Monthly Notices of the ROYAL ASTRONOMICAL SOCIETY

MNRAS **508,** 4858–4868 (2021) https://doi.org/10.1093/mnras/stab2874
Advance Access publication 2021 October 6

ISYMBA: a symplectic massive bodies integrator with planets interpolation

Fernando Roig ,[1]★ David Nesvorný,[2] Rogerio Deienno[2] and Matias J. Garcia[1]

[1]*Observatório Nacional, Rua Gal. Jose Cristino 77, Rio de Janeiro, RJ 20921-400, Brazil*
[2]*Department of Space Studies, Southwest Research Institute, 1050 Walnut St., Suite 300, Boulder, CO 80302, USA*

Accepted 2021 September 29. Received 2021 September 28; in original form 2021 August 3

ABSTRACT

A planetary instability occurring at time <100 My after formation of the giant planets (GPs) in our Solar system can be responsible for some characteristics of the inner Solar system. However, the actual influence of the instability on the terrestrial planet formation is not well understood. The simulations of terrestrial planet formation are very CPU-expensive, and this limits the exploration of different instability scenarios. To include the effects of the GPs instability in the simulations of terrestrial planets formation in a feasible way, we approach the problem in two steps. First, we model and record an evolution of the GPs that replicates the present outer Solar system in the end. Then, we use that orbital record, properly interpolated, as the input for a second step to simulate its effects on the terrestrial planet formation. For this second step, we developed ISYMBA, a symplectic massive bodies algorithm, where 'i' stands for interpolation. ISYMBA is a very useful code to accurately evaluate the effects of planetary instabilities on minor body reservoirs, whilst accounting for close encounters among massive objects. We provide a detailed description of how ISYMBA was developed and implemented to study terrestrial planet formation. Adapting ISYMBA for other problems that demand interpolation from previous simulations can be done following the method described here.

Key words: methods: numerical – software: simulations – planets and satellites: formation.

1 INTRODUCTION

Current theories of the early evolution of the Solar system invoke a temporary instability of the giant planets (GPs), which would have happened sometime after the dissipation of the gas in the protoplanetary nebula (see Nesvorný 2018, for a review). This instability led to mutual scattering of the GPs while they were radially migrating due to the interaction with an outer massive disc of remnant planetesimals (PLs). The instability is required to explain many dynamical features that are currently observed in the different populations of Solar system bodies. These include the angular momentum deficit of the GPs (e.g. Nesvorný & Morbidelli 2012; Deienno et al. 2017), the inclinations of asteroids in the main belt (e.g. Roig & Nesvorný 2015; Deienno et al. 2018), the existence of Jupiter trojans (e.g. Nesvorný, Vokrouhlický & Morbidelli 2013), the orbital architecture of the Kuiper belt (e.g. Nesvorný 2015; Gomes et al. 2018), the excited orbit of Mercury (e.g. Roig, Nesvorný & DeSouza 2016), the dynamical characteristics of the satellites of the jovian planets (e.g. Deienno et al. 2014; Nesvorný, Vokrouhlický & Deienno 2014b; Nesvorný et al. 2014a), among others.

The first instability models (Tsiganis et al. 2005) assumed that the instability may have happened around 600 My after the dissipation of the gas in the protoplanetary nebula, helping to trigger the Lunar Late Heavy Bombardment (Gomes et al. 2005; Bottke et al. 2012). Recent models, however, propose that the instability may have happened as early as ∼10 My after the dissipation of the gas (Nesvorný 2018). This means that the instability might have played a relevant role in the accretion of the terrestrial planets (e.g. Clement et al. 2018, 2019; Nesvorný, Roig & Deienno 2021).

Using fully self consistent models to study the effects of the instability in the early evolution of the Solar system may be an unfeasible task. In general, such models need to consider at least three ingredients: the GPs, the disc of massive PLs that drives the migration of the GPs, and the population of bodies affected by the instability. We refer to the latter as the *target population*, and it may be represented either by test particles or by mutually interacting massive bodies. This requires to explore a large number of model parameters and over a wide range of values, implying the need for hundreds or thousands of computationally expensive numerical simulations that, in most cases, lead to meaningless results.

An alternative to overcome these limitations is to use simplified, yet realistic, models where the amount of free parameters is smaller, and the user have more control over the range of possible evolutions of the system. This involves, for example, to drop off the disc of PLs and consider a prescribed evolution of the GPs. This prescribed evolution might be as simple as an *ad hoc* evolution, mimicked by using artificial forces, or as complex as a realistic evolution obtained from other previous simulations (e.g. Nesvorný, Roig & Bottke 2017; Deienno et al. 2018).

Here, we develop a numerical integration method that exploits the latter approach. In this method, the evolution of the GPs is stored in a file at regular intervals, and their positions and velocities for a given time are obtained by interpolation. Our aim is to construct a symplectic *N*-body integrator for the target population, which has the GPs interpolation scheme embedded in it.

Symplectic integrators are a particular class of numerical integrators, specifically designed to solve Hamiltonian problems (Yoshida

★ E-mail: froig@on.br

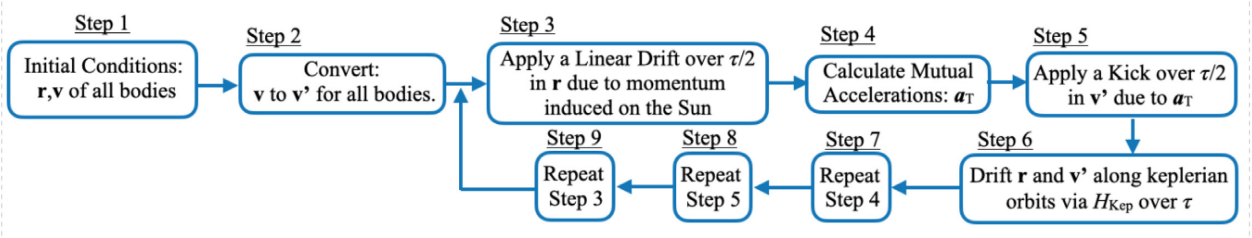

Figure 1. The detailed sequence of operations needed for a single integration time step, from t to $t + \tau$ in the SYMBA code. Primed variables are referred to the centre of mass of the whole system. Heliocentric variables are not primed.

1993). Their main property is the conservation of a quantity $\overline{H}$, referred to as the surrogate Hamiltonian, which is very close to the original Hamiltonian H of the problem. Our choice for a symplectic algorithm has two motivations: (i) these algorithms have proven to be fast and reliable, allowing for long-term simulations of planetary N-body systems with less computational cost than traditional integration algorithms (e.g. Wisdom & Holman 1991), and (ii) we intend to use the set of subroutines and packages of the well-tested and publicly available symplectic integrators SWIFT (Levison & Duncan 1993) and SYMBA (Duncan, Levison & Lee 1998), with the necessary modifications, as the basis for our algorithm.

When the target population consists of mass-less particles, the construction of a symplectic integrator with embedded interpolation is relatively straightforward (e.g. Beaugé, Roig & Nesvorný 2002; Roig & Nesvorný 2015). On the other hand, when the target populations consist of massive bodies that interact with each other and may develop close encounters, including the GPs interpolation in a symplectic way requires some specific considerations that turn the development of the algorithm more difficult. One particular problem is related to the fact that the system of GPs has a centre of mass that differs from that of the target population. This requires a specific splitting of the Hamiltonian that is not accounted for in full N-body symplectic integrators.

Here, we focus on the case when the target population is in the inner Solar system, and we describe the construction of the algorithm. For its recent applications, we refer the reader to Nesvorný et al. (2021) and DeSouza, Roig & Nesvorný (2021). The paper is organized as follows. In Section 2, we revise the basic concepts, provide the detailed description of the new algorithm, and perform some validation tests. In Section 3, we briefly discuss the possible parallelization strategies for the new code. The last section is devoted to the conclusions.

2 SYMPLECTIC INTEGRATION WITH INTERPOLATION

2.1 SYMBA

Before proceeding with the description of ISYMBA, we will briefly review the basics of SYMBA (see Duncan et al. 1998, for more details). SYMBA stands for Symplectic Massive Bodies Algorithm, and it is a second-order symplectic integrator for the planetary N-body problem, which allows for close encounters between massive bodies.

Let us assume a set of N bodies of masses m_i orbiting around the Sun (or any central star) of mass $M \gg m_i$, and introduce the Poincaré canonical coordinates $\mathbf{r}_i, \mathbf{p}'_i$ such that $\mathbf{r}_i$ are heliocentric positions and $\mathbf{p}'_i = m_i \mathbf{v}'_i$ are barycentric momenta (barycentric velocities). Throughout this work, primed variables are referred to the centre of mass of the system, and not primed variables are referred to the Sun. The Hamiltonian of the system is then composed of three parts:

$$H\left(\mathbf{r}, \mathbf{p}'\right) = H_{\rm Kep} + H_{\rm Int} + H_{\rm Sun} \tag{1}$$

where

$$H_{\rm Kep}(\mathbf{r}, \mathbf{p}') = \sum_{i=1}^{N} \left(\frac{\left|\mathbf{p}'_i\right|^2}{2m_i} - \frac{GMm_i}{|\mathbf{r}_i|} \right) \tag{2}$$

represents the two-body motion of the bodies around the Sun,

$$H_{\rm Int}(\mathbf{r}) = -\sum_{i=1}^{N-1} \sum_{k=i+1}^{N} \frac{Gm_i m_k}{|\mathbf{r}_i - \mathbf{r}_k|} \tag{3}$$

is the gravitational interaction potential between the bodies, and

$$H_{\rm Sun}(\mathbf{p}') = \frac{1}{2M} \left| \sum_{i=1}^{N} \mathbf{p}'_i \right|^2 \tag{4}$$

is the barycentric kinetic energy of the Sun.

Whenever $H_{\rm Int}, H_{\rm Sun} \ll H_{\rm Kep}$, a second-order symplectic integration over a time step τ is obtained through the following sequence of steps:

Step 1. Evolve the system only through $H_{\rm Sun}$, over $\tau/2$; this applies a linear drift (LD) to the positions $\mathbf{r}$.

Step 2. Evolve the system only through $H_{\rm Int}$, over $\tau/2$; this applies an impulse or kick (K) to the momenta $\mathbf{p}'$.

Step 3. Evolve the system only through $H_{\rm Kep}$, over τ; this applies a drift (D) to each body along a Keplerian orbit.

Step 4. Evolve the system only through $H_{\rm Int}$, over $\tau/2$; this applies again a kick (K) to the momenta $\mathbf{p}'$.

Step 5. Evolve the system only through $H_{\rm Sun}$, over $\tau/2$; this applies again a LD to the positions $\mathbf{r}$.

The integration over a time step τ is then schematized by a sequence LD - K - D - K - LD. The detailed flowchart of the algorithm is shown in Fig. 1.

However, when a close encounter between two bodies arises, the condition $H_{\rm Int} \ll H_{\rm Kep}$ is no longer satisfied and the above sequence is no longer valid. In a traditional numerical integrator, the increase of the $H_{\rm Int}$ term during the close encounter is usually compensated by a decrease of the time step τ, such as to keep the impulse $-\tau\, \partial H_{\rm Int}/\partial \mathbf{r}$ limited. However, in a symplectic integrator, the time step must be kept fixed over the whole integration. This is a well known limitation of symplectic algorithms, since the surrogate Hamiltonian $\overline{H}$, which is preserved by the algorithm, is such that $\overline{H} = H + \mathcal{O}(\tau^n)$, being n the order of the integrator. Thus, any change in the time step would break the conservation of $\overline{H}$.

The solution proposed by Duncan et al. (1998) and Chambers (1999) to circumvent this limitation consists into split the term $H_{\rm Int}$ into pieces, weighted by a function that depends on the distance

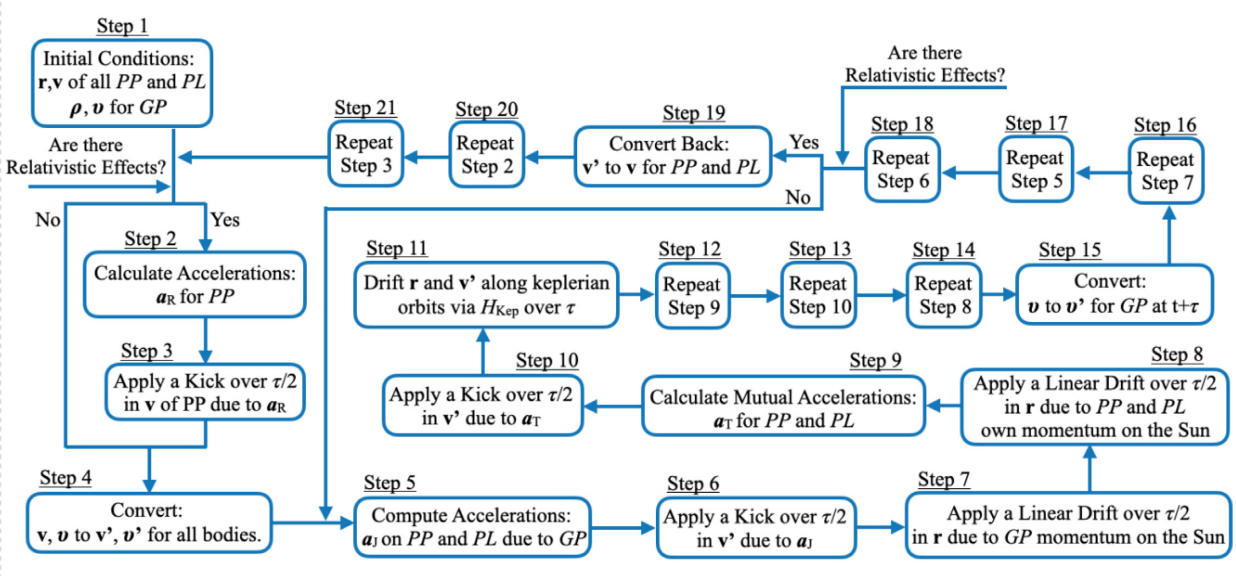

Figure 2. The detailed sequence of operations needed for a single time-step integration, from t to $t + \tau$ in the ISYMBA code. Primed variables are referred to the centre of mass of the system composed by the GPs, the terrestrial PPs, and PLs. Heliocentric variables are not primed.

between the different pairs of bodies. Consider, for example, a two-terms splitting, $H_{\rm Int} = H^{(0)}_{\rm Int} + H^{(1)}_{\rm Int}$, where:

$$\begin{aligned} H^{(0)}_{\rm Int} &= -\sum_{i=1}^{N-1}\sum_{k=i+1}^{N}\frac{Gm_im_k}{r_{ik}}\left(1-w_{1,ik}\right)\\ H^{(1)}_{\rm Int} &= -\sum_{i=1}^{N-1}\sum_{k=i+1}^{N}\frac{Gm_im_k}{r_{ik}}w_{1,ik} \end{aligned} \tag{5}$$

Here, $r_{ik} = |\mathbf{r}_i - \mathbf{r}_k|$, and $w_{l,ik}$ is a sigmoid-like weight function:

$$w_{l,ik} = \begin{cases} 1 & , \quad r_{ik} \geq R_{l,ik} \\ \phi\left(\frac{r_{ik}-R_{l+1,ik}}{R_{l,ik}-R_{l+1,ik}}\right) & , \quad R_{l+1,ik} \leq r_{ik} < R_{l,ik} \\ 0 & , \quad r_{ik} < R_{l+1,ik} \end{cases} \tag{6}$$

where ϕ is a suitable odd degree polynomial, and $R_{l,ik}$, $R_{l+1,ik}$ are defined in terms of the mutual Hill radii. Then, the sequence of steps over a time step τ becomes:

Step 1. Evolve the system only through $H_{\rm Sun}$, over $\tau/2$
Step 2. Evolve the system only through $H^{(1)}_{\rm Int}$, over $\tau/2$
Step 3. Repeat from 1 to q ($q > 1$, integer)
  Step 3.1. Evolve the system only through $H^{(0)}_{\rm Int}$, over $\tau/2q$
  Step 3.2. Evolve the system only through $H_{\rm Kep}$, over τ/q
  Step 3.3. Evolve the system only through $H^{(0)}_{\rm Int}$, over $\tau/2q$
Step 4. Evolve the system only through $H^{(1)}_{\rm Int}$, over $\tau/2$
Step 5. Evolve the system only through $H_{\rm Sun}$, over $\tau/2$

When $r_{ik} \geq R_{l,ik}$ (no close encounters), $H^{(0)}_{\rm Int} = 0$ and step 3 reduces to a single evolution of $H_{\rm Kep}$ over τ. On the other hand, when $r_{ik} < R_{l+1,ik}$ (close encounter), $H^{(1)}_{\rm Int} = 0$, and step 3 effectively performs the symplectic integration using a smaller time step τ/q. The above sequence can be represented as:

$$\underbrace{\text{LD - K}^{(1)}}_{\text{step }\tau} \text{ - } \left(\underbrace{\text{K}^{(0)}\text{-D-K}^{(0)}}_{\text{step }\tau/q}\right)^{q} \text{ - } \underbrace{\text{K}^{(1)}\text{ - LD}}_{\text{step }\tau}$$

This strategy can be recursively extended to an arbitrary sequence of weight functions for radii $R_{1,ik} > R_{2,ik} > \ldots > R_{n,ik} > R_{n+1,ik}$, associating smaller and smaller time steps to each weight function. In SYMBA, $R_{n+1,ik}$ is usually set to be of the order of the planetary radii, and $R_{1,ik}$ is of the order of a few Hill radii.

2.2 ISYMBA

In order to apply the above concepts to develop ISYMBA, we consider three different categories of bodies:

(i) GPs: represented by J bodies of masses μ_j, $j = 1, \ldots, J$. The positions $\boldsymbol{\rho}_j$ and velocities $\boldsymbol{\upsilon}_j$ of these bodies are directly read from a file, where they are stored at regular time intervals, and are interpolated down to the desired time step.[1] The GPs perturb the other two bodies categocither amongst them, nor from other bodies.

(ii) Terrestrial protoplanets (PPs): represented by T bodies of masses $m_i \geq m_{\rm tiny}$, $i = 1, \ldots, T$. The positions $\mathbf{r}_i$ and velocities $\mathbf{v}_i$ of these bodies are advanced through a second-order symplectic integrator. They feel their mutual gravitational perturbations, as well as those from the GPs and from the terrestrial PLs. Terrestrial PPs may also feel relativistic perturbations, if required.

(iii) Terrestrial PLs: represented by $N - T$ bodies of masses $m_i < m_{\rm tiny}$, $i = T + 1, \ldots, N$. The positions $\mathbf{r}_i$ and velocities $\mathbf{v}_i$ of these bodies are also advanced through a second-order symplectic integrator. They do not feel their mutual gravitational perturbations, but they are perturbed by both the GPs and the terrestrial PPs. They also perturb the terrestrial PPs.

As in the original SYMBA code, $m_{\rm tiny}$ is a constant mass threshold, that we set, for example, to 2 $M_{\rm Moon}$ in Nesvorný et al. (2021). We refer to the set of terrestrial PPs and PLs simply as the terrestrial bodies.

In the following, we describe in detail the different parts of the code. The flowchart of the algorithm is presented in Fig. 2.

[1]The storing cadence needs to be dense enough for the interpolation to work properly. A cadence of 1 yr proved to be good for interpolation to time steps of a few days. In principle, interpolation could be avoided by using a cadence equal to the time step, but this unnecessarily increases the file size and also makes the algorithm too slow due to the amount of I/O operations.

2.2.1 *GPs interpolation*

The positions and velocities of the GPs are obtained from a previous, and independent, simulation of the migration of these planets by interaction with a transneptunian disc of PLs (e.g. Nesvorný & Morbidelli 2012; Deienno et al. 2018). The output of this simulation, i.e. heliocentric positions and velocities (or heliocentric orbital elements) of the GPs, as well as their masses and the Sun mass, are stored in a file at 1 yr intervals. The masses are not necessarily constant during such evolution, since the planets and the Sun may accrete PLs while migrating and grow in mass.

Let $\boldsymbol{\rho}_b$, $\boldsymbol{\upsilon}_b$, μ_b be the position, velocity, and mass of a given GP at time t_b, and $\boldsymbol{\rho}_e$, $\boldsymbol{\upsilon}_e$, μ_e the corresponding values at a posterior time t_e. Let also M_b and M_e the corresponding masses of the Sun. Assume that we want to interpolate this trajectory over a time step $\tau = (t_e - t_b)/n$. The interpolation method consists of the following steps

Step 1. Advance $\boldsymbol{\rho}_b$, $\boldsymbol{\upsilon}_b$ from t_b to t_e along a Keplerian orbit, considering the central mass $M_b + \mu_b$, to obtain the sequence of values $\boldsymbol{\rho}_b^{(i)}$, $\boldsymbol{\upsilon}_b^{(i)}$, for times $t_i = t_b + i\tau$, $i = 0, \ldots, n$.

Step 2. Recede $\boldsymbol{\rho}_e$, $\boldsymbol{\upsilon}_e$ from t_e to t_b along a Keplerian orbit, considering the central mass $M_e + \mu_e$, to obtain the sequence of values $\boldsymbol{\rho}_e^{(j)}$, $\boldsymbol{\upsilon}_e^{(j)}$, for times $t_j = t_e - j\tau$, $j = 0, \ldots, n$.

Step 3. Compute the interpolated values at time $t_b \leq t_k \leq t_e$ through a weighted average:

$$\begin{aligned}
\boldsymbol{\rho}_k &= (1 - w_k)\boldsymbol{\rho}_b^{(k)} + w_k \boldsymbol{\rho}_e^{(n-k)} \\
\boldsymbol{\upsilon}_k &= (1 - w_k)\boldsymbol{\upsilon}_b^{(k)} + w_k \boldsymbol{\upsilon}_e^{(n-k)} \\
\boldsymbol{\mu}_k &= (1 - w_k)\boldsymbol{\mu}_b + w_k \boldsymbol{\mu}_e \\
M_k &= (1 - w_k)M_b + w_k M_e
\end{aligned} \tag{7}$$

where $w_k = k/n$, $k = 0, \ldots, n$.

We recall that these interpolated coordinates are heliocentric. A similar interpolation strategy was already implemented in the SWIFT_RMVS3 integrator to deal with encounters of test particles to massive bodies (Levison & Duncan 1993), and the corresponding subroutine has been adapted with the necessary modifications to account for mass changes.

The interpolation of a GP that disappears from the system, either by escaping or by merging to another giant, can be performed only until the last registry of that planet stored in the file. In such case, since the orbits are stored at 1 yr intervals, we loose only a few months of evolution of the giant that disappears. This is not expected to have any significant influence on the evolution of the target population over million years. Moreover, in the case of a merging between two GPs, one giant disappears but another has its mass increased. This latter giant will be properly interpolated since the interpolation scheme takes into account any variation of the planet's mass.

It is worth noting that the symplectic algorithm described in the next sections is independent of the particular interpolation scheme. Other interpolation routines, different than the one presented here, can be applied with the same result.

2.2.2 *Terrestrial bodies integration*

The key contribution to the development of the algorithm consists in the second-order symplectic integrator to advance the orbits of the terrestrial bodies over a time step τ. This is constituted by a specific sequence of Lie series, applied to a non autonomous Hamiltonian of the form:

$$H(\mathbf{r}, \mathbf{p}', t) = H_{\rm Kep} + H_{\rm Pert} + H_{\rm Jov} + H_{\rm Sun} \tag{8}$$

where, again, $\mathbf{r}_i$, $\mathbf{p}'_i$ are the Poincaré canonical coordinates. The Hamiltonian is time-dependent though the heliocentric positions $\boldsymbol{\rho}_j$ and the barycentric momenta (velocities) $\boldsymbol{\pi}'_j = \mu_j \boldsymbol{\upsilon}'_j$ of the GPs, which are computed from the interpolated heliocentric values (Section 2.2.1). It is worth noting that the barycentric momenta $\mathbf{p}'$, $\boldsymbol{\pi}'$ are referred to the centre of mass of the whole system, i.e. considering the terrestrial bodies and the GPs altogether.

The different terms of the Hamiltonian are:

$$H_{\rm Kep}(\mathbf{r}, \mathbf{p}') = \sum_{i=1}^{N} \left(\frac{|\mathbf{p}'_i|^2}{2m_i} - \frac{GMm_i}{|\mathbf{r}_i|} \right) \tag{9}$$

that represents the two-body motion of the terrestrial bodies around the Sun, of mass M,

$$H_{\rm Pert}(\mathbf{r}) = -\sum_{i=1}^{T-1} \sum_{k=i+1}^{N} \frac{Gm_i m_k}{|\mathbf{r}_i - \mathbf{r}_k|} \tag{10}$$

which is the mutual gravitational perturbation amongst the terrestrial bodies,

$$H_{\rm Jov}(\mathbf{r}, t) = -\sum_{i=1}^{N} \sum_{j=1}^{J} \frac{Gm_i \mu_j}{|\mathbf{r}_i - \boldsymbol{\rho}_j(t)|} \tag{11}$$

that gives the direct gravitational perturbation of the GPs on the terrestrial bodies, and

$$H_{\rm Sun}(\mathbf{p}', t) = \frac{1}{2M} \left| \sum_{i=1}^{N} \mathbf{p}'_i + \sum_{j=1}^{J} \boldsymbol{\pi}'_j(t) \right|^2 \tag{12}$$

that represents the linear momentum of the Sun around the centre of mass of the whole system.

The detailed sequence of operations for a single time-step integration, from t to $t + \tau$, is as follows:

Step 1. Start with the heliocentric positions and velocities $\mathbf{r}$, $\mathbf{v}$ of all the terrestrial bodies, obtained from the previous time step, as well as the heliocentric positions and velocities $\boldsymbol{\rho}$, $\boldsymbol{\upsilon}$ interpolated for the GPs, at time t.

Step 2. Calculate the heliocentric acceleration $\mathbf{a}_{\rm R}$ on the terrestrial PPs due to relativistic corrections (Quinn, Tremaine & Duncan 1991):

$$\mathbf{a}_{{\rm R},i} = \frac{GM}{c^2 |\mathbf{r}_i|^3} \left(4\,\mathbf{r}_i \cdot \mathbf{v}_i\, \mathbf{v}_i - \mathbf{v}_i \cdot \mathbf{v}_i\, \mathbf{r}_i + 4M \frac{\mathbf{r}_i}{|\mathbf{r}_i|} \right), \quad i = 1, \ldots, T \tag{13}$$

Step 3. Apply a kick, over $\tau/2$, in the heliocentric velocities $\mathbf{v}$ of the terrestrials PPs due to the relativistic acceleration correction:

$$\mathbf{v}_i \longleftarrow \mathbf{v}_i + \frac{\tau}{2} \mathbf{a}_{{\rm R},i}, \qquad i = 1, \ldots, T \tag{14}$$

Step 4. Convert the heliocentric velocities $\mathbf{v}$, $\boldsymbol{\upsilon}$ of all bodies to barycentric $\mathbf{v}'$, $\boldsymbol{\upsilon}'$, with respect to the centre of mass of the whole system:

$$\begin{aligned}
\mathbf{V}' &= -\frac{\sum_{i=1}^{N} m_i \mathbf{v}_i + \sum_{j=1}^{J} \mu_j \boldsymbol{\upsilon}_j}{M + \sum_{i=1}^{N} m_i + \sum_{j=1}^{J} \mu_j} \\
\mathbf{v}'_i &= \mathbf{v}_i + \mathbf{V}', \qquad i = 1, \ldots, N \\
\boldsymbol{\upsilon}'_j &= \boldsymbol{\upsilon}_j + \mathbf{V}', \qquad j = 1, \ldots, J
\end{aligned} \tag{15}$$

Step 5. Compute the heliocentric accelerations $\mathbf{a}_{\rm J}$ on the terrestrial bodies due to the GPs:

$$\mathbf{a}_{{\rm J},i} = -\sum_{j=1}^{J} G\mu_j \frac{\mathbf{r}_i - \boldsymbol{\rho}_j}{|\mathbf{r}_i - \boldsymbol{\rho}_j|^3}, \qquad i = 1, \ldots, N \tag{16}$$

Step 6. Apply a kick, over $\tau/2$, in the barycentric velocities $\mathbf{v}'$ of the terrestrial bodies due to these accelerations:

$$\mathbf{v}'_i \longleftarrow \mathbf{v}'_i + \frac{\tau}{2}\mathbf{a}_{\mathrm{J},i}, \qquad i = 1, \ldots, N \tag{17}$$

Step 7. Perform a LD, over $\tau/2$, of the heliocentric positions $\mathbf{r}$ of the terrestrial bodies due to the GPs contribution to the Sun momentum:

$$\mathbf{r}_i \longleftarrow \mathbf{r}_i + \frac{\tau}{2M}\sum_{j=1}^{J}\mu_j \boldsymbol{\upsilon}'_j, \qquad i = 1, \ldots, N \tag{18}$$

Step 8. Perform a LD, over $\tau/2$, of the heliocentric positions $\mathbf{r}$ of the terrestrial bodies due to their own contribution to the Sun momentum:

$$\mathbf{r}_i \longleftarrow \mathbf{r}_i + \frac{\tau}{2M}\sum_{k=1}^{N} m_k \mathbf{v}'_k, \qquad i = 1, \ldots, N \tag{19}$$

Step 9. Compute the heliocentric mutual accelerations $\mathbf{a}_{\mathrm{T}}$ of the terrestrial PPs and PLs:

$$\begin{aligned}\mathbf{a}_{\mathrm{T},i} &= -\sum_{k=i+1}^{N} Gm_k \frac{\mathbf{r}_i - \mathbf{r}_k}{|\mathbf{r}_i - \mathbf{r}_k|^3}, \qquad i = 1, \ldots, T \\ \mathbf{a}_{\mathrm{T},k} &= -\sum_{i=1}^{T} Gm_i \frac{\mathbf{r}_k - \mathbf{r}_i}{|\mathbf{r}_k - \mathbf{r}_i|^3}, \qquad k = T+1, \ldots, N\end{aligned} \tag{20}$$

Step 10. Apply a kick, over $\tau/2$, in the barycentric velocities $\mathbf{v}'$ of the terrestrial bodies due to these accelerations:

$$\mathbf{v}'_i \longleftarrow \mathbf{v}'_i + \frac{\tau}{2}\mathbf{a}_{\mathrm{T},i}, \qquad i = 1, \ldots, N \tag{21}$$

Step 11. Drift the heliocentric positions $\mathbf{r}$ and barycentric velocities $\mathbf{v}'$ of the terrestrial PPs and PLs along the Keplerian orbits generated by the Hamiltonian H_{Kep} (equation 9), over a full time step τ

Step 12. Recompute the heliocentric mutual accelerations $\mathbf{a}_{\mathrm{T}}$ of the terrestrial bodies (equation 20)

Step 13. Apply a kick, over $\tau/2$, in the barycentric velocities $\mathbf{v}'$ of the terrestrial bodies due to these accelerations (equation 21)

Step 14. Perform a LD, over $\tau/2$, of the heliocentric positions $\mathbf{r}$ of the terrestrial bodies due to their own contribution to the Sun momentum (equation 19)

Step 15. Convert the heliocentric velocities $\boldsymbol{\upsilon}$ of the GPs at time $t + \tau$ to barycentric $\boldsymbol{\upsilon}'$, with respect to the centre of mass of the whole system:

$$\begin{aligned}\mathbf{V}' &= -\frac{\sum_{i=1}^{N} m_i \mathbf{v}'_i + \sum_{j=1}^{J}\mu_j \boldsymbol{\upsilon}_j}{M + \sum_{j=1}^{J}\mu_j} \\ \boldsymbol{\upsilon}'_j &= \boldsymbol{\upsilon}_j + \mathbf{V}', \qquad j = 1, \ldots, J\end{aligned} \tag{22}$$

Step 16. Perform a LD, over $\tau/2$, of the heliocentric positions $\mathbf{r}$ of the terrestrial bodies due to the GPs contribution to the Sun momentum (equation 18)

Step 17. Recompute the heliocentric accelerations $\mathbf{a}_{\mathrm{J}}$ on the terrestrial bodies due to the GPs (equation 16)

Step 18. Apply a kick, over $\tau/2$, in the barycentric velocities $\mathbf{v}'$ of the terrestrial bodies due to these accelerations (equation 17)

Step 19. Convert back the barycentric velocities $\mathbf{v}'$ of the terrestrial bodies to heliocentric $\mathbf{v}$:

$$\begin{aligned}\mathbf{V}' &= -\frac{\sum_{i=1}^{N} m_i \mathbf{v}'_i + \sum_{j=1}^{J}\mu_j \boldsymbol{\upsilon}'_j}{M} \\ \mathbf{v}_i &= \mathbf{v}'_i - \mathbf{V}', \qquad i = 1, \ldots, N\end{aligned} \tag{23}$$

Step 20. Recalculate the heliocentric acceleration $\mathbf{a}_{\mathrm{R}}$ on the terrestrial PPs due to relativistic corrections (equation 13)

Step 21. Apply a kick, over $\tau/2$, in the heliocentric velocities $\mathbf{v}$ of the terrestrials PPs due to the relativistic acceleration correction (equation 14)

Step 22. Return to step 2.

At variance with the standard SYMBA code, which uses a LD - K - D - K - LD sequence, ISYMBA uses a $\mathrm{K_J}$ - $\mathrm{LD_J}$ - $\mathrm{LD_T}$ - $\mathrm{K_T}$ - $\mathrm{D_T}$ - $\mathrm{K_T}$ - $\mathrm{LD_T}$ - $\mathrm{LD_J}$ - $\mathrm{K_J}$ sequence, where the sub-indices J,T refer to jovian and terrestrial, respectively, with an additional outer $\mathrm{K_R}$ sequence for relativistic corrections, if required. This specific sequence has three advantages over other possible sequences:

(i) The GPs perturbations (equation 16) do not need to be recomputed at the beginning of each time step; they can be recovered from the final values of the previous time step.

(ii) If relativistic corrections are not accounted for, there is no need to recompute the barycentric velocities at the beginning of each time step (equation 15); they can be recovered from the final values of the previous time step.

(iii) The innermost $\mathrm{LD_T}$ - $\mathrm{K_T}$ - $\mathrm{D_T}$ - $\mathrm{K_T}$ - $\mathrm{LD_T}$ sequence (steps 8–14), which treats the interactions amongst the terrestrial bodies solely, could be executed by the standard SYMBA algorithm.

2.2.3 *Treatment of close encounters amongst terrestrial bodies*

Close encounters amongst terrestrial PPs, or between terrestrial PPs and PLs, are manipulated by ISYMBA using the same strategy of SYMBA, i.e. by splitting the potential term (equation 10) as:

$$H_{\mathrm{Pert}}(\mathbf{r}) = H^{(0)}_{\mathrm{Pert}} + \textstyle\sum_{s=1}^{n} H^{(s)}_{\mathrm{Pert}} \tag{24}$$

$$H^{(0)}_{\mathrm{Pert}} = -\textstyle\sum_{i=1}^{T-1}\sum_{k=i+1}^{N} \frac{Gm_i m_k}{r_{ik}} \prod_{l=1}^{n}\left(1 - w_{l,ik}\right) \tag{25}$$

$$H^{(s)}_{\mathrm{Pert}} = -\textstyle\sum_{i=1}^{T-1}\sum_{k=i+1}^{N} \frac{Gm_i m_k}{r_{ik}} w_{s,ik} \prod_{l=1}^{s-1}\left(1 - w_{l,ik}\right) \tag{26}$$

where the $w_{l,ik}$ are given by equation (6). The propagation of the orbits of any two bodies involved in an encounter during the innermost $\mathrm{LD_T}$-$\mathrm{K_T}$-$\mathrm{D_T}$-$\mathrm{K_T}$-$\mathrm{LD_T}$ sequence (steps 8–14), is carried out recursively through the nested application of a $\mathrm{K_T}$-$\mathrm{D_T}$-$\mathrm{K_T}$ sequence to the Hamiltonian $\left(\left(\left(H_{\mathrm{Kep}} + H^{(0)}_{\mathrm{Pert}}\right) + H^{(1)}_{\mathrm{Pert}}\right) + \ldots\right) + H^{(n)}_{\mathrm{Pert}}$, using smaller and smaller time steps $\tau_l = \tau/3^l$, $l = 1, \ldots, n$.

For this stage, we use the original SYMBA subroutines, with little modifications. Bodies are always merged whenever they reach the last stage of the recursion and get closer than $R_{n+1,ik}$.

2.2.4 *Treatment of close encounters between terrestrial bodies and GPs*

Close encounters with GPs are not properly manipulated by ISYMBA. The application of a splitting strategy to the term H_{Jov} (equation 11) would demand additional interpolations of the GPs orbits, down to smaller and smaller time steps. This would also require significant changes to the standard SYMBA recursive subroutines, which turns to be a quite complex task. Therefore, we leave this implementation to a future work.

Currently, the way ISYMBA deals with such close encounters is to keep tracking the distances between the PPs/PLs and the GPs, and discard the former when they get closer to a GP than the sum of their individual Hill radii. This solution proved to be adequate for our purposes, since the disc of terrestrial bodies is interior to Jupiter's orbit. Thus, we may expect that close encounters of PLs with the

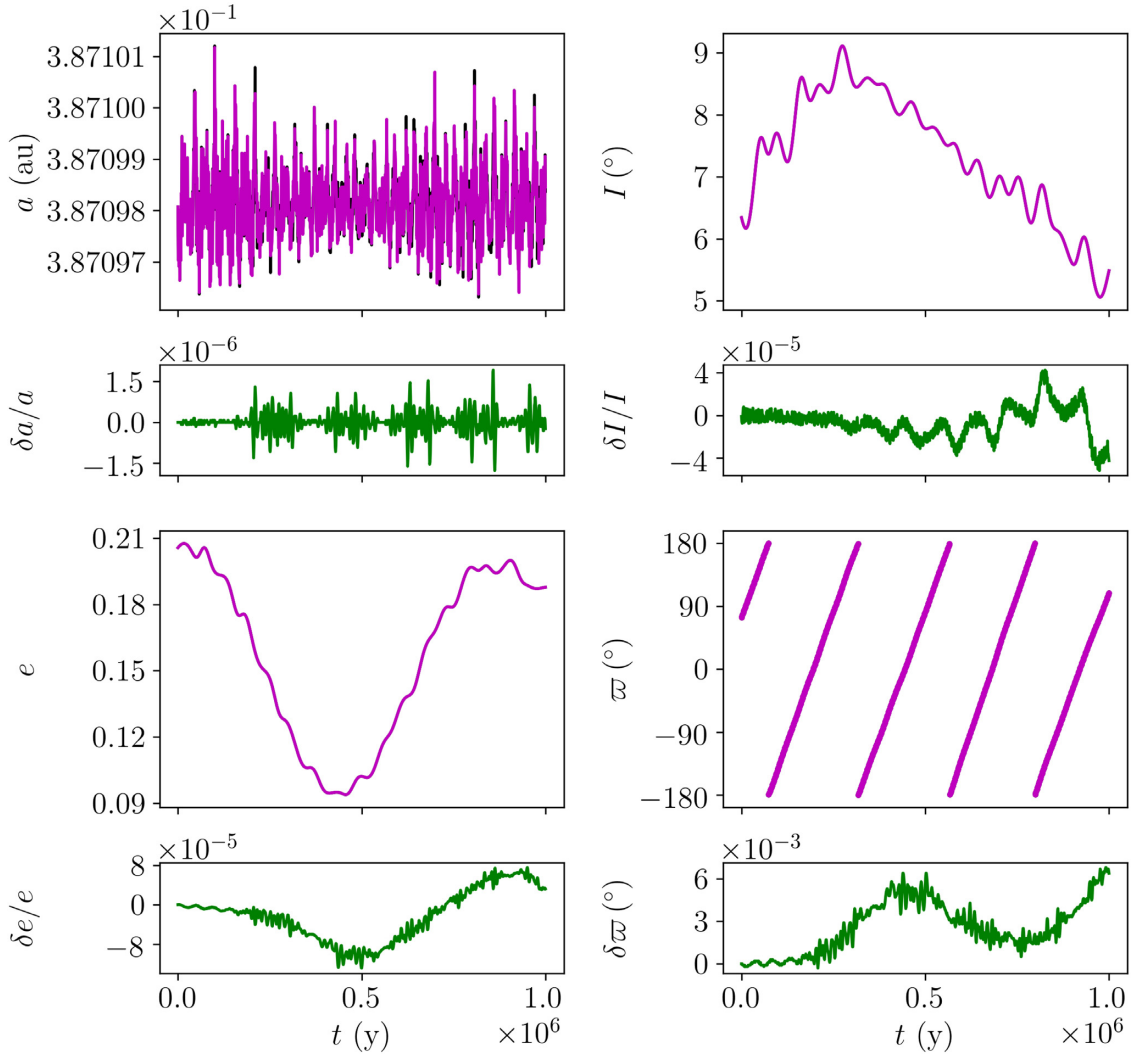

Figure 3. Comparison between the output from SYMBA (black) and ISYMBA (over-plotted in magenta) for the orbital elements of planet Mercury, over 1 My of evolution, in the 4G4T model (see text for the simulation details): a, semimajor axis, e eccentricity, I inclination, and ϖ longitude of perihelion. The smaller panels show the relative differences (in green), except for ϖ that shows the absolute difference.

Table 1. Maximum differences in the orbital elements of the terrestrial planets, over 1 My, for the validation test using the 4G4T model (see text).

	$\delta a/a$	$\delta e/e$	$\delta I/I$	$\delta\varpi$ (°)
Mercury	1.5×10^{-6}	9×10^{-5}	4×10^{-5}	6×10^{-3}
Venus	2.5×10^{-5}	6×10^{-3}	2×10^{-3}	2×10^{-1}
Earth	4×10^{-5}	1.5×10^{-2}	6×10^{-3}	5×10^{-1}
Mars	3×10^{-5}	1.5×10^{-3}	2×10^{-4}	4×10^{-2}

GPs will be rare, and they eventually will affect only the outer edge of the disc. In particular, using the results of Nesvorný et al. (2021), we verified that, when the disc extends up to 4 au with a radial surface density profile $\Sigma(r) = r^{-1}$, less than 30 per cent of the PLs experienced close encounters with Jupiter during the simulations.

2.2.5 *Treatment of small perihelion passages*

The ISYMBA code does not properly treat the $H_{\rm Sun}$ term increase (equation 12) when a terrestrial body experiences a small perihelion passage or get too close to the Sun. This limitation also exists in the standard SYMBA code. In such cases, the code relies only in the setup of a sufficiently small time step to resolve the perihelion passage without loosing too much precision. In Nesvorný et al. (2021), we setup a time step $\tau = 3$–7 d, which proved to be good enough in that application. Bodies are discarded whenever they reach heliocentric or perihelion distances smaller than a given threshold, set by the user.

2.2.6 *Symplecticity and energy conservation*

The symplecticity of ISYMBA is guaranteed by the conservation of the corresponding surrogate Hamiltonian:

$$\begin{aligned}\overline{H} &= H_{\rm Kep} + H_{\rm Pert} + H_{\rm Jov} + H_{\rm Sun} \\ &\quad - \sum_{i=0}^{n} \frac{\tau_i^2}{12} \left[\left[H_{\rm Kep}\,,\, H_{{\rm Pert},i} \right],\, H_{\rm Kep} + \frac{1}{2} H_{{\rm Pert},i} + \sum_{j=i+1}^{n} H_{{\rm Pert},j} \right] \\ &\quad - \frac{\tau_0^2}{12} \left[\left[H_{\rm Kep}\,,\, H_{\rm Sun} \right],\, H_{\rm Kep} + \frac{1}{2} H_{\rm Sun} \right] \\ &\quad - \frac{\tau_0^2}{12} \left[\left[H_{\rm Kep}\,,\, H_{\rm Jov} \right],\, H_{\rm Kep} + \frac{1}{2} H_{\rm Jov} \right] + \mathcal{O}(\tau^4) \end{aligned} \tag{27}$$

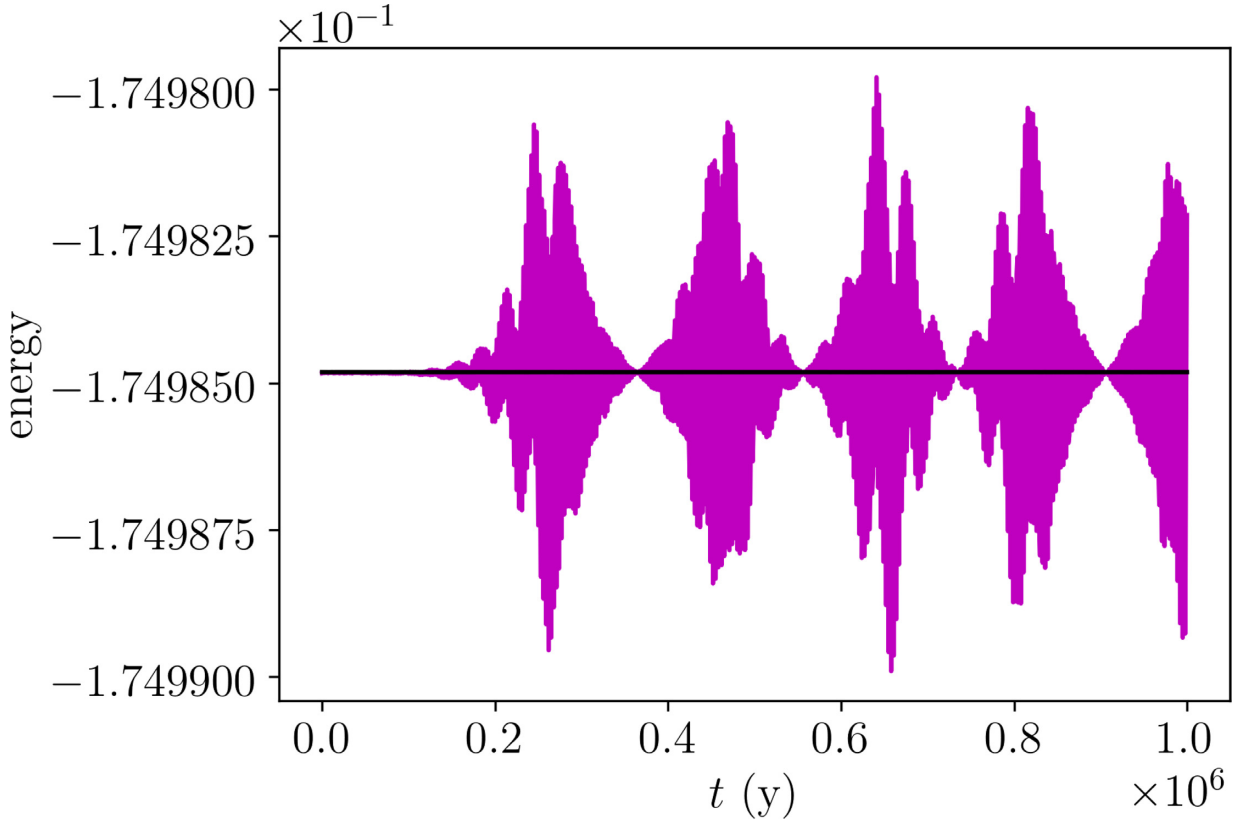

Figure 4. Comparison between the total energy of the system from SYMBA (black) and from ISYMBA (magenta), in the 4G4T model (see text). Since the ISYMBA Hamiltonian is non autonomous, the energy displays oscillations with amplitude $<10^{-5}$, with respect to the SYMBA energy.

where [,] represents the Lagrange brackets. In practice, the total energy can be computed as the sum of the barycentric kinetic and potential energies, taking into account all the three categories of bodies in the system:

$$E = \frac{1}{2}M|\mathbf{V}'|^2 + \sum_{i=1}^{N}\frac{1}{2}m_i|\mathbf{v}_i'|^2 + \sum_{j=1}^{J}\frac{1}{2}\mu_j|\boldsymbol{\upsilon}_j'|^2 - \sum_{i=1}^{N}\frac{GMm_i}{|\mathbf{r}_i|} - \sum_{i=1}^{T-1}\sum_{k=i+1}^{N}\frac{Gm_im_k}{|\mathbf{r}_i-\mathbf{r}_k|} - \sum_{j=1}^{J}\frac{GM\mu_j}{|\boldsymbol{\rho}_j|} - \sum_{j=1}^{J-1}\sum_{k=j+1}^{J}\frac{G\mu_j\mu_k}{|\boldsymbol{\rho}_j-\boldsymbol{\rho}_k|} - \sum_{i=1}^{N}\sum_{j=1}^{J}\frac{Gm_i\mu_j}{|\mathbf{r}_i-\boldsymbol{\rho}_j|} \quad (28)$$

It is worth recalling that in ISYMBA, the model Hamiltonian is non-autonomous. Therefore, the energy is not expected to be constant, but it should display periodic oscillations with bounded amplitude.

2.3 Validation tests

The ISYMBA algorithm has been validated as follows. We consider a system consisting of J GPs and T terrestrial planets, and perform a simulation over 1 My using the standard SYMBA, i.e. allowing for all the planets to be mutually perturbed. The output (orbital elements) of this simulation for the GPs is stored in a file every 1 yr, while the output for the terrestrial planets is stored in a separate file every 1000 yr. Then, we repeat the simulation using ISYMBA, with the same initial conditions for the terrestrial planets, and interpolating the GPs' orbits from the ones previously stored. The output of this simulation is recorded every 1000 yr, and it is directly compared to the previous output from SYMBA. In all the simulations, we check that the total energy of the system, computed from equation (28), is well behaved.

We apply the above validation test to two different systems. The first one is the present Solar system, with four terrestrial and four GPs. Initial conditions are taken from the JPL Ephemerides. Relativistic perturbations are not taken into account, and there are no close encounters between the planets. We call this the 4G4T model.

The second system is a fictitious system composed of five GPs and 20 terrestrial bodies. The GPs are Jupiter, Saturn, and three ice giants, initially in a mutual resonant and compact orbital configuration (see DeSouza et al. 2021, for example). The 20 terrestrial bodies are represented by planetary embryos with a total mass of 5 $M_\oplus$. These bodies are uniformly distributed in a very narrow annulus, between 0.95 and 1.05 au, with eccentricities <0.01 and inclinations <0.001. The idea of this setup is to force close encounters between the terrestrial bodies, in order to test the behaviour of ISYMBA under such conditions. Relativistic perturbations are not taken into account either. We call this the 5G20T model.

Fig. 3 shows a result from the 4G4T model. The panels display the evolution of the orbital elements of Mercury from the ISYMBA simulation (in magenta), and the SYMBA simulation (in black, but not visible due to overlapping). The differences between the two codes are shown in green. The behaviour is similar for the other terrestrial planets. Table 1 summarizes the maximum relative differences found in this validation test. The total energy of the system behaves as expected (Fig. 4), and the differences between the two codes in no larger than 10^{-5} over the whole time span.

In the 5G20T model, reproducing the exact evolution of the system with ISYMBA is not feasible, because the system is chaotic, and the small differences caused by the many collisions/mergers produce

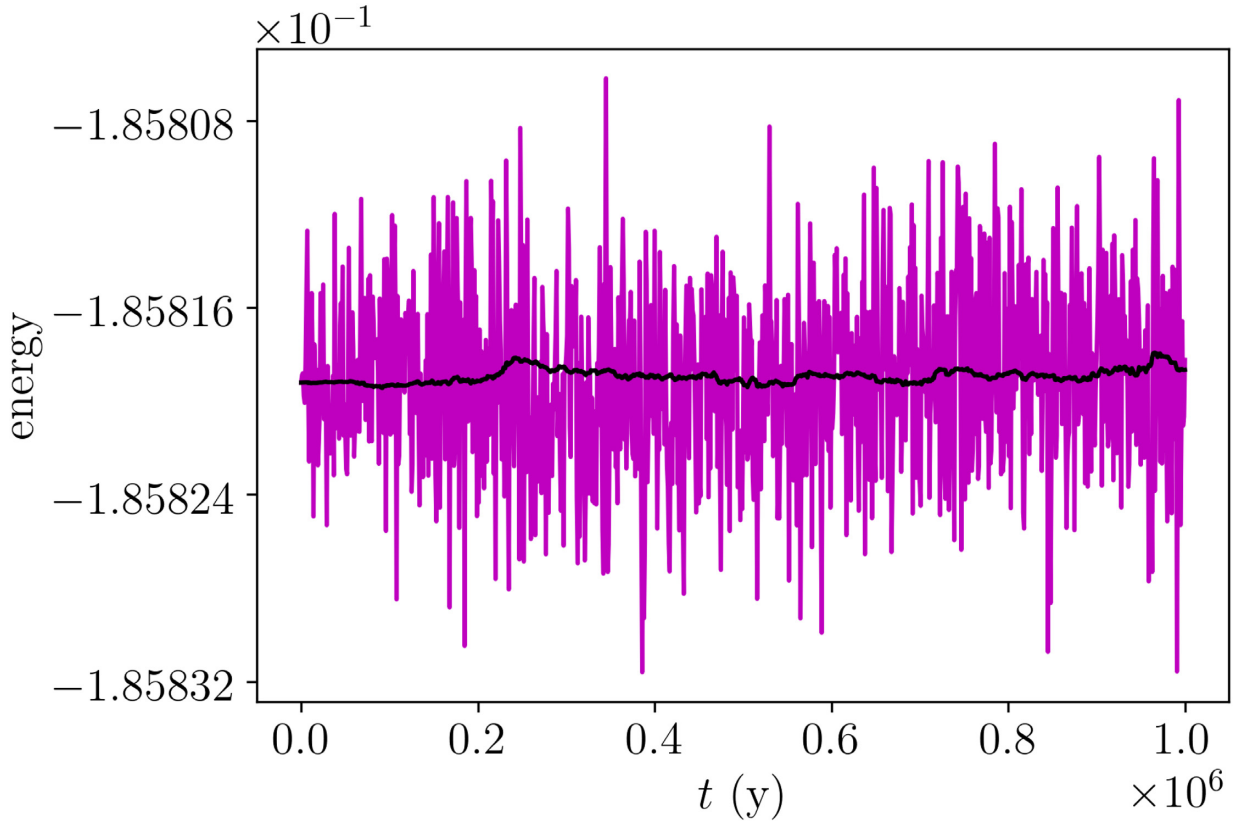

Figure 5. Comparison between the total energy of the system from SYMBA (black) and from ISYMBA (magenta), in the 5G20T model (see text). Again, the ISYMBA energy displays oscillations with amplitude $<2 \times 10^{-5}$, with respect to the SYMBA energy.

quantitatively different results. We verify, however, that the total energy of the system is well behaved, as shown in Fig. 5, with a bounded amplitude for ISYMBA that is only twice than in the 4G4T model without collisions. Although the number of collisions recorded in SYMBA and ISYMBA is not the same, the latter is able to reproduce quite well the first few collisions in the simulation. An example of this is shown in Fig. 6. We note that the merger happens at slightly different times in each case. Such small differences quickly propagate, and in a few tens of years each code starts to produce its own set of collisions, not matching each other anymore.

The above validation tests allow us to conclude that ISYMBA shows the desired behaviour in terms of simulation results.

3 PARALLELIZATION STRATEGY

Aiming to improve the performance of ISYMBA, we have implemented multithreading parallelization using OPENMP. The following discussion does not intend to be comprehensive, and it is only applicable to the specific test models described below. Our aim is to provide some clues about the possible best strategies to parallelize our code, which eventually can be also taken into account to parallelize SYMBA itself.

There is no unique strategy to parallelize a code, and sometimes the best approach is obtained by first redesigning the original serial code. Here, however, the idea is to adopt a parallelization strategy that keeps the original serial structure of the SYMBA subroutines with minimum or no changes.

The following discussion is based on two different test models: one considering 100 terrestrial PPs and 1000 PLs (e.g. Nesvorný et al. 2021), which we refer to as m100n1000 model, and another considering 10 PPs and 10 000 PLs, which we called m10n10000 model. In both cases, the number of GPs is 5. In all the tests, we use the same total time span and the same time step. We also keep the amount of I/O operations to the minimum required. We define the normalized execution time as the ratio $T_{\rm par}/T_{\rm ser}$, where $T_{\rm ser}$ is the total execution time of the serial code, without any parallelization, and $T_{\rm par}$ is the execution time of the parallelized code. All the tests have been performed in Intel Core i7 processors, using GNU FORTRAN.

There are basically two types of structures that can be parallelized in ISYMBA using OPENMP:

(i) the operations that require double loops over the terrestrial bodies, like the mutual acceleration calculations (equations 13, 16, and 20), the check for close encounters, and the energy computation (equation 28), and

(ii) the calculations that require single loops over the bodies, including the Keplerian drifts, the LD, the kicks, the coordinate changes, the loops to deal with interpolation of the GPs, etc.

We will discuss each structure separately.

3.1 Single loops

Paralellization of the single loops within the code has to be carefully evaluated, because it may provide little or no improvement of the execution speed. For example, in the LD - K - D - K - LD integration scheme, the Keplerian drifts are, in theory, the second most CPU-expensive step, after the accelerations calculation. However,

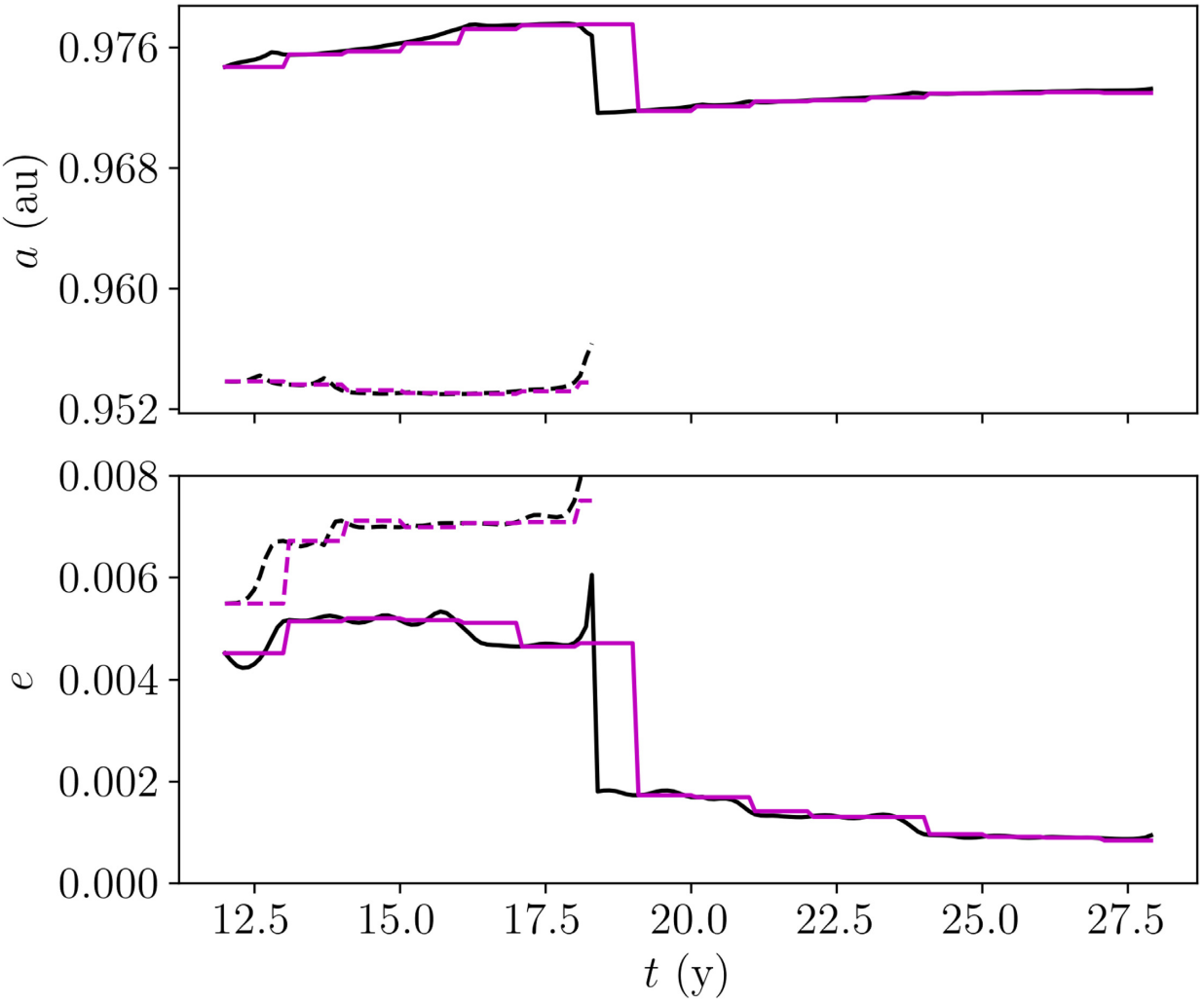

Figure 6. Detail of an encounter/merger between two terrestrial bodies in the 5G20T model. The SYMBA simulation is shown in black, and ISYMBA in magenta. Full and dashed lines identify each of the bodies, respectively.

in our simulations, the Keplerian drifts take between 5 per cent and 20 per cent of the total run time in a serial run. Therefore, their parallelization might not contribute significantly to improve performance.

We have verified that, in the m100n1000 simulations, parallelization of the single loops makes the code only ∼1.2 times faster, but in the m10n10000 simulations, it becomes ∼2.3 times faster.

We have also verified that parallelization of any loop within the recursive integration of close encounters (see Section 2.2.3) must be avoided, since it may slow down execution speed by a factor of 2. This concerns, in particular, the subroutines SYMBA7_STEP_RECUR and SYMBA7_KICK. We note, however, that there is no actual need to parallelize any part of the recursion, because it only affects the pair of bodies involved in an encounter, and these are not too frequent per time step.

After several experiments, we conclude that multithreading parallelization has to focus on the double loops, as explained below, and on the single loops that performs the most complex calculations, like the Keplerian drifts, the loops to deal with the interpolation of the GPs, the relativistic corrections, the oblateness potential, and the discard subroutines.

3.2 Double loops

The strategy applied for parallelization of the double loops influences the execution speed. A double loop to compute the accelerations between PPs and PLs typically reads:

```
do i from 1 to m
   do j from i + 1 to n
      a(j)  = a(j)  + accel_ji
      a(i)  = a(i)  + accel_ij
   end do
end do
```

where `m` is the number of self gravitating PPs, `n` is the number of PLs, and `a()` is an array of dimension `n`. In this case, one possible strategy is to parallelize the outer loop over `i`, and the other possibility is to parallelize the inner loop over `j`.

Fig. 7 shows the normalized execution time for the different models and strategies, as a function of the number of threads. The solid lines correspond to the case in which the outer loops are parallelized, while the dashed lines correspond to the case in which the inner loops are parallelized. We note that, when using few threads, there is no significant difference between one strategy or the other, although parallelization of the outer loops performs a bit better. On the other

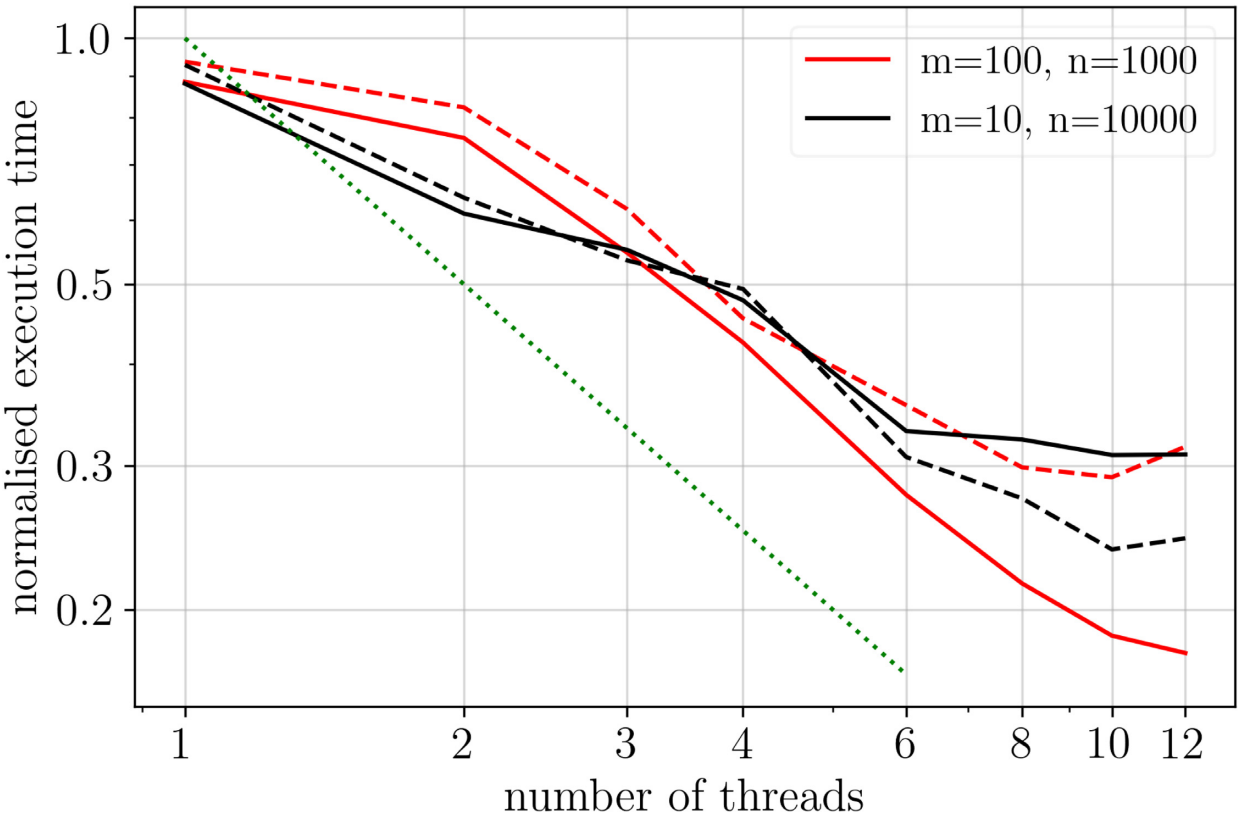

Figure 7. The performance of ISYMBA for different multithreading parallelization strategies. The solid lines correspond to the parallelization of the outer loop in double loops, while the dashed lines correspond to the parallelization of the inner loop. The colours identify the test models with different numbers of PPs (m) and PLs (n). The execution time has been normalized with respect to the serial execution time (see text for details). The green dotted line represents a linear trend, for reference purposes.

hand, when using many threads, each strategy has advantages over the other depending on the values of `m` and `n`.

For the simulations of the m100n1000 model, parallelization of the outer loops provides ∼1.5 times faster execution times with respect to the parallelization of the inner loops. On the other hand, for the simulations of the m10n10000 model, it is the parallelization of the inner loops that provides ∼1.3 times faster execution times than parallelizing the outer loops.[2]

This behaviour may be related to the fact that initialization and execution of a parallel loop involve several tasks, besides the calculations within the loop, which may produce some latency in the execution. When parallelizing an inner loop, the parallel threads have to be initialized/allocated for every iteration of the outer loop, and this may cause a lot of latency. If the outer loop is small, as in `m` = 10, the latency does not have a big impact. However, if the outer loop is bigger, as in `m` = 100, the impact of latency may become significant.

Parallelization of the outer loops has a couple of additional peculiarities that should be taken into account. The first one refers to the fact that the number of iterations over `j`, in the inner loop, is smaller for larger `i`. This means that the amount of work to be done by each iteration over `i` is different. In such case, the way of scheduling the iterations may be relevant. One possibility is choosing between a cyclic or a block scheduling. A cyclic schedule distributes the loop iterations in a round-robin fashion amongst the available threads, and should provide better performance in our case. The other possibility is choosing between static or dynamic scheduling. We have verified that using a combination of static and cyclic scheduling provides ∼1.1 times faster execution times than either a dynamic or block scheduling.

The second peculiarity in parallelizing the outer loops refers to the occurrence of data racing conditions over the accelerations array `a()`. A racing condition arises when two or more processes running in different threads try to modify or update the same variable at the same time. Fortunately, FORTRAN OPENMP has the capability of performing array reduction, which allows to properly update `a()`, avoiding data race.

Although multithreading parallelization, in our case, may improve execution speed by a factor of 3–6, the improvement is not linear with the number of threads and, as shown in Fig. 7, it tends to stabilize for $\gtrsim 10$ threads.

4 CONCLUSIONS

In this work, we described how to implement the necessary modifications to embed an orbit interpolation scheme into the symplectic planetary *N*-body integrator SYMBA. Our algorithm, named ISYMBA, allows to study the effects of a prescribed evolution of a set of planets on a target population of massive bodies, which interact with each other through close encounters.

ISYMBA is a very useful code to accurately evaluate the effects of planetary instabilities on the accretion processes in the terrestrial planets region. These include the growth of protoplanetary embryos (Nesvorný et al. 2021), the Moon-forming impact (DeSouza et al. 2021), and the origin of Mercury.

[2]These tests have been performed using eight threads.

Although ISYMBA has been primarily developed and implemented to study terrestrial planet formation, the method presented here could be easily modified to study the evolution of other populations, which requires orbit interpolation from previously developed simulations, while accounting for close encounters amongst massive objects.

ACKNOWLEDGEMENTS

We wish to thank an anonymous referee for helpful comments and suggestions. FR acknowledges financial support form the Brazilian National Council of Research (CNPq). RD acknowledges financial support from Southwest Research Institute, IR&D Targeted Research Program. The simulations have been performed at the SDumont cluster of the Brazilian System of High-Performance Computing (SINAPAD).

DATA AVAILABILITY

ISYMBA is freely available under request to the corresponding author.

REFERENCES

Beaugé C., Roig F., Nesvorný D., 2002, Icarus, 158, 483
Bottke W. F., Vokrouhlický D., Minton D., Nesvorný D., Morbidelli A., Brasser R., Simonson B., Levison H. F., 2012, Nature, 485, 78
Chambers J. E., 1999, MNRAS, 304, 793
Clement M. S., Kaib N. A., Raymond S. N., Walsh K. J., 2018, Icarus, 311, 340
Clement M. S., Kaib N. A., Raymond S. N., Chambers J. E., Walsh K. J., 2019, Icarus, 321, 778
Deienno R., Nesvorný D., Vokrouhlický D., Yokoyama T., 2014, AJ, 148, 25
Deienno R., Morbidelli A., Gomes R. S., Nesvorný D., 2017, AJ, 153, 153
Deienno R., Izidoro A., Morbidelli A., Gomes R. S., Nesvorný D., Raymond S. N., 2018, ApJ, 864, 50
DeSouza S. R., Roig F., Nesvorný D., 2021, MNRAS, 507, 539
Duncan M. J., Levison H. F., Lee M. H., 1998, AJ, 116, 2067
Gomes R., Levison H. F., Tsiganis K., Morbidelli A., 2005, Nature, 435, 466
Gomes R., Nesvorný D., Morbidelli A., Deienno R., Nogueira E., 2018, Icarus, 306, 319
Levison H. F., Duncan M. J., 1993, ApJ, 406, L35
Nesvorný D., 2015, AJ, 150, 68
Nesvorný D., 2018, ARA&A, 56, 137
Nesvorný D., Morbidelli A., 2012, AJ, 144, 117
Nesvorný D., Vokrouhlický D., Morbidelli A., 2013, ApJ, 768, 45
Nesvorný D., Vokrouhlický D., Deienno R., Walsh K. J., 2014a, AJ, 148, 52
Nesvorný D., Vokrouhlický D., Deienno R., 2014b, ApJ, 784, 22
Nesvorný D., Roig F., Bottke W. F., 2017, AJ, 153, 103
Nesvorný D., Roig F. V., Deienno R., 2021, AJ, 161, 50
Quinn T. R., Tremaine S., Duncan M., 1991, AJ, 101, 2287
Roig F., Nesvorný D., 2015, AJ, 150, 186
Roig F., Nesvorný D., DeSouza S. R., 2016, ApJ, 820, L30
Tsiganis K., Gomes R., Morbidelli A., Levison H. F., 2005, Nature, 435, 459
Wisdom J., Holman M., 1991, AJ, 102, 1528
Yoshida H., 1993, Celest. Mech. Dyn. Astron., 56, 27

This paper has been typeset from a TEX/LATEX file prepared by the author.