the new state of the system at the end of a given integration step.

The IAS15 predictor/corrector convergence algorithm monitors the change in the final coefficient in the truncated series, i.e. b_3 for a scheme of this order, from one iteration to the next. I call this change $\Delta\mathbf{b}_3$ which is a vector containing coefficients for all $3n$ equations. The maximum value of $\Delta\mathbf{b}_3$ is then compared to the maximum magnitude of the acceleration at the start of the step, $\ddot{\mathbf{q}}_0$, and the algorithm terminates when

$$\frac{\|\Delta\mathbf{b}_3\|_\infty}{\|\ddot{\mathbf{q}}_0\|_\infty} < 10^{-16}.$$

Additionally, IAS15 includes an improved step size control algorithm which monitors the truncation error for all $3n$ equations. To do this, it monitors the absolute value final coefficient in the truncated series, i.e. b_3 for a scheme of this order, for all $3n$ equations, which I call $\mathbf{b}_3$. It does this relative to the acceleration at the start of an integration step to obtain an estimate of the smoothness of the acceleration over a given step as

$$\epsilon = \frac{\|\mathbf{b}_3\|_\infty}{\|\ddot{\mathbf{q}}_0\|_\infty}.$$

The value of ϵ is then used to determine the next step size, $\mathrm{d}t_{n+1}$, as

$$\mathrm{d}t_{k+1} = \mathrm{d}t_k \left(\frac{\mathrm{tol}}{\epsilon}\right)^{1/4}$$

where tol is a dimensionless tolerance parameter.

Figure 2.4 shows the final results of IAS15 in terms of energy conservation when applied to a medium-term (one million Jupiter period) simulation of the outer planets of the solar system. There are twenty realisations of the initial conditions, each randomly perturbed on the order of 10^{-15}, shown in blue. The orange line is the RMS value of each of these realisations and it can be seen that this value follows Brouwer's law which is shown as the dashed red line. Therefore, IAS15 is considered error optimal in the sense that it follows Brouwer's law. In addition, the compensated summation scheme ensures that the random walk in relative energy error does not greatly exceed the RMS value. In contrast, the original RADAU scheme is prone to random walks of as much as two orders of magnitude from the RMS value (Rein and Spiegel, 2015).

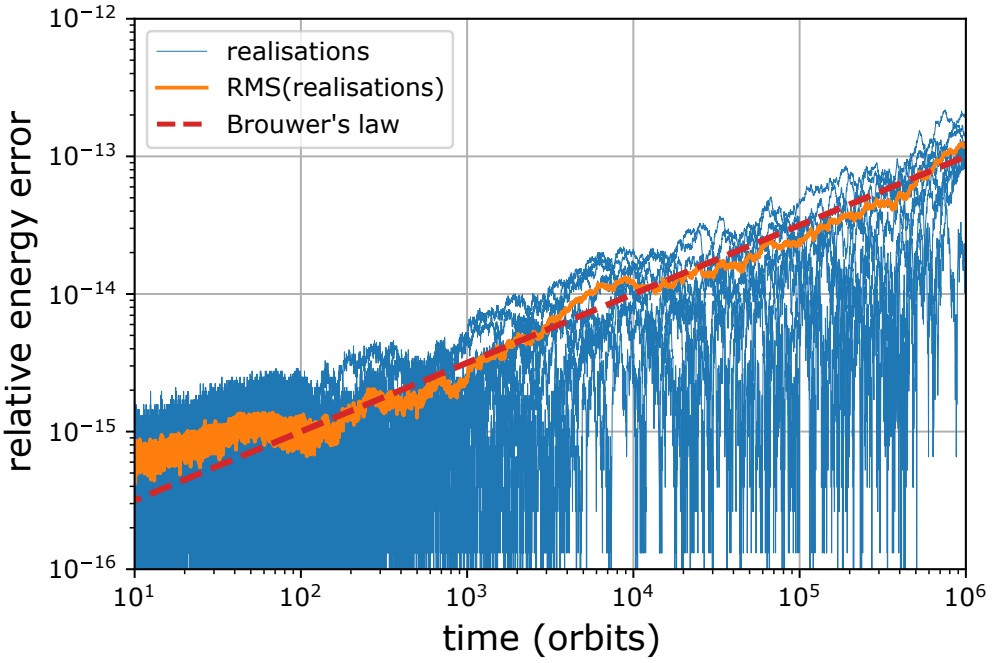


Figure 2.4: Relative energy error over time for a simulation of the outer planets of the solar system using IAS15 over a period of one million Jupiter orbits. Twenty realisations of the initial conditions are shown in blue. The orange line is the RMS of the realisations. The optimal error growth, i.e Brouwer's law, is indicated in dashed red.

2.3.2 Bulirsch-Stoer scheme

A second non-symplectic scheme that has found widespread usage in the field of modelling planetary formation is the Bulirsch and Stoer (1966) scheme. This scheme predates Everhart's RADAU by over a decade and has found popularity for its ability to handle close encounters with high precision. This is so much so that the Bulirsch-Stoer scheme is the integrator used in the MERCURY hybrid symplectic scheme during close encounters. Internally, the Bulirsch-Stoer scheme is an extrapolation technique based around the midpoint method. One of the most interesting features of the scheme is that the order of convergence within a time step can be increased dynamically to control the local integration error. Additional force evaluations are required in this case, but integration steps are not repeated with a smaller step size to achieve a given precision. This is particularly relevant to performance in the context of close encounters where it is more likely that step rejection algorithms can cause steps to be repeated if the step size control algorithm does not respond quickly enough to the changing gravitational potential. Despite being a very useful scheme historically, as will be seen in Section 2.4, the Bulirsch-Stoer scheme offers no apparent advantages over the more

modern IAS15 algorithm, and as such no further details as to its internal workings are discussed here.

2.3.3 Individual step size schemes

Non-symplectic schemes benefit greatly, both in terms of precision and runtime, from the ability to dynamically adapt the step size used throughout a simulation depending upon the current location in the phase space. In other words, the time step, $\mathrm{d}t$, at a subsequent step, $\mathrm{d}t_{k+1}$, is given by a function, τ, of the current state and, optionally, step size such that $\mathrm{d}t_{k+1} = \tau(\mathrm{d}t, \mathbf{q}, \dot{\mathbf{q}})$. As an example, in the case of a secondary orbiting in a highly eccentric orbit, the step size can be reduced at perigee when the system is at its most dynamic and then allowed to take large strides at apogee when the dynamics are easier to capture. In the general case, these schemes apply what is known as a global step size, which is to say that all particles are bound to move with a step size that is dictated by the most dynamic behaviour at a given time. As an example of why this can be inefficient, in a simulation of our solar system Neptune has an orbital period of 165 years yet is bound to take step sizes that are dictated by the orbital period of Mercury, a mere 86 days. In this example, Neptune will take 885 times more steps per orbit than Mercury and therefore a similar amount more than is necessary to capture the motion of Neptune to the same precision as Mercury.

Individual time step schemes (ITSSs) look to address this source of poor performance. To this end, they allow each body to be integrated on a separate time step, where the time step function, τ, only considers the local dynamics for each body when selecting an appropriate step size. In the example of our solar system, this would result in both Mercury and Neptune taking the same number of steps per respective orbit and would therefore cause a large decrease in the computational cost of the integration. Figure 2.5 shows the relative energy error against the computational cost, measured in runtime, for a simulation of the solar system planets over a period of 10^3 Mercury orbits. A fourth-order Hermite integration scheme (Aarseth, 2008) is used to perform fifty integrations over a range of tolerances. One set of integrations is configured to use a global step size and the other is configured to use individual time steps for each body. The individual time step scheme achieves a given precision in a much shorter runtime. At the highest levels of precision, the runtime for the global step size scheme is twice that of the individual step size scheme. These efficiency gains have led to the application of individual step size schemes to several use cases (Aarseth, 2003; Kokubo and Ida, 1996, 1998). However, their usage has not become more widespread for two reasons:

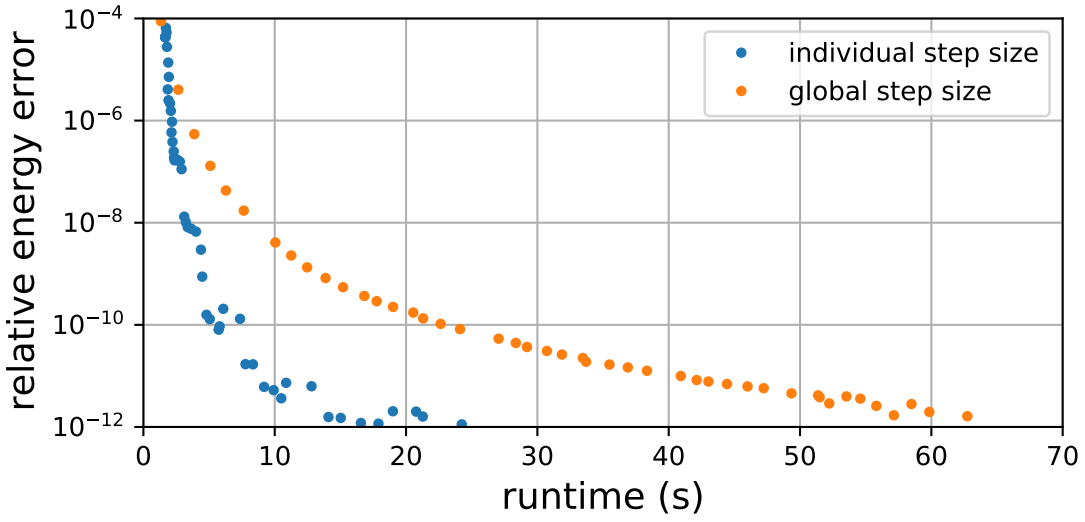


Figure 2.5: Relative energy error against computational cost for simulations of the eight planets of the solar system over 10^3 Mercury orbital periods. A fourth-order Hermite scheme was used in both cases.

1. The requirement of an individual step size for each body necessitates the use of an interpolant to capture the behaviour of slower moving bodies, and this places restrictions on the choice of integrators available.

2. Integration of bodies on individual step sizes cannot be represented by a canonical map and these schemes are therefore non-symplectic (Dehnen, 2017).

The combination of these two factors makes applying ITSSs to problems in planetary evolution challenging as one does not have the choice of using a high-order scheme nor the option to apply a symplectic mapping. Consequently, implementations of these methods usually result in a linear drift in energy, with very few exceptions (Makino et al., 2006). Before considering a potential solution to this problem, I will introduce block time step schemes (BTSSs).

BTSSs are a natural consequence of ensuring the efficiency of ITSSs and are simply an extension where the choice of available step sizes are restricted for efficiency. First, a range of appropriate step sizes for a given problem are chosen, from the smallest permissible, $\mathrm{d}t_{min}$, to the largest permissible, $\mathrm{d}t_{max}$. The ratio of $\mathrm{d}t_{max}/\mathrm{d}t_{min}$ is chosen to be a power of two, typically in the region of 2^{10} to 2^{15}. Then, each integer power of two times $\mathrm{d}t_{min}$ up to a value of $\mathrm{d}t_{max}$ becomes a permissible step size. Ergo, the choice of available step sizes is reduced to small set of 10 to 15 values. This procedure serves two purposes: reducing the computational cost of calculating predictions of

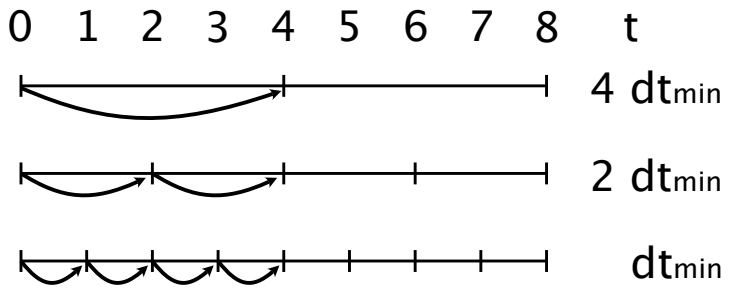


Figure 2.6: Visual representation of the hierarchical block time step scheme. Three rungs are shown here meaning that the smallest time step, dt_{min} is four times smaller than the largest.

particle positions and velocities, and allowing for approximations to gravity to be used in calculating the accelerations.

Figure 2.6 shows the step sizes allowed in a typical BTSS. In this example, there are three rungs, i.e. step sizes that bodies are permitted to take. The smallest rung represents the smallest time step permitted, dt_{min}, and each rung above is exactly twice the size of the one immediately below. Bodies are always able to move to a smaller rung but can only move to a larger rung when the body is synchronised with the rung above it, e.g. bodies on the smallest rung can only increase their step size at $t = 0, 2, 4, 6$ and 8. In this manner, it is guaranteed that the time steps of all bodies are periodically synchronised which is important for long-term conservation.

The requirement that time steps are chosen hierarchically excludes the possibility of using the majority of high-order multistage methods as the spacings of force evaluations in this case are non-uniform. Consequently, as far as I am aware, BTSSs are only used with single-stage methods, which greatly limits the choice available integrators and excludes particularly efficient schemes such as Everhart's RADAU. A commonly used integrator that can be adapted to be a BTSS is the 4^{th} order Hermite scheme. Higher order versions are available for use with BTSSs, but these versions are less efficient owing to the need to calculate higher order derivatives of the acceleration (Nitadori and Makino, 2008). In summary, the use of BTSSs excludes the possibility of using the two concepts central to precise long-term planetary system simulation: high order integration schemes, and canonical mappings. Despite these problems, the efficiency of BTSSs is alluring and work on improving their long-term conservation is ongoing. One particular avenue of active research is that of time symmetric integration (Hernandez and Bertschinger, 2018; Dehnen, 2017).

2.3.4 Time symmetry

The n-body equations of motion are symmetric with respect to time. In essence, this means that if the momentum of all bodies in a system are inverted at a point in time, then the evolution from that point in time is equivalent to if a momentum inversion had not been performed and time had been reversed instead. Mathematically, given the n-body force function

$$\dot{\mathbf{z}} = F(\mathbf{q}, \mathbf{p})$$

the equations of motion are said to be time symmetric because

$$-F(\mathbf{q}, \mathbf{p}) = F(\mathbf{q}, -\mathbf{p}).$$

Numerical integrators for n-body dynamics can also be time symmetric. Consider an integrator evolving a system forward for a given length of time from a given set of initial conditions to a set of final conditions. If time were reversed at this point and the integration were instead performed in reverse, then a scheme is considered time symmetric if the final conditions of this second integration very closely match the initial conditions of the first.

Given that the n-body equation of motion are time symmetric, it is perhaps understandable that integration methods that also respect this symmetry have long-term solutions that are similar to the true trajectory (Hairer et al., 2002). Early works with time symmetric integration (Kokubo et al., 1998; Kokubo and Makino, 2004) found that time symmetric schemes exhibit long-term behaviour similar to that of the symplectic schemes; here, the relative energy error contains no long-term drift but does exhibit the short term fluctuations synonymous with symplectic integrators. To a similar end as the symplectic shadow Hamiltonian, Hernandez and Bertschinger (2018) developed modified differential equations as a means of quantifying the errors present in a few particular time symmetric schemes. For integrations of the Hénon-Hieles problem they also found no secular energy drift. Moreover, the JANUS integrator of Rein and Tamayo (2018) achieves Brouwer's law for long-term integrations of the solar system including a modified potential approximation to general relativity.

There are many integration methods that are time symmetric over a single integration step, e.g. the leapfrog and Hermite schemes (Nitadori and Makino, 2008; Kaplan and Saygin, 2008; Kokubo et al., 1998), and also some variants of Runge-Kutta schemes (Butcher et al., 2016). However, there is a deeper problem than ensuring symmetry over a single step which is ensuring that the integrator step size is chosen in a time

symmetric manner. Succinctly, in order for an integration to be time symmetric when using an integrator that is time symmetric over a single step, evolving a system forward in time to a set of final conditions must cause integration steps to be performed at the same locations in time as if the final conditions were taken and then evolved backwards in time. This condition is met for all fixed step size schemes but is not generally the case for adaptive step size schemes. To see this, consider an arbitrary step size function, τ, that depends upon the system state at a given time such that $\mathrm{d}t_0 = \tau(\mathbf{z}_0) = \tau_0$ where the subscript 0 indicates the start of an integration step. It should be noted that although there are many options available for defining the function τ (Makino, 1991) the particular choice is unimportant in the context of this discussion on time symmetry. At the end of the integration step where $t = 1$, the time step is given by $\mathrm{d}t_1 = \tau(\mathbf{z}_1) = \tau_1$ and the integrator step size in the reverse time direction will therefore differ from that of the forward time direction as it depends upon the state at $t = 1$. Hut et al. (1995) proposed a straightforward solution to this problem: the choice of time step must consider both values of $\mathrm{d}t$. They proposed selecting

$$\mathrm{d}t_{1/2} = \frac{\tau_0 + \tau_1}{2} \tag{2.19}$$

such that the choice of time step, $\mathrm{d}t_{1/2}$, is identical in both directions in time; the notation $\mathrm{d}t_{1/2}$ indicates that the time step is time symmetrical. Therefore, the choice of a time step has become implicit as it requires knowledge of the state of the system at the end of a step, $\mathbf{z}_1$. Dehnen (2017) proposed a number of potential ways to address this implicitness when using BTSSs, including extrapolating τ_0 and implementing a try and reject strategy that ensures the condition in Eq. (2.19) is met. Whilst these method do in fact reduce the number of non-symmetric time step selections for BTSSs they do not remove them completely. Moreover, several of the proposed solutions necessitate a smooth time step function which is not always the case when modelling planetary systems, e.g. when close encounters are taking place. Therefore, the inability for time symmetry to be applied to BTSSs means that the use of BTSSs is not considered any further.

2.4 Quantifying integrator performance for solar system dynamics

This chapter has laid out the considerable diversity in approaches to accurate computationally efficient long-term n-body integration in the context of solar system dynamics. The benefits and drawbacks associated with each scheme have been presented in

isolation such that their importance to the field can be appreciated. There are four schemes that have been identified as being seminal: RADAU/IAS15, Bulirsch-Stoer, WH mapping, and hybrid. Henceforth, the version of the WH map used is the implementation in MERCURY and is referred to as the mixed variable symplectic (MVS) scheme. All that remains to be done is to benchmark all of these schemes against one another in the context of planetary dynamics modelling. High order symplectic methods that incorporate symplectic correctors are also widely used; however, as they are not suitable for modelling systems that will experience close encounters they have been excluded from the following benchmarks. As per the introduction to this chapter, there are three requirements for benchmarking planetary dynamics models, these are:

1. Ensuring that solutions obtained remain accurate over the timescales required, in solar system formation and stability studies typically 10^9 dynamical periods.
2. Ensuring that integrators can precisely model close encounters between objects.
3. Ensuring that such long duration simulations can be completed within the available computing time.

To compare the performance of each scheme in the context of the requirements above, three experiments have been performed:

1. A short-term integration of the outer planets of the solar system across a range of tolerances to understand the precision/computational cost trade-offs.
2. A long-term integration of the outer planets of the solar system to understand the long-term error growth in conserved quantities.
3. A simulation containing close approaches between Earth-mass planets to discern the performance in these challenging regions of phase space.

2.4.1 Short-term simulations of the outer planets of the solar system.

In this experiment, the outer planets of the solar system are propagated using the aforementioned integration schemes. The initial conditions of the planets are taken from the NASA horizons (NASA, 2021) database and the simulation is run for a duration of 10^3 Jupiter periods. Integrations are performed with each scheme across a

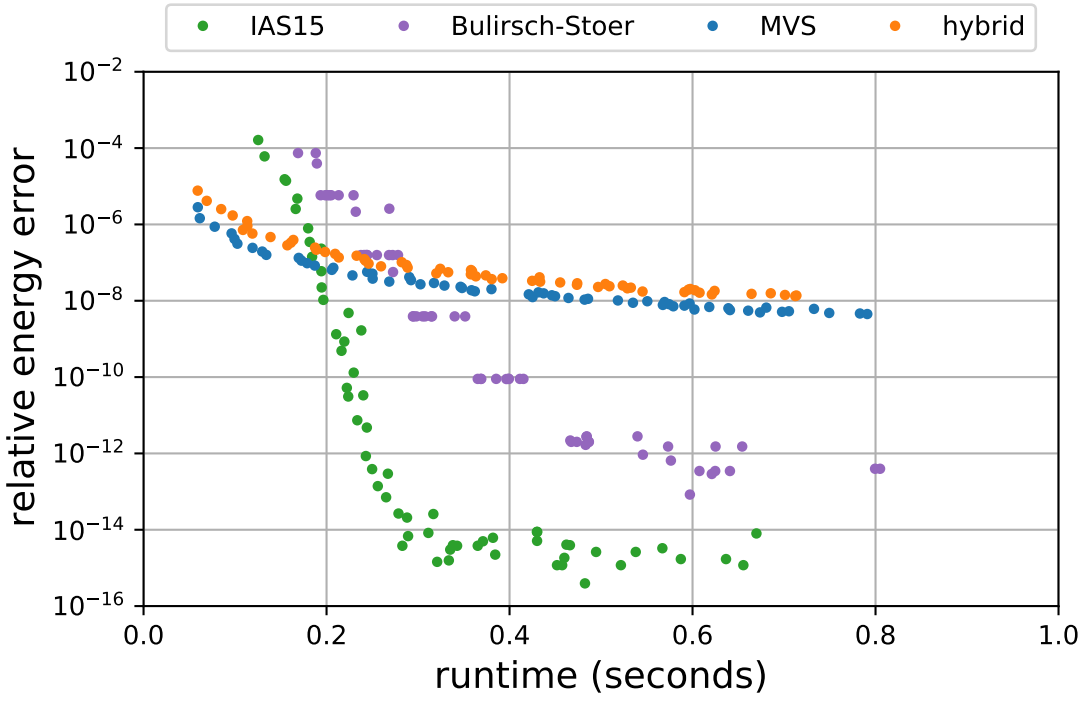


Figure 2.7: Relative energy error against computational cost for simulations of the outer planets of the solar system over 10^3 Jupiter orbital periods.

range of tolerances or step sizes. In the case of the fixed step size schemes, MVS and hybrid, the steps chosen range from twenty to two hundred and fifty per Jupiter orbit, where, in practice, twenty steps per orbit is a common use case (Lissauer and Gavino, 2021). Symplectic correctors are not used as they will later on preclude the study of close encounters. The hybrid scheme tolerance for the non-symplectic portion of the scheme is configured with a tolerance of 10^{-14}. The variable step size schemes, IAS15 and Bulirsch-Stoer, are varied across a variety of tolerances such that solutions range from highly inaccurate to the precision being dominated purely by round off error. In the case of IAS15 this means to a maximum tolerance matching the recommended operating value of 10^{-9}.

Figure 2.7 shows the relative energy error against the runtime for this experiment. Immediately, the difference between the symplectic and non-symplectic schemes are apparent: the non-symplectic schemes are much more precise for a given runtime over most of the tolerance range. As an example, for a runtime of 0.5 seconds, IAS15 is over a million times more precise in terms of energy conservation. Bulirsch-Stoer also performs well on this short term integration reaching relative energy error levels of 10^{-12} in runtimes that are roughly twice that of IAS15. For short term experiments

of this kind, the symplectic schemes are most applicable to low precision integrations, e.g. in the region where the runtime is below approximately 0.2. However, for long-term integrations the step size required to ensure favourable error growth for the entire duration of simulations must also be considered. In the symplectic cases, the furthest left data point is generated by performing an integration using the recommended default step size for most simulations. Whereas, in the non-symplectic cases, the furthest right data point is generated by performing an integration with the recommended operating tolerance. In this context, the benefits of the symplectic schemes can be seen: the computational cost per orbit is much smaller when long-term energy error growth is to be considered. In the next section, I will consider the final conservation levels achieved by these schemes using these default tolerances for long-term simulations. Finally, to add a more practical context to the runtimes with default settings, IAS15 will complete a billion-year integration of the outer solar system in approximately a week on a moderately powerful workstation, whereas the MVS scheme will take roughly a day to perform the same task. Both of these timescales are potentially acceptable for performing simulations, however, the use of a non-symplectic integrator needs to be justified accordingly.

2.4.2 Long-term simulations of the outer planets of the solar system.

Figure 2.8 shows the results of long-term integrations of the outer planets of the solar system over a period of 10^8 Jupiter orbits. Results are included for the four aforementioned integration schemes, and two additional slopes are included indicating optimal, $\sqrt{t}$, error growth and $\propto t$ error growth in dashed red and dashed grey, respectively. All schemes perform well, with a maximum relative energy error across all integrators of approximately one part in a million. However, the way in which each scheme reaches the final energy error is different for each family of schemes: the symplectic schemes exhibit no long-term energy drift, whereas the non-symplectic schemes do. The Bulirsch-Stoer scheme exhibits error growth $\propto t$ and arrives at a final energy violation approximately two orders of magnitude below the two symplectic schemes; however, as we saw previously, the computational cost for obtaining this increased precision is an increase of a factor of ten. IAS15 can be seen to be the best performer overall with an energy conservation a million times better than the symplectic schemes and, as we also saw previously, with a superior runtime when compared to the Bulirsch-Stoer scheme. Therefore, there appears to be little benefit

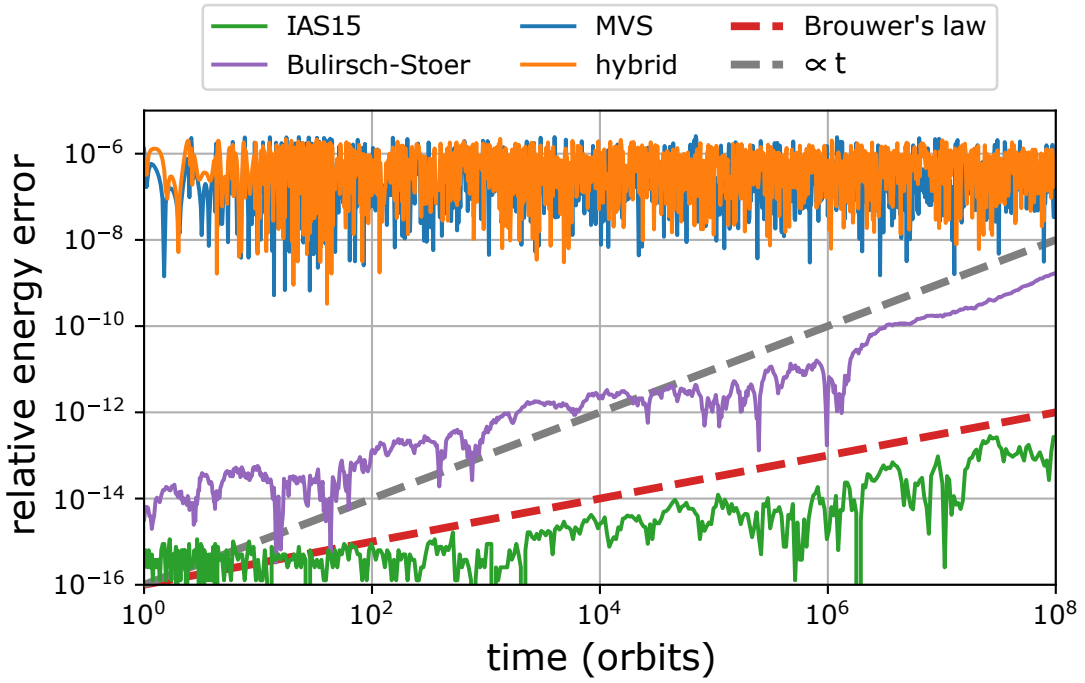
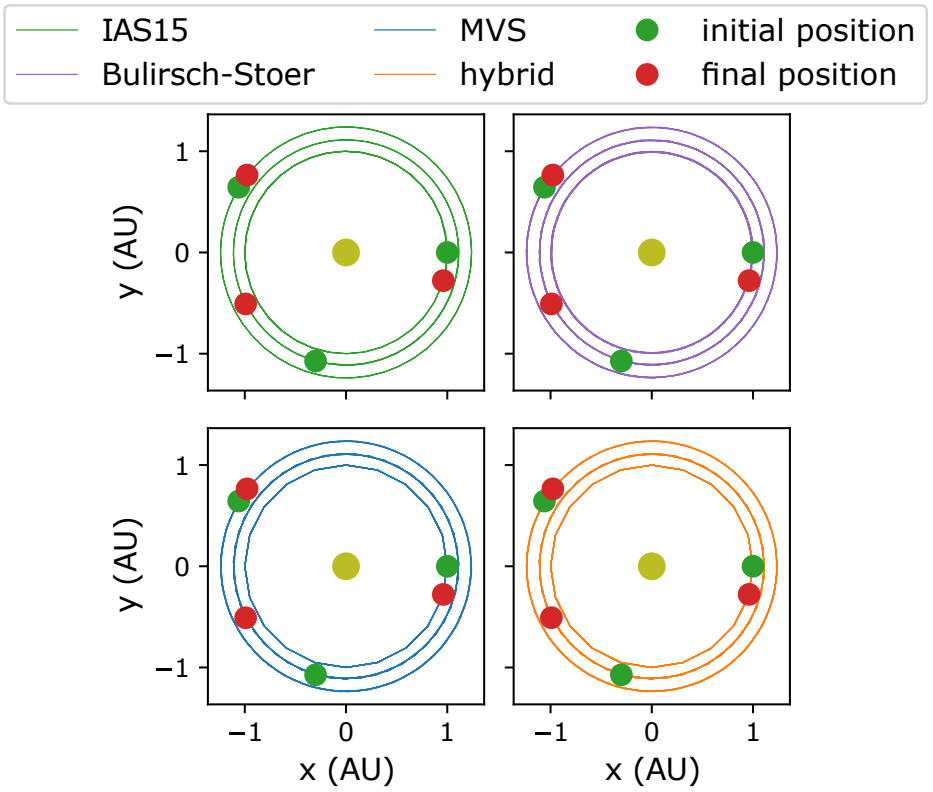


Figure 2.8: Relative energy error against time for simulations of the outer planets of the solar system over 10^8 Jupiter orbital periods. Both MVS and the hybrid scheme take twenty steps per Jupiter orbit. IAS15 and Bulirsch-Stoer use tolerances of 10^{-9} and 10^{-15}, respectively. The slopes show optimal error growth ($\sqrt{t}$) and $\propto t$ error growth in dashed red and dashed grey, respectively.

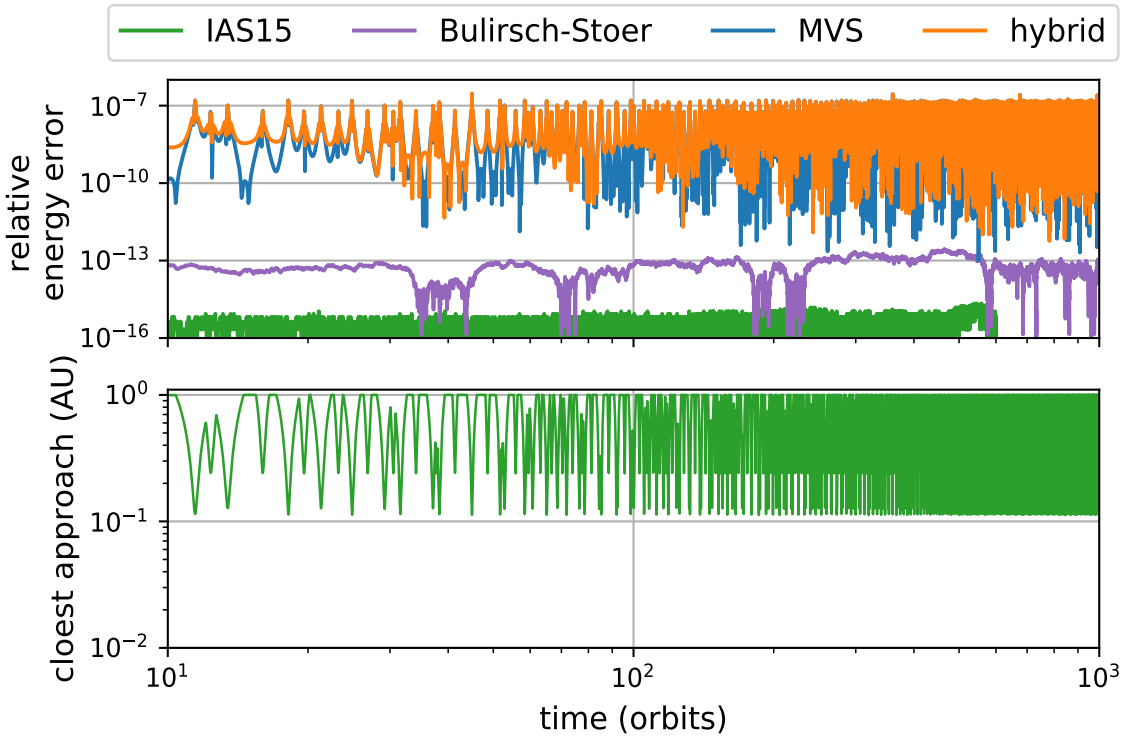
to using non-symplectic schemes that have a error growth $\propto t$ for long-term simulations of solar system dynamics. However, if a non-symplectic scheme can be made optimal in the sense that it follows Brouwer's law, then there are great precision gains to be made for long-term evolution of planetary systems compared to symplectic integration schemes.

2.4.3 The effects of planet-planet encounters on symplectically obtained solutions

The next experiment forms the basis of an argument for the use of non-symplectic schemes when modelling close encounters. This experiment is performed using a three-planet system composed of Earth-mass planets orbiting a solar-mass star. Planets are on initially circular co-planar orbits and the innermost planet is placed at 1 AU. The remaining two are given a semi-major axis such that the period ratio between each adjacent planet is identical, and the initial longitudes are chosen according to the golden ratio to avoid special configurations Smith and Lissauer (2009). The last

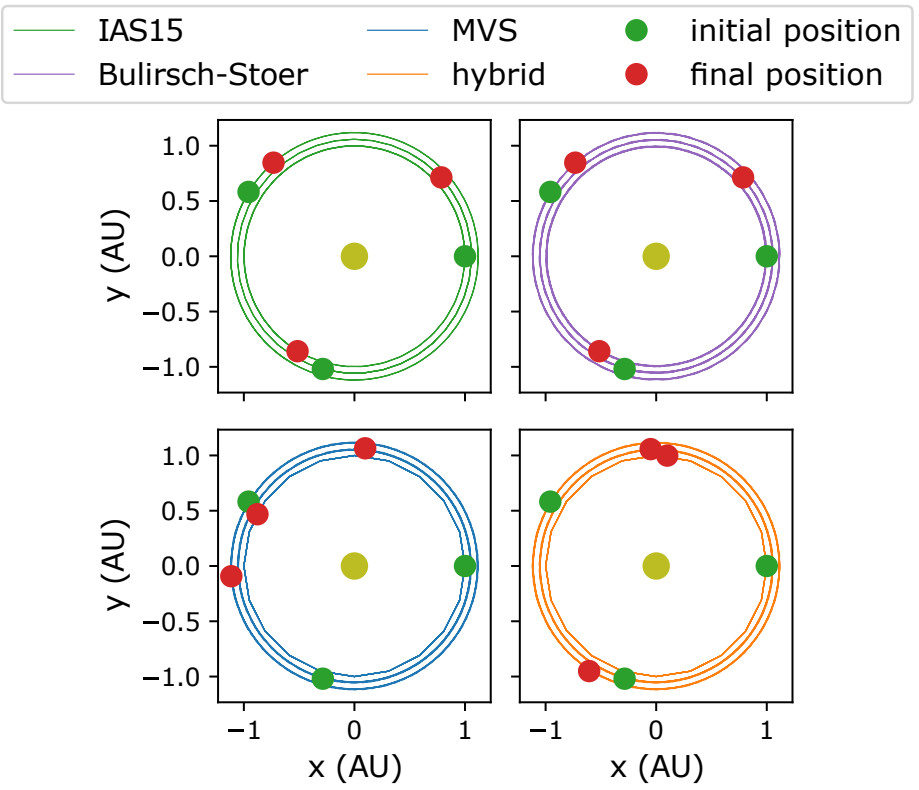


(a) Orbital diagrams resulting from integrations using the various integration routines indicated. The initial and final positions are marked with a green and red circle, respectively.

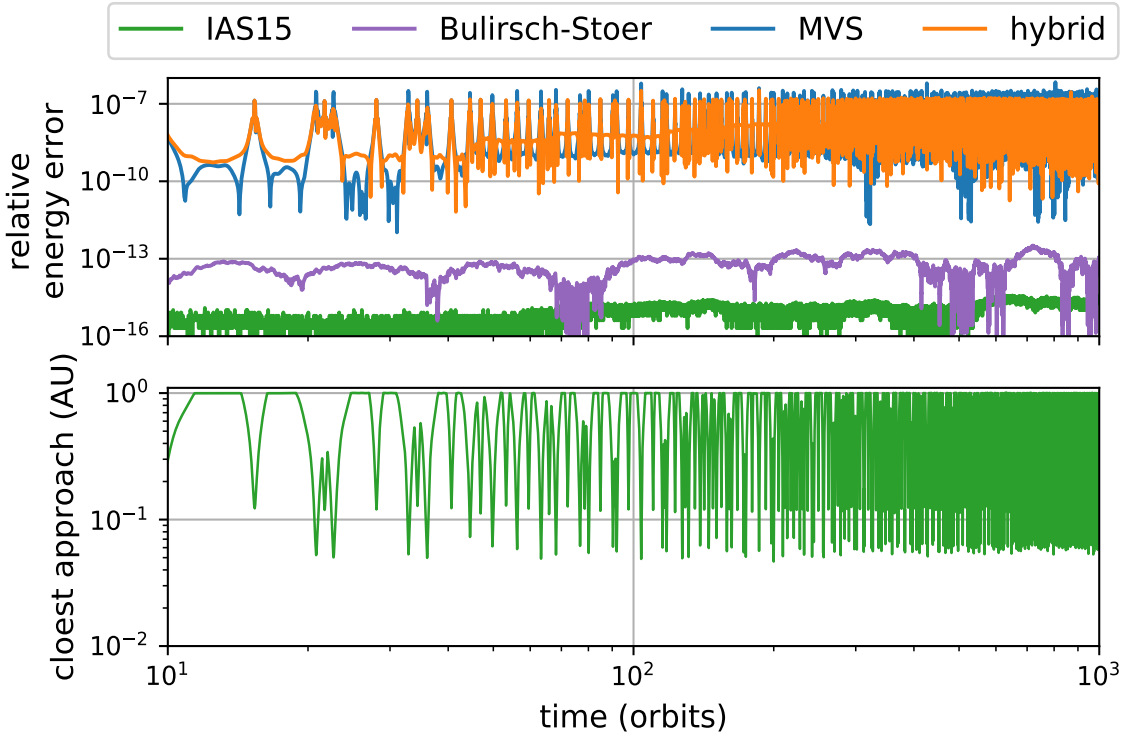


(b) The top panel shows the relative energy error over time throughout the simulations for the integrators indicated. The bottom panel shows the minimum distance between any two bodies over time.

Figure 2.9: Orbital diagrams, relative energy error and closest approach for simulations of widely-spaced, co-planar, three-planet, Earth-mass systems orbiting a solar-mass star over a period of one thousand orbital periods of the innermost planet.



(a) Orbital diagrams resulting from integrations using the various integration routines indicated. The initial and final positions are marked with a green and red circle, respectively.



(b) The top panel shows the relative energy error over time throughout the simulations for the integrators indicated. The bottom panel shows the minimum distance between any two bodies over time.

Figure 2.10: Orbital diagrams, relative energy error and closest approach for simulations of closely-spaced, co-planar, three-planet, Earth-mass systems orbiting a solar-mass star over a period of one thousand orbital periods of the innermost planet.

parameter to consider is the particular value of period ratio between planets, and for this experiment I have chosen two ratios. The first is 1.17 and leads to a system that is stable for over a hundred million orbital periods (defined as two planets approaching one another within one Hill radius); I refer to this configuration as the widely spaced system. I refer to the second configuration as the closely spaced system, which has a period ratio of adjacent planets of 1.06 and leads to a system that is only stable for a few thousand orbits. These values of the adjacent period ratio are chosen such that close encounters occur between planets at a minimum distance less than ten Hill radii in the case of the closely spaced system and greater than ten Hill radii in the case of the widely spaced system.

Figures 2.9 and 2.10 contain the results for the widely spaced and closely spaced systems, respectively. First, let us consider the widely spaced system where Fig. 2.9b shows the energy conservation during the integration for the four integration schemes. Additionally, the bottom panel here shows the minimum distance between bodies in the system and shows that no bodies come within 0.1 AU during the simulation. The energy conservation for all schemes can be seen to be quite favourable despite the clear divide between the symplectic and non-symplectic schemes. Finally, Fig 2.9a shows the resulting orbital diagrams, which are all identical for the four integration schemes. Furthermore, the initial and final positions are also marked on these diagrams, which shows that the final positions of the planets obtained are consistent between all four integration schemes. Despite not knowing what the true final positions of the planets are overall, the fact that each of the schemes has converged to the same solution gives confidence in its accuracy. Importantly, the fact that the symplectic schemes obtain solutions similar to the non-symplectic schemes which are much more precise by the metric of energy conservation, shows that they are highly capable when bodies are well separated.

Next, let us consider the closely spaced system. Figure 2.10 shows that for this set of initial conditions the closest approach is now approximately 0.05 AU, or roughly 5 Hill radii. Looking at the relative energy error plot, one can see little difference compared to the widely spaced system. The upper limit on the truncation error for the symplectic schemes is very slightly higher, but otherwise the behaviour is qualitatively identical. Figure 2.10a contains the orbital diagrams for the closely spaced system obtained with each integrator. The energy conservation for all schemes leads to the expectation that the orbital traces should look identical, and indeed they do. Moreover, the final positions of the non-symplectic schemes are consistent with one another, indicating the accuracy of the positions obtained. In contrast, the final positions of planets obtained by the symplectic integrators are not only different from the

non-symplectic final positions but they are distinct from one another as well. Given that the orbital geometry remains unchanged, there is only one possible reason for this discrepancy in the final positions: during even moderate close encounters, the phase of the planets being integrated is non-physically able to change. In general, the long timescales of simulations compared to the e-folding timescale of planetary systems means that the chaotic dynamics will cause a final position error comparable to the phase change as a result of moderately close encounters (Hernandez et al., 2021). However, there are cases where these phase changes are unacceptable, e.g. when evolving the solar system to understand the probability of asteroid impacts with the Earth. A second key area where they are unacceptable is when considering the time taken for impacts between planetary bodies to occur. Rice et al. (2018) performed a study of an array of exoplanet systems in the period of time after an instability event leading up to a collision between planets. In this work, they recognised that hybrid symplectic schemes are not adequate to capture the complex behaviour of planetary systems in the presence of repeated close encounters between Neptune-mass bodies, and they instead opted to use the Bulirsch-Stoer scheme to more accurately capture the dynamics. In Chapter 5, I perform a similar study to that of Rice et al. (2018) where, in accordance with the finding of this section, I have opted not to use a hybrid symplectic scheme and instead use a bespoke non-symplectic model.

Chapter 3

An interlude into multistep collocation methods

The content of this chapter is based upon an article in preparation for submission to the SIAM Journal of Scientific Computing. The authors of the article are Peter Bartram, Hodei Urrutxua and Alexander Wittig. I am responsible for the vast majority of the work in the article. However, the final choice of predictor and the analysis that led to this design decision are both thanks to Hodei Urrutxua; this analysis is also included here for completeness.

The title of this chapter was chosen to indicate that the work contained in it is somewhat tangential to the overall story told by this thesis. I use the word "somewhat" as the original hope was that multistep collocation methods could be efficiently applied to problems in exoplanet science. During our analysis, we discovered that these methods are best applied to problems that are globally stiff which, unfortunately, does not include the dynamics of planetary systems. As a result, MCM are unable to compete with other available methods in terms of run time, despite being able to obtain precise solutions to planetary motion. However, the analysis of these methods as applied to other problems is very interesting in its own right and therefore I have chosen to devote this chapter to presenting the findings. The integration technique that I ultimately used for modelling exoplanet dynamics can be found by proceeding to Chapter 4.

3.1 Background

Differential equations are ubiquitous in the description of physical phenomena and are not just limited to the planetary n-body problem. Their use across myriad fields is responsible for motivating a perpetual search for ever more accurate and efficient algorithms. Chapter 2 showed that one potential method for ensuring accurate solutions is the use of high-order methods, with a 15^{th} order scheme being particularly effective for two key reasons. Firstly, schemes of this nature are of a low enough order that the step size required is still a fraction of the dynamic timescale of orbital motion problems. Secondly, a high order method ensures that the number of steps taken per orbit is kept to a minimum which therefore reduces the effects of round off errors. To illustrate how schemes of roughly 15^{th} order address these two points, consider a two-body problem with an orbital period of 1 in some units. If an integration step is performed with a step size $h_{trial} = 1$ and the relative energy error, ϵ_{trial}, obtained is roughly of order unity then it is straightforward to determine what the error would be for a repeated integration step performed with a smaller step size $h_{new} = 0.1$. In this case, the relative energy error, ϵ_{new} after a single step of magnitude h_{new} is

$$\epsilon_{new} = \epsilon_{trial}\left(\frac{h_{new}}{h_{trial}}\right)^{15} = 1 \times 10^{-15} \approx \epsilon_{machine}. \tag{3.1}$$

Therefore, for methods of roughly 15^{th} order it is possible to reach machine precision by increasing the number of steps taken by only an order of magnitude thereby helping to minimise round off errors. These potential benefits have therefore prompted the analysis in this chapter into multistep collocation methods which allow for a richness of high-order schemes to be created.

All integration methods necessitate procurement of information about the behaviour of the differential equation in the nearby region of phase space. Typically, there are two information sources that can be exploited to obtain this information: firstly, through calculating higher order derivatives of the differential equation, or, secondly, one can probe the right-hand-side of the differential equation in the nearby region of phase space; this chapter focuses on the latter source of information. In this case, there are two main approaches:

1. Multistage methods perform evaluations of the right-hand-side within the current integration step.

2. Multistep methods use information from past solution points to inform the future behaviour.

Typically, schemes draw information from only one of these two sources. For example, Runge-Kutta schemes are purely multistage whereas the Adams-Bashforth scheme is purely multistep. A logical question is therefore: is it possible to combine these two sources of information in such a way as to create a scheme that can benefit from the advantages of each family of schemes?

General Linear Methods is the name given to the formalism under which these blended schemes fall, and it should be noted that this class is general enough to include purely multistage and purely multistep methods as well (Hairer and Wanner, 1991). A particular family of schemes that fall under this umbrella are known as the Multistep Collocation Methods (MCM). These methods are highly general as information can be utilised from any number of past solution points and with any number of inter-step evaluations of the right-hand-side. Therefore, these schemes can conceptually be seen as being composed of two parts: an explicit multi-step method and an implicit multi-stage Runge-Kutta method. Moreover, Radau spacings introduced in Chapter 2 enable MCM integration methods to be created that are of the order $(2s+k-2)$ where k is the number of past solution points considered and s is the number of inter-step evaluations of the right-hand-side made (Lie and Norsett, 1989). In addition to the favourable order properties, MCM integration methods also possess interesting stability properties making them particularly applicable to "stiff" problems. Specifically, MCM methods allow for past solution points to be used to increase the overall order of the scheme while still retaining the excellent stability properties of implicit Runge-Kutta methods. Each combination of $k-s$ pairing leads to a method with a distinct stability domain and computational performance metrics.

Despite all of the factors in their favour, integrators based upon MCM seem to remain of little practical interest to the broader community. The diversity owing to the degrees of freedom in the configuration space appears to have been overlooked, despite the versatility it offers. In the literature, only the case for $s = 3$ with variable k appears to have been investigated as an archetypal example of this family of methods (Hairer and Wanner, 1991). No evidence was found of any study that systematically considered other configurations. The work in this chapter attempts to fill this gap by exploring the k-s parameter space of Radau type MCM integration schemes. In particular, it considers how the performance varies for different configurations, and works to identify the causes for such variations. Heatmaps and integrator performance curves are used in the analysis, and conclusions are drawn from different performance metrics. In particular, it is found that the predictor has a notable impact on the performance of high-order MCM.

This rest of the chapter is structured as follows: Section 3.2 introduces preliminary concepts on collocation methods, so that a comprehensive derivation of MCM can be addressed in Section 3.3. Section 3.4 presents a dedicated discussion on the key practical issue of solving the implicit part of the integration method. Section 3.3.3 presents the absolute stability domains of these methods. In Section 3.5 test problems are presented (both stiff and non-stiff) for the sake of performing numerical experiments; technical and procedural aspects are clarified for the sake of reproducibility; analysis tools are presented and results are analysed. Finally, the findings are summarised in Section 3.6.

3.2 Preliminaries on collocation methods

The following initial value problem (IVP) is considered

$$\frac{\mathrm{d}y}{\mathrm{d}x} = f(x,y), \qquad y(x_0) = y_0 \tag{3.2}$$

where the function f depends, in a more general case, on both the independent integration variable, x, as well as the solution $y(x)$. When the IVP needs to be solved numerically, the solution $y(x)$ is an approximate solution to the differential equation at every discrete value of the variable x.

It is during the two decades following the publication of Radau's memoirs (Radau, 1880) that the celebrated Runge-Kutta (RK) numerical integration processes were developed. These methods yield families of integrators, both implicit and explicit, and also of varying orders, all of which can be characterised by two governing equations, namely:

$$g_i = y_n + h\sum_{j=1}^{s} a_{ij} f(x_n + c_i h, g_j), \qquad i = 1,2,...,s$$

$$y_{n+1} = y_n + h\sum_{i=1}^{s} b_i f(x_n + c_i h, g_i)$$

where y_n is the solution at the integration step x_n, h is hereafter referred to as the stepsize, and the two sets of coefficients a_{ij} and b_i define a particular instance of RK integrator. The standard method of representing the a_{ij} and b_i values is the Butcher's tableau, popularised by its namesake (Butcher, 1996). Also, a property of RK methods

is that coefficients c_i satisfy

$$c_i = \sum_{j=1}^{s} a_{ij}, \qquad i = 1, 2, ..., s. \tag{3.3}$$

Coefficients a_{ij} and b_i are normally determined by imposing order conditions, which yields nonlinear algebraic relationships that can then be solved to provide families of values of the aforementioned coefficients (Butcher, 2008); once these are known, Eq. (3.3) explicitly leads to the locations at which to sample the right-hand side function $f(x, y)$. This would seem to preclude benefiting from the increased order of a method through the selection of c_i coefficients using Radau spacings; however, the process of finding c_i from a_{ij} as per Eq. (3.3) can be reversed for an implicit Runge-Kutta (IRK) scheme when Radau spacings are used. By selecting $c_0, ..., c_s$ in accordance with Radau quadrature nodes, it is then possible to compute the corresponding a_{ij} and b_i coefficients from the following equations (Hairer and Wanner, 1999):

$$a_{ij} = \int_0^{c_i} \prod_{\substack{k=1 \\ k \neq j}}^{s} \frac{(t - c_k)}{(c_j - c_k)} \, \mathrm{d}t, \qquad b_j = \int_0^{1} \prod_{\substack{k=1 \\ k \neq j}}^{s} \frac{(t - c_k)}{(c_j - c_k)} \, \mathrm{d}t, \qquad i, j = 1, 2, ..., s.$$

Here, the coefficients can be determined so readily because Radau spacings allow for this IRK method to be expressed as a collocation method. This technique of preselecting the sampling locations and then building the integration method around these prescribed spacings is applied to multistep methods in the following section.

3.3 Multistep collocation methods (MCM)

Collocation methods are used consistently in numerical analysis and involve the determination of a polynomial $u(x)$ of degree s whose derivative $u'(x)$ coincides at s points with the vector field of the differential equation within a given interval. The derivative is said to "co-locate" with the differential equation vector field. The fact that $u'(x)$ matches the vector field of the differential equation up to order s means that, at the sampling locations, the shape of the $u(x)$ matches that of the integral of the differential equation; when this is combined with a known initial value for y, it becomes possible to accurately approximate the solution of the IVP (Eq. (3.2)). The idea of a MCM is that of not only matching the vector field within the current step, but also matching the solution at past integration steps, thus yielding a higher-order numerical integration method.

Let s represent the number of stages, and let k represent the number of previous steps to use. Then, let s real coefficients $c_i \in [0,1]$ be specified for $i = 1,...,s$, and k previous solution values $y_n, y_{n-1}, .., y_{n-k+1}$ be known. The collocation polynomial is then defined as

$$u(x_j) = y_j \qquad j = n-k+1,...,n \tag{3.4}$$

$$u'(x_n + c_i h) = f(x_n + c_i h, u(x_n + c_i h)) \qquad i = 1,...,s \tag{3.5}$$

where $u(x)$ coincides with the approximate solution y_j at the previous integration steps, and whose derivative $u'(x)$ coincides with the vector field $f(x,y)$ at s points within the interval $[x_n, x_n + h]$. The numerical solution at the new step is then obtained by evaluating the collocation polynomial at $x_{n+1} = x_n + h$, namely $y_{n+1} = u(x_{n+1})$.

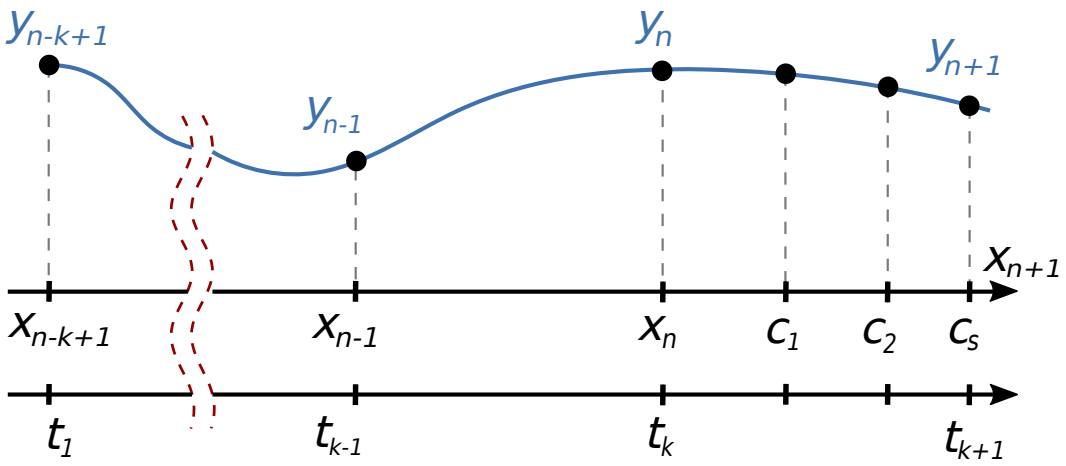


Figure 3.1: Collocation polynomial. Solution points are indicated by y_i. Collocation points are marked as C_i. The independent variable is X_i, and t_i is a non-dimensionalised version of this.

For ease of notation, the dimensionless coordinate $t = (x - x_n)/h$, can be introduced, so that $t \in [0,1]$ represents values within the current stepsize, and previous steps correspond to non-positive values of t. By doing so, $x = x_n + t\,h$, and in the case of fixed-stepsize, t takes integer values at the previous k steps, namely $t_k = 0, t_{k-1} = -1, ..., t_1 = -k+1$. The collocation strategy is illustrated in Fig. 3.1.

3.3.1 Radau spacings

As was shown previously, the sampling locations of a quadrature technique have a profound impact on the overall precision of the method; this subsection looks at how this principle can be extended for use within a MCM.

According to (Lie and Norsett, 1989), it is possible to construct General Linear Methods (GLM) of order $p = 2s+k-2$ that are "stiffly stable" so long as the value of $c_s = 1$ is fixed, and the other sampling locations $c_1, ..., c_{s-1}$ are chosen optimally. Given any prescribed set of stages, it is possible to define the fundamental interpolating polynomials $\chi_i(t), i = 1, ..., s$, which vanish at every previous step and all but one stage sampling locations, namely:

$$\chi_i(t_j) = 0, \qquad j = 1, ..., k, \qquad \chi_i(c_j) = \begin{cases} 1 & \text{if } i = j \\ 0 & \text{if } i \neq j \end{cases} \qquad j = 1, ..., s \tag{3.6}$$

These sampling locations will only be optimal (i.e. the order of the MCM will be maximised) so long as the fundamental interpolating polynomials also satisfy the following additional condition (Hairer and Wanner, 1991, p. 294):

$$\chi_i'(c_j) = 0, \qquad j = 1, ..., s. \tag{3.7}$$

These fundamental interpolating polynomials take the following form

$$\chi_i(t) = K_i \prod_{j=1}^{k} (t - t_j) \prod_{\substack{j=1 \\ j \neq i}}^{s} (t - c_j)^2, \qquad i = 1, ..., s \tag{3.8}$$

where K_i are determined from $\chi_i(c_i) = 1$. Hence, polynomials $\chi_i(t)$ readily satisfy the conditions of Eq. (3.6) for any set coefficients c_i. Enforcing that they also satisfy Eq. (3.7) yields a system of equations whose solution defines the optimal stage sampling locations for any MCM configuration of k steps and s stages:

$$\sum_{j=1}^{k} \frac{1}{c_i - t_j} + \sum_{\substack{j=1 \\ j \neq i}}^{s} \frac{2}{c_i - c_j} = 0, \qquad i = 1, 2, ..., s-1.$$

This system needs to be solved numerically for $c_1, ..., c_{s-1}$. This yields many potential solutions, most of which lead to unstable MCM constructions (Hairer and Wanner, 1991, p. 294). In order to prevent this, it is necessary to choose the spacings such that $0 < c_1 < \ldots < c_{s-1} < 1$. In practice, a set of regularly-spaced stage locations provides a good initial guess for an iterative root-finding procedure.

3.3.2 Integrator coefficients

If the derivatives of the collocation polynomial $u(x)$ are known, then Eqs. (3.4-3.5) make up a Hermite interpolation with missing data, where the solution values $u(x_n + c_i h)$ are unknown; thus, the generic Hermite interpolation formulas cannot be applied. Instead, an ad-hoc approach is taken that can be regarded as a generalized Lagrange interpolation, which requires two sets of fundamental interpolating polynomials: the polynomials $\psi_i(t)$ incorporate data from the vector field of the differential equation within the current step, while vanishing at the previous step locations; on the contrary, the polynomials $\phi_i(t)$ account for the previous steps data without interfering with the collocation of the vector field at the stage locations (Schneider, 1994). Hence, these polynomials must fulfill the following properties:

$$\psi_i(t_j) = 0, \qquad j = 1, ..., k; \qquad \psi_i'(c_j) = \delta_{ij}, \qquad j = 1, ..., s; \tag{3.9}$$

$$\phi_i(t_j) = \delta_{ij}, \qquad j = 1, ..., k; \qquad \phi_i'(c_j) = 0, \qquad j = 1, ..., s; \tag{3.10}$$

where δ_{ij} is the Kronecker delta. These fundamental interpolating polynomials allow the collocation polynomial $u(x)$ to be described by

$$u(x_n + th) = \sum_{j=1}^{k} \phi_j(t)\, y_{n-k+j} + h \sum_{j=1}^{s} \psi_j(t)\, f\left(x_n + c_j h, u(x_n + c_j h)\right) \tag{3.11}$$

The latter equation can readily be used to describe a MCM. Indeed, by evaluating Eq. (3.11) at the stage locations, $t = c_i$, and renaming $u(x_n + c_i h)$ as g_i yields the following implicit system:

$$g_i = \sum_{j=1}^{k} \phi_j(c_i)\, y_{n-k+j} + h \sum_{j=1}^{s} \psi_j(c_i)\, f\left(x_n + c_j h, g_j\right), \qquad i = 1, 2, ..., s$$

The latter equation is often written in the more usual form (Hairer and Wanner, 1991, p. 292)

$$g_i = \sum_{j=1}^{k} a_{ij}\, y_{n-k+j} + h \sum_{j=1}^{s} b_{ij}\, f\left(x_n + c_j h, g_j\right), \qquad i = 1, 2, ..., s \tag{3.12}$$

with coefficients defined in terms of the fundamental interpolating polynomials as $a_{ij} = \phi_j(c_i)$ and $b_{ij} = \psi_j(c_i)$. Hence, the solution at the new step is readily obtained

by evaluating the collocation polynomial at $t = c_s = 1$, namely

$$y_{n+1} = g_s = \sum_{j=1}^{k} a_{sj}\, y_{n-k+j} + h \sum_{j=1}^{s} b_{sj}\, f\left(x_n + h, g_j\right)$$

Note that the particular case $k = 1$ yields IRK methods of type Radau IIA, whereas $s = 1$ results in BDF multistep methods.

According to Schneider (1993, p. 334) the MCM is guaranteed to have order $2s+k-2$ and stage order $k+s-1$ as long as the coefficients a_{ij} and b_{ij} satisfy[1]:

$$\sum_{j=1}^{k} a_{ij} = 1 \quad \text{and} \quad \sum_{j=1}^{k} a_{ij} \frac{(j-k)^{l+1}}{l+1} + \sum_{j=1}^{s} b_{ij}\, c_j^l = \frac{c_i^{(l+1)}}{(l+1)} \tag{3.13}$$

for $l = 0, ..., (k+s-2)$ and $i = 1, ..., s$. The system formed by Eqs. (3.13) must be solved numerically. Note, however, that the system is decoupled in the i index, and thus not all the coefficients need to be solved at once, but instead the matrices representing the a_{ij} and b_{ij} coefficients can be computed one row at a time; this approach implies solving s non-dimensional systems of dimension $k+s-1$.

An alternative approach, perhaps more direct, to compute the integrator coefficients would be making use of the collocation polynomials. Indeed, once the stage sampling locations c_i are known, each of the collocation polynomials $\phi(t)$ and $\psi(t)$ can be computed as an Hermite interpolation with incomplete data by imposing the conditions of Eqs. (3.9-3.10), respectively. Then, as previously indicated, evaluation of these polynomials at each node readily provides the coefficients $a_{ij} = \phi_j(c_i)$ and $b_{ij} = \psi_j(c_i)$. The convenience of this approach is that Hermite interpolation with incomplete data is a linear problem, and thus the calculation of the integrator polynomials is attained by solving $k+s$ linear systems, one per collocation polynomial. It must be noted though, that for higher-order MCM instances, the calculation of the collocation polynomials yields linear system matrices that are increasingly ill-conditioned from a numerical viewpoint; likewise, the numerical solution of Eq. (3.13) is increasingly inaccurate for the computation of higher-order MCM instances, as the dimensionality of the non-linear system increases. This difficulty in the numerical calculation of the coefficients, if not treated carefully, may lead to a degradation of the performance of higher-order MCM instances. To avoid this, precalculation of coefficients with extended precision arithmetic may be advisable.

[1] Note that Eq. (3.13) corrects a typo in Lemma 2.1 of Schneider (1993).

Alternatively, calculation of the integrator coefficients can also be approached analytically, as in Lie and Norsett (1989), where an analytical procedure for constructing multistep collocation methods is presented. Although somewhat involved, this approach may be desirable for the construction of higher-order methods, as it enables an accurate calculation of the coefficients.

The integrator coefficients only need to be solved once, so they can be hardcoded in the computer implementation. Note that, for an adaptive order integration scheme, the precalculation of coefficients for all expected orders would be necessary. Table 3.1 contains the coefficients for a particular instance of MCM that is used in later sections, which the reader may find useful for validation purposes.

Table 3.1: Coefficients for a Radau MCM with $k = 8$ and $s = 2$.

$a_{11} = -8.559162 \times 10^{-4}$	$a_{12} = 9.090412 \times 10^{-3}$	$a_{13} = -4.426918 \times 10^{-2}$
$a_{14} = 1.315548 \times 10^{-1}$	$a_{15} = -2.696423 \times 10^{-1}$	$a_{16} = 4.170510 \times 10^{-1}$
$a_{17} = -5.631619 \times 10^{-1}$	$a_{18} = 1.320233 \times 10^{-0}$	$a_{21} = -8.117411 \times 10^{-5}$
$a_{22} = 9.879496 \times 10^{-4}$	$a_{23} = -5.633373 \times 10^{-3}$	$a_{24} = 2.019115 \times 10^{-2}$
$a_{25} = -5.212941 \times 10^{-2}$	$a_{26} = 1.089007 \times 10^{-1}$	$a_{27} = -2.264666 \times 10^{-1}$
$a_{28} = 1.154231 \times 10^{-0}$		
$b_{11} = 3.483501 \times 10^{-1}$	$b_{12} = -2.951588 \times 10^{-2}$	$b_{21} = 7.446506 \times 10^{-1}$
$b_{22} = 1.482532 \times 10^{-1}$		
$c_1 = 5.033967 \times 10^{-1}$	$c_2 = 1.000000 \times 10^{-0}$	

3.3.3 Stability

Radau-based MCM methods enjoy remarkable stability properties. The analysis of their stability is performed following Schneider (1993) and results are extended to higher values of s. Figure 3.2 shows the absolute stability domain diagrams for various k-s configurations. Integrations are said to be stable if each eigenvalue, λ_i, of the system of differential equations being integrated multiplied by the step size, h, fall outside of the stability boundary in these plots. The real component of each eigenvalue is shown along the x-axis and the imaginary component is shown along the y-axis.

Instances with $s = 1$ yield BDF methods, which are known to be zero-stable only for $k \leq 6$, A-stable only for $k = 1$, and $A(\alpha)$-stable otherwise, with α rapidly decreasing with k. Instances with $s = 2$ also exhibit a degrading $A(\alpha)$ stability for increasing k. However, for $s > 2$ these methods posses excellent stability properties and remain

$A(\alpha)$-stable (with a very slowly decaying α) up to very high values of k.[2] Stability properties improve for higher values of s, as such configurations inherit features of IRK methods. In particular, methods for $k = 1$ (IRK methods of type Radau IIA) are A-stable for any value of s. Interestingly, methods with $k = 2$ seem to remain A-stable for very high values of s before eventually becoming $A(\alpha)$-stable.[3] Table 3.2 provides the value of α for various MCM configurations. As mentioned before, because $c_s = 1$, these methods are also stiffly stable in the sense of Gear, which makes them appropriate for stiff problems.

Table 3.2: Stability measure α for various MCM configuration.

	$k=1$	$k=2$	$k=3$	$k=4$	$k=5$	$k=6$	$k=7$	$k=8$
$s=1$	90°	90°	86.033°	73.353°	51.851°	17.852°		
$s=2$	90°	90°	88.188°	82.770°	74.283°	61.711°	40.875°	14.983°
$s=3$	90°	90°	89.726°	88.805°	87.656°	86.454°	85.237°	83.987°
$s=4$	90°	90°	89.964°	89.775°	89.506°	89.204°	88.887°	88.562°
$s=5$	90°	90°	89.995°	89.944°	89.832°	89.667°	89.458°	89.216°
$s=6$	90°	90°	89.999°	89.983°	89.927°	89.821°	89.665°	89.471°

3.4 Implementation

This Section addresses particular aspects related to the numerical implementation of an MCM integrator and presents guidelines and recommendations on how to proceed for coding specific features.

3.4.1 Solution of the implicit system

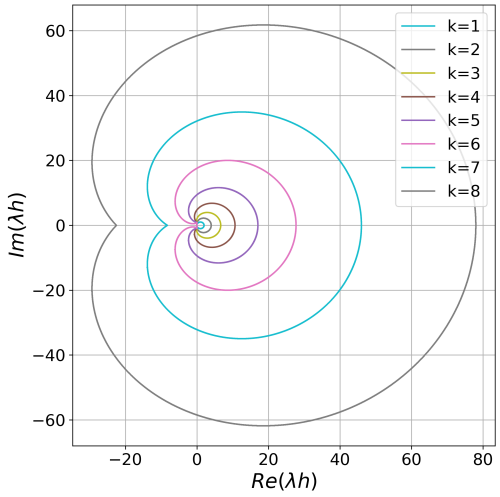
For a complete implementation of the integration scheme an effective procedure needs to be devised for solving the implicit part of the integrator. First, Eq. (3.2) needs to be extended to the computation of multivariate ODEs; using vector notation, Eq. (3.2) generalises to a system of first-order ODEs, where bold symbols represent m-dimensional vectors:

$$\frac{\mathrm{d}\mathbf{y}}{\mathrm{d}x} = \mathbf{f}(x,\mathbf{y}), \qquad \mathbf{y}(x_0) = \mathbf{y}_0. \tag{3.14}$$

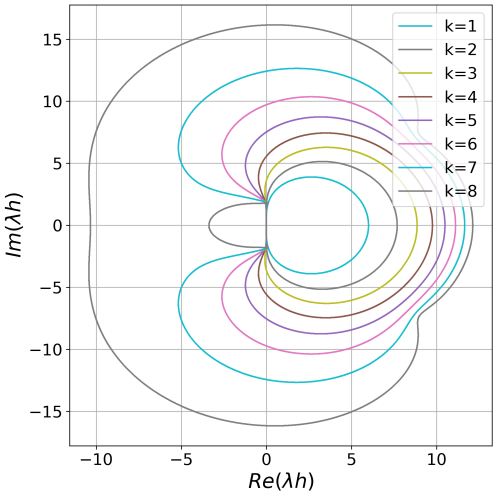
Throughout this discussion bold symbols represent column vectors.

[2] Schneider (1993) reports stability at the origin for up to $k = 28$ when $s = 3$ (a method of order 32), and we have confirmed these stability properties hold even far beyond.

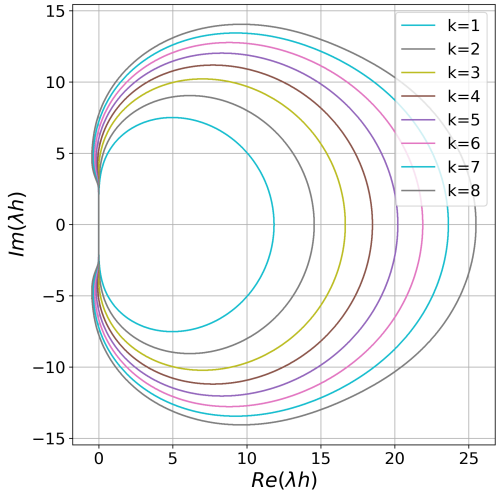
[3] We tested numerically that up to $s = 15$ the methods remain A-stable within double precision.



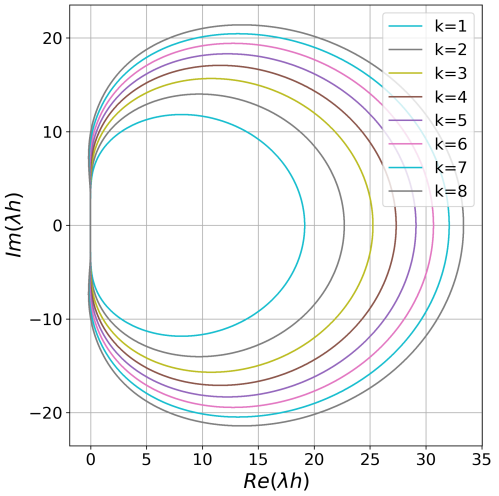
(a) s = 1



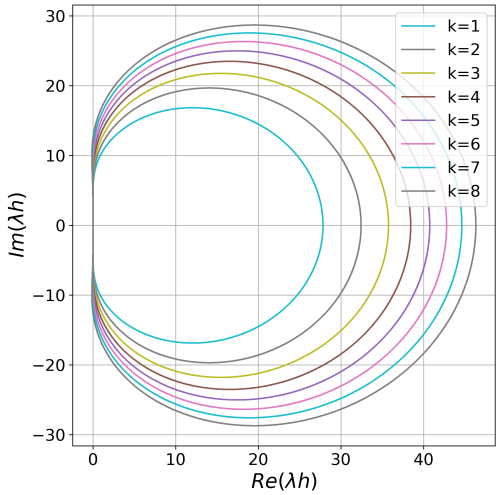
(b) s = 2



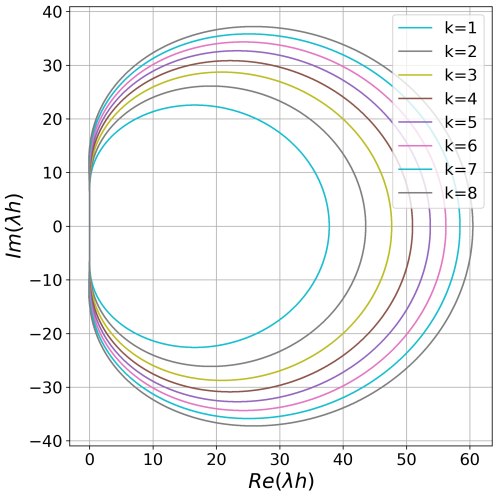
(c) s = 3



(d) s = 4



(e) s = 5



(f) s = 6

Figure 3.2: Stability domain diagrams for the configurations $s = 1, ..., 6$ and $k = 1, ..., 8$. The units are eigenvalue of the system of differential equations being integrated, λ, multiplied by the step size taken, h. The real component of the eigenvalue is shown along the x-axis and the imaginary component is shown along the y-axis.

In vector form, the collocation polynomial, $\mathbf{g}_i$, is given by

$$\mathbf{g}_i = \sum_{j=1}^{k} a_{ij}\,\mathbf{y}_{n-k+j} + h\sum_{j=1}^{s} b_{ij}\,\mathbf{f}(x_n + c_j h, \mathbf{g}_j), \qquad i = 1, 2, ..., s.$$

Since the multistep part of the MCM is explicit, defining the following quantity

$$\mathbf{w}_i = \sum_{j=1}^{k} a_{ij}\mathbf{y}_{n-k+j}, \qquad i = 1, ..., s$$

allows one to conveniently rewrite Eq. (3.12) in terms of a new variable $\mathbf{z}_i \equiv \mathbf{g}_i - \mathbf{w}_i$ that represents the implicit part of the system and helps to reduce the number of variables being worked with while allowing to iterate only on the variable $\mathbf{z}_i$ (Hairer and Wanner, 1991, p. 128):

$$\mathbf{z}_i = h\sum_{j=1}^{s} b_{ij}\,\mathbf{f}(x_n + c_j h, \mathbf{w}_j + \mathbf{z}_j), \qquad i = 1, ..., s. \tag{3.15}$$

At this point, it is convenient to define the following higher dimensional vectors, which arrange the variables $\mathbf{w}_i$ and $\mathbf{z}_i$ in (ms)-dimensional column vectors, such that

$$\mathbf{W} = \begin{bmatrix} \mathbf{w}_1 \\ \vdots \\ \mathbf{w}_s \end{bmatrix}, \qquad \mathbf{Z} = \begin{bmatrix} \mathbf{z}_1 \\ \vdots \\ \mathbf{z}_s \end{bmatrix}.$$

where, the first m components of $\mathbf{W}$ are the components of $\mathbf{w}_1$, and likewise for $\mathbf{Z}$. Similarly, the function evaluations of the right-hand side of Eq. (3.15) can be accommodated in the following (ms)-dimensional column vector:

$$\mathbf{F}(\mathbf{Z}) = \begin{bmatrix} \mathbf{f}(x_n + c_1 h, \mathbf{w}_1 + \mathbf{z}_1) \\ \vdots \\ \mathbf{f}(x_n + c_s h, \mathbf{w}_s + \mathbf{z}_s) \end{bmatrix}.$$

This allows for the system of ODEs to be compactly expressed using the Kronecker product notation

$$\mathbf{Z} = h\,(b \otimes I_{m,m})\,\mathbf{F}(\mathbf{Z}) \tag{3.16}$$

where $I_{m,m}$ is an $m \times m$ identity matrix. It is now possible to solve this system for $\mathbf{Z}$ by application of Newton's method, which at every iteration, ℓ, yields an updated

approximation

$$\mathbf{Z}^{\ell+1} = \mathbf{Z}^{\ell} + {}'\mathbf{Z}^{\ell}$$

These corrections ${}'\mathbf{Z}^{\ell}$ are obtained from solving the following linear system:

$$\left(I_{ms,ms} - h\,(b \otimes J)\right)\ {}'\mathbf{Z}^{\ell} = h\,(b \otimes I_{m,m})\,\mathbf{F}(\mathbf{Z}^{\ell}) - \mathbf{Z}^{\ell} \tag{3.17}$$

where J stands for the Jacobian of the ODE system. Its calculation might be computationally costly, so simplified Newton iterations are often used, making use of a single Jacobian evaluated at the beginning of the integration step, i.e.

$$J = \frac{\partial}{\partial \mathbf{y}}\,\mathbf{f}(x_n, \mathbf{y}_n).$$

It has to be noted that the matrix $\left(I_{ms,ms} - h\,(b \otimes J)\right)$ in Eq. (3.17) is known to posses a very special structure, which can be exploited in various ways to reduce considerably the number of calculations required to solve the linear system (Butcher, 1976). Note that in Schneider (1993) a MCM with $s = 3$ is developed on the basis that this is the smallest value of s where these techniques offer an actual gain. Alternative approaches may include transforming the matrix to Hessenberg form or exploiting the intrinsic sparsity of the matrix. In all these cases, the actual workload ultimately depends on the efficiency of the utilised linear algebra routines, specially for higher-dimensional problems, where the matrices become large and a carefully implemented algebra may offer a considerable speed-up.

The convergence criterion to stop the Newton iterations is essential from an efficiency viewpoint. The convergence criterion used in our implementation mimics that used in the RADAU code[4] and described in Hairer and Wanner (1991, p. 130, 192), and Press et al. (2007, p. 913). One key aspect is the estimation of the convergence rate $||\,{}'\mathbf{Z}^{\ell}||\,/\,||\,{}'\mathbf{Z}^{\ell-1}||$ that monitors the quality of the convergence. This allows one to reduce the computation cost by updating the Jacobian only when the convergence rate exceeds a prescribed value (we chose 10^{-3}) and allows the same Jacobian to be reused in successive steps. Analytical Jacobians were used throughout the numerical experiments in this chapter.

[4]Code available from `http://www.unige.ch/~hairer/software.html`

3.4.2 Dense output

A fixed-stepsize numerical integration method provides the solution to the IVP of Eq. (3.14) at every step. However, the solution is often needed at specific values of x that are not necessarily multiples of the stepsize. Hence, it is desirable that integrators have the ability to provide a dense output. This is achieved by computing, along with the solution at every step, an interpolation polynomial that approximates the actual solution, $\mathbf{y}(x)$, as closely as possible. Since MCM are collocation methods, they naturally possess an interpolating polynomial, $\mathbf{u}(x)$, given by Eq. (3.11) in scalar form, or in multivariate form as

$$\mathbf{u}(x_n + t\,h) = \sum_{j=1}^{k} \phi_j(t)\ \mathbf{y}_{n-k+j} + h \sum_{j=1}^{s} \psi_j(t)\,\mathbf{f}\big(x_n + c_j h, \mathbf{g}_j\big)\big), \tag{3.18}$$

which solely depends on the values $\mathbf{g}_j$, which are known once the linear system of Eq. (3.16) is solved, and the fundamental interpolating polynomials $\phi_j(t)$ and $\psi_j(t)$. The collocation polynomial can be evaluated at any value of $t \in [-k+1, 1]$, where it is accurate to order $2s + k - 2$.

Interestingly, and although the collocation polynomial readily provides the optimal interpolation for a given MCM, other (suboptimal) interpolators can also be proposed, which, occasionally, might turn out useful. In particular, note that at every integration step not only past values of the solution are available, but also their derivatives, $\mathbf{f}(x, \mathbf{y})$; similarly, once the linear system is solved, the solution is known at every stage, $\mathbf{g}_j$, and so are their derivatives, $\mathbf{f}(x_n + c_j\,h, \mathbf{g}_j)$. Thus, all this information can be used and combined as needed to obtain a Hermite interpolating polynomial. However, although it is tempting to believe that this might allow for a higher-order interpolation, one must not lose from sight that the above data is intrinsically redundant; therefore, any interpolating polynomial of an order higher than the collocation polynomial will overfit the solution, whereas a lower-order interpolation will of course be less accurate. This is analysed in Section 3.5.5.

Solving the linear system (3.16) requires an initial guess for $\mathbf{Z}^0_{n+1}$ to be used at the first iteration. $\mathbf{Z}^0_{n+1} = \mathbf{0}$ can be taken in the absence of a better guess, but one can do much better by using the collocation polynomial as a predictor (note this involves extrapolation outside the collocation domain), i.e.

$$\mathbf{Z}^0_{n+1} = [\mathbf{u}(x_{n+1} + c_1\,h), ..., \mathbf{u}(x_{n+1} + c_s\,h)]^\top$$

This point is particularly relevant for high-s configurations, since these allow larger stepsizes, which yield linear systems that are more difficult to solve. Also, providing an accurate initial guess reduces the convergence times and greatly impacts the overall efficiency of the scheme, as will be discussed in Section 3.5.5. It must be noted that this strategy for predicting the initial guess is distinct from the one proposed in Schneider (1993), which is specific to $s = 3$ and proved unsuitable for other configurations, especially those yielding high-order schemes.

3.5 Numerical experiments

The flexibility of MCM allows for integration schemes of different configurations (i.e. k and s combinations) to be created. This section presents the results of a series of numerical experiments designed to compare the performance across a wide parameter space of integrator configurations. Such that a fair comparison can be made of the intrinsic merits of MCM, fixed stepsize is used throughout all following experiments, thus removing the effects of adaptive strategies. After introducing the test problems, the experimental setup and other considerations are described, and finally the results are discussed and analysed.

3.5.1 Test problems

Three systems of ODEs have been chosen for use in this comparison: the Lorenz attractor, the Prothero-Robinson problem, and a forced Van der Pol oscillator. The first is particularly interesting due to its chaoticity, whereas the latter two pose a challenge to integration schemes due to their stiffness.

The Lorenz attractor

Represents a low dimensionality, low stiffness problem, which instead offers challenges to integration schemes due to its chaotic nature. The system is defined as

$$\frac{\mathrm{d}\mathbf{u}}{\mathrm{d}t} = \frac{\mathrm{d}}{\mathrm{d}t}\begin{bmatrix} x \\ y \\ z \end{bmatrix} = \begin{bmatrix} -\sigma x + \sigma y \\ -x z + r x - y \\ x y - b z \end{bmatrix}$$

where the parameters $b = 8/3$, $\sigma = 10$ and $r = 28$ have been selected. The initial conditions are chosen as per Hairer et al. (1987, p. 120), namely $\mathbf{u}_0 = [-8, 8, 27]^\top$ and the integration interval is $t \in [0, 5]$.

The Prothero-Robinson problem

Represents a low dimensionality, high stiffness problem (condition number on the order of 100,000), defined as (Constantinescu and Sandu, 2013)

$$\frac{\mathrm{d}\mathbf{u}}{\mathrm{d}t} = \frac{\mathrm{d}}{\mathrm{d}t}\begin{bmatrix} x(t) \\ y(t) \end{bmatrix} = \begin{bmatrix} \Gamma & \epsilon \\ \epsilon & -1 \end{bmatrix} \begin{bmatrix} \dfrac{-1 + x^2 - \cos(t)}{2\,x} \\ \dfrac{-2 + y^2 - \cos(\omega\, t)}{2\,y} \end{bmatrix} - \begin{bmatrix} \dfrac{\sin(t)}{2\,x} \\ \dfrac{\omega\, \sin(\omega\, t)}{2\,y} \end{bmatrix}$$

where $\Gamma = -2\ \times 10^5$, $\omega = 20$, $\epsilon = 0.5$ and is always integrated across the interval $t = [-3, 3]$. The analytical solution to this problem is known.

The Van der Pol oscillator

A Van der Pol oscillator with sinusoidal forcing is proposed as another example of a stiff problem, defined as

$$\frac{\mathrm{d}^2 y}{\mathrm{d}t} - \nu(1 - y^2)\frac{\mathrm{d}y}{\mathrm{d}t} + \nu\, y = A \sin(\omega\, t)$$

with $\nu = 8.5$, $A = 5$ and $\omega = \pi/50$. The initial conditions are $y(0) = 2$, $y'(0) = -0.66$, and the integration interval is $t \in [0, 200]$.

3.5.2 Experimental setup

In order to perform numerical experiments with the aforementioned problems, a few other considerations need to be taken into account.

Error calculation

As a proxy for the precision of the numerical solution, the average error across the whole integration domain is used, according to

$$E = \frac{1}{m\,n} \sum_{i=0}^{n} \sum_{j=0}^{m} \left| u_{ij} - u^*_{ij} \right|$$

where j runs across each of the m ODEs of the system, i accesses all n solution points within the domain, and u^*_{ij} are obtained from the true solution. Where no exact solution exists, a high precision set of values for u^*_{ij} is generated numerically using higher-order methods with adaptive stepsize and more stringent tolerances.

Total evaluation cost

The metric used throughout this work for the Total Evaluation Cost (TEC) of the integration methods combines the number of ODE right-hand side evaluations, `Fcalls`, and the Jacobian evaluations, `Jcalls`, which are scaled to be described in terms of the cost of a single function evaluation, i.e.

$$\texttt{TEC} = \texttt{Fcalls} + \texttt{Jcalls} \times C_{\text{scale}},$$

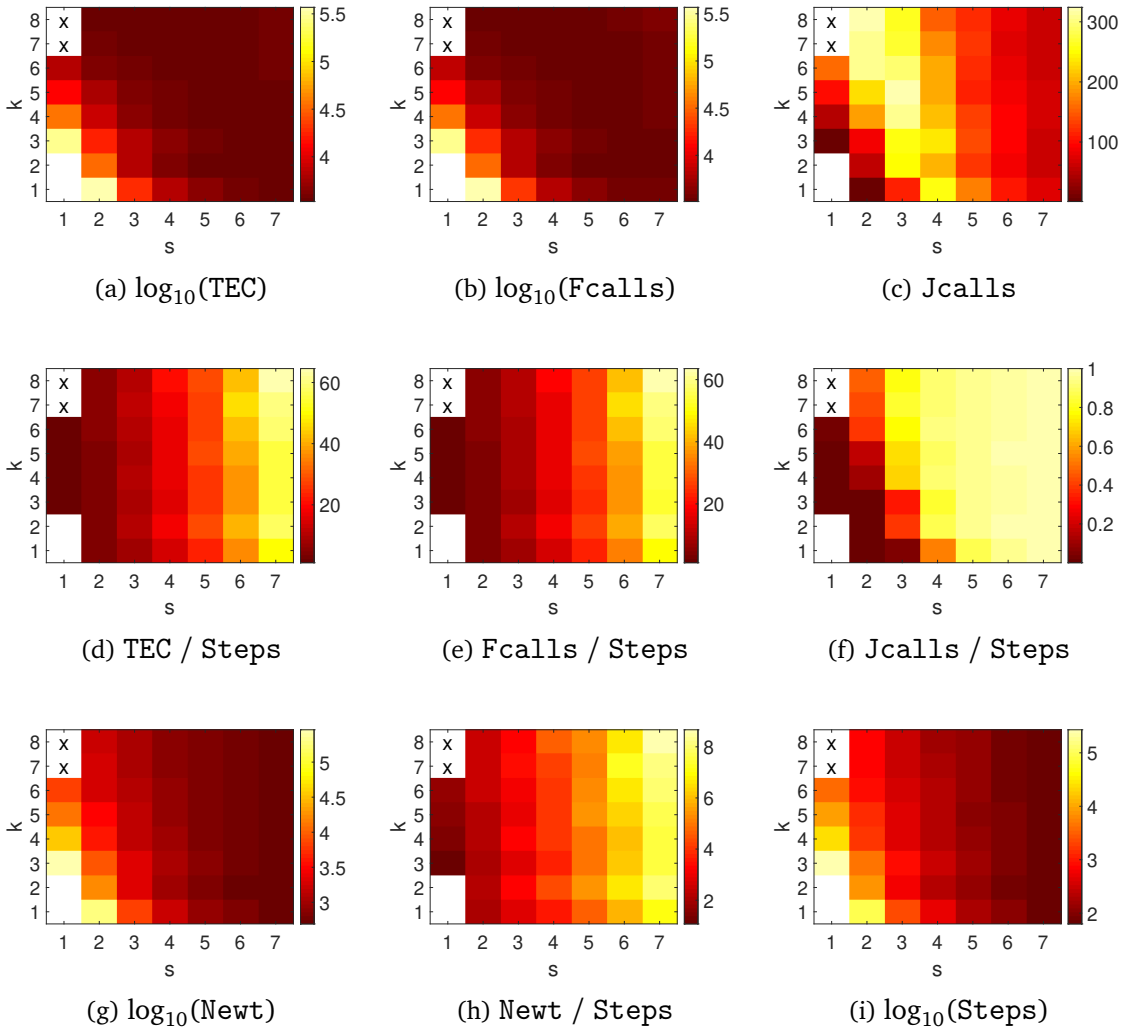
where C_{scale} is a scaling cost dependent on the specific problem. In particular, $C_{\text{scale}} = 2$ for the Prothero-Robinson problem and $C_{\text{scale}} = 1$ for the other problems.

Comparison with other methods

As a baseline for comparison and validation, the RADAU9 (Hairer and Wanner, 1999) code will be used, as in the case $k = 1$ MCM methods reduce to be of an identical nature. This routine is available online in Fortran code, and it has been purposely modified for this article so that a fixed-stepsize integration can be enforced to allow a direct comparison with MCM instances in equal conditions.

3.5.3 General comparison of MCM configurations

Figure 3.3 contains heatmaps for the Loretz problem. Heatmaps are a natural and effective way of displaying various performance metrics for different configurations



(a) $\log_{10}(\texttt{TEC})$ (b) $\log_{10}(\texttt{Fcalls})$ (c) `Jcalls`

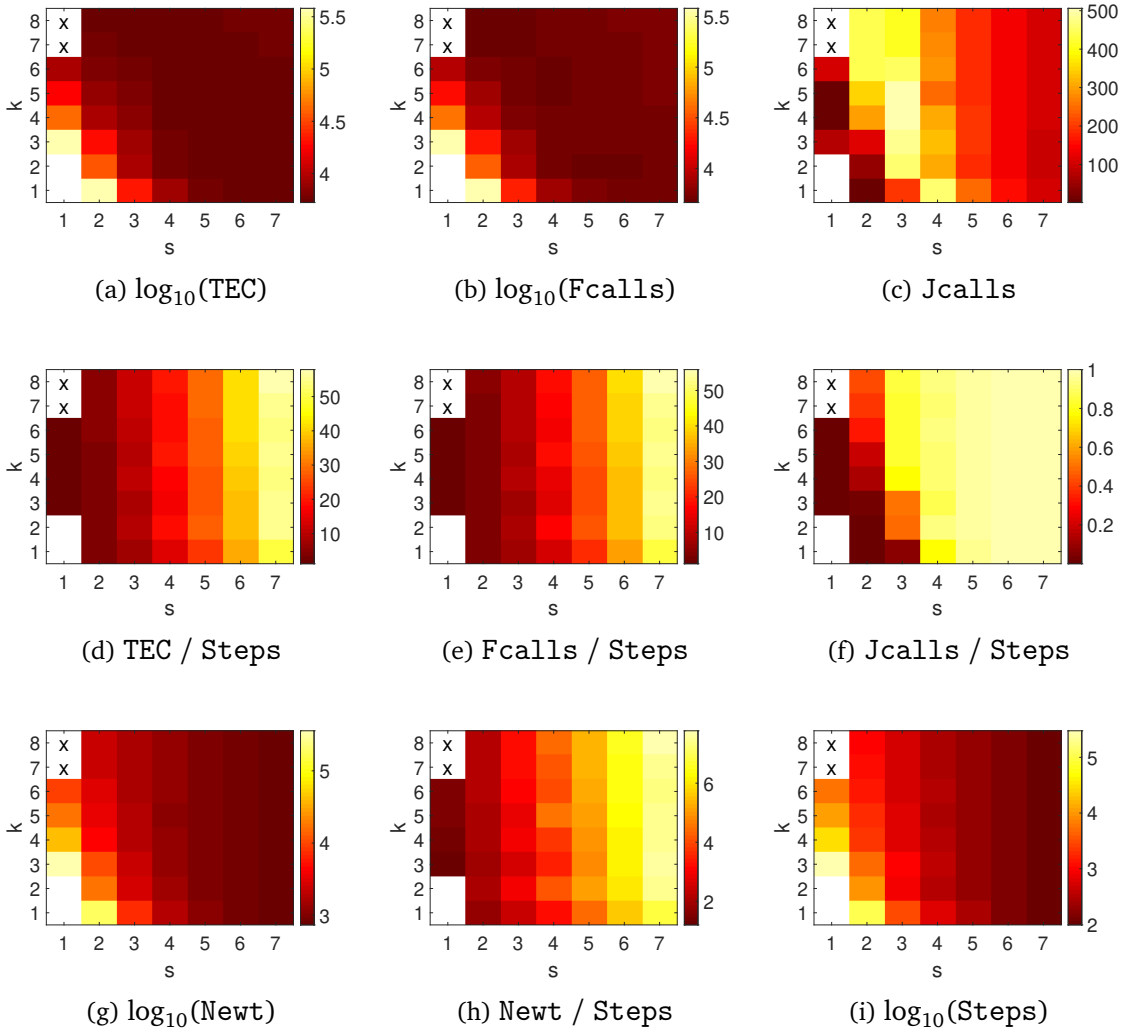
(d) `TEC / Steps` (e) `Fcalls / Steps` (f) `Jcalls / Steps`

(g) $\log_{10}(\texttt{Newt})$ (h) `Newt / Steps` (i) $\log_{10}(\texttt{Steps})$

Figure 3.3: Integrator performance in the Lorenz problem for various MCM configurations for an upper error bound of 10^{-8}. The colour map excludes methods of order lower than 3, and non zero-stable BDF methods are removed.

of MCM, i.e. different choices of k and s. Here, each location corresponds to an integration performed using a particular k-s pairing. Heatmaps are generated using a step size that, for each configuration, is determined by dividing the integration domain by the smallest integer divisor that ensures the integration error is below a prescribed threshold (specified in the figure captions). Therefore, the same accuracy is uniformly obtained across all configurations, enabling a direct comparison of their computational cost. For the sake of brevity and concision, this analysis focuses only on the results from the Lorenz problem, although the reported behaviour is consistently mirrored across the other test problems, as per Figs. 3.4 and 3.5.

Figure 3.3a clearly shows that the overall performance of a MCM instance is highly dependent upon the particular configuration used. As expected, for a prescribed upper



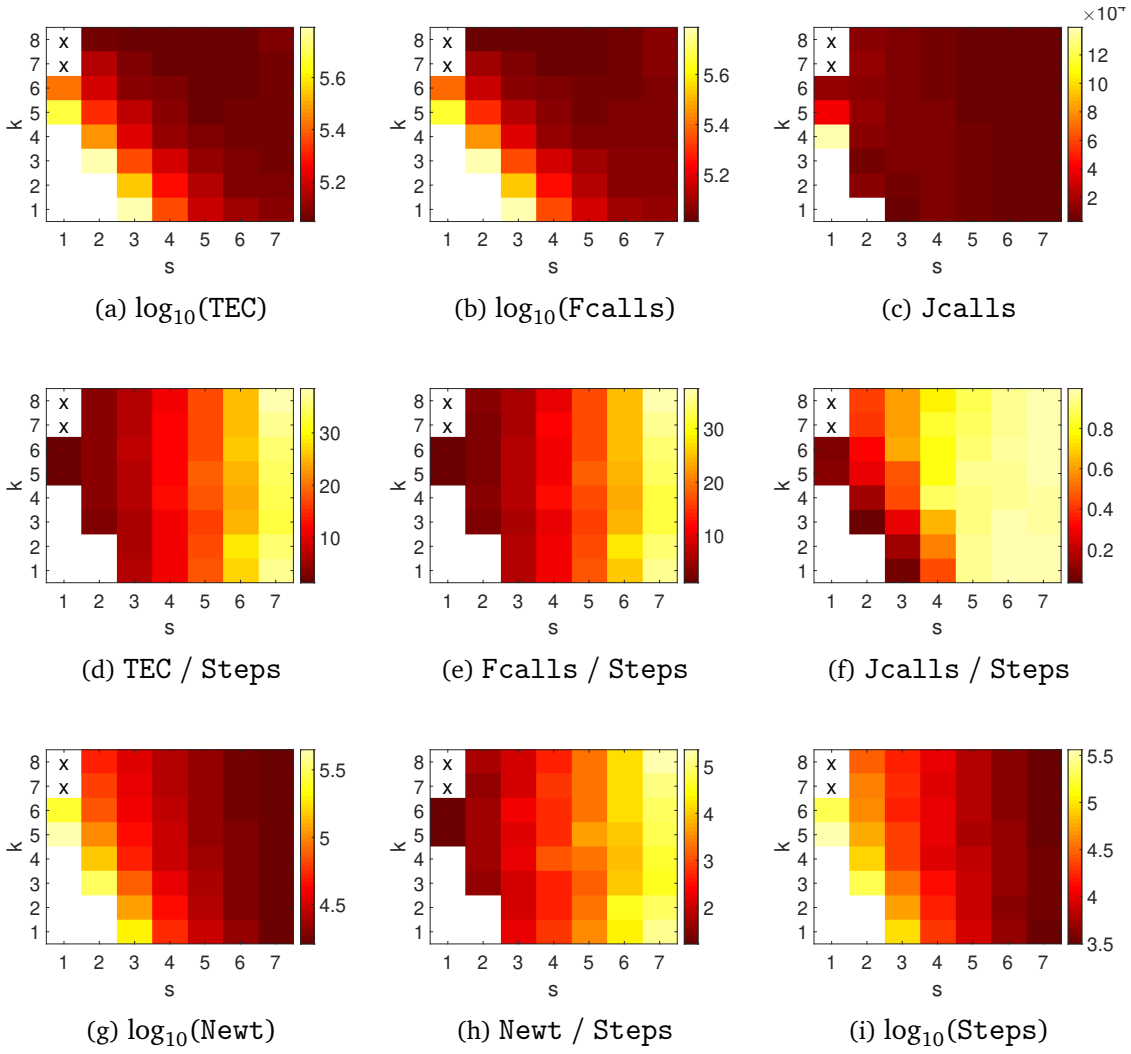
(a) $\log_{10}(\texttt{TEC})$ (b) $\log_{10}(\texttt{Fcalls})$ (c) `Jcalls`

(d) `TEC / Steps` (e) `Fcalls / Steps` (f) `Jcalls / Steps`

(g) $\log_{10}(\texttt{Newt})$ (h) `Newt / Steps` (i) $\log_{10}(\texttt{Steps})$

Figure 3.4: Integrator performance in the Prothero-Robinson problem for various MCM configurations with an upper error bound of 10^{-11}. The colour map excludes methods of order lower than 3, and non zero-stable BDF methods are removed

error bound, low-order schemes (bottom left corner) are computationally less efficient compared to high-order schemes (upper right corner). The heatmap also shows a relatively smooth transition across adjacent configurations, meaning that the performance improves as the order is raised by either moving along columns (increasing k) or along rows (increasing s). However, upon closer inspection, certain configurations stand out for offering a higher performance than their neighbouring configurations. For instance, for the case where $s = 4$, the particular value of $k = 2$ seems to perform better than $k = 3$ despite having a lower order, and the same occurs for $k = 4$. It is worth noting that $k = 2$ outperforming $k = 3$ seems to be a recurring feature for configurations with $s \geqslant 3$, which was observed in many test problems studied, both stiff and non-stiff, with the exception of the Van der Pol oscillator. The existence of such "sweet spots" or configurations that seemingly work particularly well for a given



(a) $\log_{10}$(TEC) (b) $\log_{10}$(Fcalls) (c) Jcalls

(d) TEC / Steps (e) Fcalls / Steps (f) Jcalls / Steps

(g) $\log_{10}$(Newt) (h) Newt / Steps (i) $\log_{10}$(Steps)

Figure 3.5: Integrator performance in the Van der Pol oscillator for various MCM configurations with an upper error bound of 10^{-11}. The colour map excludes methods of order lower than 4, and non zero-stable BDF methods are removed

problem, might be of great importance for repetitive tasks where the same problem may need to be integrated over and over again, as for a Monte Carlo analysis, or problems that need to be routinely performed on a regular basis. These particular configurations seem to benefit from a reduced number of steps and thus fewer Jacobian evaluations (Jcalls).

Of all of these heatmaps, the number of Jacobian evaluations (Jcalls), shown in Fig. 3.3c, is the one that exhibits the richest structure, which is a direct consequence of two competing factors. On the one hand, raising k contributes to increasing the order of the MCM, and this in turn allows larger stepsizes to be taken (Fig. 3.3i) without incurring any additional computational cost. Nor is there added overhead from a linear algebra point of view, since there is no increase in the dimensionality of the implicit part of the MCM, and the cost of solving Eq. (3.16) is, therefore, roughly

independent of k, as can be seen from the average number of Newton iterations per step Fig. 3.3h.

On the other hand, as the value of s increases, MCM instances inherit more features of multi-stage methods which also allow for increasingly larger stepsizes for a prescribed accuracy. However, increasing s also raises the dimensionality of the implicit part of the MCM, which is of dimension ms, and because root-finding is more difficult in higher dimensions, this entails two main consequences:

1. an increase in the number of Newton iterations per step necessary to achieve convergence (Fig. 3.3h),
2. that the Jacobian needs to be updated more often (Fig. 3.3f).

From these two competing factors, the former dominates for low s, thus exhibiting a reduction of `Jcalls` as k increases, and the latter becomes dominant as s raises, thus making `Jcalls` nearly invariant with k, as illustrated in Fig. 3.3c. A consequence of this strong variability of `Jcalls` makes this a very important consideration when choosing an adequate configuration for a particular problem, especially when the computational cost is driven by Jacobian evaluations, as is the case for large systems with a fully dense Jacobian, even more so when it needs to be numerically computed and function evaluations are expensive. In these cases, $C_{\text{scale}} \approx m$, and thus TEC $\approx$ `Jcalls`, meaning that the heatmap for TEC will closely match that of `Jcalls`, so choosing a configuration that minimises Jacobian evaluations may be critical. Additionally, as discussed in Section 3.3.3, increasing k has a direct impact upon the stability properties of the methods, particularly for the case $s = 2$, which suffers from an acute degradation of the $A(\alpha)$-stability, so the use of such configurations must be carefully considered for each specific problem.

Another noteworthy appreciation to be made is that single-step configurations systematically improve their performance when information from previous steps (which is available for free) is incorporated; this highlights that Radau-based IRK methods, which are highly esteemed for their high efficiency, could greatly benefit from a multistep implementation. Again, this performance improvement is an immediate consequence of larger stepsizes being possible while avoiding the added computational overhead associated with configurations comprising more stages.

These findings and conclusions were mirrored across all three of the problems tested, as can be seen in Figs. 3.4 and 3.5, where the best performing configurations are consistent for all three cases. Finally, it must be noted that instances with the highest

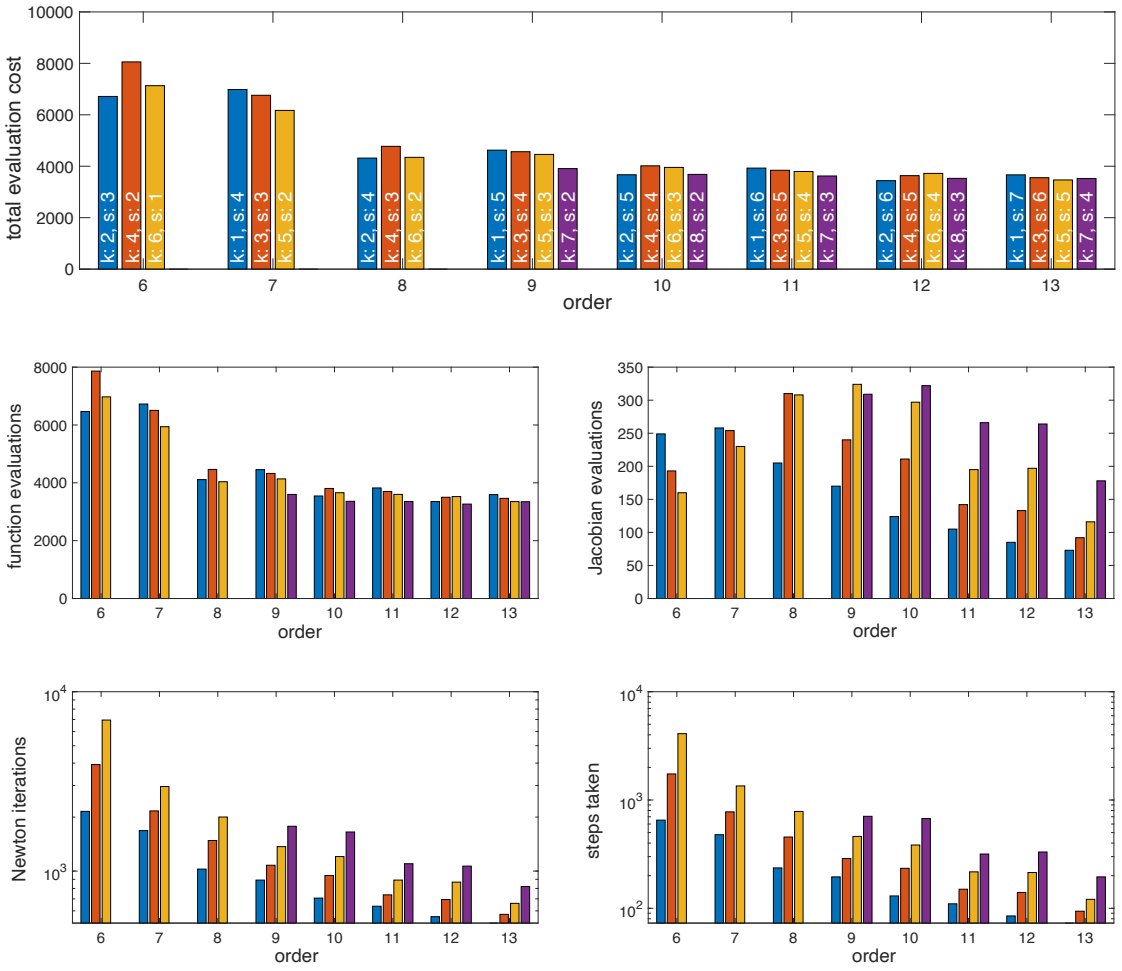
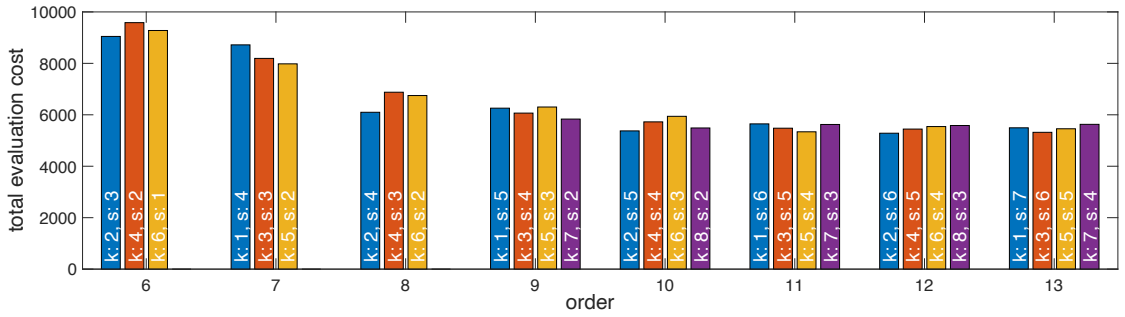


Figure 3.6: Results of Figure 3.3 (Lorenz problem) grouped in MCM configurations of equal order.

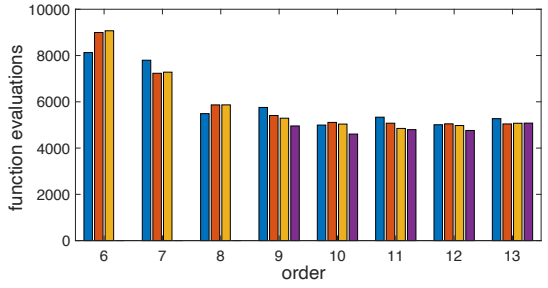
values of s and k, located at the top right corner of the heatmaps, appear to exhibit a slight degradation in their performance. This is likely so because of the natural limitations of the floating-point arithmetics of double precision, which inevitably introduces numerical errors in the computation of MCM coefficients for very high-order methods, which then propagate through the numerical integration process. This degradation can likely be mitigated by hard-coding accurately computed coefficients using extended precision arithmetics.

3.5.4 Performance as a function of the order

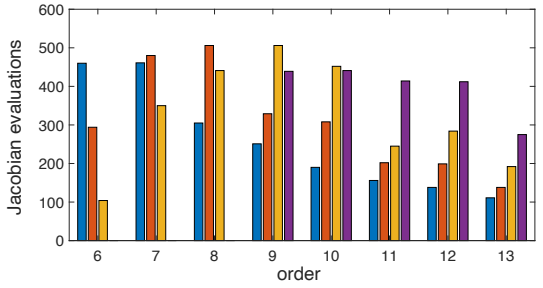
It is particularly revealing to look at the different performance metrics by comparing the various MCM instances that yield methods of the very same order. This is shown in Figure 3.6, where for a given order and prescribed accuracy, different MCM



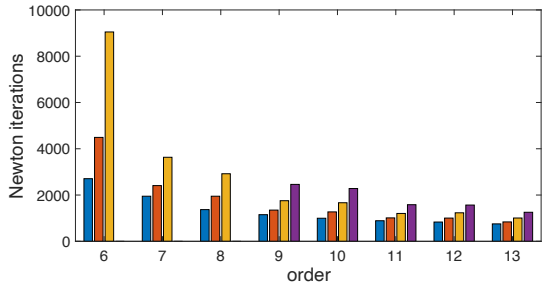
(a) TEC



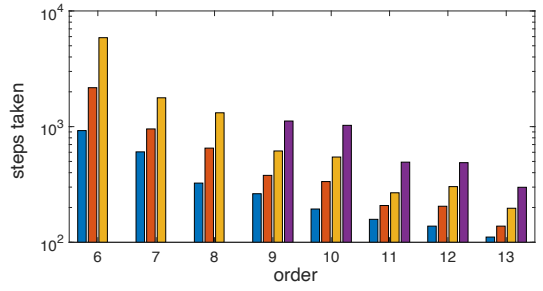
(b) Fcalls



(c) Jcalls



(d) Newt



(e) Steps

Figure 3.7: Results of Figure 3.4 (Prothero-Robinson problem) grouped in MCM configurations of equal order.

configurations may differ in performance significantly from one another. In particular, as a general rule, configurations with higher s and lower k yield larger stepsizes and require more Newton iterations per step, but the former pays off to provide a lower overall value of total Newton iterations. However, the computational cost of each iteration is proportional to the number of stages, so in the end it turns out that higher values of s require more function evaluations, except when $k = 1$, which appears to be an exception to this rule. These patterns remain valid for all studied test problems, and similar histograms for the Prothero-Robinson problem are shown in Fig. 3.7. As for the Total Evaluation Cost (TEC), however, where differences between configurations can be quite substantial (especially for low orders), its value depends on the problem-dependent parameter C_{scale}, as well as both the function and Jacobian

evaluations, the latter of which is deserving of a dedicated discussion.

Because of the competing factors discussed in the preceding section that affect the number of Jacobian evaluations, there is an involved interplay that gives rise to distinct patterns for `Jcalls` when configurations of the same order are compared. When the instances considered are of order p below a certain problem-specific cut-off value ($p \leqslant 6$ for the Lorenz test case), `Jcalls` increases with the value of s, but for p above a certain cut-off value ($p \geqslant 9$ for Lorenz) this trend reverses and `Jcalls` increases as the value s decreases. A similar pattern was observed in all the studied test problems. This pattern is of paramount importance for problems where Jacobian evaluations are expensive, and are therefore the driving aspect of the performance. In such cases, for lower-order methods it would seem preferable to favour low-s and high-k configurations; however, when higher-order methods are required, it would seem appropriate to prioritise high-s and low-k combinations. Nonetheless, the latter option should be considered judiciously, because if a high value of s is considered a requirement for a given MCM, one shouldn't just settle with a low value of k when, in practice, with the expedient of increasing k one may also increase the order of the method arbitrarily with no additional computational cost, but still with all the benefits of having a high value of s. In fact, this is the procedure we would recommend for designing a specific MCM instance, first selecting the value of s based on the problem's dimensionality and the associated linear algebra considerations, and then setting the value of k as high as needed to meet a prescribed order or performance requirements.

Interestingly, when a configuration with $k = 2$ is available for a given order, this proves to be systematically the best performing one under the proposed performance criteria. Another interesting remark is that, despite the fact that configurations where $s = 3$ have been most widely considered in the literature, under the proposed metrics there are often better performing configurations for any given order. It must be noted, however, that none of these proxies for the computational performance takes into account the cost of matrix inversion and other algebra-related aspects when solving the implicit part of the MCM (Section 3.4), which is increasingly more costly for instances with higher s. Hence, although these conclusions remain of significance for low-dimensional problems with expensive function evaluations, for higher-dimensional problems with cheap function evaluations, linear algebra consideration may outweigh the apparent high performance of high-s MCM instances. Hence, to complete the picture, not only the TEC but also the runtime would need to be considered. However, runtime considerations are intentionally omitted from this analysis for two main reasons: 1) their quantification is heavily dependent on low-level implementation details, as well as the efficiency of the linear algebra routines, aspects that

are beyond the scope of this work; and 2) the proposed performance metrics remain representative whenever the function and Jacobian evaluations are costly, and thus comprise the dominant contribution to the total computational cost when the cost of solving the linear system is marginal compared to that of function evaluations.

3.5.5 On the impact of the predictor

As highlighted before, the performance of MCM is largely dominated by the iterative process employed to solve the implicit part; in particular, the number of Newton iterations necessary for convergence depends heavily on the initial guess provided to initialise the iterative procedure. Thus, the overall performance of MCM can be greatly improved if an accurate prediction of the initial guess were available for the vector $\mathbf{Z}^0_{n+1}$. As discussed in Section 3.4.2, it makes sense to calculate such initial guesses by extrapolation of the collocation polynomial computed in the preceding step by means of Eq. (3.18); we shall refer to this predictor as $\mathscr{P}$. Table 3.3 shows how this predictor notably improves the performance metrics compared to the naive initial guess $\mathbf{Z}^0_{n+1} = \mathbf{0}$, referred to as 'None'.

$p=8$	$k=6, s=2$ (Steps: 784)				$k=4, s=3$ (Steps: 455)				$k=2, s=4$ (Steps: 236)			
	TEC	Fcalls	Jcalls	Newt	TEC	Fcalls	Jcalls	Newt	TEC	Fcalls	Jcalls	Newt
None	7909	7666	243	3810	7242	6958	284	2311	5737	5536	201	1382
$\mathscr{P}$	4346	4038	308	2002	4774	4464	310	1481	4317	4112	205	1026
$p=9$	$k=7, s=2$ (Steps: 707)				$k=5, s=3$ (Steps: 461)				$k=3, s=4$ (Steps: 288)			
	TEC	Fcalls	Jcalls	Newt	TEC	Fcalls	Jcalls	Newt	TEC	Fcalls	Jcalls	Newt
None	7218	6963	255	3451	7328	7045	283	2336	6722	6495	227	1620
$\mathscr{P}$	3908	3599	309	1778	4460	4136	324	1369	4565	4325	240	1078

Table 3.3: Impact of the predictor for the Lorenz problem with an error threshold of 10^{-8} and different 8^{th} and 9^{th} order MCM configurations.

The two key metrics to consider here are the Newton iterations, `Newt`, and `TEC`. The use of $\mathscr{P}$ systematically reduces the number of Newton iterations required for convergence when compared to the naive initial guess. A small increase in `Jcalls` is easily offset by this reduction and the combined computational cost, `TEC`, is also systematically below that of the naive guess.

3.5.6 Performance curve comparison

While heatmaps are an excellent tool for comparing integrator configurations, they only allow a comparison for a prescribed value of the integration error. To compare

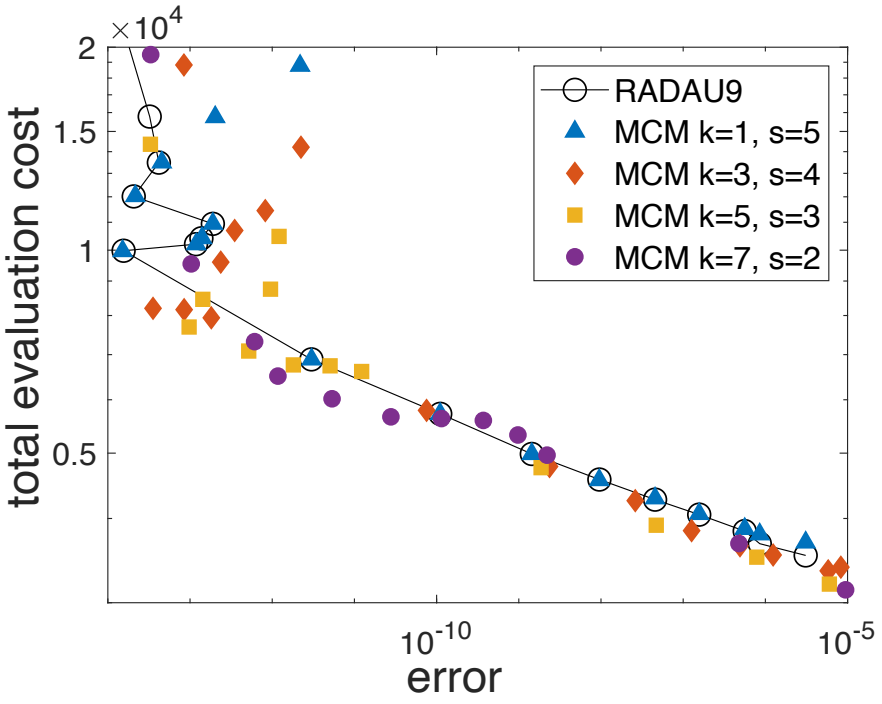


Figure 3.8: Performance curves for various MCM configurations for the Prothero-Robinson problem with instances of order 9.

the integrator performance across a range of errors and configurations, as well as to compare them with other integration procedures, performance curves are a more appropriate representation. To compute them, each integration method is used to solve a given problem with varying stepsizes, so that for each integrator a TEC vs $error$ curve can be obtained that is parametrically defined by the stepsize. Figure 3.8 displays such curves in the Prothero-Robinson problem for all MCM configurations of order 9. Similarly, Fig. 3.9 displays curves for configurations with fixed $s = 5$ and increasing k. The performance of the RADAU9 code is also displayed as a baseline and for validation purposes; as expected, it matches very closely our implementation of the 9^{th} order single-step MCM configuration, except for occasional slight deviations caused by subtle implementation differences.

Importantly, Fig. 3.8 highlights that the performance of different MCM configuration can greatly vary even for a given order. The error is comparable for all configurations, yet configurations where s is lower will allow for a reduction in the dimensionality of the implicit system to solve, a feature that may become important for computationally expensive problems dominated by Jacobian evaluations. As highlighted in preceding subsections, it can be seen in Fig. 3.9 that configurations with $k = 2$ seem to perform particularly well, and even better than higher-order configurations with the same number of stages; for instance, in this particular problem the $(k = 2, s = 5)$

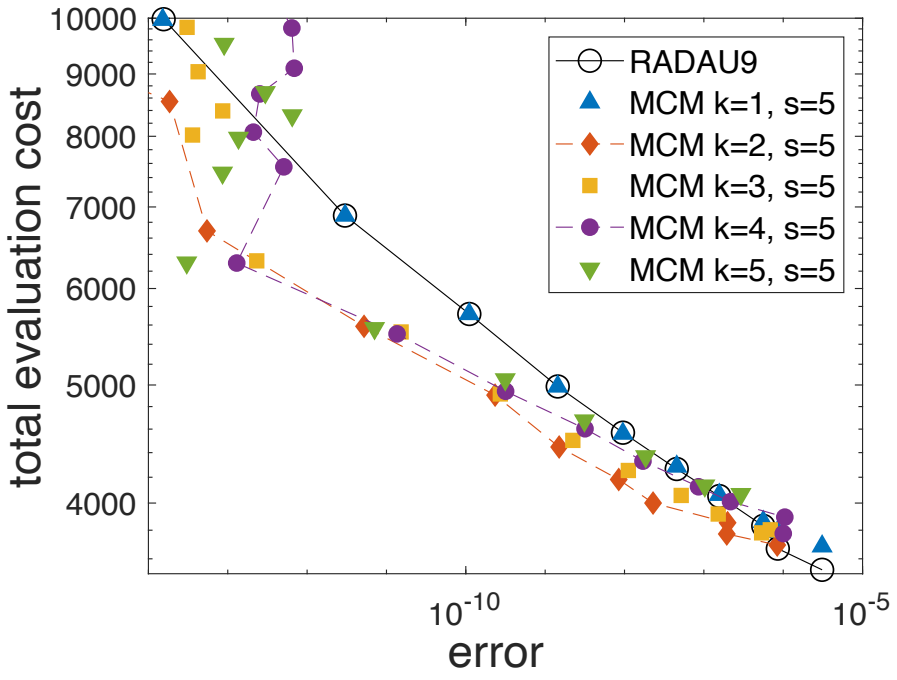


Figure 3.9: Performance curves for various MCM configurations for the Prothero-Robinson problem with $s = 5$ and varying k.

configuration even outperforms the $(k = 5, s = 5)$ configuration. Similar behaviours were observed in all test problems, thus highlighting the impact that a given MCM configuration may have on the performance, and the fact that the order of the method is not the only metric to be considered when trying to predict performance expectations.

3.6 Summary

This chapter splits in two parts: the first is devoted to providing a comprehensive, dedicated introduction to multistep collocation methods based in Radau quadratures, by bringing together a compendium of concepts and theory that in the literature are only found spread among various bibliographic resources. This information is presented with a focus on the numerical implementation, in such a way that these integration methods can be easily replicated and implemented. To this end, coefficient tables are presented along with an analysis of the stability of the schemes for various configurations, thereby showing the excellent stability properties of this family of integration methods, even when a high number of previous steps are accounted for. This stability analysis also enables the practitioner to immediately know whether this scheme might be appropriate for their intended problem.

The second part of this chapter is the primary contribution of this work and provides a detailed analysis of a large parameter space of possible integrator configurations for Radau-based MCM across a range of test problems. It has been shown that not all configurations of MCM schemes have equal performance, even for a given order. In particular, examples provided show evidence of some MCM configurations recurringly outperforming even higher-order instances, thereby exposing the existence of better performing configurations for particular integration conditions, which can be appropriately exploited when these integrations are to be performed routinely or repetitively. It is specifically in the number of Jacobian evaluations where performance differences between MCM configurations are most obvious. This is particularly relevant for problems where the computational cost is dominated by Jacobian evaluations, in which case an appropriate selection of MCM configuration that minimises the evaluation frequency could offer a significant boost in performance. Interestingly, for a given order, such configurations are shown to correspond to low-s instances for low-order methods, and transition to high-s configurations as the order rises above a problem-specific cut-off value, due to competing factors discussed in Section 3.5.3. Finally, the role of the predictor used to provide an initial guess for the Newton iterations is also shown to permit additional performance improvements for by reducing the number of Newton iterations required for convergence.

Overall, MCM exhibit a convenient versatility that allows one to tweak the configuration parameters to find a performance-wise optimal k-s set-up for a particular integration, which allows one to tune purpose-specific integrators for repetitive tasks. Future work will look into extending MCM configurations to adaptive schemes where both stepsize and order can be simultaneously changed to meet prescribed error tolerances; interestingly, for MCM instances both k and s allow one to change the order, which introduces an additional degree of freedom that can be advantageous for the design of adaptive strategies.

Chapter 4

Developing the Terrestrial Exoplanet Simulator (TES)

The content of this chapter is based upon the article published in Monthly Notices of the Royal Astronomical Society, Volume 504, Issue 1, June 2021, Pages 678-691. It is available with open access on MNRAS and also on arXiv. The authors of the article are Peter Bartram and Alexander Wittig. I am responsible for the work in the original article but owe a great deal to the Alex's guidance.

In this chapter, I present TES, a new n-body integration code for the accurate and rapid propagation of planetary systems in the presence of close encounters. TES builds upon the classical Encke method and integrates only the perturbations to Keplerian trajectories to reduce both the error and runtime of simulations. Variable step size is used throughout to enable close encounters to be precisely handled. A suite of numerical improvements are presented that together make TES optimal in terms of energy error. Lower runtimes are found in the majority of test problems considered when compared to direct integration using IAS15. I have chosen to make TES freely available. The following introduction condenses the key findings of Chapters 1 and 2 before pivoting to describe how I have taken inspiration from these concepts and reapplied them into a non-symplectic framework to arrive at my new tool, TES.

4.1 Background

Understanding and predicting the motion of the celestial bodies has been an active field of research since the times of Newton and Kepler and is still equally as important

today. Few analytical solutions exist for planetary motion and scholars have instead turned to numerical n-body techniques. N-body problems range from the most general form found in the study of plasma and star cluster dynamics (Aarseth, 1999) through to systems with more inherent structure such as protoplanetary disks (Kokubo and Ida, 1996; Kokubo et al., 1998) or exoplanet systems (Smith and Lissauer, 2009). While integrators exist for the general case, by restricting one's self to systems that exhibit more structure one can leverage it to develop more efficient integration algorithms. In this chapter, I restrict the problem domain to planetary systems whereby there exists a dominant central mass with any number of orbiting bodies. Reiterating previous discussions for clarity, the three problems that any integration method for planetary integration needs to address are:

1. Ensuring that solutions obtained remain accurate over the timescales required, in solar system formation and stability studies typically 10^9 dynamical periods.
2. Ensuring that simulations can be completed within the available computing time.
3. Ensuring that integrators can precisely model close encounters between objects.

Moreover, Chapter 2 identified the sources of numerical errors present when performing numerical integrations. In it, we saw that when round-off error can be made to dominate, specific numerical techniques can be used to ensure that the distribution of errors are symmetrical. This has led to the creation of integration schemes that are optimal in the sense that they follow Brouwer's law (Brouwer, 1937) and exhibit a growth in relative energy error over integrator time, t, proportional to $\sqrt{t}$, and are therefore suitable for long-term integrations. Despite being optimal in the sense of Brouwer's law, these schemes are computationally expensive and typically require upwards of a thousand evaluations of the force function per orbit.

There are less computationally intensive means of ensuring invariants are preserved in long-term celestial mechanics integrations. For example, symplectic methods (Forest and Ruth, 1990; Kinoshita et al., 1990; Saha and Tremaine, 1992) can be used to place an upper bound on the truncation error of integrations by solving a system governed by a Hamiltonian that is slightly perturbed from that of reality. The use of symplectic methods ensures that the Poincaré invariants are conserved which in turn has favourable properties for energy and angular momentum conservation. The WH map has, and continues to be, the workhorse of the field. In planetary systems, the WH map exploits the dominant contribution to the dynamics by the star and uses the fact that *secondaries*, i.e. bodies in orbit around a more massive *primary* body such as

the Sun, move on perturbed Keplerian trajectories to split the system Hamiltonian into separate Keplerian and perturbation terms that can be solved independently within a time step. This splitting allows for the WH map to make only twenty evaluations of the force per orbit for typical applications that require only a moderate level of precision. Obtaining the solution for the Keplerian term analytically requires the solution of Kepler's equation (Battin, 1987). There are many choices for solving Kepler's equation but universal variables (Battin, 1987) are favoured for their versatility. Stumpff functions are typically used (Danby, 1992) although other recent works shows that unbiased results can also be obtained without them (Wisdom and Hernandez, 2015).

One drawback of the WH mapping, and symplectic schemes in general, is that they must use a fixed time step to ensure that symplecticity remains unbroken. Whilst not a problem if bodies remain well separated it means integrators are unable to handle close encounters which are typically defined as encounters between bodies within one Hill radius r_H. The ability to handle close encounters is highly important in celestial mechanics for modelling many problems. This includes: the threat of asteroids to the Earth (Giorgini et al., 2008), the behaviour of exoplanet systems after an instability event (Rice et al., 2018; Bartram et al., 2021), and the planet formation process itself (Davies et al., 2014). Options exist that enable invariants to be conserved during close encounters when using the WH map (Duncan et al., 1998; Chambers, 1999; Rein et al., 2019b) but they fail to obtain the true trajectories of bodies during the encounter for the typical step sizes chosen. Therefore, to study realistic trajectories of bodies during close encounters traditional integrators such as Bulirsch-Stoer (Bulirsch and Stoer, 1966) or Everhart's RADAU (Everhart, 1985) scheme are typically used.

In this chapter, I introduce my novel method, called the Terrestrial Exoplanet Simulator (TES), that aims to combine the accuracy and performance benefits of the analytical solution of the Keplerian motion, as found in symplectic schemes, with the flexibility of traditional integration schemes. I build upon the classic scheme of Encke, see e.g. (Wiesel, 2010), to create a perturbation method that can be integrated with a traditional integrator. Importantly, I show that in this framework it is possible for close encounters to be handled precisely and with a reduction in computational cost as compared to performing a *direct integration* using the full n-body equation of motion. I show that through careful handling of round-off error through compensated summation (Kahan, 1965; Higham, 1993) the Encke method can be made optimal in the sense that it follows Brouwer's law with a relative energy error comparable to that of IAS15. I offer two implementations of TES, the *standard configuration* where the implementation is purely in double precision floating point arithmetic and the *extended configuration* where 80 bit extended precision, i.e long doubles, available on,

at least, all x86 instruction set based systems, are used in the Kepler solver. The latter implementation enables a further reduction in error of an order of magnitude for the same computational cost when compared to using IAS15. Both TES implementations are capable of accurately handling close encounters and, as we will see in Chapter 5, TES has already been used to study exoplanet evolution in the presence of collisions (Bartram et al., 2021).

During the development of TES, a similar method, called EnckeHH, has been published by Hernandez and Holman (2020). Both methods have been developed independently, and while they use similar concepts, EnckeHH focuses on achieving Brouwer's law for a fixed step size when integrating with IAS15.

I begin in Section 4.2 with a description of the components making up TES. Section 4.2.2 contains the derivation of the equations of motion used. The solution of the analytical part of our model is described in Section 4.2.3 and the numerical part in Section 4.2.4. Section 4.3 contains details about specific numerical techniques implemented to ensure TES follows Brouwer's law. I show the impact of each of these techniques in Section 4.4. Beginning in Section 4.5, the second half of this chapter contains a series of numerical experiments. In particular, Section 4.5.3 contains long-term integrations and Section 4.5.4 shows the performance in the presence of close encounters. I offer some concluding remarks in Section 5.6.

4.2 TES model

In this section, I begin by giving an overview of the method and then describe in detail the mathematical model, coordinate system, and force function used in TES.

4.2.1 General Encke method

In the Encke method, the position of a given body $\hat{\mathbf{q}}$ is made up of two terms:

1. $\mathbf{q}$, the two-body reference trajectory,
2. $\delta\mathbf{q}$, the perturbation to this two-body motion,

such that $\hat{\mathbf{q}} = \mathbf{q} + \delta\mathbf{q}$. Figure 4.1 illustrates this concept, where the reference trajectory, $\mathbf{q}$, can be obtained analytically at any future time by solving Kepler's equation

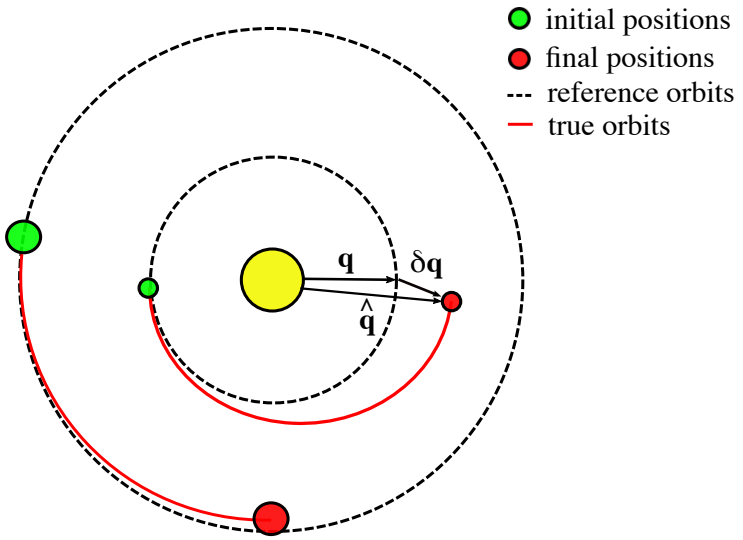


Figure 4.1: A three-body Encke method. For the inner planet, the position on the reference orbit, $\mathbf{q}$, is shown along with the perturbation from it, $\delta\mathbf{q}$. The position on the true orbit, $\hat{\mathbf{q}}$, is also shown. Deviations from the reference orbits are greatly exaggerated for clarity.

and applying the so called f and g functions (Battin, 1987, p. 156). In contrast, the perturbation term, $\delta\mathbf{q}$, must be obtained through numerical integration. When bodies are well separated, the advantages of the Encke method over, e.g., a full n-body integration, stem from the fact that the ratio of $|\delta\mathbf{q}|/|\mathbf{q}|$, henceforth called the *delta ratio*, is much smaller than unity meaning that a given absolute precision in $\hat{\mathbf{q}}$ can be obtained with lower relative precision in $\delta\mathbf{q}$ which means less computation by the numerical integrator. In order to keep the delta ratio small one occasionally needs to update the reference trajectory to the current true orbit, a process called *rectification*. With a single rectification per orbit a delta ratio of 10^{-2} can reasonably be expected for simulations of the outer solar system whereas for the inner solar system this ratio remains below 10^{-4}. With such delta ratios maintained, the analytical solution of the reference trajectory becomes the primary source of numerical error. In order for this method to work, it is crucial that this propagation is precise, in the case of this work this means down to machine precision.

I continue by introducing a general form for creating an Encke method through two arbitrary governing Hamiltonians, which therefore requires the integration of a conservative system without any dissipative effects present. Later, in Section 4.2.2, I use this form to derive our method for two specific Hamiltonians in our chosen coordinate system. I chose to work in canonical coordinates, and throughout this work the true

state vector is denoted by

$$\hat{\mathbf{z}} = (\hat{\mathbf{q}}, \hat{\mathbf{p}})^T$$

where $\hat{\mathbf{q}}$ and $\hat{\mathbf{p}}$ are conjugate position and momentum vectors. Similarly, the reference orbit state vector is

$$\mathbf{z} = (\mathbf{q}, \mathbf{p})^T$$

where $\mathbf{q}$ and $\mathbf{p}$ represent the position and momentum components of the reference orbits. Both $\hat{\mathbf{z}}$ and $\mathbf{z}$ are assumed to be in the same coordinate system. Therefore, the Encke method is

$$\hat{\mathbf{z}} = \mathbf{z} + \delta\mathbf{z}$$

and the perturbation term, henceforth called simply *the deltas*, for which the equations of motion need to be derived, is

$$\delta\mathbf{z} = \hat{\mathbf{z}} - \mathbf{z}. \tag{4.1}$$

The time evolution of $\hat{\mathbf{z}}$ and $\mathbf{z}$ for any conservative system is governed by their respective Hamiltonians $\hat{H}(\hat{\mathbf{z}})$ and $H(\mathbf{z})$. Making use of the canonical structure matrix

$$J \equiv \begin{bmatrix} 0 & +I \\ -I & 0 \end{bmatrix},$$

where I is the identity matrix of appropriate dimensions for a given problem, one can apply Hamilton's equations to yield the equations of motion as

$$\frac{\mathrm{d}}{\mathrm{d}t}\hat{\mathbf{z}} = J\nabla_{\hat{\mathbf{z}}}\hat{H}(\hat{\mathbf{z}}), \quad \frac{\mathrm{d}}{\mathrm{d}t}\mathbf{z} = J\nabla_{\mathbf{z}}H(\mathbf{z}). \tag{4.2}$$

Taking the time derivative of Eq. (4.1) and replacing appropriate terms with those from Eq. (4.2) gives a formula for finding the equations of motion for the deltas themselves as

$$\frac{\mathrm{d}}{\mathrm{d}t}\delta\mathbf{z} = \frac{\mathrm{d}}{\mathrm{d}t}(\hat{\mathbf{z}} - \mathbf{z}) = J\left(\nabla_{\hat{\mathbf{z}}}\hat{H}(\hat{\mathbf{z}}) - \nabla_{\mathbf{z}}H(\mathbf{z})\right). \tag{4.3}$$

4.2.2 Encke method: democratic heliocentric (ENCODE)

In Cartesian coordinates, I define the state vector for this system as

$$\hat{\mathbf{Z}}(t) = (\hat{\mathbf{Q}}(t), \hat{\mathbf{P}}(t))^T$$

where $\hat{\mathbf{Q}}_i(t)$ is the generalised position vector and $\hat{\mathbf{P}}_i(t)$ is the momentum vector conjugate to it. If the bodies are only interacting through mutual Newtonian gravity,

then the time evolution of the system in Cartesian coordinates is governed by the gravitational n-body Hamiltonian (Leimkuhler and Reich, 2005)

$$\hat{H}(\hat{\mathbf{Z}}) = \sum_{i=0}^{n} \frac{|\hat{\mathbf{P}}_i|^2}{2\,m_i} - \sum_{i=0}^{n} \sum_{j=i+1}^{n} \frac{G m_i\, m_j}{|\hat{\mathbf{Q}}_j - \hat{\mathbf{Q}}_i|} \tag{4.4}$$

where the subscript i refers to the i_{th} body in the system. Throughout this chapter, $i = 0$ is reserved to refer to the central body.

There are no obvious Hamiltonian splittings of Eq. (4.4) that allow for the dominant Keplerian motion of the secondary bodies about the primary, due to the dominant central mass, to be isolated from the general evolution of the system. For this, it is necessary to introduce a different coordinate system. There are several coordinate systems available and Hernandez and Dehnen (2017) give a good overview of the canonical coordinate systems for the n-body problem. The Jacobi coordinate system was used by Roy et al. (1988) to create an Encke method to simulate the outer planets of our solar system. In this coordinate system, each body has a reference orbit taken with respect to a different moving centre of mass that depends on the position and mass of all other bodies who's orbits are smaller than its own. Therefore, as the relative size of the orbits of bodies changes in a system it becomes necessary to recalculate reference trajectories with respect to a new moving centre of mass. To avoid this complication and the numerical error it could introduce I instead opt to use democratic heliocentric (DH) coordinates (Duncan et al., 1998) which are common in celestial mechanics (Chambers, 1999; Grimm and Stadel, 2014). In DH coordinates the equations of motion are such that the position of each body is expressed relative to the central body, which I denote with the index zero throughout this chapter. The momentum of each body, however, is expressed relative to the barycentre of the system. The coordinate change from Cartesian to democratic heliocentric coordinates is given by

$$\hat{\mathbf{q}}_i = \begin{cases} \hat{\mathbf{Q}}_i - \hat{\mathbf{Q}}_0, & \text{if } i \neq 0 \\ \dfrac{1}{M}\sum_{j=0}^{n} m_j \hat{\mathbf{Q}}_j, & \text{if } i = 0 \end{cases} \tag{4.5}$$

$$\hat{\mathbf{p}}_i = \begin{cases} \hat{\mathbf{P}}_i - \dfrac{m_i}{M}\sum_{j=0}^{n} \hat{\mathbf{P}}_j & \text{if } i \neq 0 \\ \sum_{j=0}^{n} \hat{\mathbf{P}}_j & \text{if } i = 0. \end{cases} \tag{4.6}$$

where M is the total system mass, i.e. $M = \sum_{j=0}^{n} m_j$. Therefore, $\hat{\mathbf{q}}_0$ and and $\hat{\mathbf{p}}_0$ are the centre of mass and momentum of the system, respectively. The Hamiltonian, $\hat{H}(\hat{\mathbf{z}})$,

as a function of the previously defined state vector, $\hat{\mathbf{z}}$, in these new coordinates is

$$\hat{H}(\hat{\mathbf{z}}) = \hat{H}_{star} + \hat{H}_{kep} + \hat{H}_{pert} \tag{4.7}$$

where

$$\begin{aligned}
\hat{H}_{star} &= \frac{1}{2m_0}\sum_{i=1}^{n} |\hat{\mathbf{p}}_i|^2, \\
\hat{H}_{kep} &= \sum_{i=1}^{n}\left(\frac{|\hat{\mathbf{p}}_i|^2}{2\,m_i} - \frac{G\,m_i\,m_0}{|\hat{\mathbf{q}}_i|}\right), \\
\hat{H}_{pert} &= \sum_{i=1}^{n-1}\sum_{j=i+1}^{n}\frac{G\,m_i\,m_j}{|\hat{\mathbf{q}}_j - \hat{\mathbf{q}}_i|}.
\end{aligned}$$

Each of the three components of $\hat{H}$ are identifiable:

1. $\hat{H}_{star}$, thus called because $-\sum_{i=1}^{n}\hat{\mathbf{p}}_i$ is the barycentric momentum of the star (Duncan et al., 1998),
2. $\hat{H}_{kep}$ is the Keplerian motion of the secondary bodies about the central body,
3. $\hat{H}_{pert}$ is the gravitational interactions between secondary bodies.

An additional fourth term that only depends upon $\hat{\mathbf{p}}_0$ and represents the motion of the centre of mass has been excluded under the assumption of a stationary barycentre, and it is therefore unnecessary to propagate $\hat{\mathbf{q}}_0$ and $\hat{\mathbf{p}}_0$.

Note that Eq. (4.4) can also be used to obtain a Hamiltonian describing a system of particles that only interact with the central body by enforcing that $j = 0$ thereby removing the gravitational interactions between secondaries. After applying the same coordinate transformation from Eqs. (4.5) and (4.6) the reference orbit Hamiltonian, $H(\mathbf{z})$, as a function of the previously defined state vector, $\mathbf{z}$, in democratic heliocentric coordinates reads

$$H(\mathbf{z}) = \sum_{i=1}^{n}\left(\frac{|\mathbf{p}_i|^2}{2\,m_i} - \frac{G\,m_0\,m_i}{|\mathbf{q}_i|}\right).$$

Inserting $\hat{H}(\hat{\mathbf{z}})$ and $H(\mathbf{z})$ into the general Encke form in Eq. (4.3) yields the equations of motion for the deltas in democratic heliocentric coordinates as

$$\delta\dot{\mathbf{q}}_i = \frac{\delta\mathbf{p}_i}{m_i} + \frac{1}{m_0}\sum_{j=1}^{n}\big(\mathbf{p}_j + \delta\mathbf{p}_j\big), \tag{4.8}$$

$$\delta\ddot{\mathbf{q}}_i = \frac{\delta\dot{\mathbf{p}}_i}{m_i} + \frac{1}{m_0}\sum_{j=1}^{n}\big(\dot{\mathbf{p}}_j + \delta\dot{\mathbf{p}}_j\big), \tag{4.9}$$

$$\begin{aligned}\delta\dot{\mathbf{p}}_i = {} & Gm_i\, m_0\left(\frac{(\mathbf{q}_i+\delta\mathbf{q}_i)}{|\mathbf{q}_i+\delta\mathbf{q}_i|^3} - \frac{\mathbf{q}_i}{|\mathbf{q}_i|^3}\right) + \\ & \sum_{\substack{j=1\\ j\neq i}}^{n} G\, m_i\, m_j\, \frac{\big(\mathbf{q}_j+\delta\mathbf{q}_j\big)-(\mathbf{q}_i+\delta\mathbf{q}_i)}{\Big|\big(\mathbf{q}_j+\delta\mathbf{q}_j\big)-(\mathbf{q}_i+\delta\mathbf{q}_i)\Big|^3}.\end{aligned} \tag{4.10}$$

Here, and throughout this work, dots are used to signify the time derivative of a variable. For the reasons discussed later in Section 4.2.4 I take an additional time derivative of $\delta\dot{\mathbf{q}}$ to obtain $\delta\ddot{\mathbf{q}}$. There are several key features to note about these equations. The summation in the trailing term in Eq. (4.10) starts at an index of 1 meaning that it only captures interactions between secondary bodies; the gravitational interaction with the central body, with an index of 0, is captured by the leading term instead. In this term, it can be seen that there is a cancellation between similar terms that depends upon only the size of $\delta\mathbf{q}_i$. Therefore, so long as this term remains small, then $\delta\dot{\mathbf{p}}_i$ will generally also remain small in comparison to the the reference trajectory derivative, $\dot{\mathbf{p}}$. The exception to this are close encounters between secondaries, where this term can grow significantly. Again, it is this reduction in the relative size of the term to be integrated that leads to the performance increase of an Encke method.

Typically, the Encke method as used in astrodynamics assumes massless secondaries, and, as such, that the centre of mass of the central body is coincident with the barycentre, i.e., the location to which the reference orbits are taken is fixed. This is not the case in celestial mechanics, however, due to the relatively high mass ratio of planets in comparison to their host stars, a ratio of approximately 10^{-3} for our solar system. As an example, consider a two-body problem consisting of Jupiter and the Sun. In this case, the elliptical motion of the Sun about the Sun-Jupiter barycentre is captured by the trailing term in Eq. (4.8) as the negative momentum of the star

$$\hat{\mathbf{p}}_{star} = -\frac{1}{m_0}\sum_{j=1}^{n}\big(\mathbf{p}_j + \delta\mathbf{p}_j\big).$$

This shows that even in a purely two-body case there will still be a deviation from the reference trajectory around the Sun over time, and the magnitude of this deviation is inversely proportional to the ratio of the mass of the star m_0 to the mass of the other bodies within the system $m_p \equiv \sum_{j=1}^{n} m_j$. I call this ratio the *system mass ratio* and define it as $\frac{m_p}{m_0}$. Practically, this means that systems with a small system mass ratio have the greatest potential performance gains. Alternatively, in systems with many bodies, an axially symmetric distribution of mass reduces the motion of the central body through a cancellation of terms in p_{star}. Clearly, a distribution such as this is non-physical for planetary systems, but typical accretion disks in planetary formation are symmetric enough to see cancellation.

For the previously discussed reasons, it is no longer necessary to evolve the motion of the central body as a separate set of equations, thereby reducing n by 1. However, whereas a pure n-body integration can take advantage of a second-order formulation of the equations of motion to reduce the number of equations to be integrated, especially in the case where the motion is not velocity dependent, this is no longer the case for these equations where two first-order ODEs, Eqs. (4.8) and (4.10), must instead be integrated. Note that this does not double the computational cost. The additional cost in the RHS is small and scales only as $O(n)$ and therefore the dominant $O(n^2)$ interaction term remains unchanged. Furthermore, the integration procedure itself performs identical operations for each first-order ODE, meaning that vectorisation, either manual or automatically performed by the compiler, can recover some of the additional computational cost. Finally, this does not have a noticeable performance penalty when calculating the reference trajectories.

4.2.3 Analytical solution

In this section, I describe how I solve the two-body problem with universal variables making use of the f and g functions (Danby, 1992). Assuming that the integration of the perturbation terms can be performed in a way that ensures the truncation error is below floating point precision for $\delta\mathbf{z}$ then the overall precision of the scheme depends upon the precision in the solution of the Keplerian motion. There are two reasons for this: firstly, the equations of motion, Eqs. (4.8) and (4.10), depend on the reference trajectories; and secondly, the rectification process is only as precise as the reference trajectories and a lack of precision would therefore accumulate over time. As such, a highly accurate solver implementation that is also non-biased is required to ensure that the energy growth follows Brouwer's law for long-duration integrations. Several modern implementations of Kepler solvers exist that are suitably accurate,

e.g. (Wisdom and Hernandez, 2015; Rein and Tamayo, 2015). I have chosen to follow the implementation presented in WHFAST (Rein and Tamayo, 2015) but with some additional numerical improvements specific to our needs, these are discussed in Section 4.3.2.

Solution of $n-1$ independent two-body problems is required for which the time evolution is governed by $\hat{H}_{kep}$ in Eq. (4.7). Typically, the f and g functions use a reduced mass parameter such that $\mu = G\left(m_0 + m_i\right)$; however, in democratic heliocentric coordinates, used here, the reduced mass is given by $\mu = Gm_0$ (Grimm and Stadel, 2014). In the following discussion I focus on a single two-body problem from the vectors $\mathbf{q}$ and $\mathbf{p}$ above. For the remainder of this section, $\mathbf{q}$ and $\mathbf{p}$ refer to the position and momentum of a single body undergoing Keplerian motion. As a result, I drop the index referring to each body: an index of zero now refers to values at the start of an integration step. After using the WHFAST algorithm to solve for the universal anomaly and obtain the G-functions (Rein and Tamayo, 2015), the f and g functions used are

$$\begin{aligned} f &= -\frac{\mu G_2}{|\mathbf{q}_0|}, \quad \dot{f} = -\frac{\mu G_1}{|\mathbf{q}_0||\mathbf{q}|}, \\ g &= \mathrm{d}t - \mu G_3, \quad \dot{g} = -\frac{\mu G_2}{|\mathbf{q}|} \end{aligned} \tag{4.11}$$

where $\mathrm{d}t$ is the time step. This allows for the solution to the Kepler problem, after a time step of $\mathrm{d}t$, to be obtained from the initial values of position and momenta and a linear combination of them to be applied to each as an update term, $\Delta\mathbf{q}$ and $\Delta\mathbf{p}$, such that

$$\mathbf{q} = \mathbf{q}_0 + \Delta\mathbf{q} = \mathbf{q}_0 + \left(f\,\mathbf{q}_0 + g\frac{\mathbf{p}_0}{m}\right), \tag{4.12}$$

$$\mathbf{p} = \mathbf{p}_0 + \Delta\mathbf{p} = \mathbf{p}_0 + m\left(\dot{f}\,\mathbf{q}_0 + \dot{g}\frac{\mathbf{p}_0}{m}\right). \tag{4.13}$$

When formulated in this manner, the smaller $\mathrm{d}t$, relative to the orbital period, the smaller the size of the bracketed terms and therefore summing them first is more numerically robust. Additionally, TES uses a value of $\mathrm{d}t$ approximately $1/300^{\text{th}}$ of an orbit, roughly an order of magnitude smaller than, e.g., WHFAST and this means that the relative sizes of $\Delta\mathbf{q}$ to $\mathbf{q}_0$ and $\Delta\mathbf{p}$ to $\mathbf{p}_0$ enables compensated summation to be used to further reduce round-off error; our algorithm for this is described in Section 4.3.4.

4.2.4 Numerical solution

While there are many numerical integrators to choose from, I showed in Chapter 2 that a particularly efficient and accurate choice for astrophysical problems is the RADAU scheme of Everhart (1974). A useful feature of RADAU is that once a polynomial is fitted to the force, it is possible to integrate it analytically one or multiple times allowing for both, e.g., velocity and position to be obtained from the same polynomial. Everhart (1985) found that his scheme is best suited for directly integrating second order ODEs, e.g. $\ddot{y} = F(y,t)$, without reduction to a pair of first-order equations. In fact, for the same number of evaluations of the force function he found the solution to second order ODEs could be as much 10^6 times more precise than if they were reduced to first order and integrated. Due to the Hamiltonian formalism I chose to use in the derivation of the equations of motion, I have had to reduce the system to a pair of first-order equations. When using the RADAU scheme to integrate the equations in this form, I also found a reduction in precision. Taking an additional derivative of Eq. (4.8) to obtain $\delta\ddot{\mathbf{q}}$ is inexpensive; however, this is not the case when taking the derivative of Eq. (4.10) to obtain $\delta\ddot{\mathbf{p}}$, and this led me to trying to integrate $\delta\ddot{\mathbf{q}}$ and $\delta\dot{\mathbf{p}}$. Heuristically, I then found that using this pairing of equations did not lead to the same reduction in precision as when both equations are integrated at first order. I cannot provide a solid theoretical reason as to why integrating $\delta\dot{\mathbf{p}}$ at first order does not cause the same reduction in precision. As discussed in Section 4.2.2, the effect of this choice on the computational cost is minor.

In Chapter 2, I described the 9^{th} order RADAU scheme. The following description is similar, however, it describes the 15^{th} order implementation used to integrate only the perturbations within the TES model. I incorporate the improvements made by Rein and Spiegel (2015) in IAS15. Only the process of integrating $\delta\ddot{\mathbf{q}}$ is discussed but a similar process is also followed for $\delta\dot{\mathbf{p}}$.

Simultaneous solution of $3n$ equations of the form

$$\delta\ddot{q} = F(\mathbf{q}, \mathbf{p}, \delta\mathbf{q}, \delta\mathbf{p}, t),$$

one for each directional component of n bodies, is required. To do this, IAS15 expands the acceleration, $\delta\ddot{q}$, in time, t, into a truncated series such that

$$\delta\ddot{q}(t) \approx \delta\ddot{q}_0 + b_0 h + b_1 h^2 + ... + b_6 h^7 \tag{4.14}$$

where $h = t/\mathrm{d}t$, $\mathrm{d}t$ is the size of an integration step, and t_0 and $\delta\ddot{q}_0$ are the time and acceleration at the start of an integration step. The b coefficients are fitted though

an iterative predictor-corrector process that performs an evaluation of $\delta\ddot{q}$ at the start of an integration step, $t = t_0$, and at seven *sub-steps* within the integration step. The sampling locations in time for each sub-step, c_i where $i = 1...7$, are chosen in accordance with Radau quadrature spacings (Radau, 1880) to maximise the order of the scheme. Once the coefficients, b_i, are obtained to a sufficient precision then Eq. (4.14) can be integrated analytically twice to obtain an estimate of $\delta\mathbf{q}$ at the end of a step, $t_1 = t_0 + \mathrm{d}t$, as

$$\delta q(t_1) \approx \delta q_0 + \mathrm{d}t\ \delta\dot{q}_0 + \mathrm{d}t^2\left(\frac{\delta\ddot{q}_0}{2} + \frac{b_0}{6} + \frac{b_1}{12} + \frac{b_2}{20} + \frac{b_3}{30} + \frac{b_4}{42} + \frac{b_5}{56} + \frac{b_6}{72}\right).$$

A similar process is also followed for $\delta\dot{\mathbf{p}}$. I expand $\delta\dot{\mathbf{p}}$ in an analogous fashion to Eq. 4.14 and also truncate at h^7. This series is then analytically integrated once to obtain an estimate of $\delta\mathbf{p}$ at time t_1.

At the start of a step, an analytical continuation of the curve fitted to $\delta\ddot{q}$ via the accurate b_i values calculated during the previous step are used to generate a predictor in the form of the values of b_i to use in the current step. To ensure convergence when iterating to obtain the coefficients b_i a *convergence criterion* is required. I opt to use a convergence criterion similar to that in IAS15 and therefore monitor the change in the final coefficient in our truncated series, i.e b_6, from one iteration to the next, I call this change $\Delta\mathbf{b}_6$ which is a vector containing coefficients for all $3n$ equations. I then compare the maximum change to the maximum magnitude of the reference orbit acceleration, $\ddot{\mathbf{q}}$, and I terminate when

$$\frac{\|\Delta\mathbf{b}_6\|_\infty}{\left\|\ddot{\mathbf{q}}_0\right\|_\infty} < 10^{-15}. \tag{4.15}$$

I find that this criterion performs better in this use case than if $\delta\ddot{\mathbf{q}}$ is used in the place of $\ddot{\mathbf{q}}$ in Eq. (4.15) which is more typical. The typical criterion would ensure that the change in the coefficients in the series expansion of the acceleration of the deltas is precise to floating point precision. By design, in TES, the deltas are much smaller than the reference orbit terms and therefore it is unnecessary to converge this far to achieve a combined relative tolerance in $\hat{\mathbf{q}} + \delta\mathbf{q}$ of 10^{-16} for use within an integration step, e.g. in the predictors.

In contrast to the convergence criterion, as I wish to suppress the truncation error across long duration integrations, it is a requirement that the step size, $\mathrm{d}t$, is chosen such that it controls the truncation error in the series expansion, Eq. (4.14), itself. I

monitor the truncation error for all $3n$ equations which is estimated by ϵ. To do this, I monitor the absolute value of the b_6 coefficients for all $3n$ equations, which I call $\mathbf{b}_6$, relative to the acceleration of the deltas at the start of an integration step to obtain an estimate of the smoothness of the acceleration of the deltas over a given step as

$$\epsilon = \frac{\|\mathbf{b}_6\|_\infty}{\left\|\delta\ddot{\mathbf{q}}_0\right\|_\infty}. \tag{4.16}$$

The value of ϵ is then used to determine the next step size, $\mathrm{d}t_{n+1}$, as

$$\mathrm{d}t_{n+1} = \mathrm{d}t\left(\frac{\mathrm{tol}}{\epsilon}\right)^{1/7} \tag{4.17}$$

where tol is a dimensionless tolerance parameter. I offer no analytical reasoning for choosing a value of tol; however, numerical experiments demonstrating its effect are shown in Section 4.5. I find that a default value of 10^{-6} is suitable for maintaining Brouwer's law for 10^9 dynamical periods for simulations of the inner solar system and for handling close encounters. I have not included a step rejection algorithm as I found little benefit in terms of precision. One drawback of this choice is that an inappropriate choice of initial step size is not automatically handled. To remedy this problem I always take an initial step size that is one hundredth of the shortest orbital period which works in the majority of cases. Further work here could also include a graceful method of handling edge cases such as detecting simulations that begin during a close encounter and appropriately notifying the user.

4.2.5 Rectification

Rectification is the name given to the process whereby a new reference trajectory is taken by adding the current reference trajectory together with the deltas. The rectification algorithm used can be found in Section 4.3.4. Rectification is important and the frequency with which is it performed has two conflicting effects on the efficiency of the scheme that must be balanced. Firstly, because rectification causes the deltas to be set to zero, the analytical continuation used to obtain a prediction of the values of b_i becomes much less precise in the integrator during the subsequent step and therefore increases the number of iterations required for convergence. Secondly, in contrast, rectifying causes the size of the deltas to be reduced to zero and therefore reduces the computational cost of the integrator in many subsequent steps. I experimented with several rectification schemes based upon the size of the deltas relative to the reference trajectories but found the optimal cutoff value depends largely upon

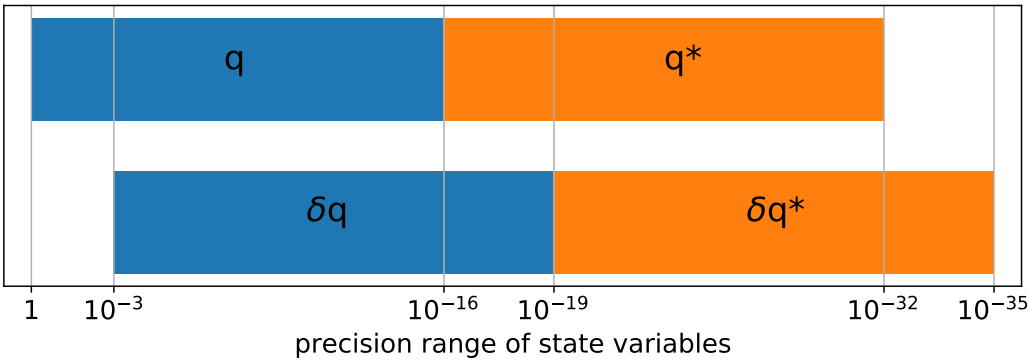


Figure 4.2: Precision ranges of terms within TES using an example where q is of unity magnitude. The size of $\delta\mathbf{q}$ is chosen to be representative of an approximate delta size for a system mass ratio of the Sun to Jupiter.. The blue area shows a double precision floating point variable and the orange area shows the range covered by the associated compensation variable. The cross-hatched area indicates the key region of precision where extended precision floating point arithmetic can be used to improve the overall performance of TES.

the mass ratio of the system. Ultimately, I found that a scheme where a rectification for all particles is performed between once and twice per orbit of the shortest period body is simple to implement and provides a good balance between the two effects. In the future, an individual rectification scheme based on, e.g., the orbital period of each planet could be worthy of investigation. I choose to rectify according to the golden ratio and perform 1.618... rectifications per orbit of the body with the smallest period in the system to help avoid any possible bias effects due to resonances. I combine this with a fall-back method whereby I also rectify if the delta ratio exceeds a given value, typically 10^{-3}.

4.3 Implementation details

In this section, I introduce a collection of numerical implementation features that improve the overall energy conservation of TES for long integration times and enable it to handle close encounters. A key component used through this work is compensated summation (Kahan, 1965). Compensated summation allows for the error made during the summation of two double precision floating point numbers to be obtained and kept as an additional double precision number, known as the compensation variable. The error made is then subtracted from any future additions. This is applicable, e.g., when performing the final update of the state vector in an integration. I use this technique to ensure a symmetrical distribution of round-off error and minimise

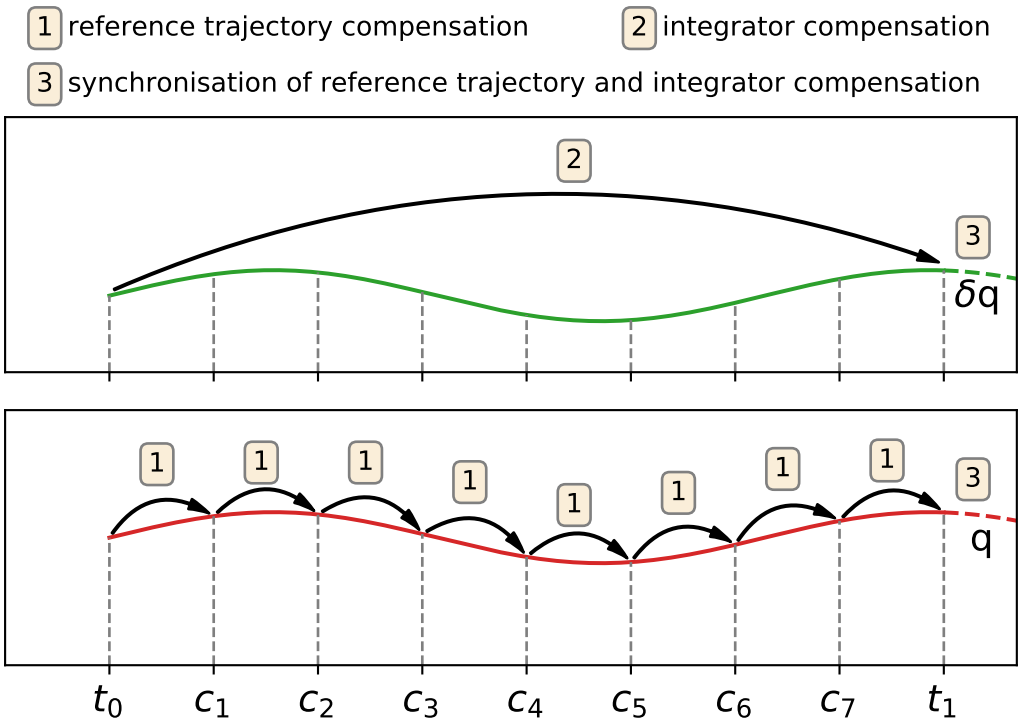


Figure 4.3: Locations in a single step of the RADAU integrator, beginning at t_0 and ending at t_1, where compensated summation is applied. Each value of c_i is an integrator sub-step location at which a reference trajectory must also be calculated. The bottom panel shows the calculation of the reference trajectories where compensated summation is used at each forward step of the Kepler solver, marked by the label 1, to maximise precision. The top panel shows the calculation of the deltas in the integrator. Here, a compensation variable is used to keep track of lost precision across an entire integration step, as shown by label 2. Finally, at the end of the integration step, t_1, label 3, compensated summation is used to combine the separate compensation terms. Compensated summation is also used to reduce error during rectification but this is not shown here.

the total energy error in simulations. Compensated summation requires an additional *compensation variable* be kept for each variable being compensated to keep track of lost precision. To this end, I define a composite datatype, written as $\{\square, \square^*\}$ and comprised of two components: the variable itself, and its compensation variable which is denoted with a star; each component is stored in the computer as a double precision floating-point number. In infinite precision, the true value represented by a composite datatype $\Sigma = \square - \square^*$ where the minus sign is due to the Kahan summation algorithm. Figure 4.2 shows the relative ranges in magnitude that are covered by the variables used in TES. Compensated addition and subtraction operations are denoted by $\oplus$ and $\ominus$ respectively. I find it advantageous to use compensated summation in three ways in addition to internally within the integrator:

1. Propagating the reference trajectories.

2. Combining the reference trajectories and deltas.

3. Ensuring precision is maintained across rectifications.

4.3.1 Encke method: democratic heliocentric (ENCODE)

Equation (4.10) contains a subtraction between two terms of similar size and, owing to the finite precision of floating point numbers, this causes cancellation of significant digits which leads to a large decrease in the relative precision of $\delta\dot{\mathbf{p}}$. The loss of relative precision here depends on how small the difference between the terms is. In particular, after a rectification, the difference can be very small, e.g. 10^{-12} has been observed. In addition to the risk of introducing numerical error when calculating the acceleration, this also poses a problem for the step size control algorithm in Eq. (4.16) and (4.17). The algorithm works by ensuring that the step size is chosen such that the acceleration approximated by the expansion in Eq. (4.14) is smooth to a precision of tol. Oftentimes, the numerical cancellation in Eq. (4.10) means that the required degree of smoothness cannot be met. This can lead to a situation where the step size will shrink uncontrollably as the algorithm tries to shrink the step size further to reach an unattainable smoothness. To remedy this situation, it is possible to reformulate the problematic term to avoid the subtraction of like terms and rewrite the term $\delta\dot{\mathbf{p}}$ as (Battin, 1987, p. 449)

$$
\begin{aligned}
u &= \frac{\delta\mathbf{q}\cdot(\delta\mathbf{q}-2\hat{\mathbf{q}})}{\hat{\mathbf{q}}\cdot\hat{\mathbf{q}}},\\
v &= \frac{-u\left(3+3u+u^2\right)}{1+(1+u)^{\frac{3}{2}}},\\
\delta\dot{\mathbf{p}}_i &= \frac{Gm_i m_0}{|\hat{\mathbf{q}}_i|^3}\left(v\hat{\mathbf{q}}-\delta\mathbf{q}\right)+\sum_{j=1}^{n} G\,m_i\,m_j\,\frac{\hat{\mathbf{q}}_j-\hat{\mathbf{q}}_i}{|\hat{\mathbf{q}}_j-\hat{\mathbf{q}}_i|^3}.
\end{aligned}
$$

where $\delta\dot{\mathbf{p}}$ is now obtained without loss of significance. This is the equation that I have implemented and it decreases the numerical error as well as preventing a step size lockup from occurring.

4.3.2 Analytical solution

Due to the relative size of the reference trajectories $\mathbf{q}$ compared to the deltas $\delta\mathbf{q}$, the dominant contribution to error growth stems from errors in $\mathbf{q}$. To minimise the potential contribution from round-off errors, I have used compensated summation in the

final update step of the f and g functions, Eq. (4.12) and (4.13). With compensated summation they become

$$\begin{aligned} \{\mathbf{q}, \mathbf{q}*\} &= \{\mathbf{q}_0, \mathbf{q}_0*\} \oplus \Delta\mathbf{q}, \\ \{\mathbf{p}, \mathbf{p}*\} &= \{\mathbf{p}_0, \mathbf{p}_0*\} \oplus \Delta\mathbf{p}. \end{aligned} \tag{4.18}$$

Due to the relatively small update terms, $\Delta\mathbf{q}$ and $\Delta\mathbf{p}$, this allows for the value of $\mathbf{q}$ and $\mathbf{p}$ to be maintained to machine precision across Kepler solver steps. Owing to non-linearity when calculating the G-functions it is more precise to take k steps of size τ than one step of size $k\tau$. I therefore go further and minimise the value of $\mathrm{d}t$ in Eq. (4.11) (the simulation time passed since the f and g function basis vectors were calculated) which in turn decreases the size of $\Delta\mathbf{q}$ and $\Delta\mathbf{p}$. To do this, I only ever take a single step in the Kepler solver before calculating new basis vectors, i.e., I calculate new basis vectors at the start of each step as well as at each sub-step required by the integrator. The locations in time that I perform both the compensation in Eq. (4.18) and a recalculation of basis vectors are illustrated in Fig. 4.3 and are marked by the label 1. Here, it is shown that the universal variables compensation is used at the start of a step, t_0, at the end of a step t_1, and also at all sub-steps required by the integrator, c_i.

While the standard TES configuration only makes use of double precision floating-point arithmetic throughout, there is also a build configuration that allows for the selected use of extended precision arithmetic through the *C long double* datatype. I only use extended precision to perform the entirety of the reference trajectory calculations, and I achieve an improvement of energy conservation of an order of magnitude for long duration integrations. Using extended precision sparingly like this allows the compiler to optimise the majority of the code to use single instruction multiple dispatch (SIMD) operations which are not generally available for extended precision variables on Intel or AMD64 hardware.

Figure 4.2 shows an example of the relative scales of the various state variables used within TES: the state vector variables are in blue and the compensation variables are in orange. The cross-hatched area shows the extra precision required in the reference trajectory, $\mathbf{q}$, such that the extra relative precision in $\delta\mathbf{q}$ can be used to create a scheme with a round-off error, per step, of 10^{-19}, assuming that $\delta\mathbf{q}$ remains suitably small. The cross-hatched area is also exactly the extra precision that can be obtained through the use of long doubles in the calculation of the reference trajectories, and, as will be shown in Section 4.4, allows for a reduction in energy violation of over an order of magnitude.

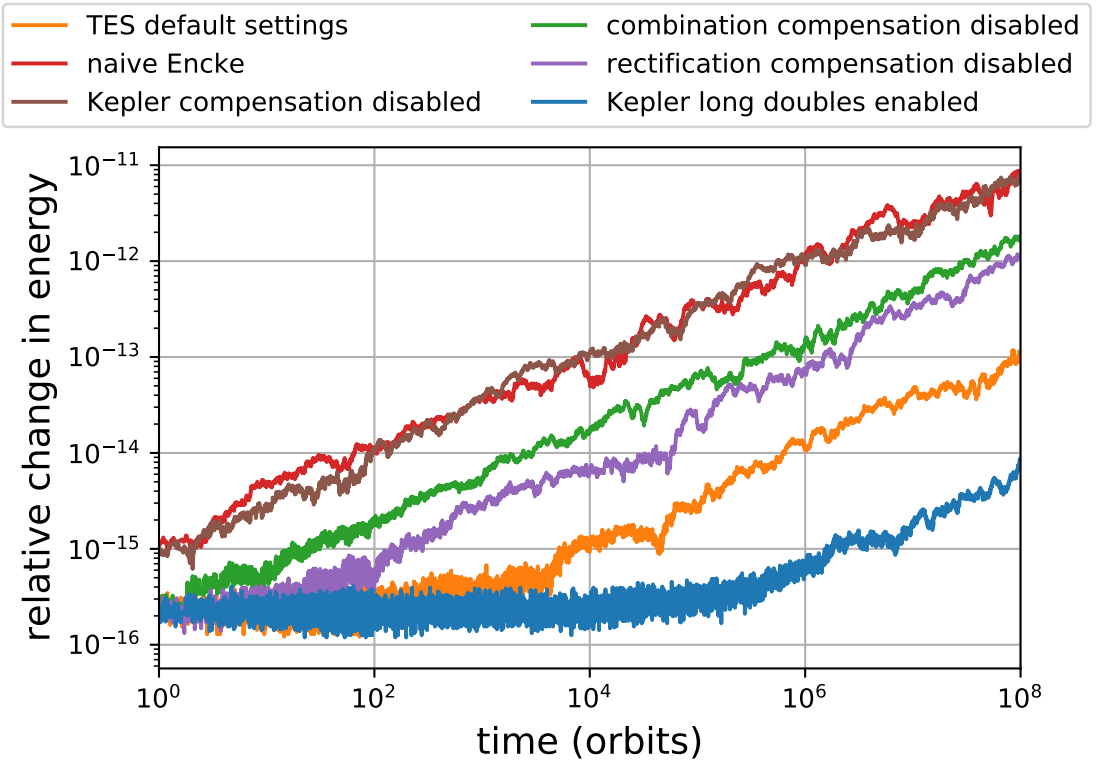


Figure 4.4: The effect of each numerical implementation technique on the relative change in energy, dE/E, for simulations of the inner solar system over 10^8 Mercury orbits. All results plotted are the RMS of twenty realisations of the initial conditions randomly perturbed on the order of 10^{-15}. All results use the double precision implementation of TES unless stated otherwise. "TES default settings" has all compensation features enabled, while "naive Encke" has all compensation features disabled.

4.3.3 Numerical solution

In a similar fashion to Eq. (4.18), the numerical integrator also obtains a compensation variable at the end of a step for each variable being integrated, as is shown in the top panel of Fig. 4.3. In our case, this means that at the end of an integration step I obtain $\{\delta\mathbf{q}, \delta\mathbf{q}*\}$ and $\{\delta\mathbf{p}, \delta\mathbf{p}*\}$. I therefore have two separate sets of compensation variables: one for the reference trajectories, $\mathbf{q}*$ and $\mathbf{p}*$, and one for the deltas, $\delta\mathbf{q}*$ and $\delta\mathbf{p}*$; however, these two sets must be combined at the end of a step to ensure the correct error is used in the subsequent step. Combination of compensation variables is performed at the end of each step as can be seen by label 3 in Fig. 4.3. It is achieved

through the following algorithm:

$$\begin{aligned}
&\boldsymbol{\psi}* \leftarrow 0,\\
&\{\mathbf{q}, \boldsymbol{\psi}*\} \leftarrow (\{\mathbf{q}, \boldsymbol{\psi}*\} \ominus \delta\mathbf{q}*) \ominus \mathbf{q}*,\\
&\delta\mathbf{q}* \leftarrow 0, \quad \mathbf{q}* \leftarrow 0,\\
&\{\delta\mathbf{q}, \mathbf{q}*\} \leftarrow \{\delta\mathbf{q}, \mathbf{q}*\} \ominus \boldsymbol{\psi}*
\end{aligned}$$

where $\boldsymbol{\psi}*$ is a temporary summation variable for each body. This algorithm begins by combining the two compensation variables $\mathbf{q}*$ and $\delta\mathbf{q}*$ into a new compensation variable $\psi*$. This is done as a compensated subtraction into the reference trajectory $\mathbf{q}$ in case the subtraction causes the range of the double precision variable $\psi*$ to overlap with $\mathbf{q}$. After this, the range of $\psi*$ now overlaps with the least significant region of $\delta\mathbf{q}$ and a third compensated subtraction is therefore used to update $\delta\mathbf{q}$ accordingly. In this final subtraction, the reference trajectory compensation variable, $\mathbf{q}*$, is used to store the final summation error such that it can be used immediately in the subsequent Kepler step. The same process is also used for the momentum terms, $\mathbf{p}$ and $\delta\mathbf{p}$. Note that this process differs from rectification, described next, as only the reference trajectories are used here.

4.3.4 Rectification

Compensated summation can also be applied to maintain precision across a rectification by following a very similar algorithm:

$$\begin{aligned}
&\boldsymbol{\psi}* \leftarrow 0,\\
&\{\mathbf{q}, \boldsymbol{\psi}*\} \leftarrow \{\mathbf{q}, \boldsymbol{\psi}*\} \oplus \delta\mathbf{q},\\
&\{\mathbf{q}, \boldsymbol{\psi}*\} \leftarrow \{\mathbf{q}, \boldsymbol{\psi}*\} \oplus \mathbf{q}*,\\
&\delta\mathbf{q} \leftarrow -\boldsymbol{\psi}*\\
&\mathbf{q}* \leftarrow 0, \quad \delta\mathbf{q}* \leftarrow 0
\end{aligned}$$

where again $\boldsymbol{\psi}*$ is a temporary summation variable for each body. This algorithm begins by performing the rectification process by summing the reference trajectories and deltas together and using temporary summation variables to capture any lost precision from the addition. The reference trajectories, $\mathbf{q}$, are then refined using the compensation variables $\mathbf{q}*$ and $\delta\mathbf{q}*$ with any lost precision being captured again by $\boldsymbol{\psi}*$. Due to the sign convention used in the compensated summation this means that

$-\boldsymbol{\psi}*$ now contains the rectified value of the deltas which is then placed into $\delta\mathbf{q}$. This process is also used to rectify the momentum $\mathbf{p}$.

4.4 Validation of implementation details

In the previous section, I described four key numerical features present in TES, in summary they are:

1. Kepler compensation described in Section 4.3.2.
2. Kepler long doubles described in Section 4.3.2.
3. Combining compensation variables described in Section 4.3.3.
4. Rectification compensation described in Section 4.3.4.

Except for Kepler long doubles, the default configuration of TES enables all of these features. Figure 4.4 shows the effect of disabling compensation at various points in TES upon the energy conservation in a simulation of the inner planets of our solar system over a period of 10^8 Mercury orbits. The data plotted is the RMS of twenty realisations of the initial conditions randomly perturbed on the order of 10^{-15}. The modification to the force function in Section 4.3.1 to avoid numerical issues due to cancellation of similar size terms is enabled for all configurations as without it a step size lockup can occur.

As a baseline for the performance without any numerical improvements a *naive Encke* implementation is included that has no numerical improvements other than the reformulation of the acceleration. The default configuration of TES using only double precision with all compensated summation schemes enabled is shown under *TES default settings*. Here, the conservation of energy for TES is two orders of magnitude better than for the naive Encke scheme. If the use of extended precision floating point variables is permitted in the Kepler solver, then the energy conservation in TES can be further improved by an order of magnitude compared to the default configuration. Disabling individual compensation schemes results in energy conservation at least ten times worse than that of TES with default settings. In the worst case, when compensated summation is not used in the final update step of the f and g functions, the conservation of energy can be as poor as just using the naive Encke method which

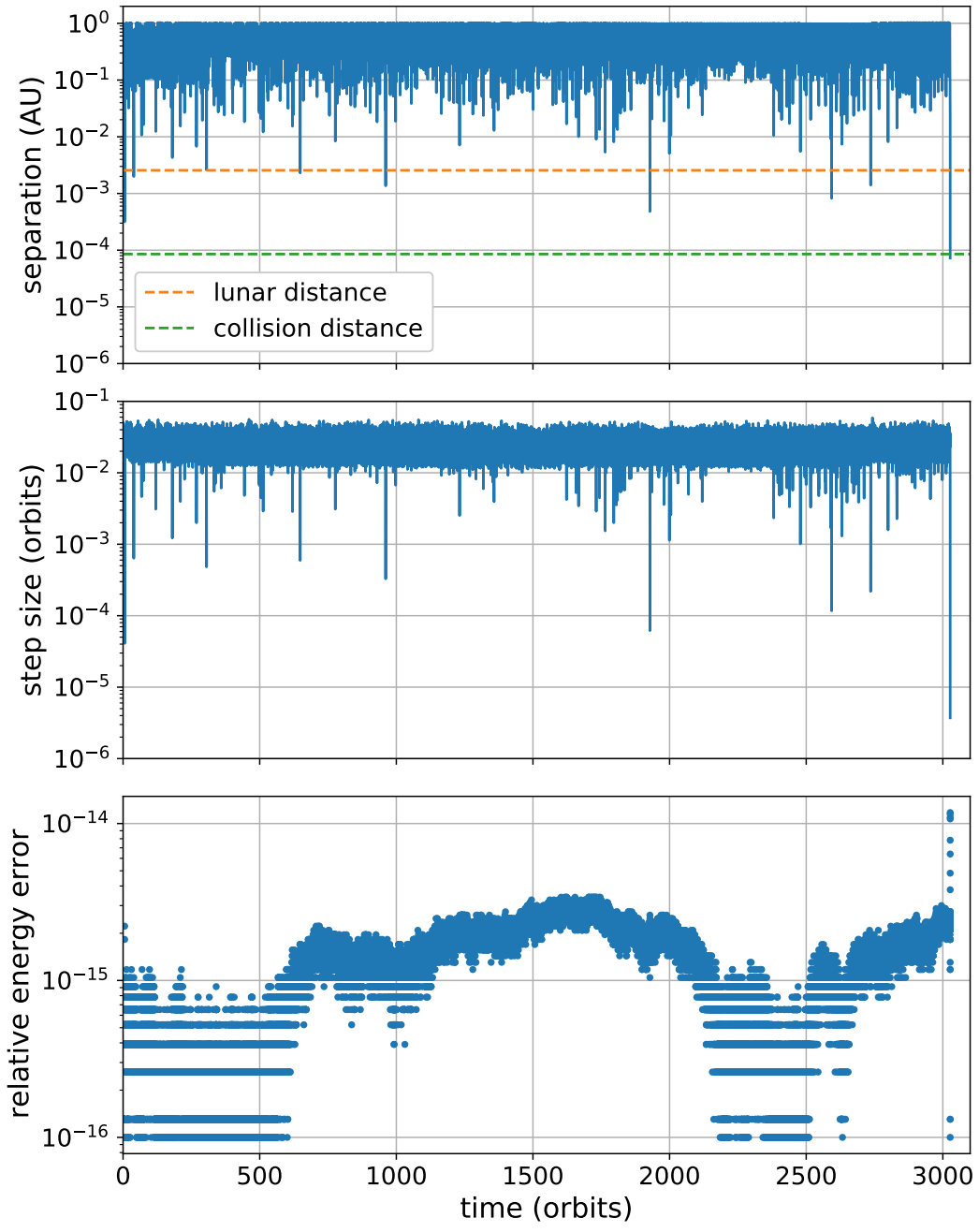


Figure 4.5: A series of close encounters leading up to a collision between Earth mass and radius planets orbiting a solar mass star at roughly 1 AU. The top panel shows the minimum separation between bodies over time. The orange line shows the separation between the Earth and Moon, and the green line shows the separation representing a collision between planets. The central panel shows the step size used by TES. The bottom panel shows the relative energy error over the same time span. The final small change in energy is due to integrating all the way to collision.

highlights how important a precise solution to the dominant Keplerian motion is to schemes of this nature.

To ensure that TES can handle close encounters, I ran a simulation of three Earth mass planets orbiting a solar mass star at 1 AU as per Bartram et al. (2021). Planets are tightly packed and over time the system becomes unstable causing close encounters between them. Figure 4.5 shows a series of these encounters leading up to a collision. Firstly, in the top panel, nine separate encounters cause two of the planets to pass closer to one another than the Moon is to the Earth. The last of these encounters results in a collision between planets. Next, in the central panel, the step size controller shrinks and subsequently expands the step size appropriately to cope with the close approaches. Finally, in the bottom panel, the relative energy error can be seen to perform a random walk close to machine precision throughout all but the final encounter where a small increase in energy is present owing to integrating all the way to collision.

These examples validate that the TES model derived in Section 4.2 and implemented as described in Section 4.3 is indeed capable of performing highly accurate long-term integration as well as handling close encounters between bodies effectively.

4.5 Numerical experiments

To further investigate the performance of TES in a variety of settings, in this section I perform a series of numerical experiments. I also provide comparisons with a number of other integrators. The schemes used are: TES (double); TES (long double), which makes use of long doubles in the Kepler solver; naive Encke; IAS15 (Rein and Spiegel, 2015) from the REBOUND package (Rein and Liu, 2012); and Bulirsch-Stoer (Bulirsch and Stoer, 1966) as well as the hybrid (Chambers, 1999) integrators from the MERCURY package. Table 4.1 contains the default tolerances used throughout these experiments unless otherwise specified. In the case of the TES, naive Encke and IAS15 schemes, the tolerances used are the recommended defaults. All runtime measurements were performed on an Intel Core i7-6700 CPU running at 3.4 GHz.

4.5.1 Efficiency mass dependence

As discussed previously, the magnitude of the deltas in comparison to the magnitude of the reference trajectory, the delta ratio, must be kept small to maximize the efficiency

Table 4.1: Summary of all default integrator tolerances used. TES, naive Encke and IAS15 tolerances are the recommended defaults.

Tool	Default Tolerance
TES (double)	10^{-6}
TES (long double)	10^{-6}
naive Encke	10^{-6}
IAS15	10^{-9}
Bulirsch-Stoer	10^{-14}
hybrid	10^{-14} with 20 steps per orbit

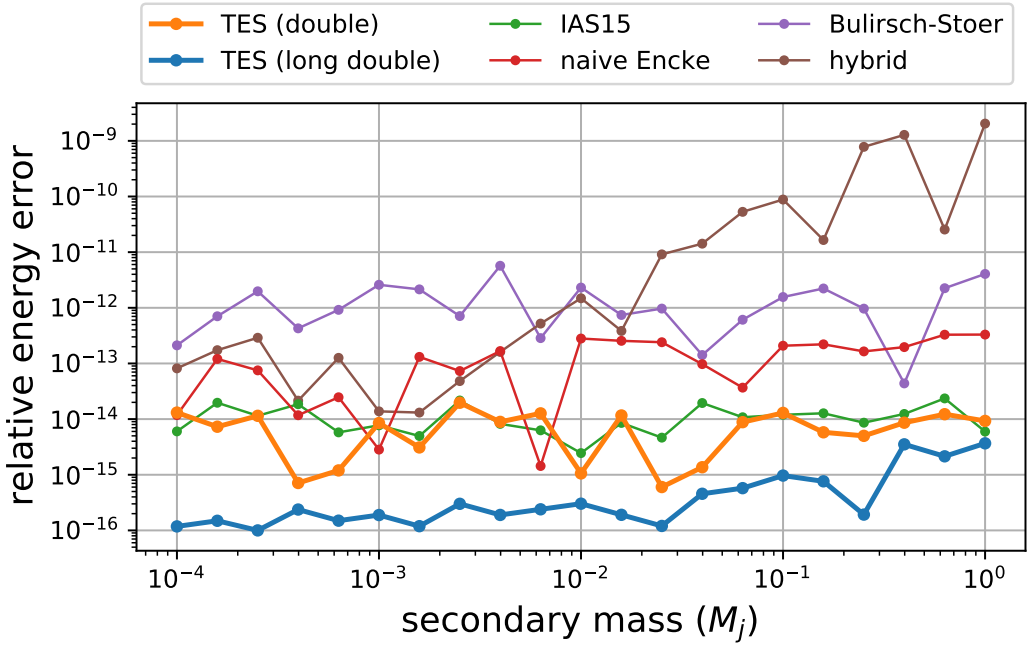


Figure 4.6: Relative energy error of simulations of circular two-body systems over 10^4 orbits for all integrators. The primary is a solar mass star and the mass of the secondary is varied across a range coincident with that of our solar system. The secondary mass is expressed in units of Jupiter's mass, M_j. The Encke based methods must account for the motion of the central body and the two-body problem is therefore still an appropriate test case.

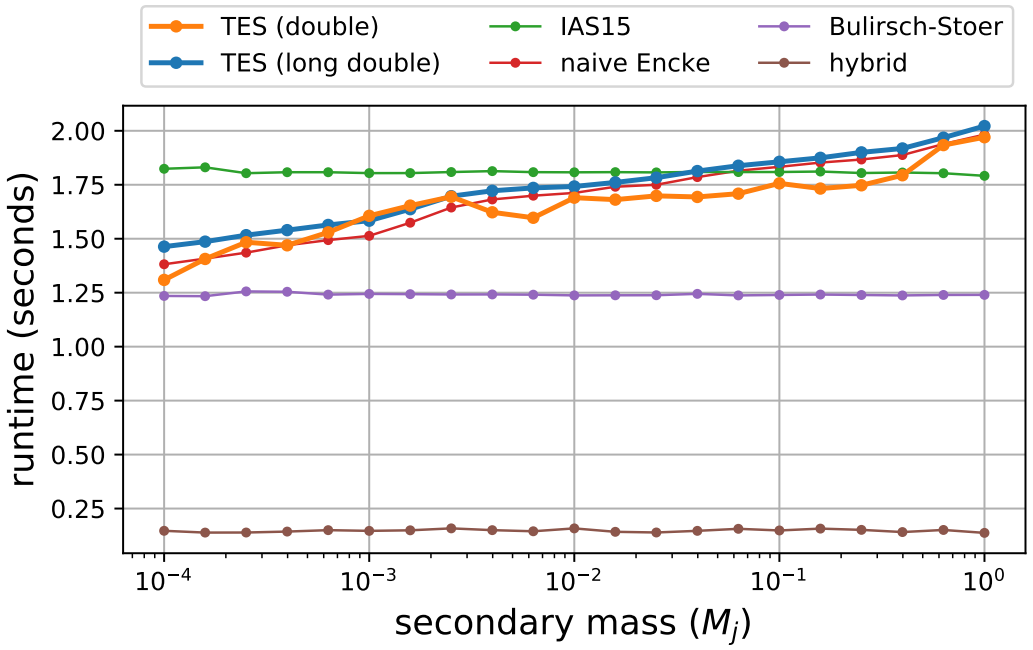


Figure 4.7: Runtime of simulations of two-body systems over 10^4 orbits for all integrators. The primary is a solar mass star and the mass of the secondary is varied across a range coincident with that of our solar system. The secondary mass is expressed in units of Jupiter's mass, M_j. The Encke based methods must account for the motion of the central body and the two-body problem is therefore still an appropriate test case. Each data point is the average of twenty identical integrations.

of an Encke method. Simultaneously, one must not rectify too frequently as rectifications degrade the precision of the predictor in the subsequent step and thus incur a performance penalty. In the absence of close encounters, the dominant contribution to the acceleration of the deltas is related to the motion of the central body which in turn depends on the system mass ratio. Therefore, this experiment is designed to understand in which region of system mass ratio TES is most effective.

I perform integrations of twenty-one two-body problems for our full selection of integration packages. I have opted to examine the range of system mass ratios that can be found in our own solar system if each planet is taken in isolation with the Sun. Therefore, this experiment ranges from a secondary of Jupiter mass, M_j, down to a mass of $10^{-4}M_j$, approximately equal to that of Mercury. The samples across the range of masses are logarithmically spaced, and the primary is always a solar mass star. The secondary body is placed on a circular, co-planar orbit at 1 AU and integrations are performed for 10^4 orbits. Runtime is calculated as the mean of twenty identical integrations for each two-body problem for each integrator.

Figure 4.6 shows the relative energy error achieved in these experiments. Here, except for the hybrid scheme, I find no dependence between the relative energy error and the system mass ratio, simply meaning that the error control algorithms within each integrator are performing as expected. However, I do find large differences in the precision of the various integration schemes. In particular, the Bulirsch-Stoer, hybrid and naive Encke schemes all fail to reach the regions of highest energy conservation. In contrast, the schemes that are floating point arithmetic aware, i.e, TES and IAS15, are much more precise. TES (double) and IAS15 have almost identical performance across the entire range of system mass ratios examined. TES (long double) is the best performer overall and outperforms TES (double) and IAS15 by up to two orders of magnitude.

Figure 4.7 shows the runtime for the same experiments where a cluster of curves can be seen as well as the hybrid scheme which is between six and eight times faster than the non-symplectic schemes. I find that the standard deviation in runtime across all TES realisations is 146 ms. The Bulirsch-Stoer, hybrid and IAS15 schemes can be seen to exhibit no dependency of the runtime on the system mass ratio. However, all Encke based schemes, i.e., TES and naive Encke, show a positive correlation between the runtime and the system mass ratio, as predicted. An interesting comparison is that of TES (double) and IAS15, where for systems with a smaller mass ratio, TES is able to achieve the same level of precision with only 75% of the computational cost. TES (double) remains more efficient until the mass of the secondary is roughly $10^{-1}M_j$. Therefore, for maximum benefit, TES should be applied to systems with a system mass ratio below this value. Finally, TES (long double) can also be seen to perform well with a runtime slightly greater than that of TES (double) despite having a better conservation of energy of up to two orders of magnitude.

4.5.2 Convergence and runtime comparisons

Next, I study the convergence and runtime of TES in comparison to the wider field of integrators over a period of ten thousand orbits of the innermost planet. In the previous section, I showed that TES is most effective in systems with a low system mass ratio, and I have therefore chosen the inner planets of our solar system as a test problem using initial conditions taken from the NASA Horizons database (NASA, 2021). The tolerance of each of the non-symplectic integrators is varied over a range of values such that the relative energy error no longer converges. TES and IAS15 use a range ending at the recommended operating tolerances in Table 4.1. The hybrid scheme uses a fixed tolerance of 10^{-14} throughout but the step size, which must be

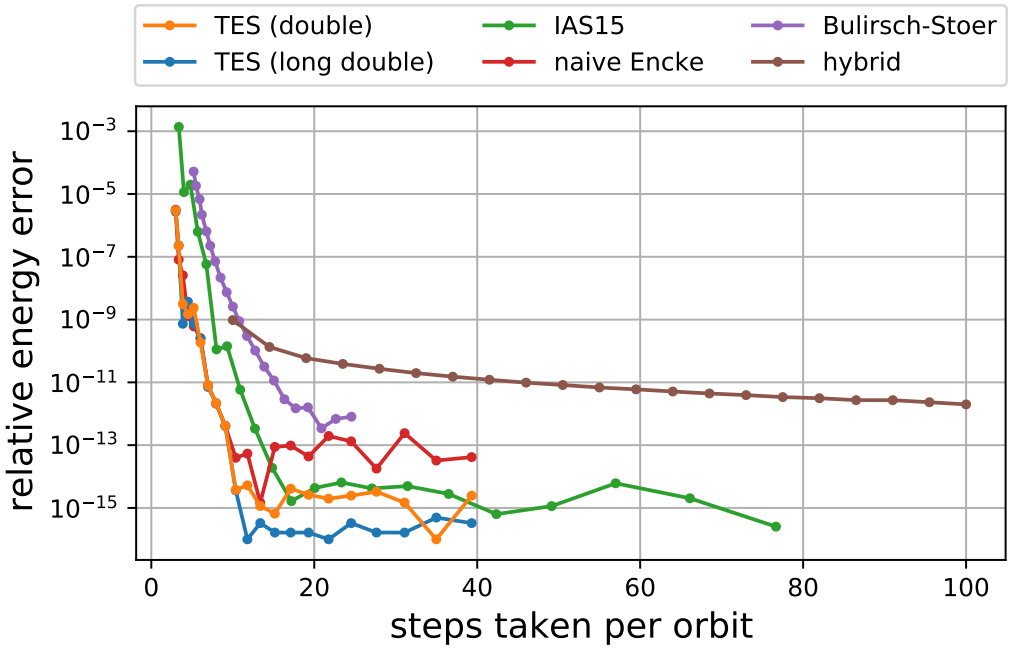


Figure 4.8: Relative energy error against average number of steps per orbit for the inner solar system for 10^4 Mercury orbits.

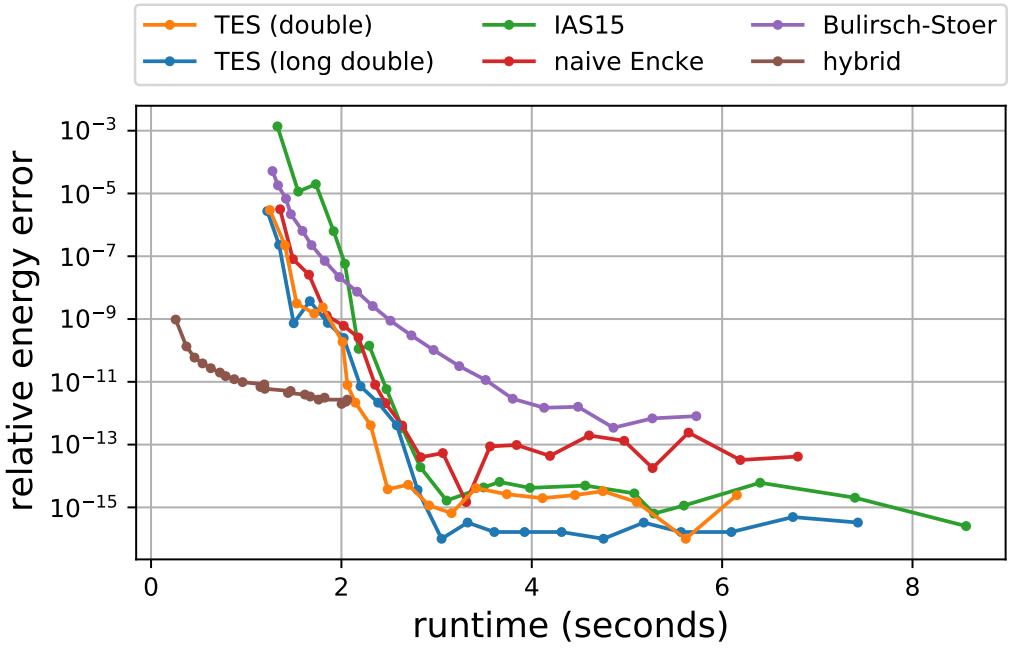


Figure 4.9: Relative energy error against runtime for the inner solar system for 10^4 Mercury orbits. Each data point is the average of twenty identical integrations.

kept constant within an integration, is varied until the relative energy error no longer converges. Runtime is calculated as the mean of twenty identical integrations for each tolerance for each integrator.

Figure 4.8 shows the relative energy error against the number of steps per orbit. Once again, there is a divide between the optimal and non-optimal schemes, TES and IAS15 clearly conserve energy more precisely than the other schemes, with TES (long double) being the most precise by roughly an order of magnitude once the round-off error dominated regime is entered at roughly fifteen steps per orbit. The power of the Encke method can be seen in two places in this plot. Firstly, in the truncation error dominated region, i.e., the region below roughly fifteen steps, where the relative change in energy for a given step size is approximately three orders of magnitude smaller than any of the direct integrations. Secondly, in the furthest right data point for TES and IAS15 where the recommended default tolerances for the Encke based methods yield a reduction in the number of steps taken per orbit when compared to a direct integration. Figure 4.9 shows how these benefits manifest themselves in the runtime. Immediately, the hybrid scheme can be seen to stand apart from the others and is indeed much faster than any of the non-symplectic integrators; however, as I will show in Section 4.5.4 the relatively low precision of the hybrid scheme is not entirely suitable for modelling exoplanet evolution in the presence of close encounters. The Bulirsch-Stoer scheme has a poor runtime in comparison to the other integrators and does not reach the highest levels of precision either. The naive Encke method has a reasonable runtime in the truncation error dominated regime but is not capable of the energy conservation of the optimal floating-point implementation aware methods. For the three remaining integrators, TES (double), TES (long double) and IAS15, the performance is similar. However, for the recommended default tolerances, the furthest right data point for each integrator, TES (double) is the fastest and is approximately 20% faster than the slowest scheme. Interestingly, TES (long double) has the best energy conservation by up to an order of magnitude and has very comparable runtime to IAS15 despite the disadvantage of not being able to use vectorisation in the Kepler solver.

4.5.3 Long-term integrations of the inner solar system

In the field of exoplanet modelling, it is typical for simulations to span a billion dynamical periods. It is therefore of great importance to ensure that propagation schemes follow the optimal error growth of Brouwer's Law, i.e. $\propto \sqrt{N}$ for the relative energy error, where N is the number of steps taken. I have therefore chosen to perform

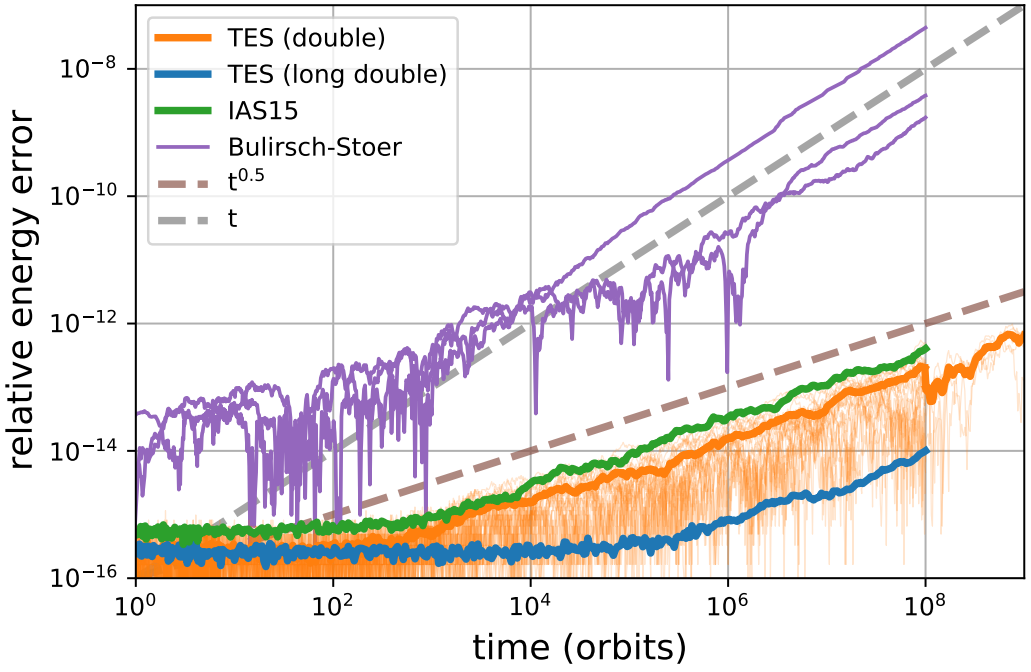


Figure 4.10: Relative energy error of long-term simulations of the inner solar system lasting either 10^8 or 10^9 Mercury orbits using default tolerances for TES and IAS15. Bulirsch-Stoer is included for comparison with manually chosen tolerances of $10^{-13,-14,-15}$ to maximise precision. For TES and IAS15 lines plotted up to 10^8 orbits are the RMS of twenty realisations of the initial conditions perturbed on the order of 10^{-15}. Beyond 10^8 orbits, the line plotted is the RMS of five realisations. Individual realisations are also shown for the TES (double) integrator. Slopes show optimal ($\sqrt{t}$) and linear error growth in brown and grey, respectively.

long-term simulations of the inner solar system to ensure TES conforms to this requirement. I use the tolerance in Table 4.1 for TES and IAS15. In keeping with the original IAS15 experiments (Rein and Spiegel, 2015) and to generate a statistical sample I perform twenty integrations for TES and IAS15 with a perturbation in the initial conditions on the order of 10^{-15}. The RMS of these twenty realisations is plotted in Fig. 4.10. However, only five realisations were used in the region between 10^8 and 10^9 orbits. Additionally, results of three integrations highlighting the performance of the Bulirsch-Stoer integrator are also shown for tolerances of 10^{-13}, 10^{-14} and 10^{-15}. Integrations performed with TES (double) span the full 10^9 Mercury orbital periods whereas all other schemes are terminated after 10^8 Mercury orbital periods to save on computation. Finally, two slopes are included: one in grey marking the linear error growth typically associated with truncation dominated regimes, and another, in brown, showing the optimal error growth associated with the symmetrical distribution of round-off error required for Brouwer's law.

Figure 4.10 highlights that TES (double) and TES (long double) both follow Brouwer's

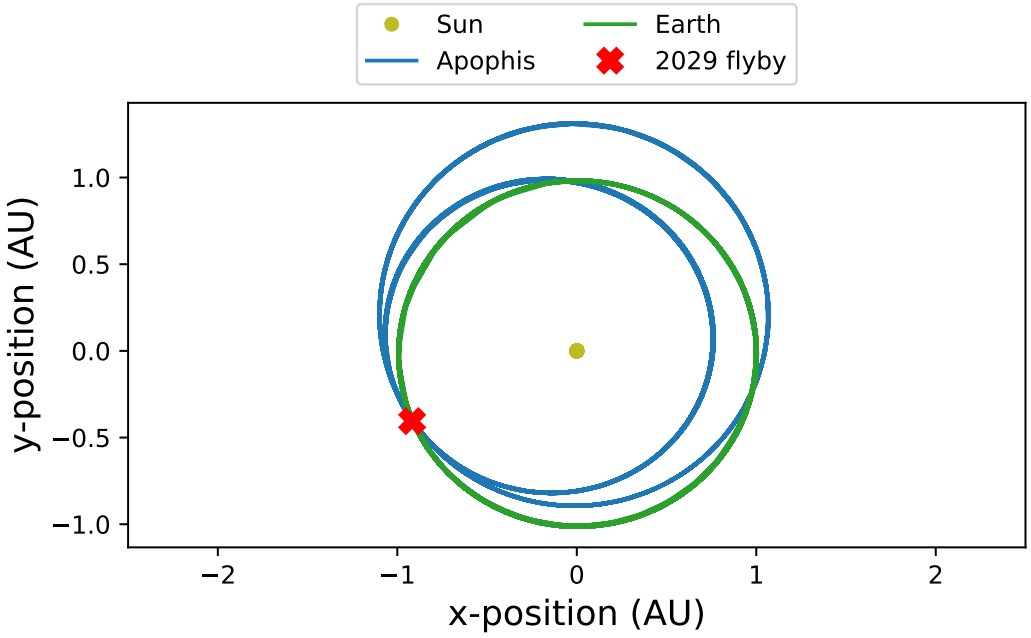


Figure 4.11: Orbits of the Sun, Earth and Apophis over a one-hundred year period from 1979 to 2079. The closest approach is marked and causes a transition of Apophis from Apollo to Atens group.

law for the full integration duration and therefore show that these schemes are well suited to the long duration integrations required in exoplanet modelling. Note that despite the larger steps taken by TES (double), the use of the Encke method has enabled the truncation error growth to be suppressed for the entirety of integrations. When performed by an integrator that follows Brouwer's law, the RMS relative energy error of a suitably large number of realisations $\epsilon \approx C\sqrt{N}$ where C is a constant approximately at floating point precision, i.e. $\approx 10^{-16}$, and N is the number of steps taken. In double precision, TES performs marginally better than IAS15 and I believe this is due to the larger step size taken by the Encke method reducing the $\sqrt{N}$ term. TES (long double) performs only the solution of the Keplerian motion in extended precision, i.e., the integrator and force models use only double precision. However, even with this sparing use of extended precision, TES is able to attain a relative energy error of over an order of magnitude better than if only double precision is used throughout, and importantly, it does this without excessive computational cost. Most notably, TES (long double) is able to integrate for up to 10^5 orbits before there is any noticeable growth in the relative energy error above the floating point floor. Both TES (double) and IAS15 already start to show error growth after just hundreds of orbits.

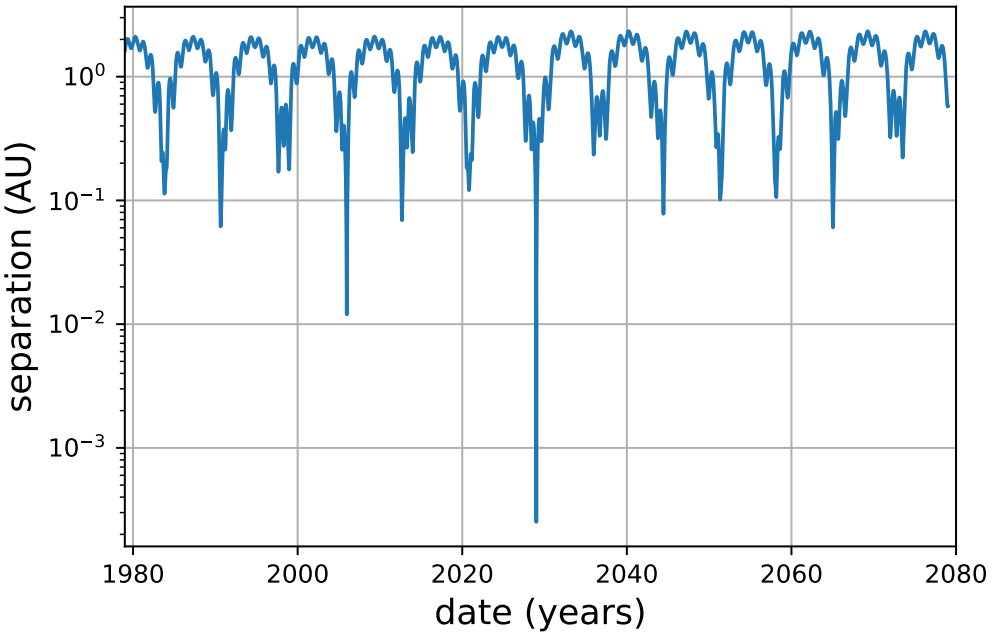


Figure 4.12: Relative separation between the Earth and Apophis over a one hundred year period from 1979 to 2079. The closest approach is approximately 2.5×10^{-4} AU or roughly 17,000 km, well within the geosynchronous orbital altitude at 35,786 km.

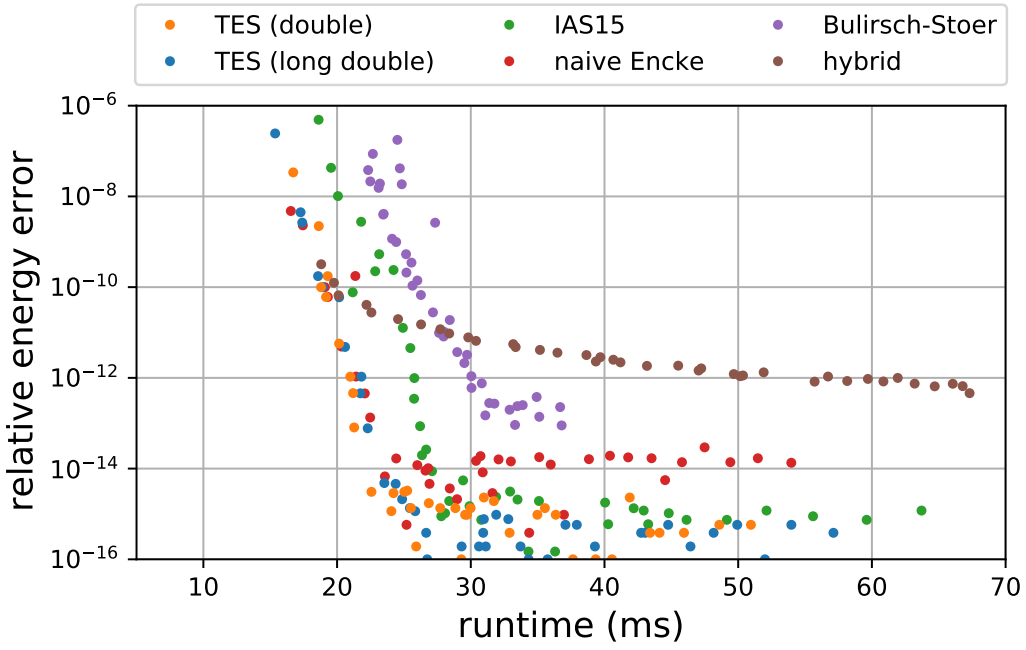


Figure 4.13: Relative energy error for a given runtime at the end of a one hundred year integration of the Sun, Earth and Apophis, including the 2029 close encounter with Earth. Each data point is the mean of twenty identical integrations.

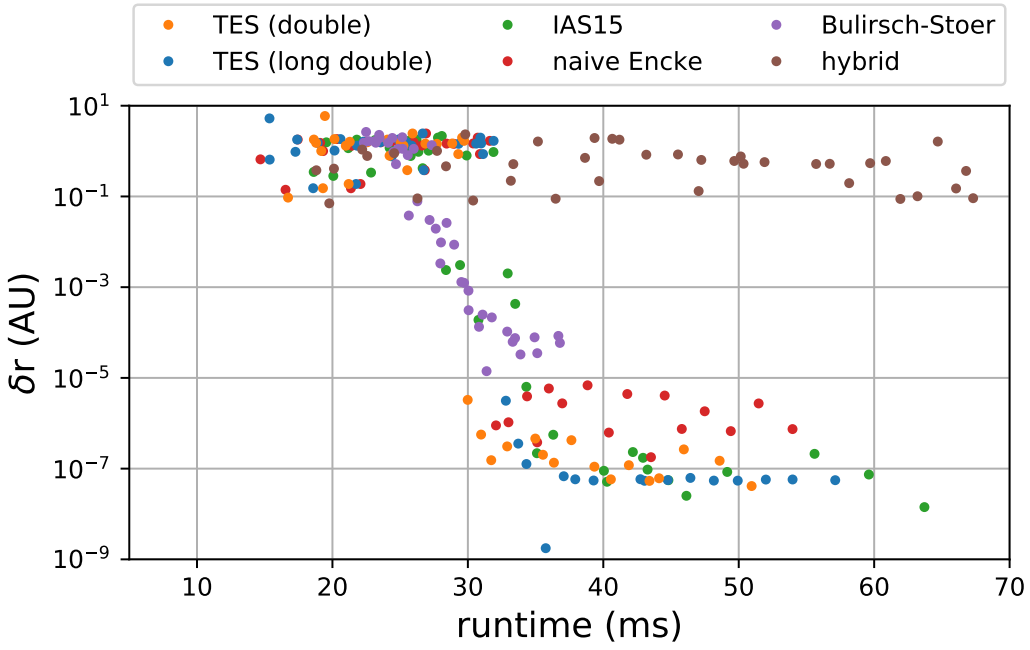


Figure 4.14: Final position error for a given runtime of Apophis after a one hundred year integration of the Sun, Earth and Apophis, including the 2029 close encounter with Earth. Each data point is the mean of twenty identical integrations.

4.5.4 Apophis 2029 encounter

The last example presented studies the performance of TES in the presence of close encounters. When the dynamics of planetary systems are such that bodies do not remain well separated indefinitely, it is important that close encounters are handled accurately, e.g. when modelling the behaviour of systems after an instability event (Rice et al., 2018; Bartram et al., 2021). Closer to home, another example is the accurate modelling of near-Earth objects (NEOs) to understand potential hazards to humanity. Of the known NEOs, one of particular interest is Apophis owing to the fact that it will pass within 100,000 km of the Earth in 2029 bringing it closer than the Moon.

The simplified model I use (Amato et al., 2017) evolves only the Sun, Earth, and Apophis and makes use of purely Newtonian dynamics throughout. While not a realistic model, it allows us to showcase the behaviour of different integrators in the presence of strongly perturbing close encounters. To obtain a set of initial conditions and a high precision reference solution trajectory, first the position of the three bodies were obtained from the NASA Horizons system at the point of closest approach during the 2029 encounter. Next, a Dormand-Prince integrator (Dormand and Prince, 1986) operating in quadruple precision was used to integrate backwards for fifty years to

obtain the initial conditions. Finally, the same integrator was used to propagate the initial conditions forward for one hundred years to obtain the final conditions. I consider this high precision integration truth, such that the final positions can also be used as an error metric in addition to energy conservation.

The trajectories followed by the three bodies are plotted in Fig. 4.11 and the transition of Apophis from the Apollo to the Atens group after its flyby of Earth can clearly be seen. Figure 4.12 shows the separation between Apophis and the Earth over the simulated period of 100 years. Here, Apophis makes many close approaches within 10 Earth Hill radii, or 0.1 AU, but only one very close approach which is in 2029. This approach distance is approximately 2.5×10^{-4} AU or roughly $17,000$ km, well within the geosynchronous orbital altitude at $35,786$ km. I perform integrations at 61 different tolerance settings for each of the integrators in Table 4.1. The tolerances used are chosen in the same way as in Section 4.5.2. At each tolerance, I perform twenty identical integrations and take the mean value for the runtime.

Figure 4.13 shows the relative energy error for a given runtime. First, consider the behaviour of TES (double) in this plot. In the truncation dominated regime, it is among the fastest schemes competing only with the naive Encke and TES (long double). TES (double) can also be seen to be fastest at the default tolerance (Table 4.1) which is represented by the furthest right orange data point; this is the recommended default setting for users. Once TES (double) has converged such that only round-off error is present, it is clear that the numerical implementation described in Section 4.3 has improved the performance by between one and two orders of magnitude in comparison to a naive Encke method (red). Due to the very short integration timescale of only one hundred orbits, TES (long double) shows no advantage over TES (double) and the runtime for the default tolerance setting is comparable to that of IAS15. The MERCURY integrators, Bulirsch-Stoer and hybrid, fail to conserve energy as precisely as the schemes based upon Everhart's Radau and reach final values of relative energy error of 10^{-13} and 10^{-12}, respectively. Interestingly, in contrast to Fig. 4.9, in the presence of repeated close encounters, the hybrid scheme runtime increases to be comparable to that of the non-symplectic integrators.

While the conservation of energy is a common metric for the accuracy, its value is usually dominated by the most massive objects in a given system. Given the low mass of Apophis in comparison to the Earth, I also look at the final position of Apophis as output by each integrator in comparison to our quadruple precision numerical solution. Figure 4.14 shows the error in the position of Apophis, δr, for a given runtime. Again starting with TES (double), there are two regions of performance: one where

the runtime is low, below 32 ms, and it performs poorly, and one where the runtime is higher, above 32 ms, where the scheme performs well. What is interesting is the lack of a transition period between these two regions which is present in, e.g., IAS15; instead, all Encke based methods are either highly precise or highly inaccurate. Fortunately, the recommended default tolerance for both implementations of TES is well within the highly precise solution region. Given a suitable tolerance, both implementations of TES and IAS15 are comparable in precision, with TES being slightly faster for the default tolerances (Table 4.1).

For this problem, TES (long double) shows interesting behaviour once it has converged to roughly $\delta r = 10^{-7}$ AU. At this point, its performance becomes highly consistent regardless of decreases in tolerance. This contrasts with all other Everhart based schemes which exhibit variance in the precision of the final position of Apophis with changing tolerance. The precisions seen here are highly consistent with Amato et al. (2017) who find their implementation of Everhart's Radau scheme integrating using Cowell's formulation converging to roughly 10^{-7} AU. To obtain more precise solutions, the authors had to employ regularisation techniques on the equations of motion.

The degree of positional precision obtained after a close encounter is paramount if one wishes to model repeated encounters. Deviations from a true trajectory are greatly amplified during a close encounter and any inaccuracies in modelling the first encounter in a series are therefore increased in all subsequent encounters. The precision achieved by TES in these experiments equates to a positional error at the time of closest approach of approximately 10 cm whereas the naive Encke method achieves an error of 10 m. Given that the size of keyholes, i.e., regions of space on the b-plane formed by the separation vector at the point of closest approach, found by Farnocchia et al. (2013) that can lead to a resonant return trajectory for Apophis are between 6 cm and 600 m, the precision of the naive Encke makes it unsuitable for use in this challenging application domain. Therefore, the numerical improvements in Section 4.3 that comprise TES are sufficient to improve the naive Encke method to the point where it can now be used in the study of NEO asteroid dynamics.

The hybrid scheme is not precise enough for the step sizes presented to accurately model the trajectory of Apophis during its flyby of Earth. However, I mirror the findings of Amato et al. (2017) and find that if the hybrid scheme step size is reduced by roughly a factor of one hundred then it is possible to obtain positional errors as low as 10^{-5} AU, although the computational cost for doing this means this option is of little practical importance.

4.6 Summary

In this chapter, I introduced TES, a new integrator for planetary systems that follows Brouwer's law and permits close encounters between massive bodies. TES builds upon the classical Encke method and takes advantage of the dominant nature of the star in planetary systems. I showed that TES is effective across a wide range of planet-to-star mass ratios but found that the more dominant the central body, the more effective the scheme is, with excellent improvements in speed being seen in simulations of the inner solar system.

In Section 4.2.2 I derived a new version of the Encke method in democratic heliocentric coordinates (ENCODE) and presented the equations of motion in this coordinate system. Additionally, I implemented a series of numerical improvements that reduced the round-off error by two orders of magnitude compared to the naive Encke method. TES is optimal in that it follows Brouwer's law and has an RMS energy error slightly below that of IAS15.

I performed extensive comparisons with IAS15 in REBOUND, and the Bulirsch-Stoer and symplectic hybrid schemes within the MERCURY package. I found that for well-separated systems, TES is the fastest non-symplectic scheme for a given precision. In the presence of close encounters, I found that TES is able to reach a precision much greater than either of the MERCURY schemes, in terms of both conservation of energy and final position, with a precision comparable to IAS15.

TES is open source and accessible at `https://github.com/PeterBartram/TES`. It is available in a "double" version using only double precision floating point arithmetic and a "long" version using extended, 80 bit, floating point arithmetic in the Kepler solver. In double precision, I found that TES is only 15% faster than the extended precision implementation which is two orders of magnitude more precise, but for portability I recommend the double precision version as the default. Regardless of the version used, I found that TES is faster than IAS15 for the same energy conservation in the majority of the problems examined, although some of this performance is lost for systems with more massive secondaries. I also found that TES can handle close encounters such as the Apophis flyby in 2029 efficiently and accurately.

In the next chapter, I use TES to study the stability of planetary systems. In particular, I use it to model the dynamics of compact exoplanet systems right up until a collision between planets occurs.

Chapter 5

Post-instability impact behaviour of compact three-planet systems

The contents of this chapter is based upon the article published at Monthly Notices of the Royal Astronomical Society. It is available with open access on MNRAS and also on arXiv. The authors of the article are Peter Bartram, Alexander Wittig, Jack J. Lissauer, Sacha Gavino and Hodei Urrutxua. I am responsible for performing the work in the original article although I owe a great deal to helpful conversations with my co-authors. Additionally, results included in the chapter using the WH map and hybrid scheme are courtesy of Sacha Gavino and Jack Lissauer.

This chapter brings together all previous discussions and work in this thesis. In it, I take the TES algorithm developed in the previous chapter, and use it to perform a large simulation campaign to understand the behaviour of compact exoplanet systems. This simulation campaign actually therefore provides two benefits. Firstly, it enables a deeper understanding of the post-instability behaviour of compact three-planets systems. And secondly, it provides a testing ground for TES to ensure that it can handle the precision demands required by very long-term simulations in the presence of repeated close encounters and a collision between Earth analogues.

I perform over 25,000 integrations of a Sun-like star orbited by three Earth-like secondaries for up to a billion orbits to explore a wide parameter space of initial conditions in both the co-planar and inclined cases, with a focus on the initial orbital spacing. I calculate the probability of collision over time and determine the probability of collision between specific pairs of planets. I find systems that persist for over 10^8 orbits after an orbital crossing and show how the post-instability survival time of systems depends upon the initial orbital separation, mutual inclination, planetary radius, and

the closest encounter experienced. Additionally, I examine the effects of very small changes in the initial positions of the planets upon the time to collision and show the effect that the choice of integrator can have upon simulation results. I generalise the results throughout to show the behaviour of systems with an inner planet initially located at 1 AU and 0.25 AU.

5.1 Background

The now retired NASA Kepler Space Telescope is responsible for observations leading to the confirmation of hundreds of multi-planet systems (Lissauer et al., 2014; Rowe et al., 2014). Of these systems, as many as six percent are thought to be compact (Wu et al., 2019), containing planets that are much more closely spaced than the inner planets of our own Solar System. These discoveries have naturally led to many questions being asked about the long-term stability of compact exoplanet systems. Indeed, it is even possible that compact planetary embryos existed interior to Venus's current orbit that have subsequently been ejected from this region due to orbital instabilities (Volk and Gladman, 2015). Within the class of observed compact systems, a large population of planets have been observed with a mass (Mayor et al., 2011) and radius (Petigura et al., 2013) between that of Earth and Neptune. Moreover, the observed orbital architecture is such that mutual inclinations are small, typically in the region of 1° to 2° (Fabrycky et al., 2014), while eccentricities are also found to be small, on average $\bar{e} \approx 0.04$ (Xie et al., 2016). An archetypal example of these systems, albeit containing six planets, is Kepler-11 (Lissauer et al., 2011). Exoplanet systems with orbital spacings much greater than that required for stability are also present in the Kepler dataset. It is a favourable hypothesis that this orbital architecture is a result of dynamical instabilities in much more compact systems leading to close encounters and orbital reconfiguration (Pu and Wu, 2015). Understanding of the stability and evolution of compact exoplanet systems is therefore not only important for making sense of observations but also for understanding the planetary formation process as a whole.

Characterisation of the stability of three or more planet systems can be approached in several ways. Analytical models have been built that can predict the lifetime of three-planet systems based upon resonance overlap (Wisdom, 1980; Petit et al., 2020). Recently, machine learning approaches have also been developed that, after being guided by a training set of 10^9 year integrations, can use far shorter integrations to predict with surprisingly high accuracy which given exoplanet systems will remain

stable for a billion orbital periods (Tamayo et al., 2016, 2020a). However, the most common approach to the problem, and the one employed in this chapter, is the use of n-body simulation (Chambers, 1999; Smith and Lissauer, 2009; Obertas et al., 2017; Hussain and Tamayo, 2020; Lissauer and Gavino, 2021).

The majority of studies performed take a subset of the possible input parameter space for a compact, near-circular, near co-planar system of a given number of planets, and then evolve this system forward in time checking for either the first close approach, typically specified as one Hill radius, r_H, or waiting for an orbital crossing to occur: this is then termed the instability event. Throughout this work, I will use orbital crossing as the definition of an instability event and refer to the time at this point as the crossing time. Rice et al. (2018) found that systems containing four Neptune-mass planets can continue to evolve after an instability event for over ten million dynamical periods before a collision of planets, meaning that the commonly used instability metric may not capture the entire evolution of the system. Given that the manner in which these planets collide determines the final orbital architecture, it is important properly to understand this phase of the exoplanet system life cycle.

This study builds upon the work done by Rice et al. (2018) and Lissauer and Gavino (2021) by considering the post-instability evolution of compact, Earth-analogue, three-planet systems across a large range of initial orbital separations equally spaced in units of mutual Hill radii. I create three integration suites called the standard suite, perturbed suite, and inclined suite, and perform 4,800 integrations each in the first two and a further 16,800 in the final one. I continue integrations up until the time of first collision between planets or for 10^8 or 10^9 orbits depending on the experiment.

In Section 5.2 of this chapter I describe the methodology used for the integrations including the initial conditions for each integration suite, the integration packages used, and the termination criteria. Section 5.3 contains the results of all standard suite integrations: Section 5.3.1 details the timescales for orbital crossing and collision between pairs of planets, and details collision probabilities over time for various initial configurations of systems; the effects of small changes in initial orbital longitude upon these results are then examined in Section 5.3.2; and, finally, Section 5.3.3 examines the probabilities of particular pairs of planets colliding. Section 5.4 introduces the results of inclined suite integrations: in Section 5.4.1 I explore the heating of what are initially dynamically cold systems that eventually enables orbital crossing and collision; here, I find that the three-planet Earth-mass systems behave in a similar manner to the four-planet Neptune-mass case but follow a different power law. Section 5.4.2 examines the timescales leading to collision in the inclined case and shows

that the survival time after crossing can be a non-trivial fraction of the main-sequence lifetime of stars. In addition, this section also looks at the effects on the lifetime of systems dependent on the distance from the innermost planet to the star and the initial inclination. I finally summarise my findings in Section 5.6.

5.2 Methods

I have chosen to simulate compact three-planet systems comprising of analogues of our own Solar System which are consistent with systems identified in the Kepler dataset. The central body in each system is a one solar-mass star, $m_0 = 1\ \mathrm{M}_\odot$. Each of the planets within the systems are Earth mass, $m_j = 1\ \mathrm{M}_\oplus$ where $j = 1, 2, 3$ with a planetary radius also equal to that of Earth, $\mathrm{R}_P = \mathrm{R}_\oplus$. Planets are placed on initially circular orbits orbiting the star in a common direction with the innermost planet located at 1 AU. Time throughout this work is provided in units of initial orbital period of the innermost planet, this means that the crossing time is invariant to rescaling of the system so long as the initial orbital period ratios between bodies are maintained along with the mass-ratios of planets and star.

5.2.1 Initial semi-major axes

Initial semi-major axes a_j of systems are evenly spaced in terms of mutual Hill radii. The mutual Hill radii is defined as

$$r_{H_{j,j+1}} = \left(\frac{m_j + m_{j+1}}{m_0 + \sum_{k=1}^{j-1} m_k} \right)^{\frac{1}{3}} \left(\frac{a_j + a_{j+1}}{2} \right).$$

This allows for a dimensionless value β to be defined to specify the even spacing of adjacent planetary orbits in units of their mutual Hill radii as

$$\beta \equiv \frac{a_{j+1} - a_j}{r_{H_{j,j+1}}}.$$

Therefore, the initial semi-major axes of adjacent planets are chosen to be such that

$$\begin{aligned} a_{j+1} &= a_j + \beta r_{H_{j,j+1}} \\ &= a_j \left[1 + \frac{\beta}{2} \left(\frac{m_j + m_{j+1}}{m_0 + \sum_{k=1}^{j-1} m_k} \right)^{\frac{1}{3}} \right] \left[1 - \frac{\beta}{2} \left(\frac{m_j + m_{j+1}}{m_0 + \sum_{k=1}^{j-1} m_k} \right)^{\frac{1}{3}} \right]^{-1} . \end{aligned} \tag{5.1}$$

The innermost planet is placed such that it has a semi-major axis of 1 AU, and all other semi-major axes are chosen through Eq. (5.1). I refer to this configuration as a *system at* 1 *AU*. Likewise, later on, when results are generalised to include systems with an innermost planet located at 0.25 AU with other planets spaced as per Eq. (5.1) I refer to it as a *system at* 0.25 *AU*.

5.2.2 Stopping criteria and integration packages

I use TES (Bartram and Wittig, 2021) with a non-dimensional tolerance of 1×10^{-8} to perform all integrations. This has ensured that the relative energy error in all simulations, even after collision and for the longest lived systems, is maintained below 1×10^{-13}. To validate these results, I also repeated all standard suite integrations making use of IAS15 (Rein and Spiegel, 2015) within the REBOUND package (Rein and Liu, 2012). The results from this comparison can be found in Section 5.5.

As previously mentioned, time is measured by initial periods of the innermost planet in the system throughout this work, meaning that all times are specified in units of orbits or dynamical periods. Integrations run until either a collision is detected or the simulation reaches a maximum time of 10^8 or 10^9 dynamical periods, depending on the experiment.

To detect an orbital crossing, the orbital elements of each planet are calculated at every step within each integration. These are then compared to determine the time at which the apoapsis of a planet crosses the periapsis of the exterior adjacent planet. I define this as the *crossing time* and denote it t_c. Moreover, also at each step, the mutual separations of each of the planets are calculated so that collisions can be detected. The metric of two planets coming within $2R_\oplus$ of one another is used for collision detection. I define the time at which this occurs as the *impact time* and denote it t_i. I also define the *post-crossing survival time*, t_s, of a system to be the time that the system

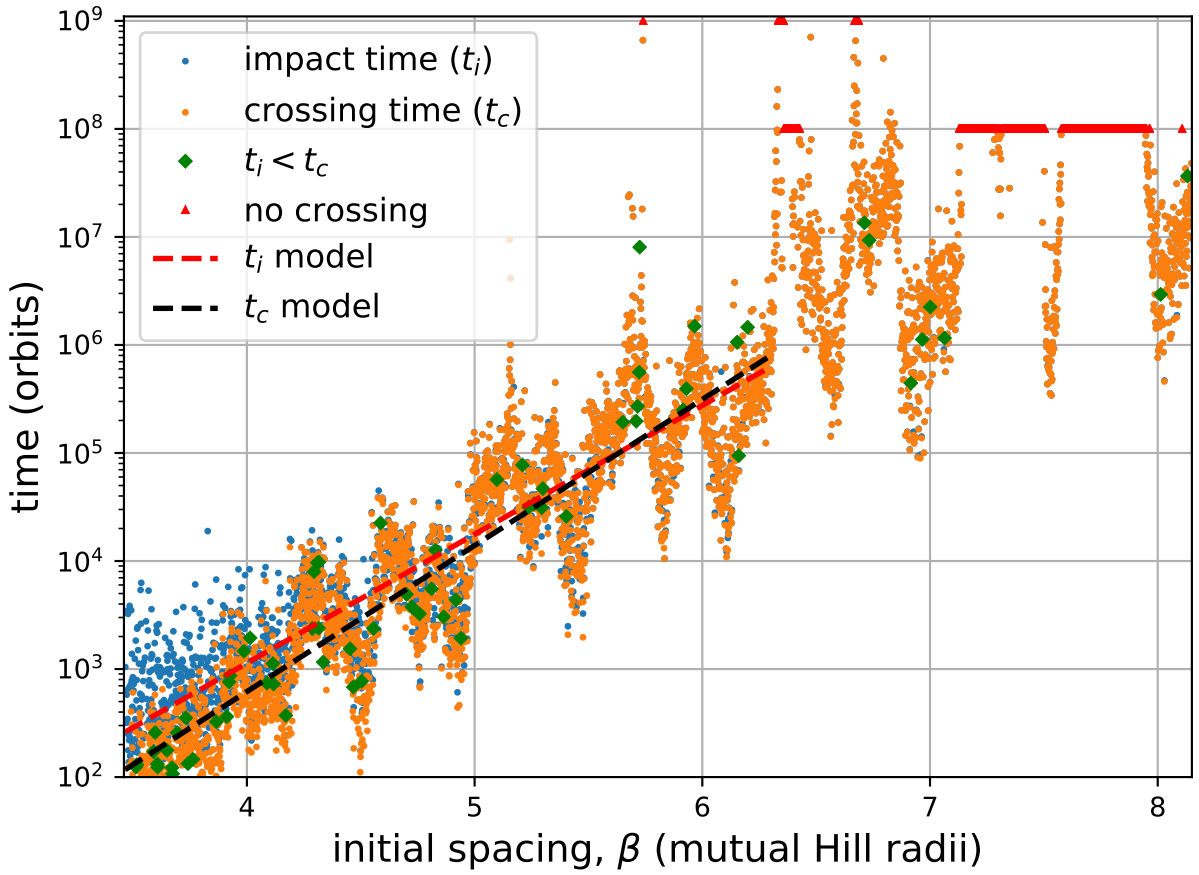


Figure 5.1: Plot showing the crossing time, t_c, and impact time, t_i, for all integrations in the standard suite for systems at 1 AU. Simulations are run for up to 10^9 orbits in general but some are terminate at 10^8 orbits to save on computation. Orbits are specified by the initial period of the innermost planet. Impacts that take place before a crossing are highlighted by a green diamond whereas systems that did not cross within the maximum simulation time are marked with a red triangle. Models fitted to the crossing and impact times according to Eq. 5.2 are shown as a dashed black and a dashed red line, respectively.

persists without a collision after the point of orbital crossing:

$$t_s \equiv t_i - t_c.$$

All encounters closer than any experienced previously are recorded such that it is possible to generalise the collision results to systems with planetary radii greater than that of the Earth or, equivalently, initial orbital radii closer than 1 AU. I use this generalisation to consider systems at 0.25 AU and 1 AU for all integration suites. I also define the time of closest encounter prior to collision as the *closest encounter time*, t_e. To ensure bitwise identical initial conditions as in Lissauer and Gavino (2021), initial conditions are specified as orbital elements which are then entered in to the

Table 5.1: Summary of all simulation event time symbols used.

Symbol	Definition
t_c	crossing time
t_i	impact time
t_s	post-crossing survival time
t_e	closest encounter time

MERCURY (Chambers, 1999) integration package to generate an initial state vector which is then provided to either TES or REBOUND. Table 5.1 contains a summary of all symbols related to simulation event times.

5.2.3 Standard integration suite

The first suite of integrations is composed of $4,800$ orbital configurations and is termed the standard suite. In this suite, systems are on initially circular, co-planar orbits with an initial mean anomaly for the j_{th} planet $M_j = 2\pi j\lambda$ radians where $\lambda \equiv \frac{1}{2}\left(1+\sqrt{5}\right)$, i.e., the golden ratio, and are merely chosen to avoid special orientations. As I wish to study the effects of the initial spacing of planets upon impact timescales, I choose a high resolution in β such that there are 1×10^3 integrations per unit β over the range $\beta = [3.465, 8.3]$. Generally, integrations are terminated after 10^9 orbits if a collision is not encountered. However, in certain areas I have chosen to limit integrations to 10^8 orbits to save on computation; these regions are clearly marked on any plots.

5.2.4 Perturbed integration suite

The second integration suite is termed the perturbed suite and is also composed of $4,800$ integrations. The only difference between the initial conditions of the standard suite and the perturbed suite is that in the latter case the innermost planet is perturbed by 100 m along its orbital arc. I strictly terminate integration at 1×10^8 orbital periods of the innermost planet in this suite. This suite is used to examine the effects of very small changes in initial conditions upon crossing and impact time.

5.2.5 Inclined integration suite

The final integration suite is the inclined suite and is composed of 16,800 integrations. I choose initial conditions across a subset of the available parameter space manually rather than randomly and perform integrations for a maximum simulation time of 1×10^8 orbital periods of the innermost planet. To make best use of computational resources, I limit this study to the range $\beta = [3.5, 6.3]$ and perform experiments uniformly spaced in β with fifty values per unit β. At each value of β I perform one hundred and twenty experiments where the initial values of semi-major axis, eccentricity and mean longitude are the same as in the standard suite.

Planets are, however, inclined relative to each other in one of four ways: one of inner, middle or outer planet inclined above the orbital plane of the system, and also with the middle planet above and the outer planet below. For each such configuration of relative inclination fifteen initial values of inclination are logarithmically spaced between $i_0 = 0.06°$ and $i_0 = 0.58°$, yielding an initial orbital height ranging from $0.10\ r_H$ to r_H. The distribution of initial inclinations within this range is such that ten values are used between $i_0 = 0.24°$ and $i_0 = 0.58°$ and five values are used over the region $i_0 = 0.06°$ and $i_0 = 0.24°$. Finally, two values are chosen for the ascending nodes Ω: either according to the golden ratio in Section 5.2 such that $M_j = \Omega_j$ or equally spaced such that $\Omega_j = [0°, 120°, 240°]$.

The full state vector of each simulation is output to file once every ten thousand orbital periods; additionally, each planetary flyby closer than any other previously observed is also recorded.

5.3 Standard integration suite

This section contains the results of the standard integration suite described in Section 5.2.3.

5.3.1 Timescale to planet-planet collision

The crossing and impact times for the standard suite are plotted in Fig. 5.1. Inspection of the crossing time with respect to the initial orbital spacing shows the clear upwards trend present in other works (Smith and Lissauer, 2009; Obertas et al., 2017; Hussain and Tamayo, 2020; Tamayo et al., 2020a; Lissauer and Gavino, 2021). I also capture

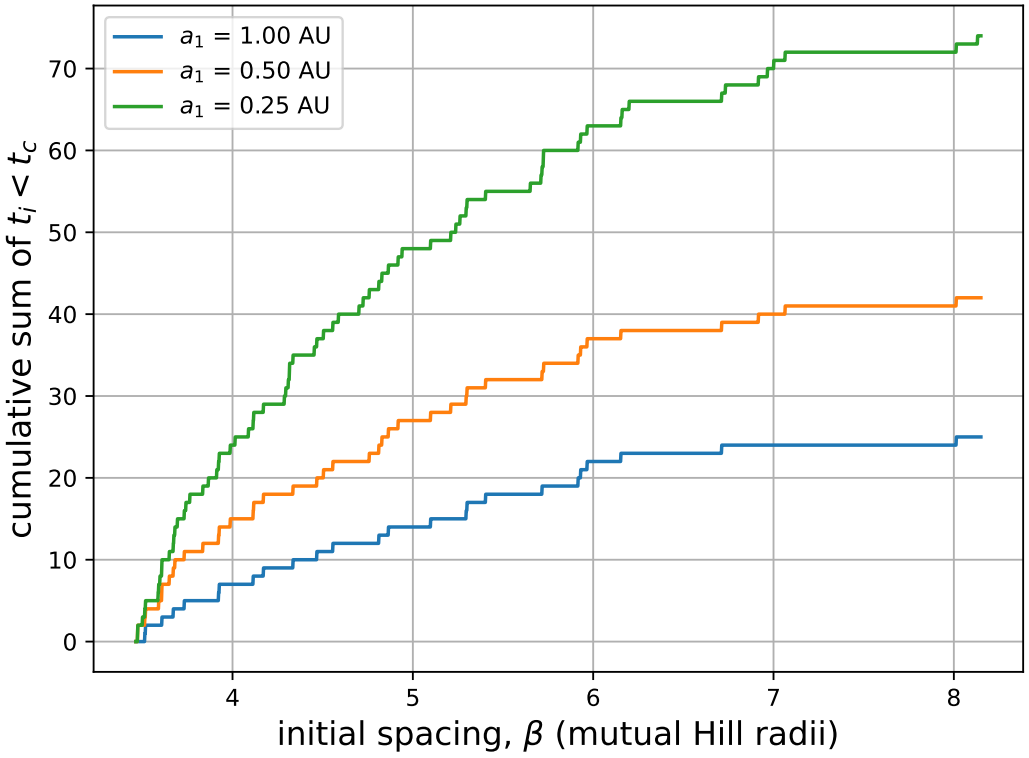


Figure 5.2: Cumulative sum of integrations with a collision before orbital crossing for various initial values for semi-major axis of the innermost planet. The flat region between beta $\beta = 7$ and $\beta = 8$ is due to systems not experiencing an orbital crossing within the maximum simulation time in that region (see the red triangles in Figure 5.1).

the large scale variations about the trend which for the most part are a result of mean motion resonances as discussed in Obertas et al. (2017). Additionally, I replicate the finding of Lissauer and Gavino (2021) in the discovery of a highly stable configuration around $\beta = 5.74$ which they attribute to the distance of this configuration from any strong resonances.

Throughout this work I use a linear logarithmic fit of the form

$$\log_{10}(t) = b'\beta' + c' \tag{5.2}$$

in several places where $\beta' = \beta - 2\sqrt{3}$ and is used to reduce the dependency of the y-intercept upon the slope. I fit this model to three datasets such that $t = t_c$, t_i or t_s and state explicitly which at the time of use. Unless otherwise stated, I only include data points in the region $\beta = [3.465, 6.3]$ in the fits to avoid biasing the results due to systems that did not experience an orbital crossing within the maximum simulation time. For $t = t_c$, over this region, I find that $b' = 1.352$ and $c' = 2.067$ which is in

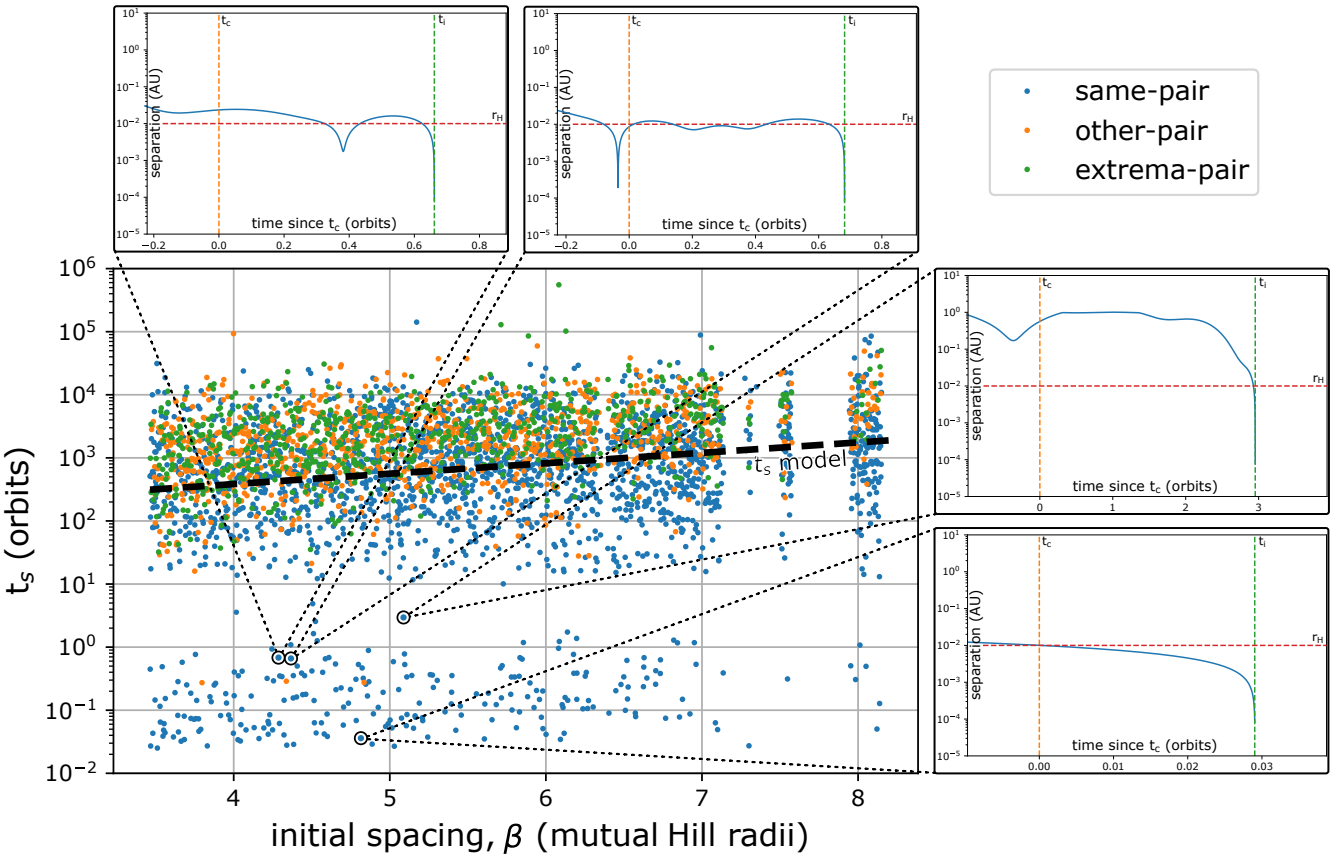


Figure 5.3: Post-crossing survival time of systems initially at 1 AU against β. Blue dots indicate the same pair both crossed orbits and collided; orange indicates the pair that collided was not the pair that crossed; green indicates a collision between the inner and outer planets. The t_s model (bold dashed black) is fitted to all data points with a survival time greater than two orbits. The insets show the planet separation for the marked systems between crossing time, t_c, (dashed orange) and collision time, t_i, (dashed green). Additionally, the Hill radius at 1 AU is shown (dashed red).

strong agreement with Lissauer and Gavino (2021) and confirms the functionality of TES. For impact times t_i, I find $b' = 1.192$ and $c' = 2.42$.

Figure 5.1 highlights that the post-crossing survival time is very small compared to the crossing time for the majority of systems observed. The log scale of the plot and the relatively small magnitude of t_s means the bulk of the impact time data points are hidden in this figure. The only exception is in the region of small β where the ratio t_i/t_c is large due to the relatively small size of t_c.

Finally, it can be seen that for a small subset of integrations collisions can occur before an orbital crossing has taken place. A cumulative sum showing the number of occurrences is shown in Fig. 5.2 where I believe that the increase between systems at 1 AU and 0.25 AU is not dependent purely on the physical cross-sectional area of planets

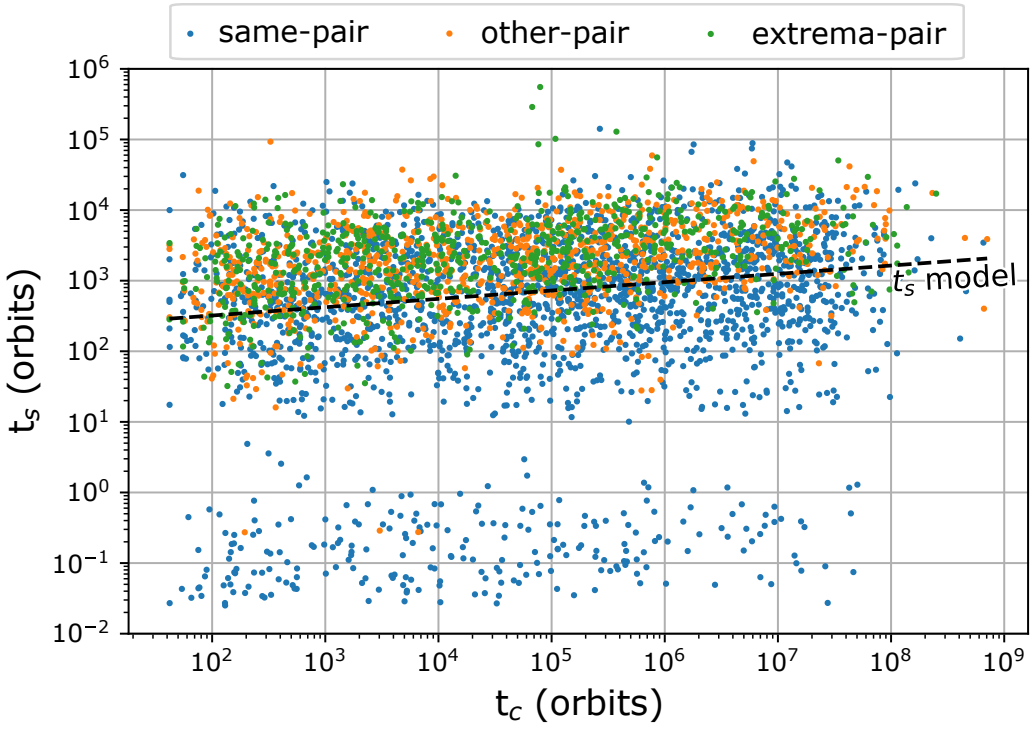


Figure 5.4: Post-crossing survival time, t_s, against orbital crossing time, t_c, for systems initially at 1 AU. Blue dots indicate the same pair both crossed orbits and collided; orange indicates the pair that collided was not the pair that crossed; green indicates a collision between the inner and outer planets. The t_s model (dashed black) is fitted to all data points with a survival time greater than two orbits.

but rather the enhanced cross-sectional area due to gravitational focusing (Safronov, 1969, p. 147).

It is likely that a symplectic integrator, configured to use the standard step size of $1/20$ th of the smallest dynamical period, would miss these collisions. However, given the small number of occurrences relative to the number of integrations typically performed in stability studies, it is unlikely that these missed collisions will have biased the datasets in any statistically meaningful way.

Figure 5.3 shows the post-crossing survival time for all systems within the standard suite against β; Figure 5.4 is identical but plotted against t_c. I find two main populations of post-crossing survival times present: those surviving for less than two orbits, and those surviving for more than ten orbits with very few outliers in between. Within the long surviving population, it can be seen that there is a clear increase in the post-crossing survival time of systems with respect to both β and t_c. I fit models of the form of Eq. (5.2) to both the long-lived population and the population in its entirety, I call these datasets *long* and *all*, respectively. The model coefficients b' and c' can be found in the top two rows of Table 5.2. Similarly, I also fit linear models to the two

Table 5.2: Fitted model coefficients for t_s against β and t_c. Plotted models are fitted to the long-lived population, long, only but fitted models for the full dataset, all, are included as well. PCC is the Pearson correlation coefficient. σ is the standard deviation of the dataset from the fitted model.

t_s model	dataset	b	c	b'	c'	PCC	σ
Figure 5.3	long	–	–	0.111	2.84	0.197	0.680
	all	–	–	0.165	2.496	0.176	1.13
Figure 5.4	long	0.0781	2.693	–	–	0.183	0.682
	all	0.118	2.27	–	–	0.167	1.13

datasets present in Fig. 5.4 for $\log_{10}(t_s)$ against $\log_{10}(t_c)$. The model coefficients b and c can be found in the bottom two rows of Table 5.2. In all cases, I calculate the Pearson correlation coefficient (PCC) and also calculate the standard deviation, σ, of the data minus the fitted model, e.g. $\sigma(\log_{10}(t_s) - (b'\beta' + c'))$. Clearly, there is a tendency for systems to persist for longer after an orbital crossing when the initial mutual spacing between them is greater, with a difference of a factor of three in median post-crossing survival time over the entire beta range. However, even given this increase, the post-crossing survival time for systems simulated did not ever exceed one million orbits. Given that this represents roughly one ten-thousandth of the main sequence lifetime of solar-mass stars it is possible, although very unlikely, that we could observe a compact exoplanet system that has undergone an orbital crossing but has not yet experienced a collision between planets, even if it were a truly co-planar system.

In the case of the short-lived population, there is a further subdivision of different behaviours: those systems that experience a collision almost immediately following a crossing, e.g. those bodies whereby $t_s < 10^{-1}$, and those which persist for longer than this but less than a couple of orbits. In the former case, I have observed that the trajectories of two planets about the star simply cross, leading to straightforward collisions, and triggering an orbital crossing in the process. However, in the latter case, I find that the trajectories of the planets about the star are such that a very close encounter occurs, which causes the two planets to become temporarily gravitationally captured. These two planets then remain within approximately a Hill radius of one another before finally experiencing a fatal collision a fraction of an orbit later. These behaviours are shown in the satellite images in Fig. 5.3. It can be seen here that temporary gravitational capture is not the cause of collision in the case of outliers with a post-crossing survival time between two and ten orbits.

To consider whether these results generalise to other systems of planets, I have calculated t_i and t_s for a system with the inner planet initially placed at 0.25 AU. This is

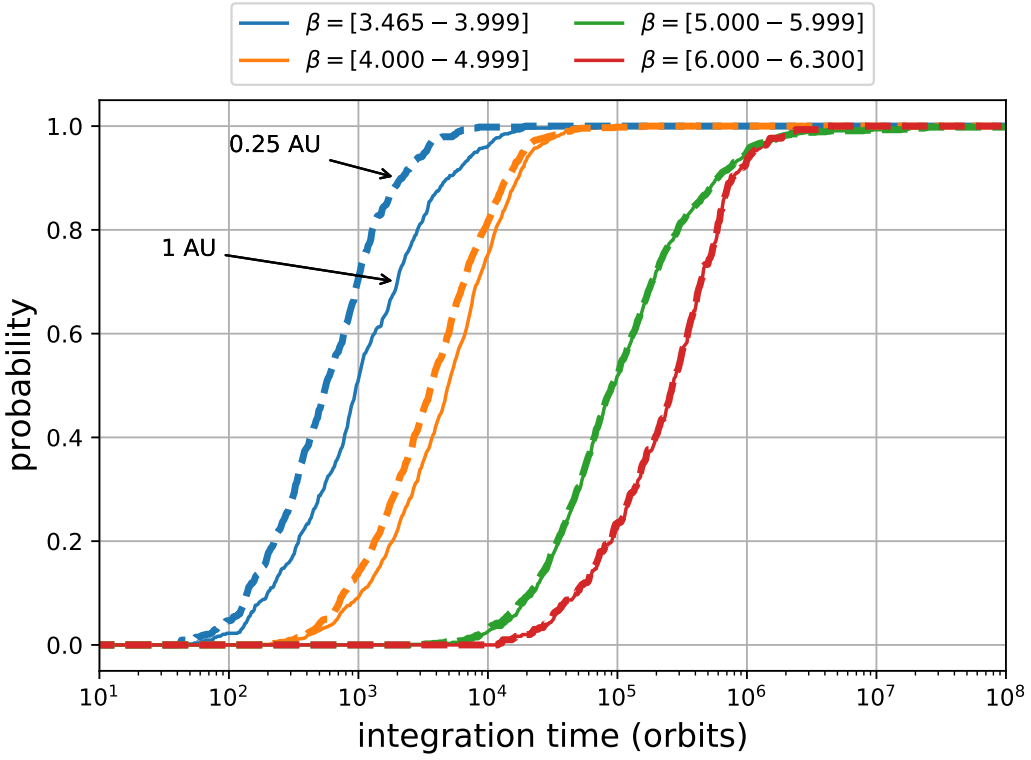


Figure 5.5: Probability of having experienced a collision over time for various regions of initial spacing, β. The probability is calculated as the cumulative fraction of systems that have experienced collisions over the total number of systems. Solid lines show the probabilities for systems initially at 1 AU while the dashed lines are initially at 0.25 AU.

equivalent to artificially inflating the radius of all planets in systems at 1 AU by a factor of four. When thought of this way, this is akin to placing planets with a radius approximately the same size as Neptune at 1 AU; t_c is invariant to the initial location of the inner planet. The probability, calculated as the cumulative fraction of systems that have experienced collisions over the total number of systems, of collision over time for both settings are shown in Fig. 5.5. The separation between dashed and solid lines indicates that a given collision probability is reached sooner in systems composed of planets with a larger radius. The difference in time remains approximately constant over all values of β, even if the log scale suggests otherwise.

Figure 5.6 contains normalised histograms of t_s within different regions of β for systems with the inner planet initially at 1 AU and 0.25 AU. I find that the distribution of post-crossing survival times is log-skew-normal distributed across all systems; I confirmed this using a Kolmogorov-Smirnov test with a precision parameter of $\alpha = 0.005$. The skew-normal distribution is a generalisation of the normal distribution that allows the class to be extended to include distributions with non-zero skewness through the

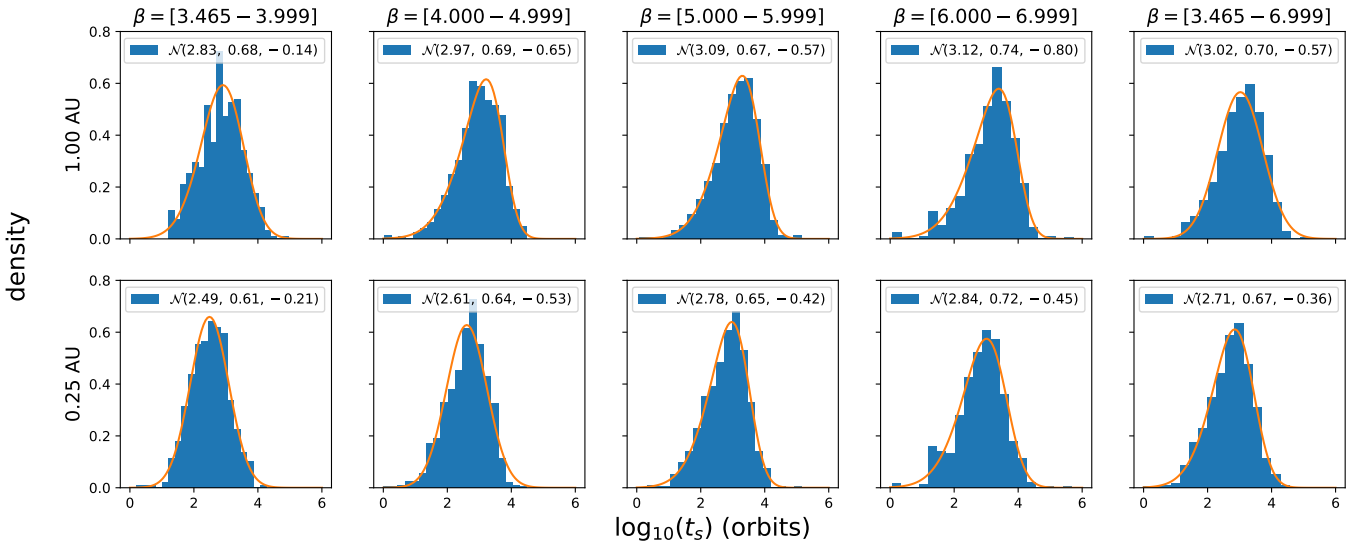


Figure 5.6: Normalised histograms of post-crossing survival time, $\log(t_s)$, for different regions of initial spacing, β. The top row of plots is for systems initially at 1 AU while the bottom one is at 0.25 AU. Log-skew-normal probability density functions, shown in orange, are fitted to the data through a maximum likelihood estimator. The mean μ, standard deviation σ, and the skew ζ are included for each distribution as $N(\mu, \sigma, \zeta)$. Systems that did not experience a crossing were excluded from these distributions.

addition of a shape parameter (Azzalini and Capitanio, 1999). Log-skew-normal probability density functions, shown in orange, are fitted to the data through a maximum likelihood estimator. I calculated the mean μ, standard deviation σ, and the skew ζ for each distribution; I use the Fisher-Pearson coefficient of skewness throughout. I find that μ increases with increasing β range, and also find the same pattern for σ in all but one case. In all cases, ζ is negative indicating a skew towards shorter post-crossing survival times compared to a normal distribution. This means that there is a preference for systems to collide sooner rather than later after an orbital crossing as compared to the most frequent survival times. There is a slow build up in the number of systems experiencing collisions over time after an orbital crossing but a much sharper cut-off after the peak density of collisions. This highlights the difficulty for systems to persist for long timescales after an orbital crossing in the co-planar case. Systems with a shorter mean post-crossing survival time show a skew of a smaller magnitude than those with a longer survival time, e.g. at 1 AU $\zeta = -0.14$ for $\beta < 4.0$ whereas for $\beta >= 4.0$ the smallest, in magnitude, value observed is $\zeta = -0.57$. I find that the distributions of post-crossing survival times at 0.25 AU are less skewed than those at 1 AU, indicating that the survival times of systems in this case are closer to a log-normal distribution.

Table 5.3: Comparison of *crossing times* of systems using identical values of initial spacing, β, in mutual Hill radii for the standard and perturbed initial longitudes.

Interval:	[3.465, 3.999]	[4.0, 4.999]	[5.0, 5.999]	[6.0, 6.33]	[3.465, 6.33]
number of runs in the range	535	1000	1000	331	2866
$< \log_{t_c}(\text{standard}) - \log_{t_c}(\text{perturbed}) >$	0.006	−0.001	−0.011	−0.014	−0.004
$< \lvert\log_{t_c}(\text{standard}) - \log_{t_c}(\text{perturbed})\rvert >$	0.039	0.182	0.306	0.356	0.219
$t_c(\text{perturbed}) < 0.5t_c(\text{standard})$	7 (1.31%)	92 (9.20%)	200 (20.00%)	75 (22.66%)	374 (13.05%)
$0.5t_c(\text{standard}) < t_c(\text{perturbed}) < 2t_c(\text{standard})$	524 (97.94%)	812 (81.20%)	580 (58.00%)	173 (52.27%)	2089 (72.89%)
$t_c(\text{standard}) < 0.5t_c(\text{perturbed})$	4 (0.75%)	96 (9.60%)	220 (22.00%)	83 (25.08%)	403 (14.06%)
within 10% of standard systems	398 (74.39%)	217 (21.70%)	100 (10.00%)	27 (8.16%)	742 (25.89%)
within 1% of standard systems	333 (62.24%)	68 (6.80%)	10 (1.00%)	7 (2.11%)	418 (14.58%)

Table 5.4: Comparison of *collision times* of systems using identical values of initial spacing, β, in mutual Hill radii for the standard and perturbed initial longitudes both with the innermost planet initially at 1 *AU*.

Interval:	[3.465, 3.999]	[4.0, 4.999]	[5.0, 5.999]	[6.0, 6.33]	[3.465, 6.33]
number of runs in the range	535	1000	1000	331	2866
$< \log_{t_i}(\text{standard}) - \log_{t_i}(\text{perturbed}) >$	0.015	−0.012	−0.010	−0.012	−0.006
$< \lvert\log_{t_i}(\text{standard}) - \log_{t_i}(\text{perturbed})\rvert >$	0.429	0.302	0.294	0.349	0.328
$t_i(\text{perturbed}) < 0.5t_i(\text{standard})$	145 (27.10%)	189 (18.90%)	185 (18.50%)	76 (22.96%)	595 (20.76%)
$0.5t_i(\text{standard}) < t_i(\text{perturbed}) < 2t_i(\text{standard})$	260 (48.60%)	614 (61.40%)	598 (59.80%)	175 (52.87%)	1647 (57.47%)
$t_i(\text{standard}) < 0.5t_i(\text{perturbed})$	130 (24.30%)	197 (19.70%)	217 (21.70%)	80 (24.17%)	624 (21.77%)
within 10% of standard systems	108 (20.19%)	110 (11.00%)	99 (9.90%)	25 (7.55%)	342 (11.93%)
within 1% of standard system	79 (14.77%)	17 (1.70%)	8 (0.80%)	5 (1.51%)	109 (3.80%)

Table 5.5: Comparison of *collision times* of systems using identical values of initial spacing, β, in mutual Hill radii for the standard and perturbed initial longitudes both with the innermost planet initially at 0.25 *AU*.

Interval:	[3.465, 3.999]	[4.0, 4.999]	[5.0, 5.999]	[6.0, 6.33]	[3.465, 6.33]
number of runs in the range	535	1000	1000	331	2866
$< \log_{t_i}(\text{standard}) - \log_{t_i}(\text{perturbed}) >$	−0.005	−0.010	−0.010	−0.011	−0.009
$< \lvert\log_{t_i}(\text{standard}) - \log_{t_i}(\text{perturbed})\rvert >$	0.297	0.243	0.301	0.353	0.286
$t_i(\text{perturbed}) < 0.5t_i(\text{standard})$	98 (18.32%)	141 (14.10%)	187 (18.70%)	75 (22.66%)	501 (17.48%)
$0.5t_i(\text{standard}) < t_i(\text{perturbed}) < 2t_i(\text{standard})$	335 (62.62%)	701 (70.10%)	592 (59.20%)	176 (53.17%)	1804 (62.94%)
$t_i(\text{standard}) < 0.5t_i(\text{perturbed})$	102 (19.07%)	158 (15.80%)	221 (22.10%)	80 (24.17%)	561 (19.57%)
within 10% of standard systems	142 (26.54%)	130 (13.00%)	99 (9.90%)	25 (7.55%)	396 (13.82%)
within 1% of standard system	108 (20.19%)	19 (1.90%)	10 (1.00%)	7 (2.11%)	144 (5.02%)

5.3.2 Sensitivity to initial conditions

To examine the sensitivity to initial conditions of the results of the simulations, I use the perturbed suite of integrations described in Section 5.2.4. The crossing and collision times of each integration between the standard suite and the perturbed suite are compared to determine the effect of the perturbation. Table 5.3 contains the results of that comparison for the *time of orbital crossing*. Tables 5.4 and 5.5 contain the same comparison for but for the *impact time* of systems at 1 AU and 0.25 AU, respectively.

In general, the comparison between crossing times in Table 5.3 aligns closely with Lissauer and Gavino (2021). Percentages between the two studies rarely differ by more than a few points despite the different integration tools used: TES and MERCURY. One notable difference between the two studies is in the initially wider spaced systems. In the regions $\beta = [5.0, 5.999]$ and $\beta = [6.0, 6.33]$ I find roughly double the number of initial orbital spacings where the standard and perturbed suite integrations experience orbital crossing times within 10% of one another. Given the precise orbital evolution required in order for standard and perturbed suite systems to experience a crossing at the same time, it is unlikely that numerical error would ever cause an increase in this statistic. I therefore take this as an indication that TES has maintained a higher precision than the symplectic Wisdom-Holman (Wisdom and Holman, 1991) scheme within MERCURY. To further validate TES in this setting I have also repeated the standard suite integrations with IAS15 from the REBOUND package for comparison. I find very good agreement in results between the two routines.

The right-most summary column for the full range of $\beta = [3.464, 6.300]$ in Table 5.4 shows there is a marked decrease in the number of collisions occurring within a factor of two, and within ten and one percent of one another; as compared to the orbital crossing times in Table 5.3. The largest reduction is seen in the within-a-factor-of-two row where a reduction of over 15 percentage points highlights the sensitivity to close approaches in this setting. The majority of this difference in collision times is seen in the initially closely spaced systems where a reduction of almost 50 percentage points can be seen for integrations finishing within 10% of one another. However, once the crossing times exceed approximately 1×10^4 orbits at $\beta = 5$ the effect of the perturbation disappears and values between crossing and collision times for the two datasets converge.

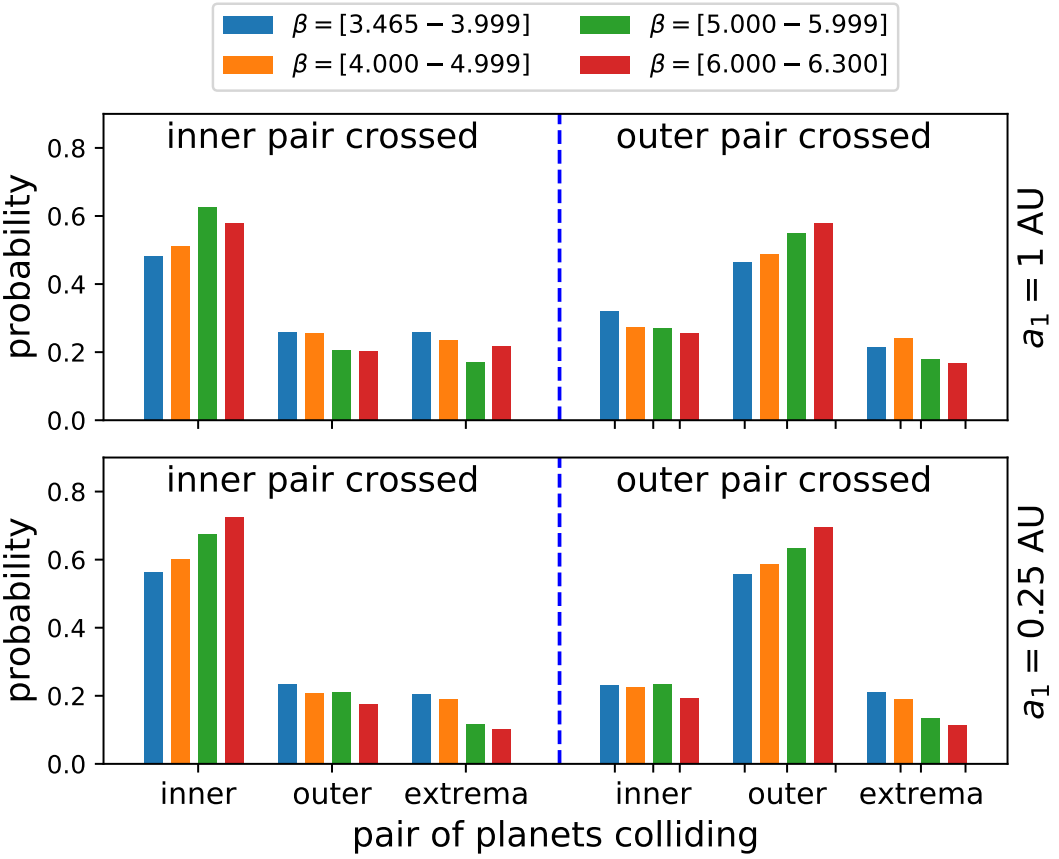


Figure 5.7: Probability of collision per pair of planets broken down by the pair of orbits that initially crossed and initial spacing, β, range. Probability is calculated as the fraction of collisions between a given pair of planets over the total number of collisions.The top panel is for systems initially at 1 AU while the bottom panel is initially at 0.25 AU. Inner and outer refer to the innermost and outermost pairs of planets, respectively. Extrema refers to the pair comprising the innermost and outermost planets.

5.3.3 Which planets collide?

I find a slight discrepancy between the prevalence of orbital crossings, with the innermost pair triggering 48% of crossings compared to 52% for the outermost pair. These percentages were calculated using $n = 4,800$ integrations and the expected stochastic variation, about the mean, i.e. 50%, is therefore approximately 0.72% (Dobrovolskis et al., 2007).

In the following, I designate the specific pair of planets that collide as the *collision pair*, and analogously I refer to the pair of planets that experienced an orbital crossing as the *crossing pair*. I find that across all values of β a collision between two planets is almost twice as likely if the same two planets were also involved in the orbital crossing. Figure 5.7 highlights clearly that this is the case with between 48% and 62%

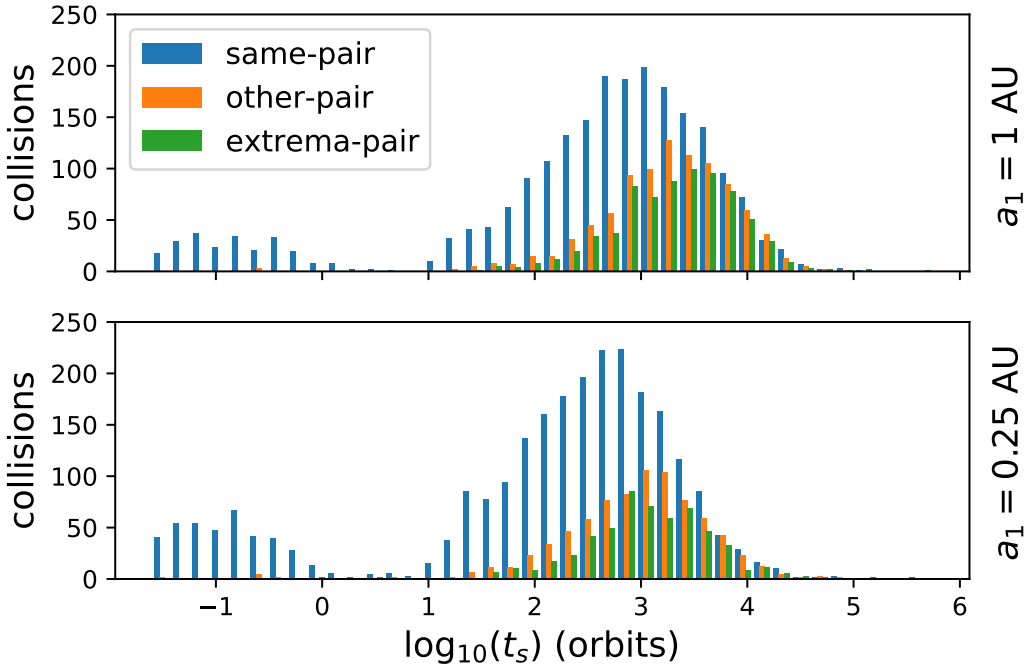


Figure 5.8: Post-crossing survival time distribution of collisions between different pairs of planets. Blue bars indicate the same pair both crossed orbits first and collided; orange indicates the pair that collided was not the pair that crossed; green indicates a collision between the inner and outer planets. The top panel is initially at 1 AU while the bottom panel is initially at 0.25 AU.

of collision events occurring between the crossing pair for systems at 1 AU depending on the initial orbital spacing. Moreover, these percentages appear to be invariant as to whether the inner or outer pair is involved in the orbital crossing. A clear trend can be seen with respect to β, where an increase in the initial orbital spacing between planets leads to an increased probability of collision between the crossing pair.

Figures 5.3 and 5.4 are coloured based on the collision and crossing pair for each system. As first crossings can only ever occur between neighbouring planets, it is possible to use only three colours for this: blue for *same-pair systems* whereby the same pair was involved in both the first orbit crossing and the collision, orange for *other-pair systems* to indicate that the colliding pair was the neighbouring pair that did not first cross, and green for *extrema-pair systems* to indicate a collision between the inner and outer planets. Across the whole range of β it can be seen that for systems with a t_s below the t_s model fit line collisions are predominantly between the first crossing pair. Figure 5.8 shows how these three combinations of events are distributed over time. Collisions that take place within a single orbit of orbit crossing are almost exclusively found in same-pair systems due either to simple immediate collisions or to the temporary gravitational bounding of planets discussed previously. Same-pair

collisions are the most likely outcome for all systems at 1 AU, shown in the top pane, until $t_s \approx 10^4$ orbits, followed by other-pair systems, with extrema-pair systems being the least likely. However, after this period the probability of collision between any combination of planets becomes almost identical, indicating that the mixing of planetary orbits after crossing is sufficient to overcome the increased probability of same-pair integrations due to the initial orbital configuration. Interestingly, the peak of other-pair and extrema-pair systems do not align, instead the former peaks first. This can be understood as the mixing process taking longer to cause the inner and outer planets orbits to overlap than to excite the middle planet enough to cross the orbits of both of its neighbours. In the bottom pane, it can be seen that at 0.25 AU the behaviour is similar; however, the number of collisions taking place within a single orbit roughly doubles.

5.4 Inclined Integration Suite

In the co-planar case, no system survived for more than a million orbits after the first orbital crossing. However, Rice et al. (2018) observed a number of non-co-planar systems that survived for their maximum simulation time of ten million orbits. Therefore, I now go on to examine the behaviour in the non-co-planar case described by the inclined suite of initial conditions in Section 5.2.5. As a reminder, these initial conditions include fifteen initial inclinations ranging from an initial orbital height of 0.10 r_H to r_H.

5.4.1 Dynamic heating

The systems studied in the inclined integration suite begin with modest inclinations and no eccentricities, making them dynamically cold. Figure 5.9 shows how the system heats up over time by plotting the root-mean-square (RMS) inclination and eccentricity over time. I calculate the mean over all runs that have experienced an orbital crossing and fit a linear model to this mean which is shown as the solid green line. Individual integrations are shown in purple until they experience an orbital crossing and in grey thereafter. For clarity, in Fig. 5.9 results of individual integrations are only shown for eighty integrations in the inclined suite for $\beta = 5.98$. When generating the data for this plot, the eccentricity and inclination were sampled uniformly in time which has the effect of causing an apparent increase in the variation over time of the

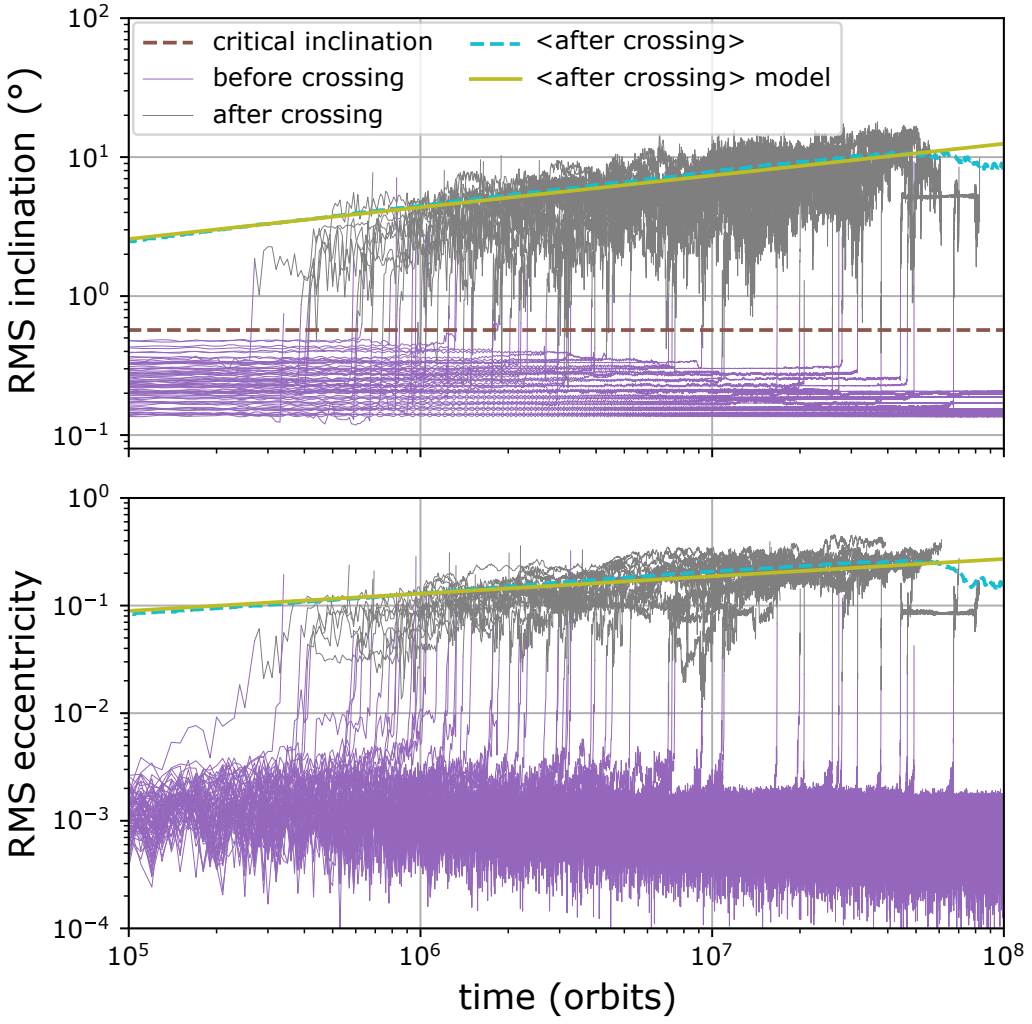


Figure 5.9: Inclination and eccentricity growth for individual systems from the inclined suite with $\beta = 5.98$. Only eighty configurations are included to aid clarity. Systems are shown in purple until they experience an orbital crossing and in grey thereafter. The RMS inclination and eccentricity values for all systems that have experienced an orbital crossing are shown (dashed blue). A linear model fitted to the mean of all systems that have experienced an orbital crossing is also shown (solid green).

eccentricity for systems that have not yet undergone an orbital crossing. This variation appears to indicate a decrease in the eccentricity for these systems over time, however, this is not the case and is simply an artifact of the sampling and plotting.

Rice et al. (2018) found that, for four-planet Neptune-mass systems, there are two distinct growth modes of RMS eccentricity before and after an instability event: Eccentricity evolves rapidly to a quasi-equilibrium at a value of 10^{-2} at which point encounters begin. After a period of mixing as a result of close approaches, systems transition into a new evolutionary phase during which eccentricity growth follows a

power-law form approximately $\propto t^{1/6}$. In the three-planet Earth-mass case, systems reach a quasi-equilibrium value of $e \approx 10^{-3}$ before a period of chaotic mixing and rapid growth, which finally settles into the new growth phase approximately $\propto t^{1/6}$.

The RMS inclinations in Fig. 5.9, on the other hand, while similar, are different to the four-planet Neptune-mass case. I also observe that the inclination of the systems remains at roughly the initial value until the first encounter, at which point they are rapidly excited before entering a new growth mode. This rapid excitation is in keeping with the findings of Matsumoto and Kokubo (2017). These behaviours can be seen by the horizontal inclination lines in the population of systems before crossing and in the power-law growth in the population afterward. Rice et al. (2018) stated that the trend towards long-lived systems depends upon only the RMS inclination being greater than the averaged ratio of Hill radius to semi-major axis, this is called the critical inclination and is marked on this plot. I also find this to be the case across all systems within the inclined suite: any systems that have experienced orbital crossing and have their RMS inclination damped below this threshold rapidly experience a collision. The key difference in results from simulations as compared to the four-planet, Neptune-mass case is that the power-law growth rate appears to be $\propto t^{1/4}$ as opposed to $\propto t^{1/3}$. I offer two possible explanations for this: 1) the dataset could be biased due to the non-random initial conditions used; or 2) there could be an underlying dependence between either the planetary mass or the number of planets within the system and the growth rate. Further investigation is needed to distinguish between these two possibilities.

5.4.2 Timescale to planet-planet collision

Figure 5.10 shows the crossing time for systems within the inclined suite. I find a large variance in crossing time across the inclined suite with a difference between the maximum and minimum crossing times at each value of β as large as two orders of magnitude in many cases. The spikes seen in Fig. 5.1 are also present in some of the inclined cases. A model of the type in Eq. (5.2) is fitted to the mean values of crossing time observed at each value of β, yielding coefficients $b' = 1.39$ and $c' = 2.18$. These values are in very good agreement with those from the standard suite. This is, however, where the similarities between the co-planar and inclined cases end. Figure 5.11 shows the post-crossing survival time for systems within the inclined suite, the times are much higher than in the co-planar case where the longest surviving system after crossing survived for roughly one million orbits. Here, the majority of systems survive for longer than this and, there are twenty-three systems

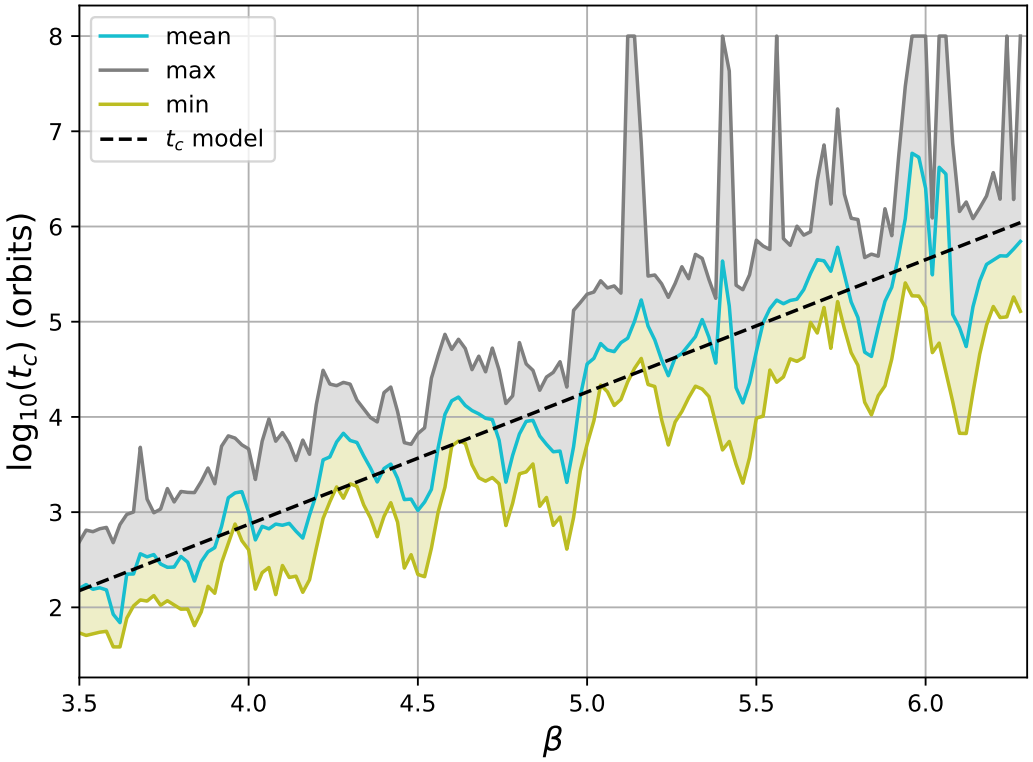


Figure 5.10: Time to orbital crossing against β for the inclined integration suite. The minimum, maximum and mean values of the one hundred and twenty integrations performed at each value of β are shown. Additionally, the t_c model is fitted to the mean values.

that do not experience any collision at all within the maximum simulation time (100 million orbits), equivalent to 0.14% of all integrations. Given that the post-crossing survival time is now approaching one percent of the lifetime of the Sun, it is much more likely that we actually could observe an inclined system between a crossing and a collision. However, at 0.25 AU no integrations survived for the full simulation duration after an integration.

Figure 5.12 shows the probability of a collision across all integrations within the inclined suite. The probability is calculated as the cumulative fraction of systems that have experienced collisions over the total number of systems. Results are included for systems initially at 1 AU as well as at 0.25 AU. Decreasing the initial distance to the star by this amount is identical to having artificially inflated the planetary radius R_P by a factor of four, i.e., made R_P approximately equal to that of Neptune, whilst keeping the innermost planet initially at 1 AU. It is therefore expected that the collision probability over time should increase with decreasing initial distance to the star. However, the increase is striking: for Earth analogues, the probability of a collision for

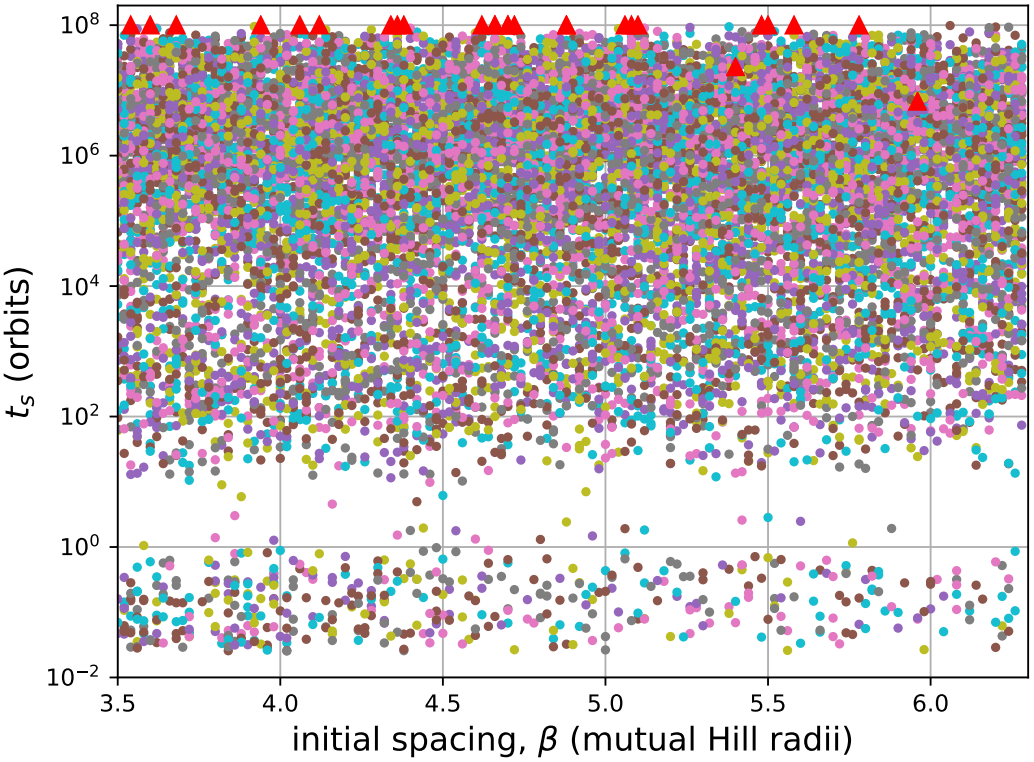


Figure 5.11: Post-crossing survival time of inclined integration suite with systems at 1 AU. Colours of data points are used only to aid in visualisation. The twenty-three systems that persisted for the full 10^8 orbits are highlighted via a red triangle, independent of their initial inclination. Note that most of these surviving systems had their initial orbital crossing in far less than 10^8 years, so they survived for almost 10^8 years post-crossing before the simulation was terminated and appear as triangles at the top of the plot; the two exceptions, which survived for $< 3 \times 10^7$ years, both had initial orbital separations $\beta > 5.3$.

a given system after one million orbital periods is roughly 50%, but for a Neptune radius (1 $M_\oplus$) planet at 1 AU that probability increases to over 75% across all β ranges, reaching almost 90% in all but one range. Furthermore, for the 1 AU systems it can be seen that the various β regions converge after roughly a million orbits. This indicates that the evolution after the first close encounter has reconfigured the system such that any prior collision probabilities due to initial orbital spacing are lost. To understand this, I can look at the collision probability in Figure 5.12 at one million orbits for 0.25 AU systems. These systems are equivalent to to a Neptune radius planet being placed at 1 AU and roughly 90% have experienced a collision within this timescale. I can therefore infer that the same roughly 90% of Earth radius planets at 1 AU must have experienced a close encounter within $4R_P$. The loss of prior collision probabilities due to orbital spacing after this point in time therefore appears to be driven by these particularly close encounters.

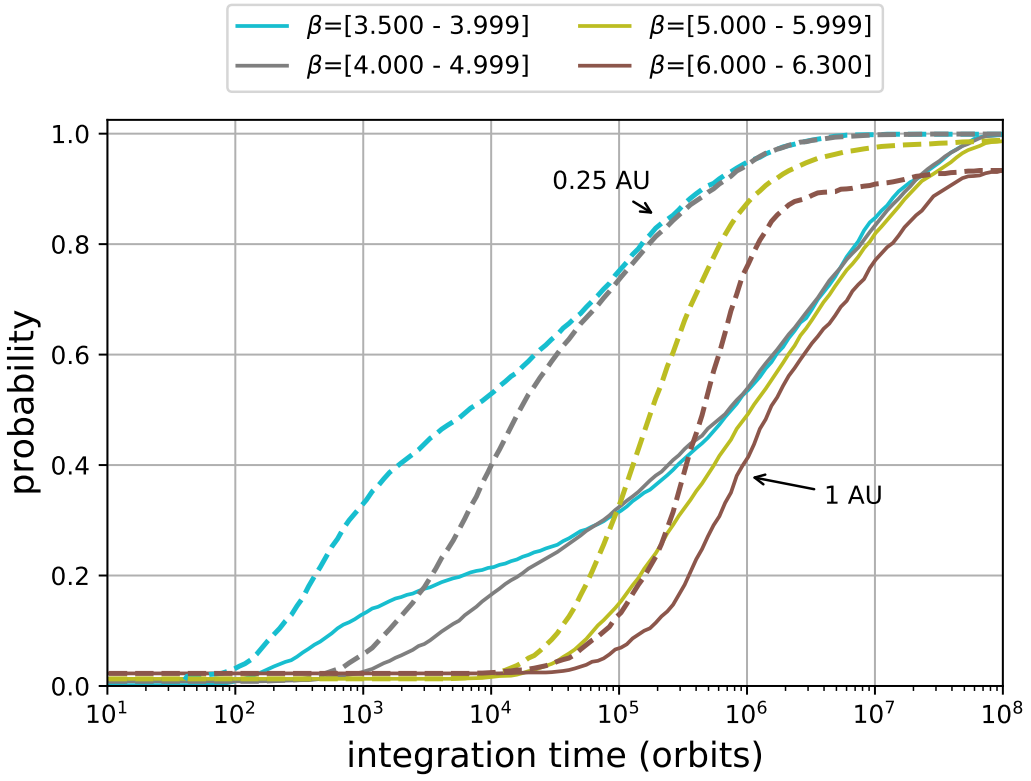


Figure 5.12: Probability of having experienced a collision over time for various regions of β in the inclined integration suite. The probability is calculated as the cumulative fraction of systems that have experienced collisions over the total number of systems. Solid lines show the probabilities for systems initially at 1 AU while the dashed lines are initially at 0.25 AU.

Figure 5.13 contains the distribution of post-crossing survival times for two subsets of the inclined suite results: the subsets each contain 1120 configurations, one at the minimum initial inclination (0.06°) and the other at the maximum initial inclination (0.58°). It can be seen that the distributions are different at each initial orbital radii and inclination. Firstly, the population of collisions taking place within several orbits of an orbital crossing decreases with increasing initial inclination. In both most highly inclined cases, there is only a single peak present in the distribution; however, this distribution is much more negatively skewed in systems initially at 1 AU. In the lowest inclination cases, there are two peaks present in addition to the one caused by immediate collisions. One peak is collocated with those found in the more inclined case. The second peak is centered at approximately $t_s = 10^{2.5}$. In the co-planar case I have seen that the distribution of post-crossing survival times are centered at approximately $10^{2.5}$ orbits and it is also known that if the inclination is below the critical threshold $i = r_H$ the number of collisions occurring within a factor of three of the orbital crossing increases (Rice et al., 2018). Both of these factors combined explain

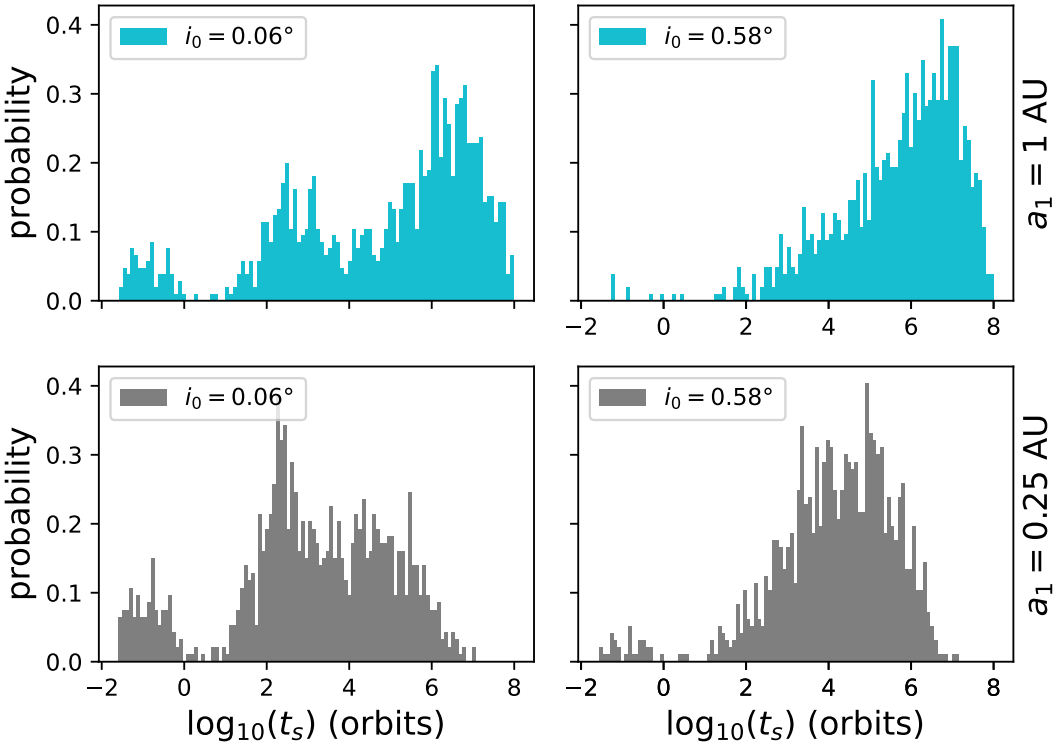


Figure 5.13: Distribution of post-crossing survival times in the inclined integration suite for systems after a close encounter. Each plot contains data from 1120 integrations across the entire inclined β range where $\beta = 3.5-6.3$. The upper two plots, in cyan, are for systems initially at 1 AU and the lower two plots, in grey, are for 0.25 AU. The two leftmost plots contain data for systems with the minimum initial inclination, $i_0 = 0.06°$, whereas the two rightmost plots contain data for systems with the maximum initial inclination, $i_0 = 0.58°$. Two systems survived for the full simulation time after an orbital crossing in the low inclination case at 1 AU whereas one survived in the high inclination case. No systems in the 0.25 AU case survived for the full simulation duration after an orbital crossing in any of the integrations.

the appearance of this second peak. Additionally, a larger proportion of systems at 0.25 AU experience a collision in this second peak.

The effect of increased initial inclination across the whole inclined integration suite can be seen in Fig. 5.14, where an increase in inclination, shown here in terms of orbital height, leads to a moderate increase in the median post-crossing survival times for systems at both 0.25 AU and 1 AU. The RMS inclination in compact three-body systems has been seen to stay approximately constant up until the time of the first close encounter, which means that the observed inclinations of actual planetary systems could provide information about the probable survival times of systems after an orbital crossing.

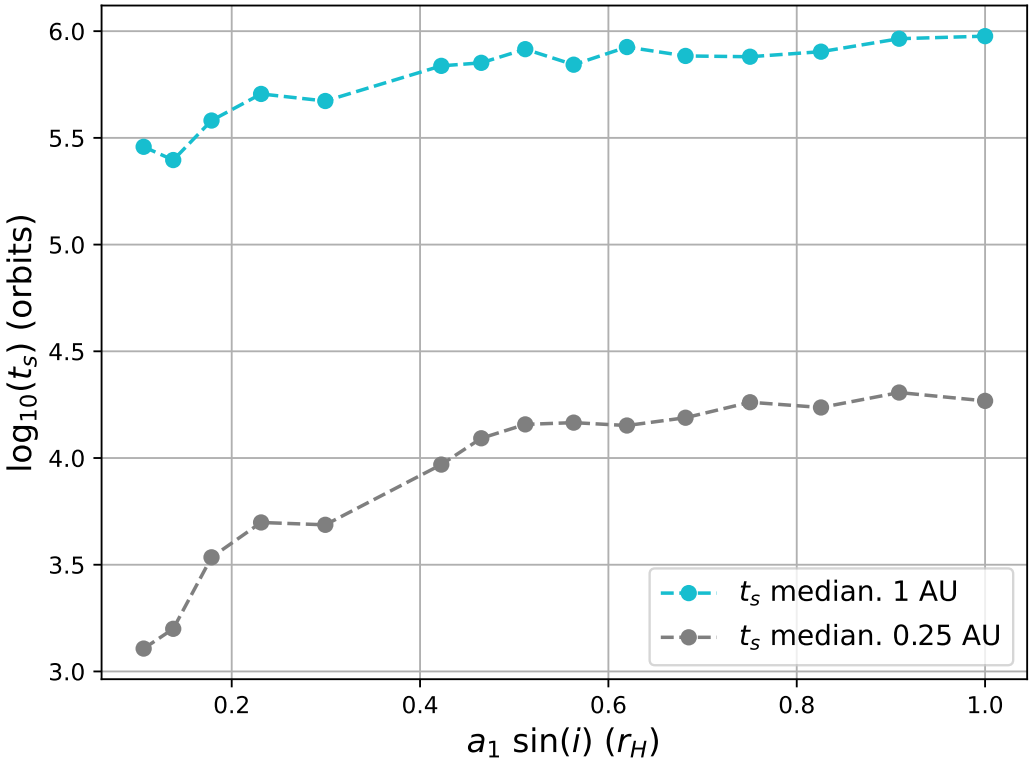


Figure 5.14: Median of the log post-crossing survival time for each value of initial inclination within the inclined suite represented by the orbital height as a fraction of the Hill radius. There are fifteen values of inclination used meaning that each data point plotted is the average of up to 1120 integrations; the only systems excluded are those that did not experience a collision in the maximum integration time.

The parameter that dominates the post-crossing survival time of systems in the inclined suite is the ratio of the planetary radius to the Hill radius at 1 AU. Figure 5.15 shows the median of the log post-crossing survival times for all systems in the suite at 1 AU. I find almost two orders of magnitude difference in the average survival time of systems with planets where $R_P/r_H = 0.017$ as compared to systems with planets where $R_P/r_H = 0.004$. This outweighs the effect of initial inclination on the survival times. Interestingly, systems surviving for the full 10^8 orbits can be seen all the way down to a value of $R_P/r_H = 0.0157$ where a rapid decrease in the lifetime of the longest lived systems is seen. This is equivalent to a planet initially located at 1 AU with a radius 3.5 times that of Earth.

In addition to the dependence of the post-crossing survival time upon the orbital elements of the system, I also find a correlation with the distance of the closest approach. Figure 5.16 shows the time taken for a collision to occur after the closest encounter experienced prior to it, at time denoted t_e, against the distance between the surfaces of the planets. Data points in the shaded grey area are excluded from the fitted models,

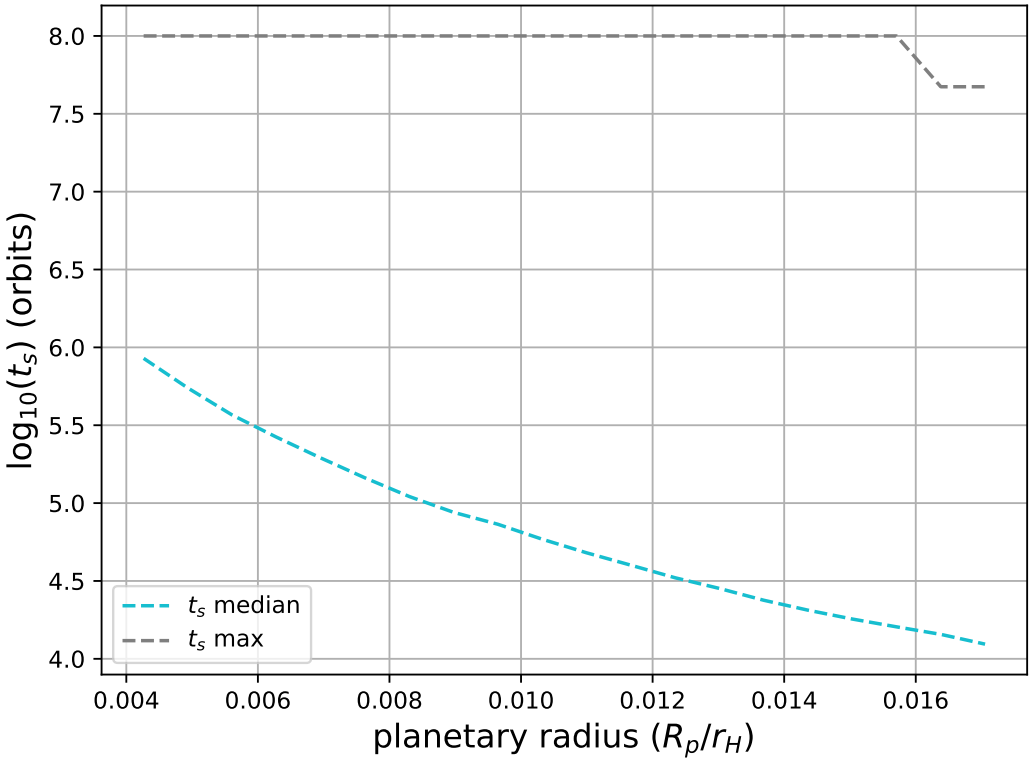


Figure 5.15: Median and maximum post-crossing survival time for systems as a function of the radius of planets relative to the Hill radius at 1 AU for systems in the inclined integration suite at 1 AU. Simulation times are capped at 10^8 orbits.

and this area corresponds to the boundary seen in Figure 5.3 at approximately eight orbits. Here, I see a strong negative correlation where a least squares model fitted to the log of $t_i - t_e$ and the miss distance of the encounter has a slope of -0.26 with a y-intercept of 1.6. Ergo, the closer an encounter experienced by a system the longer it is likely to survive afterwards. In this plot, each point is also coloured according to the post-crossing survival time of the system. Looking vertically from top to bottom at the colouring it can also be seen that the absolute post-crossing survival time of systems depends upon the miss distance of the closest encounter. It seems that for planetary systems to survive for a long time after an orbital crossing they must risk collision.

I find that the closest encounters are responsible for driving the largest changes in both inclination and eccentricity, and I believe that it is the increase in inclination that causes the trend seen in Figure 5.16. Figure 5.17 shows the time taken for a collision to occur after the closest encounter experienced prior to it against the time-averaged inclination range, i.e., the difference between the maximum and minimum inclinations. Systems with the largest inclination range survive for the longest after a close encounter, and the minimum miss distance, indicated through colouring, is key

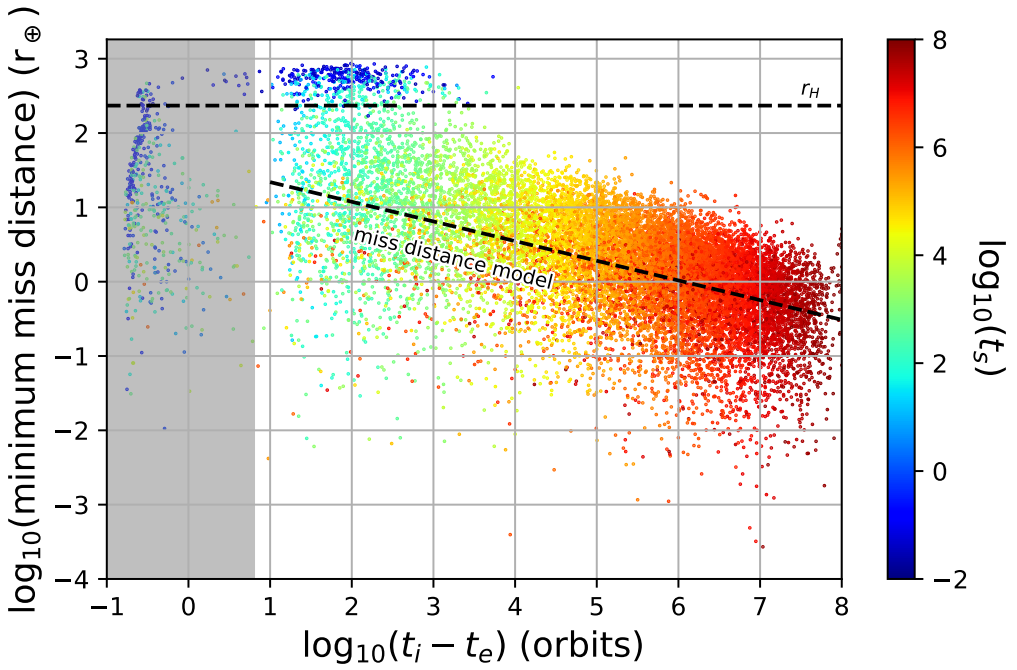


Figure 5.16: Time between closest encounter prior to impact and impact against the distance between the surfaces of the planets involved for systems at 1 AU in the inclined integration suite. The post-crossing survival time of each system is indicated through colouring. The grey shaded area indicates impacts that are possibly due to temporary gravitational capture which are excluded from the fitted model shown as a bold dashed black line. The horizontal dashed black line shows the Hill radius at 1 AU.

to increasing this range. Figure 5.18 is identical except it shows the time-averaged maximum eccentricity in a system. Again, the minimum miss distance can be seen to be responsible for the increases in eccentricity. These increases in eccentricity will also work to increase the lifetime of systems through a reduction in the effect of gravitational focusing on the combined physical/gravitational cross-sectional area of planets (Safronov, 1972).

5.4.3 Which planets collide?

Figure 5.19 is the equivalent to Fig. 5.8 but for the inclined suite. Similarly to the coplanar case, I find that collisions within a single orbit, due to immediate impacts and gravitational capture, are almost exclusively between the same pair involved in the crossing. I also find an increase in the number of collisions within this time frame in the 0.25 AU case compared to the 1 AU case. However, at a factor of approximately 3, here I see that the increase is more substantial. The distributions of survival times for

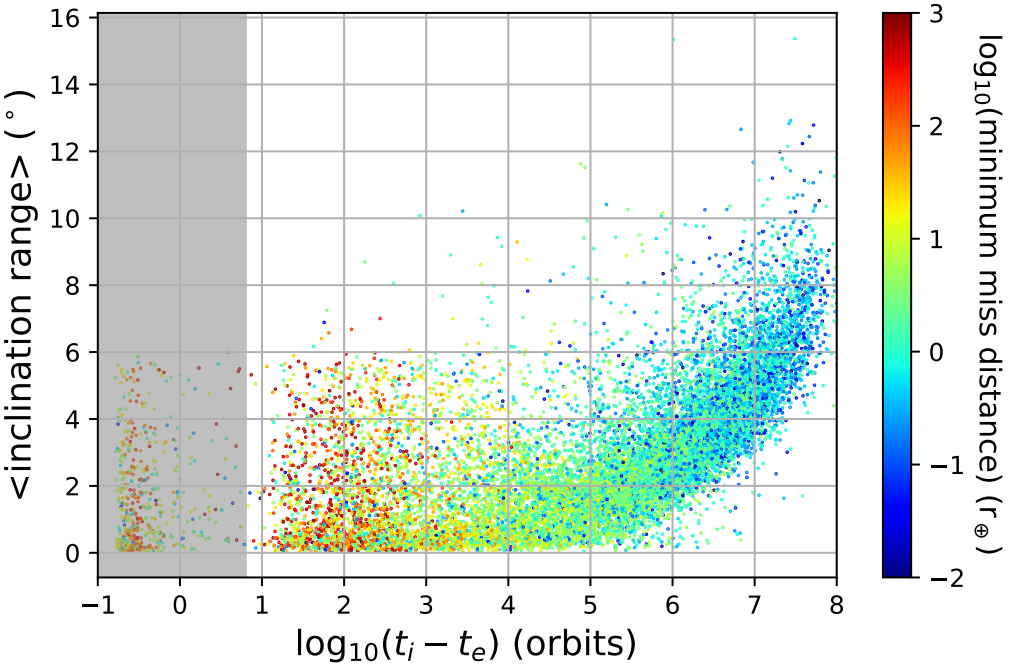


Figure 5.17: Time between closest encounter prior to impact and impact against the time-averaged inclination range, i.e. the difference between the smallest and largest inclinations, for systems at 1 AU in the inclined integration suite. The closest encounter experienced by a system is indicated through colouring.

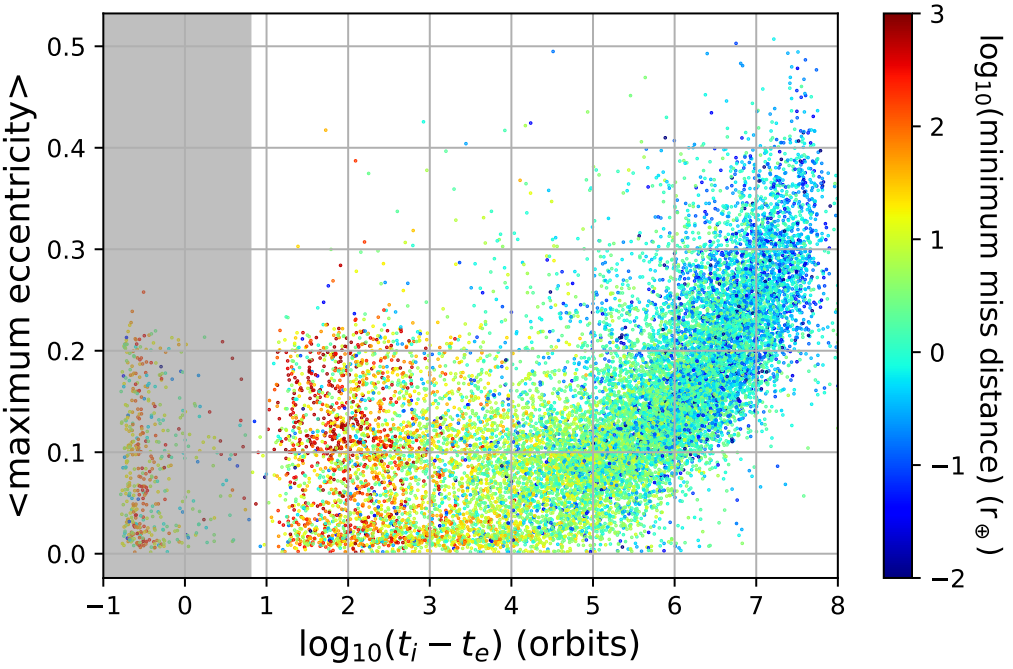


Figure 5.18: Time between closest encounter prior to impact and impact against the time-averaged maximum eccentricity for systems at 1 AU in the inclined integration suite. The closest encounter experienced by a system is indicated through colouring.

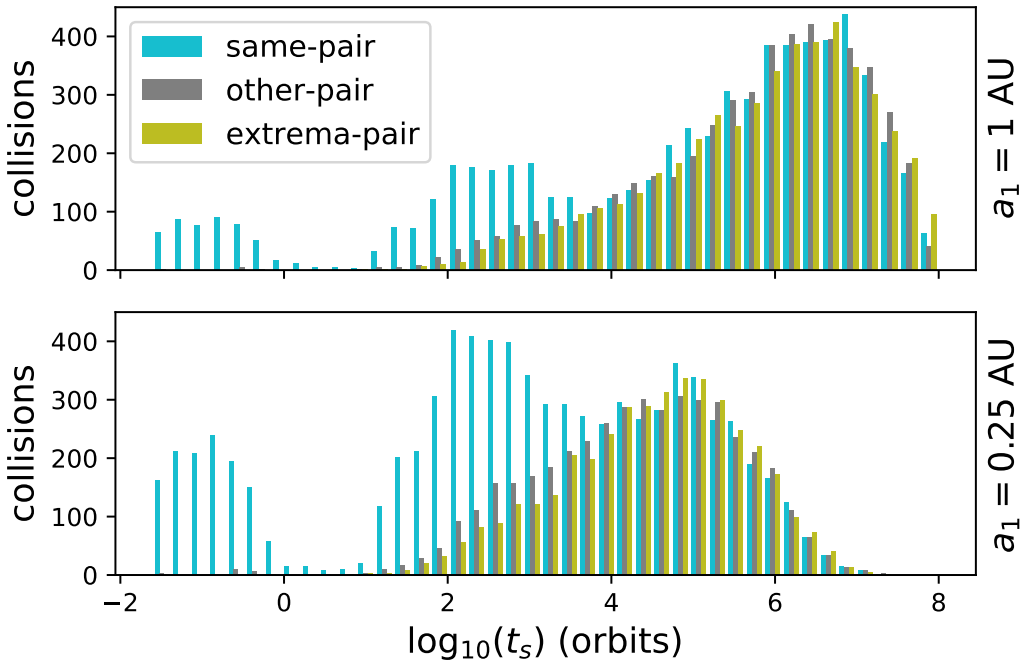


Figure 5.19: Time distribution of collisions between different pairs of planets in the inclined integration suite. Cyan bars indicate the same pair both crossed orbits and collided; dark grey indicates the pair that collided was not the pair that crossed; yellow indicates a collision between the inner and outer planets. The top panel is initially at 1 AU while the bottom panel is initially at 0.25 AU.

systems surviving after crossing for longer than a single orbit appear very different to the co-planar case. Nonetheless, some similarities in behaviour are present: in both the co-planar and inclined case there is a peak present of same-pair collisions between 10^2 and 10^3 orbits which are largely due to systems having a low inclination and therefore behaving similarly to the co-planar case. Adjusting for the number of systems in each suite I find that the fraction of systems colliding at this point is roughly five times smaller at 1 AU in the inclined case. The period for mixing in the inclined case is approximately 10^4 orbits, slightly longer than in the co-planar case, after which collisions between any pair of planets become equally likely.

5.5 Integrator comparison

To validate the performance of TES, I have chosen to perform additional integrations making use of IAS15 such that crossing and collision times can be compared. I performed integrations using IAS15 for all runs in the standard integration suite over the

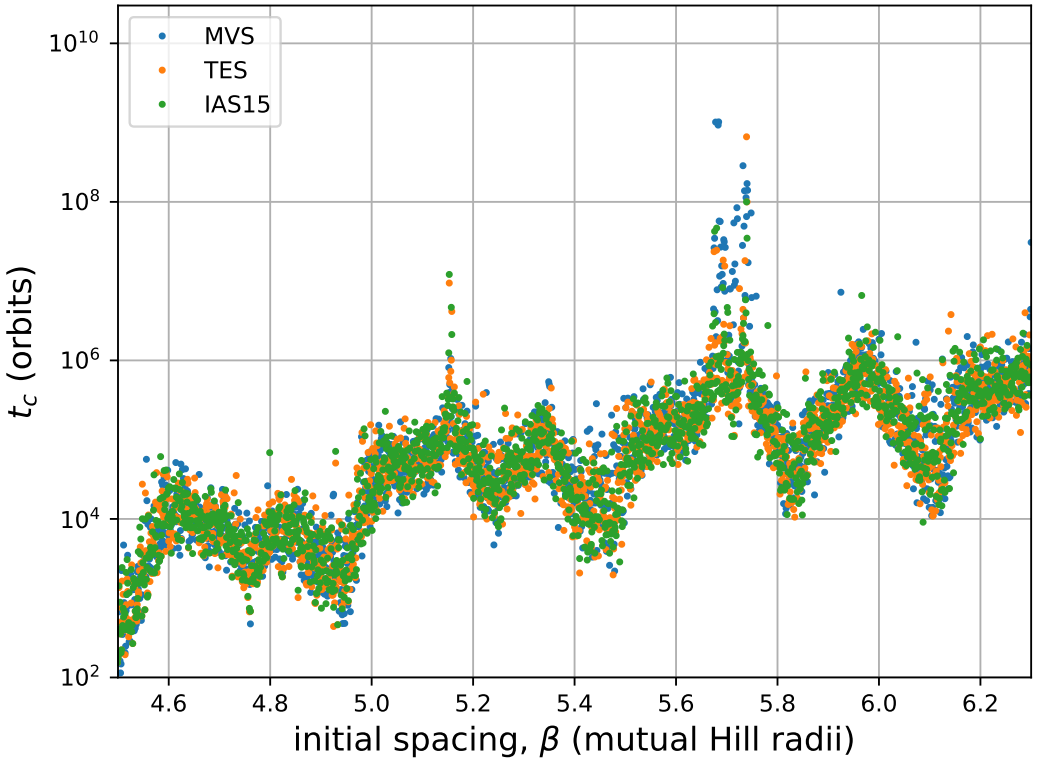


Figure 5.20: Plot showing a comparison of crossing times for three integration routines making use of the initial conditions in the standard integration suite.

range $\beta = 3.5$ to 6.3 using the standard configuration where the tolerance parameter $\epsilon = 10^{-9}$. Additionally, I also perform identical comparisons with the dataset of crossing times in Lissauer and Gavino (2021) obtained using the MVS implementation within MERCURY. Throughout this section, TES and IAS15 check for an orbital crossing on every integration step whereas the MVS and hybrid schemes check for an orbital crossing once every ten years. Table 5.6 compares the crossing time obtained by TES and IAS15; I find very good agreement in results especially in lower β ranges where the lifetime of systems is shorter. In particular, for systems where $\beta < 4.0$ I find that 68% of systems experience an orbital crossing within 1% of one another, increasing to 79% if the tolerance is relaxed to being within 10% of one another. These percentages can be compared to the data presented in Table 5.3 comparing the performance of TES under the influence of an initial 100 m perturbation in the position of the innermost planet along its orbital arc. Comparison of the summary columns in these two tables reveals that the difference in crossing time between TES and IAS15 is smaller than the effect of the 100 m perturbation. Tables 5.7 and 5.8 compare the crossing times found in TES and IAS15, respectively, against those obtained with MERCURY. Unsurprisingly, I find that the comparison yields very similar statistics in both cases. In particular, the runs finishing within 1% and 10% of each other drop

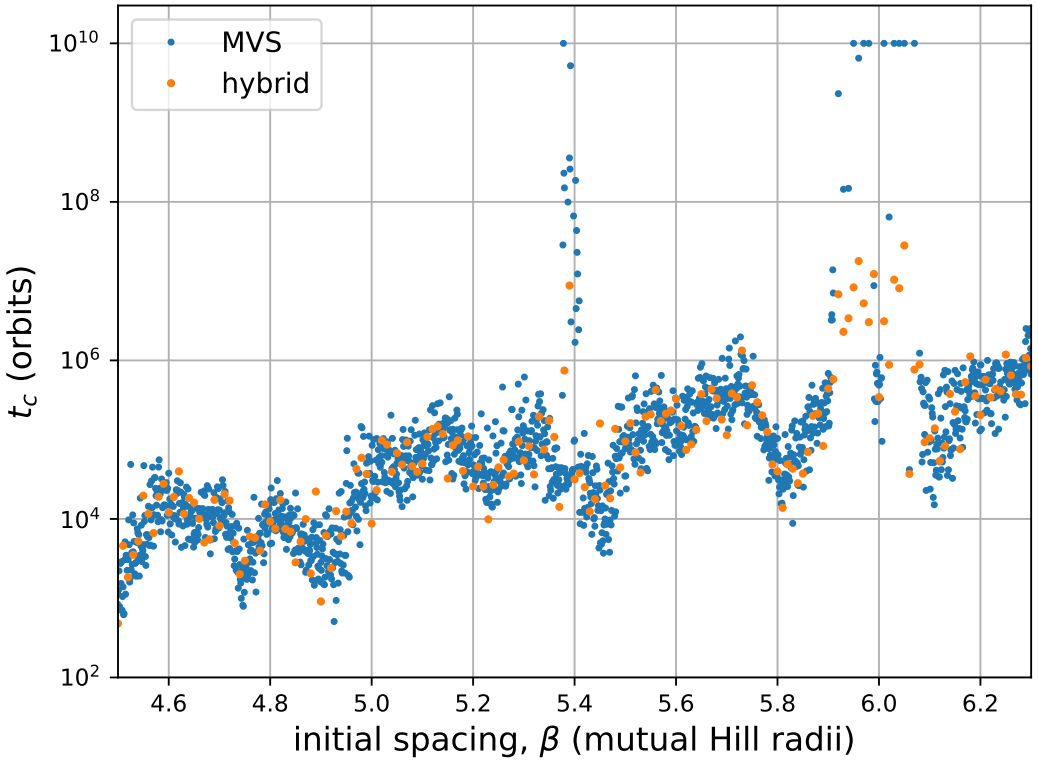


Figure 5.21: Plot showing a comparison of crossing time for two integrations routines with shifted initial longitudes described in the main body of text. The MVS and hybrid schemes have a density of one thousand and one hundred runs per unit β, respectively.

by at least 54% in the lowest region of β, although the reduction is much less pronounced at higher β. I also find that the difference in crossing times between TES and IAS15 at higher β is very similar to that of TES and IAS15, and MVS. Finally, Table 5.9 compares the collision times for TES and IAS15, and this is where I find the largest differences between the two schemes. In the summary column, over the entire β range, I see that the number of runs finishing within 1%, 10% and a factor of two have decreased substantially when compared to Table 5.6 performing the same comparison but for crossing times. The majority of the reduction in these statistics comes from the integrations where $\beta < 5$ and the evolution after crossing is still a substantial fraction of the overall simulation time. These differences highlight the sensitivity of integrations to close encounters.

In addition to the quantitative comparisons between integrators thus far, I have also encountered some qualitative differences in behaviour between the schemes examined so far and the hybrid integration scheme within MERCURY. Figure 5.20 visualises the comparison between integration schemes found in Table 5.6 to 5.8. It shows how tightly the crossing times for the three schemes are clustered to one another, and in

particular it shows that in the region located around $\beta = 5.7$, where a region of high stability is found, the schemes all perform identically capturing the long-lived system behaviour. The hybrid integration scheme within MERCURY combines the MVS scheme with a non-symplectic integrator to allow for close approaches to be handled, and it would therefore seem like an ideal candidate for performing all experiments in this article. Figure 5.21 shows the results of an experiment identical to that described in Section 5.2.3 except with initial longitudes of $M_j = [0, 10.17, 20.33]^\circ$. In this experiment, the MVS scheme has a density of one thousand integrations per unit β whereas the hybrid scheme has a reduced density of one hundred per unit β. Both the MVS and hybrid schemes use what is considered a conservative step size of 18 days resulting in slightly over 20 steps per orbit. For the most part, the schemes agree well with each other; however, a key difference between the schemes can be seen in the region $\beta = 6$ where no integrations performed by the hybrid scheme lasted for longer than 10^8 orbits despite the majority of MVS scheme integrations lasting for 10^{10} orbits. It appears that the hybrid scheme is not accurate enough properly to capture the dynamics in this region which has led to a population of short-lived systems that should have been stable for a lot longer. Therefore, this is a particular example of when a hybrid symplectic integration scheme is not necessarily precise enough to fully resolve the dynamics and a scheme such as TES should instead be used.

Table 5.6: Comparison of *crossing times* of systems using identical values of β for the standard initial longitudes using *TES* and *IAS15*. Systems have the innermost planet initially placed at 1 AU.

Interval:	[3.5, 3.999]	[4.0, 4.999]	[5.0, 5.999]	[6.0, 6.3]	[3.5, 6.3]
number of runs in the range	500	1000	1000	301	2801
$< \log_{t_c}(\text{TES}) - \log_{t_c}(\text{IAS15}) >$	−0.014	−0.005	0.004	−0.041	−0.007
$< \lvert\log_{t_c}(\text{TES}) - \log_{t_c}(\text{IAS15})\rvert >$	0.034	0.156	0.289	0.352	0.203
$t_c(\text{IAS15}) < 0.5t_c(\text{TES})$	2 (0.40%)	77 (7.70%)	200 (20.00%)	56 (18.60%)	335 (11.96%)
$0.5t_c(\text{TES}) < t_c(\text{IAS15}) < 2t_c(\text{TES})$	489 (97.80%)	848 (84.80%)	608 (60.80%)	164 (54.49%)	2109 (75.29%)
$t_c(\text{TES}) < 0.5t_c(\text{IAS15})$	9 (1.80%)	75 (7.50%)	192 (19.20%)	81 (26.91%)	357 (12.75%)
within 10% of one another	395 (79.00%)	291 (29.10%)	105 (10.50%)	27 (8.97%)	818 (29.20%)
within 1% of one another	340 (68.00%)	120 (12.00%)	9 (0.90%)	1 (0.33%)	470 (16.78%)

Table 5.7: Comparison of *crossing times* of systems using identical values of β for the standard initial longitudes using *TES* and *MVS* with the innermost planet initially at 1 AU. Data marked with a * are likely to be somewhat erroneous due to the MVS scheme integrations only checking for an orbital crossing once every ten orbits.

Interval:	[3.5, 3.999]	[4.0, 4.999]	[5.0, 5.999]	[6.0, 6.3]	[3.5, 6.3]
number of runs in the range	500	1000	1000	301	2801
$< \log_{t_c}(\text{TES}) - \log_{t_c}(\text{MVS}) >$	−0.045	0.002	−0.081	−0.044	−0.041
$< \lvert\log_{t_c}(\text{TES}) - \log_{t_c}(\text{MVS})\rvert >$	0.244	0.303	0.355	0.371	0.318
$t_c(\text{MVS}) < 0.5t_c(\text{TES})$	64 (12.80%)	215 (21.50%)	172 (17.20%)	56 (18.60%)	507 (18.10%)
$0.5t_c(\text{TES}) < t_c(\text{MVS}) < 2t_c(\text{TES})$	335 (67.00%)	589 (58.90%)	558 (55.80%)	162 (53.82%)	1644 (58.69%)
$t_c(\text{TES}) < 0.5t_c(\text{MVS})$	101 (20.20%)	196 (19.60%)	270 (27.00%)	83 (27.57%)	650 (23.21%)
within 10% of one another	67 (13.40%)	99 (9.90%)	81 (8.10%)	21 (6.98%)	268 (9.57%)
within 1% of one another	12 (2.40%)*	4 (0.40%)*	6 (0.60%)	4 (1.33%)	26 (0.93%)*

Table 5.8: Comparison of *crossing times* of systems using identical values of β for the standard initial longitudes using *IAS15* and *MVS* with the innermost planet initially at 1 AU. Data marked with a * are likely to be somewhat erroneous due to the MVS scheme integrations only checking for an orbital crossing once every ten orbits.

Interval:	[3.5, 3.999]	[4.0, 4.999]	[5.0, 5.999]	[6.0, 6.3]	[3.5, 6.3]
number of runs in the range	500	1000	1000	301	2801
$< \log_{t_c}(\text{IAS15}) - \log_{t_c}(\text{MVS}) >$	−0.031	0.008	−0.086	−0.002	−0.034
$< \lvert\log_{t_c}(\text{IAS15}) - \log_{t_c}(\text{MVS})\rvert >$	0.243	0.291	0.359	0.360	0.314
$t_c(\text{MVS}) < 0.5t_c(\text{IAS15})$	72 (14.40%)	201 (20.10%)	189 (18.90%)	74 (24.58%)	536 (19.14%)
$0.5t_c(\text{IAS15}) < t_c(\text{MVS}) < 2t_c(\text{IAS15})$	333 (66.60%)	605 (60.50%)	552 (55.20%)	155 (51.50%)	1645 (58.73%)
$t_c(\text{IAS15}) < 0.5t_c(\text{MVS})$	95 (19.00%)	194 (19.40%)	259 (25.90%)	72 (23.92%)	620 (22.13%)
within 10% of one another	66 (13.20%)	112 (11.20%)	79 (7.90%)	27 (8.97%)	284 (10.14%)
within 1% of one another	10 (2.00%)	5 (0.50%)	10 (1.00%)	2 (0.66%)	27 (0.96%)

Table 5.9: Comparison of *collision times* of systems using identical values of β for the standard initial longitudes using *TES* and *IAS15* with the innermost planet initially at 1 AU.

Interval:	[3.5, 3.999]	[4.0, 4.999]	[5.0, 5.999]	[6.0, 6.3]	[3.5, 6.3]
number of runs in the range	500	1000	1000	301	2801
$< \log_{t_c}(\text{TES}) - \log_{t_c}(\text{IAS15}) >$	−0.080	−0.024	0.003	−0.039	−0.026
$< \lvert\log_{t_c}(\text{TES}) - \log_{t_c}(\text{IAS15})\rvert >$	0.485	0.296	0.280	0.352	0.330
$t_c(\text{IAS15}) < 0.5t_c(\text{TES})$	125 (25.00%)	170 (17.00%)	189 (18.90%)	59 (19.60%)	543 (19.39%)
$0.5t_c(\text{TES}) < t_c(\text{IAS15}) < 2t_c(\text{TES})$	207 (41.40%)	626 (62.60%)	622 (62.20%)	156 (51.83%)	1611 (57.52%)
$t_c(\text{TES}) < 0.5t_c(\text{IAS15})$	168 (33.60%)	204 (20.40%)	189 (18.90%)	86 (28.57%)	647 (23.10%)
within 10% of one another	62 (12.40%)	117 (11.70%)	115 (11.50%)	29 (9.63%)	323 (11.53%)
within 1% of one another	33 (6.60%)	20 (2.00%)	14 (1.40%)	1 (0.33%)	68 (2.43%)

5.6 Summary

I performed more than 25,000 integrations of compact three-planet systems with the TES integration tool for a maximum time of 10^9 orbits of the innermost planet or until the first collision of planets. In all of these integrations, I found that TES performed extremely well and conserved energy to a very high degree of precision. Importantly, the results obtained with TES were very similar to those obtained with IAS15 which further validates the new tool. Finally, TES was also able to resolve regions of high stability that could not be captured with a hybrid symplectic integrator.

During these integrations, I chose to focus my attention on the effects of orbital spacing and therefore distributed system configurations across a wide range of initial values evenly spaced in β. Efforts were initially focused on the co-planar case where it is easier to isolate the effects of increasing β but then extended to include the inclined case as well.

I find in the co-planar suite that planetary systems are doomed after an orbital crossing: they rapidly experience a collision within a maximum observed time of less than one million orbits. However, despite this prognosis, I found that systems with a wider initial spacing of planets do survive longer, exhibiting a median post-crossing survival time following a slope $\log_{10}(t_s) \propto 0.12\beta$. Additionally, I show that three distinct populations of post-instability impact behaviour are present, with very few outliers:

1. immediate collisions within a tenth of an orbit,
2. prompt collisions between a tenth of an orbit and two orbits,
3. those surviving for much longer than ten orbits.

The pathologies of these different behaviours have been identified and each of them are also observed in the inclined suite.

The probabilities of a collision between specified planetary pairs were also calculated and it was found that collisions will occur between the same pair of planets that initially crossed in the majority of cases, ranging from 48% to 62% depending on the region of β. These probabilities increase further depending on the radius of the planet, with Neptune-radius planets experiencing probabilities as high as 76%. Despite this increase in probabilities in the co-planar case, the post-crossing survival time only weakly depends upon the planetary radius, causing an increase of only 10^3 orbits. In the inclined suite, however, I observe that the planetary radius is the main driver

of the post-crossing survival time. I find a decrease in median post-crossing survival time of almost two orders of magnitude between Earth and Neptune radius planets. Additionally, the initial orbital inclinations have been shown to also influence the post-crossing survival times across the full range of β by as much as an order of magnitude.

Additionally, I looked at the RMS eccentricity and inclination growth of all systems within the inclined suite after an orbital crossing. Here, I replicate the eccentricity growth rate $e \propto t^{1/6}$ found in other studies. I do, however, find the growth rate of the inclination to be $i \propto t^{1/4}$ instead of the $i \propto t^{1/3}$ observed in previous work.

Finally, I have shown that systems that experience the closest encounters also survive for the longest, and planetary systems that wish to survive must therefore live dangerously.

Chapter 6

Conclusions and future work

6.1 Conclusions

Our understanding of the formation and evolution of exoplanet systems has improved astronomically over the past thirty years, but there is still much that we are yet to discover. The powerful combination of order of magnitude improvements in observations coupled with the creation of more sophisticated models is largely responsible for our widened comprehension, and this synergy looks set to continue yielding new understanding long into the future. Observations of exoplanet systems have revealed many unexpected features, such as the existence of compact planetary systems, with very different dynamics to that of our own solar system. As new observatories come online, these datasets will slowly become larger and less biased, and likely yield new enigmas that will be left for theorists to interpret. In this regard, n-body simulations are the natural choice of tool for theorists, and improving the sophistication and precision of these models is therefore key to improving our understanding of the cosmos.

This thesis has concerned itself with further understanding the behaviour of compact three-planet systems. In particular, the analysis performed in Chapter 5 provides a deeper understanding of the behaviour of such systems in the period of time after an instability event leading up to a collision between planets. This period of time is particularly dynamic, and repeated close encounters between planets drive up the eccentricity and inclination of bodies, ultimately leading to them being placed on a collision trajectory with one another. Clearly, this imposes acute constraints on any n-body integrator wishing to model these dynamics. Not only must integrators be able to handle close encounters at machine precision, but they must do this after already

having integrated for hundreds of millions of dynamical periods leading up to the first close encounter.

In Chapter 4, I developed a novel tool, the Terrestrial Exoplanet Simulator (TES), to address these modelling challenges more efficiently. TES is currently the fastest tool available for accurately modelling the behaviour of terrestrial exoplanet systems after an instability event. TES preserves energy over long timescales in a way that respects Brouwer's law and is therefore considered error optimal in terms of this precision metric. To confirm the performance of TES, I performed extensive comparisons with the leading tools in this field. Here, I confirmed that TES has very favourable energy conservation properties both during close encounters and over integrations spanning up to one billion dynamical periods.

After quantifying the performance of TES, I then used it to study the post-instability behaviours of compact three-planet systems in Chapter 5. I performed over $25,000$ long-term integrations across a wide parameter space of initial conditions to understand the influences upon the survival time once a system is deemed unstable. In particular, I found that the radius of planets is key to their longevity in this period. However, other factors do play a lesser role, e.g. the initial spacing of planets in terms of relative orbital periods. Interestingly, temporary gravitational capture between planets is responsible for a small number of very rapid collisions after an instability event. Moreover, a final revelation was that planetary systems that experience the closest encounters also survive for the longest, and planetary systems that wish to survive must therefore live dangerously.

6.2 Future work

Whilst the TES algorithm in Chapter 4 is already an improvement on other similar schemes, there are additional features that are worthy of further investigation. An example of this is investigating the effects of alternative coordinate systems upon the performance. Likewise, there are still likely areas of the code base that could be further optimised to take further advantage of modern processor features such as the vectorization registers. Overall, these are likely to be marginal improvements but could possibly result in an additional speed up of $5-10\%$. However, a key area that could be improved with regards to TES is making it more accessible and user friendly. To this end, preliminary work is being undertaken at the moment to incorporate the TES algorithm into the REBOUND package which would make the code more easily accessible to people wishing to use it while also improving usability features.

Chapter 5 provided a detailed analysis of the post-instability behaviours of compact three-planet systems leading up to a collision between planets. The analysis performed here was specific to compact three-planet systems and also to Earth-mass planets orbiting solar-mass stars. It is of interest to understand how these effects generalise to different planetary masses, multiplicities and initial conditions. There is still much work to be done on understanding the probability of impacts between planets in unstable systems.

The Chapter 3 analysis into multistep collocation methods has shown that they are a highly capable class of integrator for modelling problems where the dynamics vary on significantly different timescales. Additionally, particular configurations of MCM have been shown decrease the computational cost of a given integration. Therefore, it is of interest to investigate the possibility of creating a variable step size and/or variable order implementation of the MCM to enable a wider variety of problems to be tackled efficiently.

6.3 Environmental impact of this research

Simulation based research projects, by their very nature, oftentimes require expensive computations to be carried out, and an often overlooked aspect of these calculations is the environmental impact that they can have. Carbon dioxide (CO_2) emissions are a widely used proxy for the environmental impact of our behaviors and I therefore adopt this metric in the following discussion.

I estimate that during my PhD I have performed approximately $720,000$ hours of computation using the Iridis supercomputing cluster at the University of Southampton. The HPC team were kind enough to provide an estimate for the energy usage per core in their datacentre, which is roughly 0.0125 kWh. Finally, in the UK the CO_2 created as a byproduct of electricity generation is approximately 0.223 kg kWh^{-1}. Combining these values yields a total carbon emission of two tonnes, which, for context, is similar to an economy seat on a transatlantic flight from London to New York. Given the volume of CO_2 produced in astrophysical simulations, an additional benefit of producing more computationally efficient algorithms, such as TES, is therefore a reduction in the overall environmental cost of understanding the universe.

References

S. J. Aarseth and K. Zare. A regularization of the three-body problem. *Celestial Mechanics*, 10(2):185–205, 1974. ISSN 00088714. .

Sverre Aarseth. *Graviational N-body simulations*. Cambridge Press, Cambridge, 1st edition, 2003. ISBN 9780521432726.

Sverre Aarseth. *The Cambridge Nbody Lectures*. Cambridge Press, Cambridge, 2008. ISBN 9781402084317.

Sverre J. Aarseth. From NBODY1 to NBODY6: The Growth of an Industry. *Publications of the Astronomical Society of the Pacific*, 111(765):1333–1346, 1999. ISSN 0004-6280. .

V. Akimkin, S. Zhukovska, D. Wiebe, D. Semenov, Ya Pavlyuchenkov, A. Vasyunin, T. Birnstiel, and Th Henning. Protoplanetary disk structure with grain evolution: The ANDES model. *Astrophysical Journal*, 766(1), 2013. ISSN 15384357. .

Davide Amato, Giulio Baù, and Claudio Bombardelli. Accurate orbit propagation in the presence of planetary close encounters. *Monthly Notices of the Royal Astronomical Society*, 470(2):2079–2099, 2017. ISSN 13652966. .

Philip J. Armitage. *Astrophysics of planet formation*. Cambridge University Press, 2009. ISBN 9780511802225. .

A. Azzalini and A. Capitanio. Statistical applications of the multivariate skew normal distribution. *Journal of the Royal Statistical Society. Series B: Statistical Methodology*, 61(3):579–602, 1999. ISSN 13697412. .

Elizabeth Bailey and Konstantin Batygin. The hot jupiter period-mass distribution as a signature of in situ formation. *The Astronomical Journal Letters*, 866(1):L2, 2018. .

Thomas Barclay, Joshua Pepper, and Elisa V. Quintana. A revised exoplanet yield from the Transiting Exoplanet Survey Satellite (TESS). *The Astronomical Journal Supplement Series*, 239(2):15, 2018. ISSN 23318422. .

Josh Barnes and Piet Hut. A hierarchical O(N log N) force-calculation algorithm. *Nature*, 324(6096):446–449, 1986. ISSN 00280836. .

Peter Bartram and Alexander Wittig. Terrestrial Exoplanet Simulator (TES): an error optimal planetary systems integrator that permits close encounters. *Monthly Notices of the Royal Astronomical Society*, 504(1):678–691, 2021. .

Peter Bartram, Alexander Wittig, Jack J Lissauer, Sacha Gavino, and Hodei Urrutxua. Orbital stability of compact three-planet systems, II : Post-instability impact behaviour. *In print, Monthly Notices of the Royal Astronomical Society*, 2021.

Richard H. Battin. *An Introduction to the Mathematics and Methods of Astrodynamics*. AIAA, New York, 2nd edition, 1987. ISBN 1-56347-342-9.

Konstantin Batygin and Gregory Laughlin. Jupiter's decisive role in the inner Solar System's early evolution. *Proceedings of the National Academy of Sciences of the United States of America*, 112(14):4214–4217, 2015. ISSN 10916490. .

Konstantin Batygin, Peter H Bodenheimer, and Gregory P Laughlin. In Situ Formation and Dynamical Evolution of Hot Jupiter Systems. *The Astrophysical Journal*, 829 (2):114, 2016a. ISSN 1538-4357. .

Konstantin Batygin, Peter H. Bodenheimer, and Gregory P. Laughlin. In Situ Formation and Dynamical Evolution of Hot Jupiter Systems. *The Astrophysical Journal*, 829 (2):114, 2016b. ISSN 1538-4357. .

D.G. Bettis and V. Szebehely. Treatment of Close Approaches in the Numerical Integration of the Gravitational Problem of n-Bodies. *International Journal of Solids and Structures*, 51(9):1646–1661, 1971. ISSN 00207683. .

Jürgen Blum. Dust Evolution in Protoplanetary Discs and the Formation of Planetesimals: What Have We Learned from Laboratory Experiments? *Space Science Reviews*, 214(2):1–19, 2018. ISSN 15729672. .

Dirk Brouwer. On the accumulation of errors in numerical integration. *The Astronomical Journal*, 46(1072):149, 1937. ISSN 00046256. .

Roland Bulirsch and Josef Stoer. Numerical treatment of ordinary differential equations by extrapolation methods. *Numerische Mathematik*, 8(1):1–13, 1966. ISSN 0029599X. .

J. C. Butcher, A. T. Hill, and T. J.T. Norton. Symmetric general linear methods. *BIT Numerical Mathematics*, 56(4):1189–1212, 2016. ISSN 15729125. .

J.C Butcher. On the implementation of implicit Runge-Kutta methods. *BIT Numerical Mathematics*, 16:237–240, 1976. ISSN 0010485X. .

J.C Butcher. A History of Runge-Kutta Methods. *Applied Numerical*, 20:247–260, 1996.

J.C Butcher. *Numerical Methods for Ordinary Differential Equations*. John Wiley and Sons, 2nd edition, 2008. ISBN 978-0-470-72335-7.

J E Chambers. A hybrid symplectic integrator that permits close encounters between massive bodies. *Monthly Notices of the Royal Astronomical Society*, 304(4):793–799, 1999. ISSN 00358711. .

J E Chambers. Giant planet formation with pebble accretion. *Icarus*, 233:83–100, 2014. ISSN 00191035. .

John Chambers. Rapid Formation of Jupiter and Wide-Orbit Exoplanets in Disks with Pressure Bumps. *arXiv*, 2021. URL http://arxiv.org/abs/2104.10704.

John E. Chambers. Planetary accretion in the inner Solar System. *Earth and Planetary Science Letters*, 223(3-4):241–252, 2004. ISSN 0012821X. .

Rohit Chandra, Dagum Leonardo, Dave Kohr, Dror Maydan, Jeff McDonald, and Ramesh Menon. *Parallel Programming in OpenMP*. 2001. ISBN 1558606718.

Sourav Chatterjee, Eric B Ford, Soko Matsumura, and Frederic A Rasio. Dynamical Outcomes of Planet-Planet Scattering. *The Astrophysical Journal*, 686:580, 2008. ISSN 2041-8213.

Matthew S. Clement, Sean N. Raymond, Nathan A. Kaib, Rogerio Deienno, John E. Chambers, and André Izidoro. Born eccentric: Constraints on Jupiter and Saturn's pre-instability orbits. *Icarus*, 355:1–35, 2021. ISSN 10902643. .

Emil M Constantinescu and Adrian Sandu. Extrapolated Multirate Methods for Differential Equations with Multiple Time Scales. *Journal of Scientific Computing*, pages 28–44, 2013.

J.M.A Danby. *Fundamentals of Celestial Mechanics*. Willmann-Bell Inc, 3rd edition, 1992. ISBN 0-943396-20-4.

M. B. Davies, F. C. Adams, P. Armitage, J. Chambers, E. Ford, A. Morbidelli, S. N. Raymond, and D. Veras. The Long-Term Dynamical Evolution of Planetary Systems. In *Protostars and Planets VI*, chapter 3. The University of Arizona Press, 2014.

Rebekah I. Dawson and John Asher Johnson. Origins of hot Jupiters. *Annual Review ofAstronomy and Astrophysics*, 56:175–221, 2018. ISSN 23318422.

Walter Dehnen. Towards time symmetric N-body integration. *Monthly Notices of the Royal Astronomical Society*, 472(1):1226–1238, 2017. ISSN 0035-8711. .

Anthony R. Dobrovolskis, José L. Alvarellos, and Jack J. Lissauer. Lifetimes of small bodies in planetocentric (or heliocentric) orbits. *Icarus*, 188(2):481–505, 2007. ISSN 00191035. .

J. R. Dormand and P. J. Prince. A reconsideration of some embedded Runge-Kutta formulae. *Journal of Computational and Applied Mathematics*, 15(2):203–211, 3 1986. ISSN 03770427. .

Martin J. Duncan, Harold F. Levison, and Man Hoi Lee. A Multiple Time Step Symplectic Algorithm for Integrating Close Encounters. *The Astronomical Journal*, 116 (4):2067–2077, 1998. ISSN 00046256. .

EPCC. Archer - Hardware, 2017. URL https://www.archer.ac.uk/training/course-material/2015/03/intro/slides/L15-ARCHER-Hardware.pdf.

Leandro Esteves, André Izidoro, Sean N. Raymond, and Bertram Bitsch. The origins of nearly coplanar, non-resonant systems of close-in super-Earths. *Monthly Notices of the Royal Astronomical Society*, 497(2):2493–2500, 2020. ISSN 13652966. .

Edgar Everhart. Implicit Single-Sequence Methods For Integrating Orbits. *Celestial Mechanics*, 10(1972):35–55, 1974. .

Edgar Everhart. An efficient integrator that uses Gauss-Radau spacings. *International Astronomical Union Colloquium*, 83:185–202, 1985. ISSN 0252-9211. .

Daniel C. Fabrycky, Jack J. Lissauer, Darin Ragozzine, Jason F. Rowe, Jason H. Steffen, Eric Agol, Thomas Barclay, Natalie Batalha, William Borucki, David R. Ciardi, Eric B. Ford, Thomas N. Gautier, John C. Geary, Matthew J. Holman, Jon M. Jenkins, Jie Li, Robert C. Morehead, Robert L. Morris, Avi Shporer, Jeffrey C. Smith, Martin Still, and Jeffrey Van Cleve. Architecture of Kepler's multi-transiting systems. II. New investigations with twice as many candidates. *Astrophysical Journal*, 790(2), 2014. ISSN 15384357. .

D. Farnocchia, S. R. Chesley, P. W. Chodas, M. Micheli, D. J. Tholen, A. Milani, G. T. Elliott, and F. Bernardi. Yarkovsky-driven impact risk analysis for asteroid (99942) Apophis. *Icarus*, 224(1):192–200, 5 2013. ISSN 00191035. .

Etienne Forest and Ronald D Ruth. Fourth-order symplectic integration. *Physica D: Nonlinear Phenomena*, 43(1):105–117, 1990. ISSN 01672789. .

Jon D. Giorgini, Lance A.M. Benner, Steven J. Ostro, Michael C. Nolan, and Michael W. Busch. Predicting the Earth encounters of (99942) Apophis. *Icarus*, 193(1):1–19, 2008. ISSN 00191035. .

Peter Goldreich. Disk Satellite Interactions. *The Astrophysical Journal*, 241:425–441, 1980.

R. Gomes, H. F. Levison, K. Tsiganis, and A. Morbidelli. Origin of the cataclysmic Late Heavy Bombardment period of the terrestrial planets. *Nature*, 435(7041):466–469, 2005. ISSN 0028-0836. .

K. R. Grazier, W. I. Newman, James M. Hyman, Philip W. Sharp, and David J. Goldstein. Achieving Brouwer's law with high-order Stormer multistep methods. *ANZIAM Journal*, 46:786, 2005. ISSN 1445-8810. .

Simon L. Grimm and Joachim G. Stadel. The genga code: Gravitational encounters in N-body simulations with GPU acceleration. *Astrophysical Journal*, 796(1), 2014. ISSN 15384357. .

Evgeni Grishin, Hagai B. Perets, and Yael Avni. Planet seeding through gas-assisted capture of interstellar objects. *Monthly Notices of the Royal Astronomical Society*, 487(3):3324–3332, 2019. ISSN 13652966. .

Sam Hadden and Yoram Lithwick. A Criterion for the Onset of Chaos in Systems of Two Eccentric Planets. *The Astronomical Journal*, 156(3):95, 2018. ISSN 0004-6256. . URL http://dx.doi.org/10.3847/1538-3881/aad32c.

E Hairer, S.P Norsett, and G Wanner. *Solving Ordinary Differential Equations I*. Springer, 3rd edition, 1987. ISBN 9783540566700.

Ernst Hairer and Gerhard Wanner. *Solving Ordinary Differential Equations II*, volume 14. Springer, 1991. ISBN 3-540-53775-9.

Ernst Hairer and Gerhard Wanner. Stiff differential equations solved by Radau methods. *Journal of Computational and Applied Mathematics*, 111:93–111, 1999.

Ernst Hairer, Christian Lubrich, and Gerhard Wanner. *Geometric Numerical Integration*, volume 267. Springer, 2002. ISBN 978-3-540-30666-5.

Chushiro Hayashi. Structure of the Solar Nebula, Growth and Decay of Magnetic Fields and Effects of Magnetic and Turbulent Viscosities on the Nebula. *Supplement of the Progress of Theoretical Physics*, 70, 1981. .

Douglas C Heggie. A global regularization of the gravitational N-body problem. *Celestial Mechanics*, 10(1972):217–241, 1974.

R. Helled, P. Bodenheimer, M. Podolak, A. Boley, F. Meru, S. Nayakshin, J. J. Fortney, L. Mayer, Y. Alibert, and A. P. Boss. Giant Planet Formation, Evolution, and Internal Structure. *Protostars and Planets VI*, 2014. ISSN 9780816531240. .

David M. Hernandez. Should N-body integrators be symplectic everywhere in phase space? *Monthly Notices of the Royal Astronomical Society*, 486(4):5231–5238, 2019. ISSN 13652966. .

David M Hernandez and Edmund Bertschinger. Time-symmetric integration in astrophysics. *Monthly Notices of the Royal Astronomical Society*, 475(4):5570–5584, 2018. ISSN 13652966. .

David M Hernandez and Walter Dehnen. A study of symplectic integrators for planetary system problems: Error analysis and comparisons. *Monthly Notices of the Royal Astronomical Society*, 468(3):2614–2636, 2017. ISSN 13652966. .

David M. Hernandez, Eric Agol, Matthew J. Holman, and Sam Hadden. Significant Improvement in Planetary System Simulations from Statistical Averaging. *Research Notes of the AAS*, 5(4):77, 2021. .

M Hernandez and Matthew J Holman. EnckeHH: an integrator for gravitational dynamics with a dominant mass that achieves optimal error behaviour. *Monthly Notices of the Royal Astronomical Society*, pages 5–12, 2020. .

Nicholas J. Higham. The accuracy of floating point summation. *SIAM journal of Scientific Computing*, 14(4):783–799, 1993. ISSN 00063835. .

Naireen Hussain and Daniel Tamayo. Fundamental limits from chaos on instability time predictions in compact planetary systems. *Monthly Notices of the Royal Astronomical Society*, 491(4):5258–5267, 2020. ISSN 0035-8711. .

Piet Hut, Jun Makino, and Steve McMillan. Building a Better Leapfrog. *The Astrophysical Journal*, 443, 1995. ISSN 00390895.

Shigeru Ida and Junichiro Makino. Scattering of Planetesimals by a Protoplanet: Slowing Down of Runaway Growth. *Icarus*, 106(1):210–227, 1993. ISSN 10902643. .

IEEE. *754-2019 - IEEE Standard for Floating-Point Arithmetic*. IEEE, 2019. ISBN 9781504459242.

Maria Jimenez and Frederic Masset. Improved torque formula for low and intermediate mass planetary migration. *Monthly Notices of the Royal Astronomical Society*, 471(4):4917–4929, 2017.

Mario Jurić and Scott Tremaine. Dynamical Origin of Extrasolar Planet Eccentricity Distribution. *The Astrophysical Journal*, 686(1):603–620, 2008. ISSN 0004-637X. .

W Kahan. Further remarks on reducing truncation errors. *Communications of the ACM*, 8(1):40–40, 1965. .

Immanuel Kant. *Universal natural history and theory of the heavens*. 1755.

Murat Kaplan and Hasan Saygin. Hermite Integrations with TSBTS Algorithm for N-Body Problems. In *1st WSEAS International Conference on Multivariate Analysis*, 2008. ISBN 9789606766657.

Hiroshi Kinoshita, Haruo Yoshida, and Hiroshi Nakai. Symplectic integrators and their application to dynamical astronomy. *Celestial Mechanics and Dynamical Astronomy*, 50(1):59–71, 1990. ISSN 09232958. .

E Kokubo. Formation of Protoplanets from Planetesimals in the Solar Nebula. *Icarus*, 143(1):15–27, 2000. ISSN 00191035. .

E Kokubo and S Ida. Formation of Protoplanet Systems and Diversity of Planetary Systems. *The Astrophysical Journal*, 581:666, 2002. ISSN 1538-4357. .

Eiichiro Kokubo and Shigeru Ida. Orbital Evolution of Protoplanets Embedded in a Swarm of Planetesimals. *Icarus*, 114(2):247–257, 1995. ISSN 00191035. .

Eiichiro Kokubo and Shigeru Ida. On Runaway Growth of Planetesimals. *Icarus*, 123: 180–191, 1996. ISSN 00191035. .

Eiichiro Kokubo and Shigeru Ida. Oligarchic Growth of Protoplanets. *Icarus*, 131(1): 171–178, 1998. ISSN 00191035. .

Eiichiro Kokubo and Junichiro Makino. A Modified Hermite Integrator for Planetary Dynamics. *PASJ: Publ. Astron. Soc. Japan*, 56:861–868, 2004. ISSN 0004-6264. .

Eiichiro Kokubo, Keiko Yoshinaga, and Junichiro Makino. On a time-symmetric Hermite integrator for planetary N-body simulation. *Monthly Notices of the Royal Astronomical Society*, 297(4):1067–1072, 1998. ISSN 0035-8711. .

Eiichiro Kokubo, Junko Kominami, and Shigeru Ida. Formation of Terrestrial Planets from Protoplanets. I. Statistics of Basic Dynamical Properties. *The Astrophysical Journal*, 642(2002):1131–1139, 2006. ISSN 0004-637X. .

Janusz Kowalik. MPI : The Complete Reference Scientic and Engineering Computation. 1996.

P Kustaanheimo and E Stiefel. Perturbation theory of Kepler motion based on spinor regularization. *Journal für die reine und angewandte Mathematik*, 1965(218), 1965.

M. Lambrechts and A. Johansen. Rapid growth of gas-giant cores by pebble accretion. *Astronomy & Astrophysics*, 544:A32, 2012. ISSN 0004-6361. .

Pierre Laplace. *Exposition du système du monde*. 1835.

J. Laskar and M. Gastineau. Existence of collisional trajectories of Mercury, Mars and Venus with the Earth. *Nature*, 459(7248):817–819, 2009. ISSN 00280836. .

Benedict Leimkuhler and Sabastian Reich. *Simulating Hamiltonian Dynamics*. Cambridge University Press, Cambridge, 1st edition, 2005. ISBN 9780521772907.

Zoë M. Leinhardt and Derek C. Richardson. Planetesimals To Protoplanets - I . Effect of Fragmentation on Terrestrial Planet Formation. *The Astrophysical Journal*, 625 (3):427–440, 2005. ISSN 0004-637X. .

T. Levi-Civita. Sur la regularisation du probleme des trois corps. *Acta Mathematica*, 42(1):99–144, 1920. ISSN 00015962. .

Harold F. Levison and Martin J. Duncan. The Long-Term Dynamical Behaviour of Short-Period Comets. *Icarus*, 108(1), 1994.

By Ivar Lie and Syvert P Norsett. Superconvergence for Multistep Collocation. *Mathematics of Computation*, 52(185):65–79, 1989.

Jack J. Lissauer. Planet formation. *Annual Review ofAstronomy and Astrophysics*, 31: 129–174, 1993. ISSN 00758450. .

Jack J Lissauer and Sacha Gavino. Orbital stability of compact three-planet systems, I: Dependence of system lifetimes on initial orbital separations and longitudes. *Icarus*, 364, 2021.

Jack J. Lissauer, Daniel C. Fabrycky, Eric B. Ford, William J. Borucki, Francois Fressin, Geoffrey W. Marcy, Jerome A. Orosz, Jason F. Rowe, Guillermo Torres, William F. Welsh, Natalie M. Batalha, Stephen T. Bryson, Lars A. Buchhave, Douglas A. Caldwell, Joshua A. Carter, David Charbonneau, Jessie L. Christiansen, William D. Cochran, Jean Michel Desert, Edward W. Dunham, Michael N. Fanelli, Jonathan J. Fortney, Thomas N. Gautier, John C. Geary, Ronald L. Gilliland, Michael R. Haas, Jennifer R. Hall, Matthew J. Holman, David G. Koch, David W. Latham, Eric Lopez, Sean McCauliff, Neil Miller, Robert C. Morehead, Elisa V. Quintana, Darin Ragozzine, Dimitar Sasselov, Donald R. Short, and Jason H. Steffen. A closely packed system of low-mass, low-density planets transiting Kepler-11. *Nature*, 470 (7332):53–58, 2011. ISSN 00280836. .

Jack J. Lissauer, Geoffrey W. Marcy, Stephen T. Bryson, Jason F. Rowe, Daniel Jontof-Hutter, Eric Agol, William J. Borucki, Joshua A. Carter, Eric B. Ford, Ronald L. Gilliland, Rea Kolbl, Kimberly M. Star, Jason H. Steffen, and Guillermo Torres. Validation of kepler's multiple planet candidates. II. refined statistical framework and descriptions of systems of special interest. *Astrophysical Journal*, 784(1), 2014. ISSN 15384357. .

Masahiro N. Machida, Eiichiro Kokubo, Shu ichiro Inutsuka, and Tomoaki Matsumoto. Gas accretion onto a protoplanet and formation of a gas giant planet. *Monthly Notices of the Royal Astronomical Society*, 405(2):1227–1243, 2010. ISSN 13652966. .

Junichiro Makino. Optimal Order and Time-Step Criterion for Aarseth-Type N-Body Integrators. *The Astrophysical Journal*, (1985):200–212, 1991.

Junichiro Makino and Sverre J. Aarseth. On a Hermite Integrator with Ahmad-Cohen Scheme for Gravitational Many-Body Problems. *Publications of the Astronomical Society of Japan*, 44(2):141–151, 1992. URL http://articles.adsabs.harvard.edu/cgi-bin/nph-iarticle_query?1992PASJ...44..141M&data_type=PDF_HIGH&whole_paper=YES&type=PRINTER&filetype=.pdf.

Junichiro Makino, Piet Hut, Murat Kaplan, and Hasan Saygin. A time-symmetric block time-step algorithm for N-body simulations. *New Astronomy*, 12(2):124–133, 2006. ISSN 13841076. .

Christian Marois, B Zuckerman, Quinn M Konopacky, Bruce Macintosh, and Travis Barman. Images of a fourth planet orbiting HR 8799. *Nature*, 468(7327):1080–1083, 2010. ISSN 00280836. .

Yuji Matsumoto and Eiichiro Kokubo. Formation of Close-in Super-Earths by Giant Impacts: Effects of Initial Eccentricities and Inclinations of Protoplanets. *The Astronomical Journal*, 154(1):27, 2017. ISSN 0004-6256. .

M. Mayor and D. Queloz. A Jupiter-mass companion to a solar-type star. *Nature*, 378 (7):603–605, 1995.

M. Mayor, M. Marmier, C. Lovis, S. Udry, D. Ségransan, F. Pepe, W. Benz, J. L. Bertaux, F. Bouchy, X. Dumusque, G. Lo Curto, C. Mordasini, D. Queloz, and N. C. Santos. The HARPS search for southern extra-solar planets XXXIV. Occurrence, mass distribution and orbital properties of super-Earths and Neptune-mass planets. *eprint arXiv:1109.2497*, (2007):1–25, 2011.

Michael R Meyer, Lynne A Hillenbrand, Dana Backman, Steve Beckwith, Jeroen Bouwman, Tim Brooke, John Carpenter, Martin Cohen, Stephanie Cortes, Uma Gorti, Thomas Henning, Dean Hines, David Hollenbach, Jinyoung Serena, Jonathan Lunine, Renu Malhotra, Eric Mamajek, Stanimir Metchev, Amaya Moro, Pat Morris, Joan Najita, Deborah Padgett, Ilaria Pascucci, Jens Rodmann, Murray Silverstone, Michael R Meyer, Lynne A Hillenbrand, Dana Backman, Steve Beckwith, Jeroen Bouwman, Tim Brooke, John Carpenter, Martin Cohen, Stephanie Cortes, Nathan Crockett, Uma Gorti, Thomas Henning, Dean Hines, David Hollenbach, Jinyoung Serena Kim, Jonathan Lunine, Renu Malhotra, Eric Mamajek, Stanimir Metchev, Amaya Moro-martin, Pat Morris, Joan Najita, Deborah Padgett, Ilaria Pascucci, Steve Strom, Dan Watson, Stuart Weidenschilling, Sebastian Wolf, and Erick Young. The Formation and Evolution of Planetary Systems : Placing Our Solar System in Context with Spitzer Published by : Astronomical Society of the Pacific Stable URL : http://www.jstor.org/stable/10.1086/510099 The Formation and Evolution of Planetary Systems. 2006.

Yohei Miki and Masayuki Umemura. GOTHIC: Gravitational oct-tree code accelerated by hierarchical time step controlling. *New Astronomy*, 52:65–81, 2017. ISSN 13841076. .

Seppo Mikkola. A practical and regular formulation of the N-body equations. *Monthly Notices of the Royal Astronomical Society*, 215(2):171–177, 1985. ISSN 0035-8711. .

Seppo Mikkola and Sverre J. Aarseth. A chain regularization method for the few-body problem. *Celestial Mechanics and Dynamical Astronomy*, 47(4):375–390, 1989. ISSN 09232958. .

Seppo Mikkola and Sverre J. Aarseth. An Implementation of N-Body Chain Regularisation. *Celestial Mechanics & Dynamical Astronomy*, 57:439–459, 1993.

Alexander Moore and Alice C. Quillen. QYMSYM: A GPU-accelerated hybrid symplectic integrator that permits close encounters. *New Astronomy*, 16(7):445–455, 2011. ISSN 13841076. .

A Morbidelli, J. Chambers, J. I. Lunine, J. M. Petit, F. Robert, G. B. Valsecchi, and K. E. Cyr. Source regions and timescales for the delivery of water to the Earth. *Meteoritics and Planetary Science*, 35(6):1309–1320, 2000. ISSN 10869379. .

Alessandro Morbidelli and Sean N. Raymond. Challenges In Planet Formation. *Journal of Geophysical Research: Planets*, 121:1962–1980, 2016. .

Smadar Naoz, Will M. Farr, Yoram Lithwick, Frederic A. Rasio, and Jean Teyssandier. Hot Jupiters from secular planet-planet interactions. *Nature*, 473(7346):187–189, 2011. ISSN 00280836. .

NASA. Kepler Mission Overview Page, 2018. URL https://www.nasa.gov/mission_pages/kepler/overview/index.html.

NASA. NASA Jet Propulsion Laboratory Solar System Dynamics Group, JPL HORIZONS System, 2021. URL URL:https://ssd.jpl.nasa.gov/?horizons.

Keigo Nitadori and Junichiro Makino. Sixth- and eighth-order Hermite integrator for N-body simulations. *New Astronomy*, 13(7):498–507, 2008. ISSN 13841076. .

NVIDIA. *NVIDIA: GPU Gems 3*. 2008.

Alysa Obertas, Christa Van Laerhoven, and Daniel Tamayo. The stability of tightly-packed, evenly-spaced systems of Earth-mass planets orbiting a Sun-like star. *Icarus*, 293:52–58, 2017. ISSN 10902643. .

David P. O'Brien, Alessandro Morbidelli, and Harold F. Levison. Terrestrial planet formation with strong dynamical friction. *Icarus*, 184(1):39–58, 2006. ISSN 00191035. .

Keiji Ohtsuki, Glen R Stewart, and Shigeru Ida. Evolution of planetesimal velocities based on three-body orbital integrations and growth of protoplanets. *Icarus*, 155 (2):436–453, 2002. ISSN 00191035. .

David Patterson. *In Praise of Programming Massively Parallel Processors : A Hands-on Approach*. 2010. ISBN 9780123814722.

Erik A Petigura, Andrew W Howard, and Geoffrey W Marcy. Correction for Li et al., Mechanism of E-cadherin dimerization probed by NMR relaxation dispersion. *Proceedings of the National Academy of Sciences*, 110(48):19651–19651, 11 2013. ISSN 0027-8424. .

Antoine C. Petit, Gabriele Pichierri, Melvyn B. Davies, and Anders Johansen. The path to instability in compact multi-planetary systems. *Astronomy & Astrophysics*, 641: A176, 9 2020. ISSN 0004-6361. .

William Press, Saul Teukolsky, William Vetterling, and Brian Flannery. *Numerical Recipes: The Art of Scientific Computing*. Cambridge University Press, 3rd edition, 2007. ISBN 9780521880688.

Bonan Pu and Yanqin Wu. Spacing of Kepler Planets: Sculpting by Dynamical Instability. *Astrophysical Journal*, 807(1):44, 6 2015. ISSN 15384357. . URL https://iopscience.iop.org/article/10.1088/0004-637X/807/1/44.

Alice C. Quillen. Three-body resonance overlap in closely spaced multiple-planet systems. *Monthly Notices of the Royal Astronomical Society*, 418(2):1043–1054, 2011. ISSN 13652966. .

R. Radau. Étude Sur Les Formules D'Approximation Qui Servent À Calculer La Valeur Numérique D'Une Intégrale Définie. *Journal de mathématiques pures et appliquées*, 6(3):283–336, 1880.

Sean N. Raymond and Christophe Cossou. No universal minimum-mass extrasolar nebula: Evidence against in situ accretion of systems of hot super-Earths. *Monthly Notices of the Royal Astronomical Society: Letters*, 440(1):11–15, 2014. ISSN 17453933. .

Sean N Raymond, Thomas Quinn, and Jonathan I Lunine. High-resolution simulations of the final assembly of Earth-like planets I. Terrestrial accretion and dynamics. *Icarus*, 183(2):265–282, 2006. ISSN 00191035. .

Sean N. Raymond, David P. O'Brien, Alessandro Morbidelli, and Nathan A. Kaib. Building the terrestrial planets: Constrained accretion in the inner Solar System. *Icarus*, 203(2):644–662, 2009. ISSN 00191035. . URL http://dx.doi.org/10.1016/j.icarus.2009.05.016.

Hanno Rein. Embedded operator splitting methods for perturbed systems. *Monthly Notices of the Royal Astronomical Society*, 492(4):5413–5419, 2020. ISSN 13652966. .

Hanno Rein and S.F. Liu. REBOUND: An open-source multi-purpose N-body code for collisional dynamics. *Astronomy and Astrophysics*, 537:1–10, 2012. .

Hanno Rein and David S Spiegel. IAS 15 : a fast , adaptive , high-order integrator for gravitational dynamics accurate to machine precision over a billion orbits. *Monthly Notices of the Royal Astronomical Society*, 1437:1424–1437, 2015. .

Hanno Rein and Daniel Tamayo. WHFAST: A fast and unbiased implementation of a symplectic Wisdom-Holman integrator for long-term gravitational simulations. *Monthly Notices of the Royal Astronomical Society*, 452(1):376–388, 2015. ISSN 13652966. .

Hanno Rein and Daniel Tamayo. JANUS: A bit-wise reversible integrator for N-body dynamics. *Monthly Notices of the Royal Astronomical Society*, 473(3):3351–3357, 2018. ISSN 13652966. .

Hanno Rein, Garett Brown, and Daniel Tamayo. On the accuracy of symplectic integrators for secularly evolving planetary systems. *Monthly Notices of the Royal Astronomical Society*, 490(4):5122–5133, 2019a. ISSN 13652966. .

Hanno Rein, David M. Hernandez, Daniel Tamayo, Garett Brown, Emily Eckels, Emma Holmes, Michelle Lau, Réjean Leblanc, and Ari Silburt. Hybrid symplectic integrators for planetary dynamics. *Monthly Notices of the Royal Astronomical Society*, 485 (4):5490–5497, 2019b. ISSN 13652966. .

David R. Rice, Frederic A. Rasio, and Jason H. Steffen. Survival of non-coplanar, closely packed planetary systems after a close encounter. *Monthly Notices of the Royal Astronomical Society*, 481(2):2205–2212, 2018. ISSN 13652966. .

Jason F. Rowe, Stephen T. Bryson, Geoffrey W. Marcy, Jack J. Lissauer, Daniel Jontof-Hutter, Fergal Mullally, Ronald L. Gilliland, Howard Issacson, Eric Ford, Steve B. Howell, William J. Borucki, Michael Haas, Daniel Huber, Jason H. Steffen, Susan E. Thompson, Elisa Quintana, Thomas Barclay, Martin Still, Jonathan Fortney, T. N. Gautier, Roger Hunter, Douglas A. Caldwell, David R. Ciardi, Edna Devore, William Cochran, Jon Jenkins, Eric Agol, Joshua A. Carter, and John Geary. Validation of Keplers Multiple Planet Candidates. III. Light Curve Analysis and Announcement of Hundreds of New Multi-Planet Systems. *The Astrophysical Journal*, 784(1):45, 3 2014. ISSN 0004-637X. .

A E Roy, I W Walker, J Macdonald, I P Williams, K Fox, C D Murray, A Milani, A M Nobili, P J Message, A T Sinclair, and M Carpino. Project Longstop. *Vistas in Astronomy*, 32:95–116, 1988. .

Victor Safronov. Evolution of Protoplanetary Cloud and Formation of the Earth and Planets. *Israel Program for Scientific Translations*, 1972.

Viktor Safronov. Evolution of Protoplanetary Cloud and Formation of the Earth and Planets. *NASA Tech. Trans.*, 1969.

Prasenjit Saha and Scott Tremaine. Symplectic Integrators for Solar System Dynamics. *The Astronomical Journal*, 104(4):1633–1640, 1992. .

Matthew Scarpino. *OpenCL in Action*. 2012. ISBN 9781617290176.

Stefan Schneider. Numerical Experiments with a Multistep Radau Method. *BIT Numerical Mathematics*, 33(May 1992):332–350, 1993.

Stefan Schneider. Efficient Implementation of Multistep Collocation Methods. 1994.

P W Sharp and W I Newman. GPU-enabled N -body simulations of the Solar System using a VOVS Adams integrator. *Journal of Computational Science*, 16:89–97, 2016. ISSN 1877-7503. .

Philip W Sharp. N-Body Simulations : The Performance Of Some Integrators. *ACM Transactions on Mathematical Software*, 32(3):375–395, 2006.

Andrew W. Smith and Jack J. Lissauer. Orbital stability of systems of closely-spaced planets. *Icarus*, 201(1):381–394, 2009. ISSN 00191035. .

Stanly Steinberg. *Lie series, Lie transformations, and their applications*. 2008. .

T Stuchi. Symplectic Integrators Revisited. *Brazillian Journal of Physics*, 32, 2002.

Kate Y.L. Su, Alan P. Jackson, András Gáspár, George H. Rieke, Ruobing Dong, Johan Olofsson, G. M. Kennedy, Zoë M. Leinhardt, Renu Malhotra, Michael Hammer, Huan Y.A. Meng, W. Rujopakarn, Joseph E. Rodriguez, Joshua Pepper, D. E. Reichart, David James, and Keivan G. Stassun. Extreme debris disk variability – Exploring the diverse outcomes of large asteroid impacts during the era of terrestrial planet formation. *The Astronomical Journal*, 157(5), 2019. ISSN 23318422. .

Gerald Jay Sussman and Jack Wisdom. Numerical evidence that the motion of Pluto is chaotic. *Science*, 241(4864):433–437, 1988. ISSN 00368075. .

J. Szulágyi, F. Masset, E. Lega, A. Crida, A. Morbidelli, and T. Guillot. Circumplanetary disc or circumplanetary envelope? *Monthly Notices of the Royal Astronomical Society*, 460(3):2853–2861, 2016. ISSN 13652966. .

Daniel Tamayo, Ari Silburt, Diana Valencia, Kristen Menou, Mohamad Ali-Dib, Cristobal Petrovich, Chelsea X. Huang, Hanno Rein, Christa van Laerhoven, Adiv Paradise, Alysa Obertas, and Norman Murray. a Machine Learns To Predict the Stability of Tightly Packed Planetary Systems. *The Astrophysical Journal*, 832(2):L22, 2016. ISSN 2041-8213. . URL http://dx.doi.org/10.3847/2041-8205/832/2/L22.

Daniel Tamayo, Miles Cranmer, Samuel Hadden, Hanno Rein, Peter Battaglia, Alysa Obertas, Philip J. Armitage, Shirley Ho, David N. Spergel, Christian Gilbertson, Naireen Hussain, Ari Silburt, Daniel Jontof-Hutter, and Kristen Menou. Predicting the long-term stability of compact multiplanet systems. *Proceedings of the National Academy of Sciences*, 117(31):18194–18205, 8 2020a. ISSN 0027-8424. .

Daniel Tamayo, Hanno Rein, Pengshuai Shi, and David M. Hernandez. REBOUNDx: A library for adding conservative and dissipative forces to otherwise symplectic N-body integrations. *Monthly Notices of the Royal Astronomical Society*, 491(2): 2885–2901, 2020b. ISSN 13652966. .

Hidekazu Tanaka and William R. Ward. Three-dimensional Interaction between a Planet and an Isothermal Gaseous Disk. II. Eccentricity Waves and Bending Waves. *The Astrophysical Journal*, 602(1):388–395, 2004. ISSN 0004-637X. .

K. Tsiganis, R. Gomes, A. Morbidelli, and H. F. Levison. Origin of the orbital architecture of the giant planets of the Solar System. *Nature*, 435(7041):459–461, 2005. ISSN 0028-0836. .

Kathryn Volk and Brett Gladman. Consolidating and Crushing Exoplanets: Did It Happen Here? *Astrophysical Journal Letters*, 806(2):L26, 2015. ISSN 20418213. .

Kevin J. Walsh, Alessando Morbidelli, Sean N. Raymond, David P. O'Brien, and Avi M. Mandell. A low mass for Mars from Jupiter's early gas-driven migration. *Nature*, 475(7355):206–209, 2012. ISSN 0028-0836. .

Ji Wang, Debra A. Fischer, Elliott P. Horch, and Xu Huang. On the occurrence rate of hot Jupiters in different stellar environments. *Astrophysical Journal*, 799(2), 2015. ISSN 15384357. .

William R. Ward. Protoplanet migration by Nebula Tides. *Icarus*, 126(2):261–281, 1997. ISSN 00191035. .

Lewis Watt, Zoe Leinhardt, and Kate Y L Su. Planetary embryo collisions and the wiggly nature of extreme debris discs. *Monthly Notices of the Royal Astronomical Society*, 502(2):2984–3002, 2021. ISSN 0035-8711. .

S J Weidenschilling. THE DISTRIBUTION OF MASS IN THE PLANETARY SYSTEM AND SOLAR NEBULA. *Astrophysics and Space Science*, 51:153–158, 1977.

William E. Wiesel. *Modern Astrodynamics*. Aphelion Press, 2nd edition, 2010. ISBN 978-145378-1470.

Jack Wisdom. The resonance overlap criterion and the onset of stochastic behavior in the restricted three-body problem. *The Astronomical Journal*, 85(8):1122, 8 1980. ISSN 00046256. .

Jack Wisdom. The Origin of the Kirkwood Gaps: a Mapping for Asteroidal Motion Near the 3/1 Commensurability. *The Astronomical Journal*, 87(3):577–593, 1982.

Jack Wisdom and David M. Hernandez. A fast and accurate universal Kepler solver without Stumpff series. *Monthly Notices of the Royal Astronomical Society*, 453(3): 3015–3023, 2015. ISSN 13652966. .

Jack Wisdom and Matthew Holman. Symplectic maps for the n-body problem. *The Astronomical Journal*, 102(4), 1991. ISSN 00046256. .

Alex Wolszczan and D.A Frail. A planetary system around the millisecond pulsar PSR1257+12. *Nature*, 355:242–244, 1992. ISSN 0028-0836. .

Dong Hong Wu, Rachel C. Zhang, Ji Lin Zhou, and Jason H. Steffen. Dynamical instability and its implications for planetary system architecture. *Monthly Notices of the Royal Astronomical Society*, 484(2):1538–1548, 2019. ISSN 13652966. .

Ji Wei Xie, Subo Dong, Zhaohuan Zhu, Daniel Huber, Zheng Zheng, Peter De Cat, Jianning Fu, Hui Gen Liu, Ali Luo, Yue Wu, Haotong Zhang, Hui Zhang, Ji Lin Zhou, Zihuang Cao, Yonghui Hou, Yuefei Wang, and Yong Zhang. Exoplanet orbital eccentricities derived from LAMOST-kepler analysis. *Proceedings of the National Academy of Sciences of the United States of America*, 113(41):11431–11435, 2016. ISSN 10916490. .

Haruo Yoshida. Symplectic Integrators for Hamiltonian Systems: Basic Theory. *Chaos, Resonance, and Collective Dynamical Phenomena in the Solar System: Proceedings of the 152nd Symposium of the International Astronomical Union*, 1991.