Institutional Repository - Research Portal
Dépôt Institutionnel - Portail de la Recherche
researchportal.unamur.be
University of Namur

THESIS / THÈSE

MASTER EN SCIENCES MATHÉMATIQUES

Développement d'un intégrateur symplectique pour les systèmes binaires. Application à la formation des systèmes de planètes géantes.

Roisin, Arnaud

Award date:
2017

Awarding institution:
Universite de Namur

Link to publication

General rights
Copyright and moral rights for the publications made accessible in the public portal are retained by the authors and/or other copyright owners and it is a condition of accessing publications that users recognise and abide by the legal requirements associated with these rights.

- Users may download and print one copy of any publication from the public portal for the purpose of private study or research.
- You may not further distribute the material or use it for any profit-making activity or commercial gain
- You may freely distribute the URL identifying the publication in the public portal ?

Take down policy
If you believe that this document breaches copyright please contact us providing details, and we will remove access to the work immediately and investigate your claim.

Download date: 25. Apr. 2024

UNIVERSITE DE NAMUR

Faculté des Sciences

DÉVELOPPEMENT D'UN INTÉGRATEUR SYMPLECTIQUE POUR LES SYSTÈMES BINAIRES. APPLICATION À LA FORMATION DES SYSTÈMES DE PLANÈTES GÉANTES.

Promotrice :

Anne-Sophie LIBERT

Mémoire présenté pour l'obtention

du grade académique de master en « Sciences mathématiques à finalité approfondie »

Arnaud ROISIN

Juin 2017

Remerciements

Au moment d'écrire ces remerciements, mes premières pensées vont à ma promotrice Anne-Sophie Libert qui m'a accompagné durant l'ensemble du travail relatif à ce manuscrit. Sans elle, ce mémoire ne serait pas ce qu'il est. Je tiens à la remercier chaleureusement pour la patience dont elle a fait preuve lors de nos nombreuses entrevues. Je la remercie également pour ses relectures ainsi que ses précieux conseils qui m'ont permis d'améliorer ce manuscrit. Notre collaboration dans le cadre de ce travail m'a donné goût à la recherche et m'a fait progresser sur ma manière de l'aborder.

Je voudrais également remercier Kleomenis Tsiganis, mon maître de stage, qui m'a accompagné et soutenu lors de mon séjour à Thessalonique. A ses côtés, mes connaissances en mécanique céleste se sont améliorées. De plus, il m'a apporté un regard neuf sur ce domaine. Je remercie également l'ensemble de l'équipe de mécanique céleste de l'Université d'Aristote à Thessalonique pour leur accueil chaleureux durant mon stage de recherche. Je les remercie pour les nombreuses discussions extrêmement intéressantes que nous avons eues. Ce séjour à leurs côtés m'a réellement donné goût à ce domaine de recherche.

Je profite de ces quelques lignes pour également remercier Sotiris Sotiriadis avec qui j'ai énormément échangé et qui a eu la patience de répondre à l'ensemble de mes questions à propos du code SyMBA. Son expérience et les codes qu'il m'a généreusement fournis m'ont permis de réaliser ce mémoire.

Je remercie également Alexandre Mauroy, André Füzfa, Anne Lemaître et Anne-Sophie Libert pour le temps consacré à la lecture de ce travail.

De plus, je tiens à remercier les corps enseignants de l'Université de Namur et de l'Université de Cardiff qui m'ont fait progresser durant mes études tant mathématiquement qu'humainement. Sans eux, je ne serais pas là où j'en suis et ce manuscrit ne serait pas ce qu'il est.

Je remercie tous mes amis et, en particulier, les étudiants de master qui m'ont soutenu et avec qui j'ai passé d'excellentes sessions de travail.

Enfin je remercie ma famille pour leur soutien inconditionnel et pour les relectures orthographiques.

Abstract

Actuellement, les scientifiques estiment que plus de la moitié des systèmes stellaires sont multiples. Des exoplanètes ont déjà été détectées dans une soixantaine de ces systèmes. Dans ce mémoire, nous avons développé un intégrateur symplectique calculant l'évolution des systèmes binaires abritant des planètes de type S, c'est-à-dire en orbite autour d'une des deux étoiles. L'intérêt de tels intégrateurs est de permettre, sur base de la structure hamiltonienne des équations d'évolution, de limiter la perte d'énergie, même avec un pas d'intégration élevé. L'adaptation du code existant SyMBA aux systèmes binaires a nécessité l'introduction d'un jeu de coordonnées adapté, ainsi qu'un découpage différent de l'hamiltonien. Notre code est également adapté au problème des rencontres proches entre les planètes et à celui des rencontres proches entre les planètes et le corps central. La question de la migration des systèmes de planètes géantes dans les binaires a ensuite été abordée. Le code a été modifié pour simuler la migration de type II des planètes géantes et étudier l'influence d'un compagnon binaire éloigné sur les configurations finales des corps. Pour ce faire, nous avons étudié l'évolution de plus de 1300 simulations. Notre attention s'est portée, en particulier, sur l'influence des paramètres initiaux du compagnon binaire sur le processus de migration.

Mots-clés : Systèmes binaires - Intégrateurs symplectiques - Problème à n corps - Théorie hamiltonienne - Exoplanètes - Formation - Migration

Currently, scientists estimate that more than half of the stellar systems are multiple. Exoplanets have already been detected in about sixty of those systems. In the master thesis, we have developed a symplectic integrator for the evolution of binary systems containing S-type planets in orbit around one of the two stars. The interest of such integrators is to allow, using the hamiltonian structure of the evolution equations, to limit the loss of energy, even when using a large stepsize. The adaptation of the SyMBA code to binary systems has required the introduction of an adapted set of coordinates as well as a different partition of the hamiltonian. Our code is also adapted to the problem of close encounters between the planets and the one between the planets and the central body. The formation of giant planet systems in binary stars has also been addressed. The code has been modified to include Type II migration of giant planets, in order to study the influence of a wide binary companion on the final architecture of the planetary systems. To do so, we have realized a parametric study of more than 1300 simulations. The influence of the initial parameters of the binary companion during the migration process has been extensively studied.

Keywords : Binary systems - Symplectic integrators - n-body problem - Hamiltonian theory - Exoplanets - Formation - Migration

Table des matières

I MÉMOIRE **1**

Introduction **3**

1 Systèmes binaires **5**
1.1 Etoiles binaires . . . 5
1.2 Classification . . . 7
1.3 Compagnons planétaires . . . 8

2 SyMBA **11**
2.1 Jeux de cordonnées . . . 11
2.1.1 Repère barycentrique . . . 11
2.1.2 Repère héliocentrique . . . 13
2.1.3 Repère héliocentrique canonique . . . 14
2.1.4 Repère de Jacobi . . . 14
2.2 Intégrateurs symplectiques . . . 15
2.2.1 Définition . . . 16
2.2.2 Familles d'intégrateurs symplectiques . . . 17
2.2.3 Décomposition en trois morceaux . . . 21
2.2.4 WHM . . . 22
2.2.5 RMVS . . . 23
2.2.6 Helio . . . 24
2.2.7 SyMBA . . . 30
2.3 Code . . . 35
2.3.1 Anal . . . 35
2.3.2 Coord . . . 36
2.3.3 CPP_test . . . 36
2.3.4 Discard . . . 37
2.3.5 Helio . . . 38
2.3.6 IO . . . 38
2.3.7 MVS . . . 44
2.3.8 RMVS . . . 46
2.3.9 SyMBA . . . 46
2.3.10 Orbel et Util . . . 47
2.4 Organigramme . . . 47
2.4.1 Initialisation . . . 48
2.4.2 Intégration . . . 49

3 Algorithme pour les systèmes binaires **53**
3.1 Coordonnées . . . 55
3.2 Intégrateur . . . 60
3.2.1 Sans rencontres proches . . . 60
3.2.2 Avec rencontres proches . . . 69
3.3 Code . . . 75

3.3.1 Fonctions . . . 75
3.3.2 Organigramme . . . 76
3.4 Validation . . . 79
3.4.1 Ordre de l'intégrateur . . . 79
3.4.2 Comparaison avec la littérature . . . 80

Conclusion **95**

II STAGE **97**

Introduction **99**

4 Rencontres proches avec le corps central **101**
4.1 Systèmes contenant une seule étoile . . . 103
4.1.1 Algorithme sans rencontres proches entre les planètes . . . 103
4.1.2 Algorithme avec rencontres proches entre les planètes . . . 112
4.1.3 Code . . . 114
4.1.4 Organigramme . . . 115
4.1.5 Validation . . . 117
4.2 Systèmes binaires . . . 117
4.2.1 Algorithme sans rencontres proches entre les planètes . . . 117
4.2.2 Algorithme avec rencontres proches entre les planètes . . . 121
4.2.3 Code . . . 122
4.2.4 Organigramme . . . 123
4.2.5 Validation . . . 125

5 Formation des systèmes de planètes géantes **131**
5.1 Migration de type II . . . 131
5.2 Modifications de SyMBA . . . 139
5.2.1 Code . . . 139
5.2.2 Organigramme . . . 140
5.3 Adaptation aux systèmes binaires . . . 143
5.3.1 Code . . . 143
5.3.2 Organigramme . . . 143
5.3.3 Validation . . . 145

6 Simulations **151**
6.1 Conditions initiales des systèmes planétaires . . . 151
6.2 Exemples d'évolutions . . . 152
6.3 Analyse des résultats . . . 159

Conclusion **165**

A Annexe : codes **171**
A.1 Coordonnées . . . 171
A.1.1 coord_posbin2h.f . . . 171
A.1.2 coord_posh2bin.f . . . 172
A.1.3 coord_vbin2h.f . . . 173
A.1.4 coord_vh2bin.f . . . 173
A.2 Intégrateur pour systèmes binaires . . . 175
A.2.1 symba7_helio_getacch_bin.f . . . 175
A.3 Influence du disque de gaz . . . 179
A.3.1 dissipative_kick_bin.f . . . 179

Première partie

MÉMOIRE : Développement d'un intégrateur symplectique pour les systèmes binaires

Introduction

Ce mémoire s'inscrit dans le cadre de la mécanique céleste et du problème des n corps. Il s'intéresse plus précisément aux systèmes binaires dans lesquels des planètes évoluent autour de deux étoiles en interaction gravitationnelle. Ces systèmes seront décrits plus en détail au chapitre 1.

L'objectif de ce travail est de créer un intégrateur numérique permettant d'étudier l'évolution de systèmes à n corps contenant deux objets massifs. Pour ce faire, nous utiliserons un intégrateur numérique existant connu sous le nom de SyMBA. Cet intégrateur fait partie du package d'intégration Swifter permettant de calculer les trajectoires de n corps au sein d'un système contenant un unique corps massif. Une adaptation de ce code aux systèmes binaires sera donc réalisée. En d'autres termes, le but est d'introduire un deuxième objet massif dans les systèmes manipulés par SyMBA et d'effectuer les changements nécessaires à une telle introduction dans le code. Les changements à effectuer sont explicités dans l'article [10] sur lequel nous nous basons pour ce travail.

De façon plus détaillée, dans la première partie de ce mémoire, nous nous intéresserons à ce que sont les étoiles binaires (chapitre 1) et nous présenterons le package Swifter et plus précisément l'algorithme SyMBA avant toute modification (chapitre 2). Ensuite, nous expliciterons les modifications effectuées afin d'incorporer un deuxième objet massif dans le code (chapitre 3). Pour terminer, nous exploiterons le code ainsi obtenu de deux manières. Premièrement, afin de valider notre code, nous réaliserons des simulations permettant de visualiser les mouvements des n corps et de comparer nos évolutions dynamiques à celles de la littérature. Deuxièmement, le code sera utilisé lors du stage de recherche pour étudier les processus de formation planétaire dans les systèmes binaires (partie II).

Chapitre 1

Systèmes binaires

Dans ce premier chapitre, nous commençons par présenter le sujet au centre du présent travail, à savoir les systèmes binaires. Il s'agit de systèmes de n corps célestes contenant une étoile binaire formée de deux étoiles en interaction gravitationnelle. Les étoiles binaires sont définies à la section 1.1 et leur classification explicitée à la section 1.2. Les mouvements décrits par les compagnons planétaires d'étoiles binaires font l'objet de la section 1.3. Notons que la plupart des informations reprises dans ce chapitre provient du site de l'observatoire de Paris ([19]).

1.1 Etoiles binaires

Une étoile binaire est un groupe de deux étoiles qui s'influencent, significativement, au travers de la force de gravité. Nous insistons sur le caractère significatif de l'interaction car toute paire de corps dans l'univers s'attire gravitationnellement. Cependant, la force gravitationnelle étant proportionnelle à $\frac{1}{r^2}$, où r est la distance entre les deux corps en question, lorsque les corps sont très éloignés, cette force est proche de zéro et n'aura donc pas d'influence perceptible.

Concernant la dénomination des étoiles dans les systèmes binaires, l'étoile la plus brillante sera appelée étoile A ou étoile primaire et la seconde sera nommée étoile B ou étoile secondaire. Cependant si les deux étoiles ont la même intensité lumineuse, l'étoile A sera celle qui, historiquement, a été découverte en premier.

Par ailleurs, remarquons qu'il ne faut pas généraliser les étoiles binaires aux étoiles doubles. Les étoiles doubles sont deux étoiles que nous voyons comme proches lorsque nous regardons le ciel la nuit, à l'oeil nu ou au télescope ([49]). Cependant, être proche dans le ciel ne signifie pas forcément être proche physiquement. En effet, par un jeu de perspective illustré à la figure 1.1, certaines étoiles doubles ne sont pas des étoiles binaires. Elles se nomment alors étoile double optique. Un exemple d'étoile double optique sont les étoiles α^1-capricorni et α^2-capricorni qui sont proches sur la voûte céleste, comme nous pouvons le constater à la figure 1.2, mais éloignées physiquement. Celles-ci sont situées, respectivement, à 690 et 109 années lumières de la Terre ([49]). Etant trop éloignées l'une de l'autre, elles ne peuvent donc pas être considérées comme une étoile binaire.

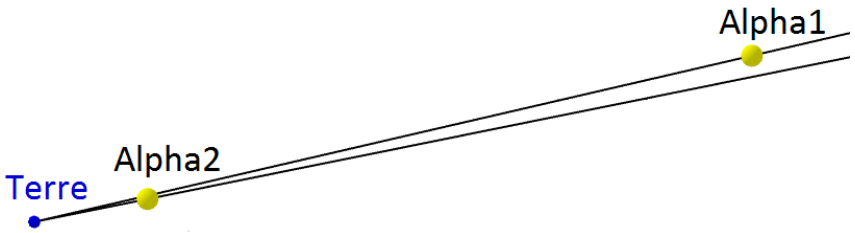


FIGURE 1.1 – Disposition schématique de α-capricorni qui sont proches sur la voûte céleste mais très éloignées l'une de l'autre physiquement

FIGURE 1.2 – Photo de l'étoile double α-capricorni (Source : [5])

Néanmoins certaines étoiles doubles sont bien des étoiles binaires. C'est, par exemple, le cas de Sirius. En effet cette étoile binaire est composée de Sirius A et Sirius B ([49]). Ces deux étoiles ont une distance moyenne de 19,5 unités astronomiques (UA), ce qui correspond à, approximativement, 2.8×10^9 kilomètres et sont donc suffisamment proches pour être considérées comme une étoile binaire. Nous pouvons observer cette étoile binaire à la figure 1.3.

Enfin, afin de motiver notre travail, notons que les scientifiques considèrent qu'approximativement la moitié des étoiles de l'univers sont des étoiles binaires ([17] et [38]). Les étudier nous paraît donc pertinent. Pour terminer, notons également que dans le cadre de ce travail nous étudions les systèmes binaires mais il existe également des systèmes multiples possédant plus de deux étoiles orbitant au sein du même système. C'est par exemple le cas de α-centauri qui est un système ternaire ([32]). En d'autres termes, ce système est composé de trois étoiles qui orbitent les unes autour des autres. Celles-ci sont entourées en jaune sur la figure 1.4.

FIGURE 1.3 – Photo de l'étoile binaire Sirius (Source : [4])

FIGURE 1.4 – Photo du système ternaire α-centauri (Source : [49])

1.2 Classification

Ce paragraphe explicite les différentes classes d'étoile binaire détaillées dans [19] et [49]. Cette classification est basée sur la manière dont nous pouvons observer ces étoiles dans le ciel et compte quatre classes :

- *binaire visuelle* : les deux étoiles sont visibles, à l'œil nu ou au télescope, sur la voûte céleste. Sirius est une étoile binaire visuelle comme nous pouvons le constater à la figure 1.3.

- *binaire astrométrique* : la détection des deux corps se fait grâce au mouvement induit d'une des deux étoiles par l'influence de l'autre. L'étoile semble trembler dans le ciel lorsque nous l'observons. Elle ne se déplace donc pas en ligne droite mais en sinusoïde sur la voûte céleste comme illustré à la figure 1.5.

- *binaire spectroscopique* : l'étoile binaire est repérée grâce à la variation de son spectre de lumière. Cette variation est engendrée par l'effet Doppler induit par le déplacement des deux étoiles l'une par rapport à l'autre. Cet effet induit un décalage dans le spectre de lumière en changeant régulièrement la fréquence de l'onde lumineuse. Ce décalage est illustré à la figure 1.6.

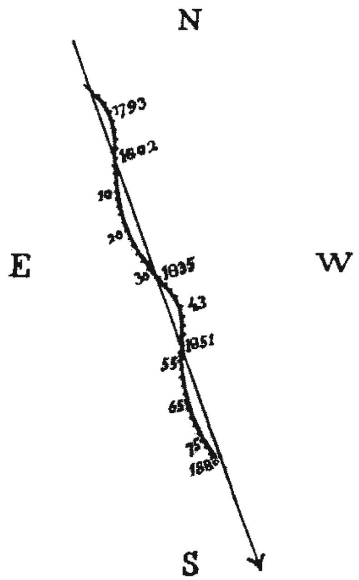


FIGURE 1.5 – Mouvement d'une binaire astrométrique dans le ciel (Source : [49])

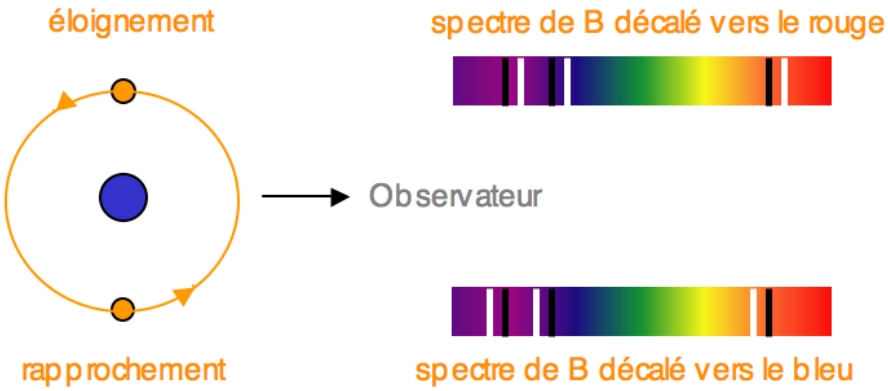


FIGURE 1.6 – Décalage dans le spectre de lumière dû au mouvement relatif du système binaire (Source : [49])

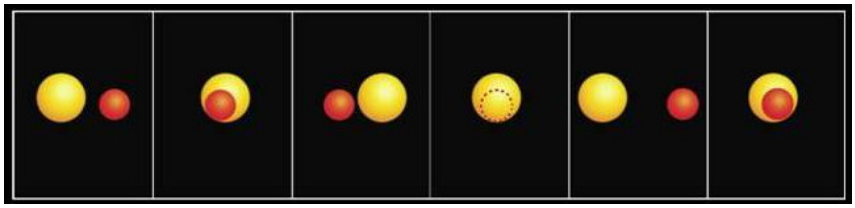


FIGURE 1.7 – Illustration du phénomène d'éclipse (Source : [43])

- *binaire à éclipse* : ce système est mis en évidence par l'éclipse d'une des deux étoiles du système binaire. En effet, la disposition du système est telle que, régulièrement, les deux étoiles du système binaire et la Terre s'alignent. Cet alignement engendre l'observation d'une éclipse depuis la Terre. Ce phénomène est illustré à la figure 1.7.

1.3 Compagnons planétaires

Intéressons-nous maintenant au mouvement des corps dans un système binaire. Commençons par nous pencher sur le mouvement des deux étoiles. Selon la première loi de Kepler, les deux étoiles ont un mouvement elliptique dont un des foyers est le centre de masse du système. Nous avons donc un mouvement similaire à celui représenté à la figure 1.8. De plus, dans le cas des systèmes binaires, le centre de masse se situe entre les deux étoiles tandis que dans le cas d'un système planétaire autour d'une étoile unique, le centre de masse se trouve généralement dans l'étoile. Cette différence est due au fait que les planètes sont de masse bien inférieure à celle des étoiles et ont, de ce fait, peu d'influence sur la détermination du centre de masse. Dans les systèmes formés d'une seule étoile, le mouvement s'effectue donc autour de l'étoile centrale

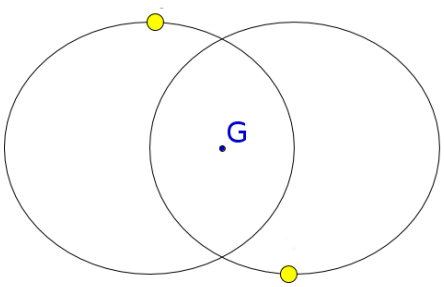
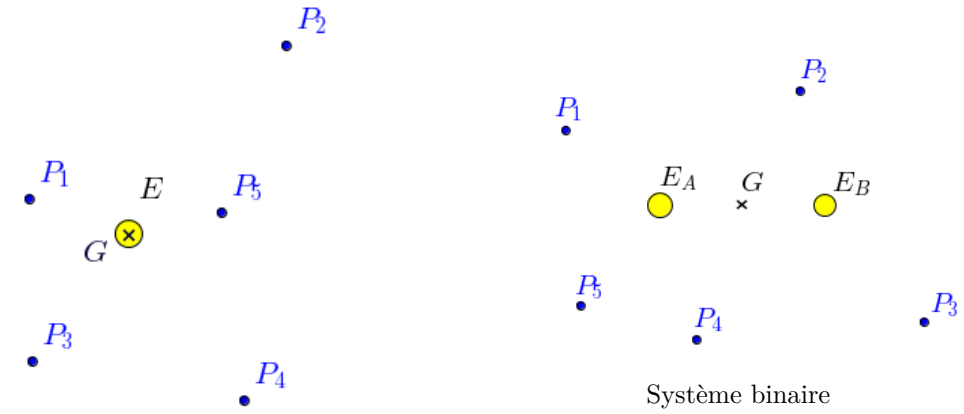


FIGURE 1.8 – Mouvement des composantes d'une étoile binaire



Système contenant une étoile

Système binaire

FIGURE 1.9 – Position du centre de masse G dans les systèmes à une ou deux étoiles

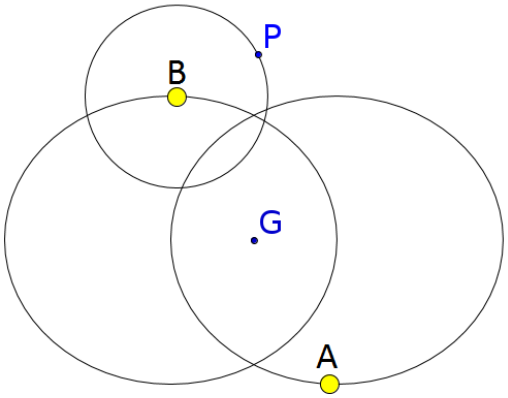


FIGURE 1.10 – Planète de type S

tandis que dans les systèmes binaires le mouvement se fera autour du centre de masse situé quelque part entre les deux étoiles. Ces deux configurations sont illustrées à la figure 1.9.

Nous allons maintenant nous intéresser au mouvement des planètes dans un tel système. En effet, il a été découvert récemment que certaines étoiles binaires abritent également des compagnons planétaires. Le mouvement de ces compagnons planétaires peut être de deux types ([44]). D'une part, les planètes de type S sont en mouvement autour d'une des deux étoiles. Ce type de mouvement est représenté à la figure 1.10. D'autre part, les planètes de type P sont des planètes en mouvement autour de deux étoiles. Cette disposition est représentée à la figure 1.11. Dans le cadre de ce mémoire, seules les planètes de type S seront étudiées.

Pour terminer, notons que de nombreux compagnons planétaires d'une étoile binaire ont été découverts par le télescope spatial Kepler de la NASA ([33]). Nous pouvons, par exemple, citer la planète Kepler-14b qui est une planète de type S et Kepler-16b qui est une planète de type P. De plus, jusqu'à présent, plus de 60 planètes de type S ont été découvertes comme le précise [41]. Pour plus d'informations, un catalogue d'exoplanètes peut être consulté à l'adresse `http://www.univie.ac.at/adg/schwarz/bincat_binary_star.html` ([45]) et les caractéristiques des planètes découvertes par le télescope Kepler sont disponibles à l'adresse `http://kepler.nasa.gov/Mission/discoveries` ([34]).

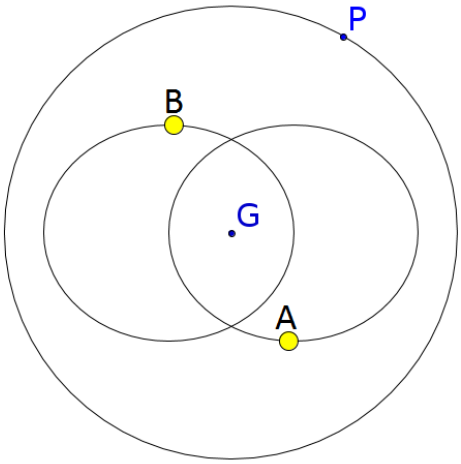


FIGURE 1.11 – Planète de type P

Chapitre 2

SyMBA

Dans ce chapitre, nous allons détailler l'intégrateur SyMBA contenu dans le package d'intégration Swifter. Celui-ci contient plusieurs intégrateurs permettant de calculer l'évolution de systèmes de n corps en interaction gravitationnelle avec un corps massif. Ceux-ci sont détaillés sur le site [21]. Une version simplifiée du code appelée SWIFT est également téléchargeable sur ce site.

Afin de présenter SyMBA, nous commencerons par expliciter ce que sont les intégrateurs symplectiques ainsi que les différents intégrateurs symplectiques présents dans le package Swifter (section 2.2). Nous présenterons ensuite les fonctions du package Swifter utilisées par SyMBA (section 2.3). L'étude des différents dossiers de Swifter utilisés par SyMBA se fera par ordre alphabétique. Pour terminer, nous réaliserons un organigramme afin de clairement identifier la structure de SyMBA et la façon dont les fonctions des différents dossiers interviennent dans le code (section 2.4). Commençons tout d'abord par introduire les différents jeux de coordonnées dont nous aurons besoin.

2.1 Jeux de cordonnées

Nous allons à présent détailler différents jeux de coordonnées qui sont utilisés dans le package Swifter. Nous nous basons sur [23] pour cette description.

2.1.1 Repère barycentrique

Dans ce jeu de coordonnées, le centre du repère est le barycentre ou centre de masse du système. Sa position est donnée par la formule

$$r_{CM} = \frac{1}{\sum_{j=1}^{n} m_j} \sum_{i=1}^{n} m_i r_i,$$

où r_i est la position du corps i dans un repère inertiel. La table 2.1 reprend les notations des vecteurs utilisées dans les différentes formules du chapitre.

Dans le repère barycentrique, chaque corps i est repéré au moyen d'un vecteur $\overrightarrow{OP_i}$ où O est le centre de masse du système et P_i le corps en question. Ce jeu de coordonnées est représenté à la figure 2.1. Mathématiquement, les positions dans ce repère s'expriment par

$$r_i^B = r_i - r_{CM} = r_i - \frac{1}{\sum_{k=1}^{n} m_k} \sum_{j=1}^{n} m_j r_j, \qquad i = 1...n.$$

a_i^H	accélération du corps i dans un repère héliocentrique
G	constante gravitationnelle
m_i	masse du corps i
$\tilde{p}_i$	quantité de mouvement du corps i dans un repère inertiel.
$\tilde{p}_i^B$	quantité de mouvement du corps i dans un repère barycentrique
$\tilde{p}_i^H$	quantité de mouvement du corps i dans un repère héliocentrique
p_i^B	moment associé du corps i dans un repère barycentrique
p_i^J	moment associé du corps i dans un repère de Jacobi
p_i^{HC}	moment associé du corps i dans un repère héliocentrique canonique
r_i	position du corps i dans un repère inertiel
r_i^B	position du corps i dans un repère barycentrique
r_i^H	position du corps i dans un repère héliocentrique
r_i^{HC}	position du corps i dans un repère héliocentrique canonique
r_i^J	position du corps i dans un repère de Jacobi
r_{ij}	position du corps i par rapport au corps j
v_i	vitesse du corps i dans un repère inertiel
v_i^B	vitesse du corps i dans un repère barycentrique
v_i^H	vitesse du corps i dans un repère héliocentrique
ω	vecteur vitesse angulaire
$a \wedge b$	produit vectoriel entre le vecteur a et le vecteur b
$a \cdot b$	produit scalaire entre le vecteur a et le vecteur b

TABLE 2.1 – Notations utilisées dans ce chapitre

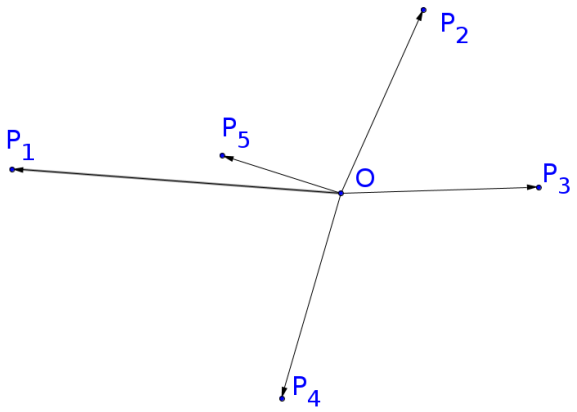


FIGURE 2.1 – Illustration d'un repère barycentrique avec O comme centre de masse

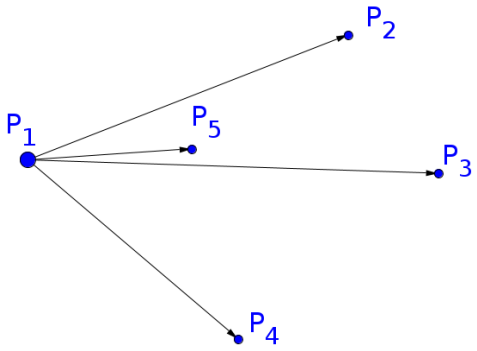


FIGURE 2.2 – Illustration d'un repère héliocentrique où P_1 est l'étoile du système planétaire

De plus, les moments associés sont les quantités de mouvement. Ainsi, par la définition de la quantité de mouvement, nous trouvons (la notation $\tilde{p}$ représentera dans la suite la quantité de mouvement, afin de distinguer cette quantité du moment conjugué p)

$$\begin{aligned}
\tilde{p}_i^B &= m_i v_i^B \\
&= m_i \dot{r}_i^B \\
&= m_i \frac{d}{dt}\left(r_i - \frac{1}{\sum_{k=1}^{n} m_k} \sum_{j=1}^{n} m_j r_j \right) \\
&= m_i \left(\dot{r}_i - \frac{1}{\sum_{k=1}^{n} m_k} \sum_{j=1}^{n} m_j \dot{r}_j \right) \\
&= m_i \dot{r}_i - \frac{m_i}{\sum_{k=1}^{n} m_k} \sum_{j=1}^{n} m_j \dot{r}_j .
\end{aligned}$$

Comme les moments associés sont égaux aux quantités de mouvement dans le repère barycentrique, nous avons

$$p_i^B = \tilde{p}_i - \frac{m_i}{\sum_{k=1}^{n} m_k} \sum_{j=1}^{n} \tilde{p}_j, \qquad i = 1...n. \tag{2.1.1}$$

Notons que ce jeu de coordonnées permet une formulation hamiltonienne du problème des n corps. En d'autres termes, la dynamique peut s'exprimer au moyen des équations d'Hamilton qui seront détaillées dans la section 2.2.

2.1.2 Repère héliocentrique

Dans ce repère, chacun des corps du système est repéré par rapport au corps le plus massif du système, à savoir l'étoile. Si cette dernière est notée P_1, tout corps P_i sera repéré par $\overrightarrow{P_1 P_i}$. Ce jeu de coordonnées est représenté à la figure 2.2. Mathématiquement, ce jeu de coordonnées s'exprime par

$$r_i^H = r_i - r_1, \qquad i = 1...n,$$

où r_1 est la position de l'étoile dans un repère inertiel.

De plus, comme précédemment, nous pouvons calculer les quantités de mouvement des corps :

$$\begin{aligned}
\tilde{p}_i^H &= m_i v_i^H \\
&= m_i \left(\dot{r}_i - \dot{r}_1 \right)
\end{aligned}$$

$$= m_i\dot{r}_i - m_i\dot{r}_1$$
$$= \tilde{p}_i - m_i v_1.$$

Malheureusement ces quantités de mouvement ne peuvent être utilisées comme moments associés. En effet, la transformation de coordonnées permettant de passer des coordonnées barycentriques (r_i^B, p_i^B) aux coordonnées héliocentriques $(r_i^H, \tilde{p}_i^H)$ n'est pas symplectique et ne conserve donc pas la structure hamiltonienne du système qui est primordiale dans la conception des intégrateurs symplectiques.

2.1.3 Repère héliocentrique canonique

Suite à la non-symplecticité de la transformation précédente, un nouveau jeu de coordonnées a été développé. Il s'agit des coordonnées héliocentriques canoniques. Dans ce système, la première coordonnées correspond au centre de masse du système et les autres corps sont représentés par les cordonnées héliocentriques telles que présentées précédemment. Nous avons alors le jeu de coordonnées suivant :

$$r_i^{HC} = \begin{cases} r_i - r_1 & \text{si } \ i = 2...n \\ \dfrac{1}{\sum\limits_{k=1}^{n} m_k} \sum\limits_{j=1}^{n} m_j r_j & \text{si } \ i = 1. \end{cases}$$

Par ailleurs, afin que la transformation soit canonique, les moments associés sont tels que :

$$p_i^{HC} = \begin{cases} \tilde{p}_i - \dfrac{m_i}{\sum\limits_{k=1}^{n} m_k} \sum\limits_{j=1}^{n} \tilde{p}_j & \text{si } \ i = 2...n \\ \sum\limits_{j=1}^{n} \tilde{p}_j & \text{si } \ i = 1. \end{cases}$$

La détermination des moments pour avoir une transformation canonique peut s'effectuer au moyen de fonctions génératrices. Plus de précisions peuvent être trouvées dans l'article [15]. Rappelons qu'une transformation canonique est nécessaire afin de garder la structure Hamiltonienne de notre problème ([28]). Cette structure aura toute son importance dans l'élaboration des intégrateurs symplectiques décrite à la section suivante.

Remarquons que les moments associés dans le repère héliocentrique canonique correspondent aux moments associés dans le repère barycentriques repris à la formule (2.1.1) pour les planètes et à la quantité de mouvement totale du système dans un repère inertiel pour la première coordonnée. Notons qu'ils sont, dans ce cas-ci, différents des quantités de mouvement dans le repère héliocentrique canonique.

2.1.4 Repère de Jacobi

Dans le repère de Jacobi, chaque corps est repéré par rapport au centre de masse des corps précédents. Le $n^{\text{ème}}$ corps P_n sera donc repéré par $\overrightarrow{O_{n-1}P_n}$ où O_{n-1} est le centre de masse des $n-1$ masses précédentes. Ce jeu de coordonnées est représenté à la figure 2.3.

Nous allons à présent exprimer ce système mathématiquement. Pour ce faire, nous définissons

$$\mu_i = \sum_{j=1}^{i} m_j. \tag{2.1.2}$$

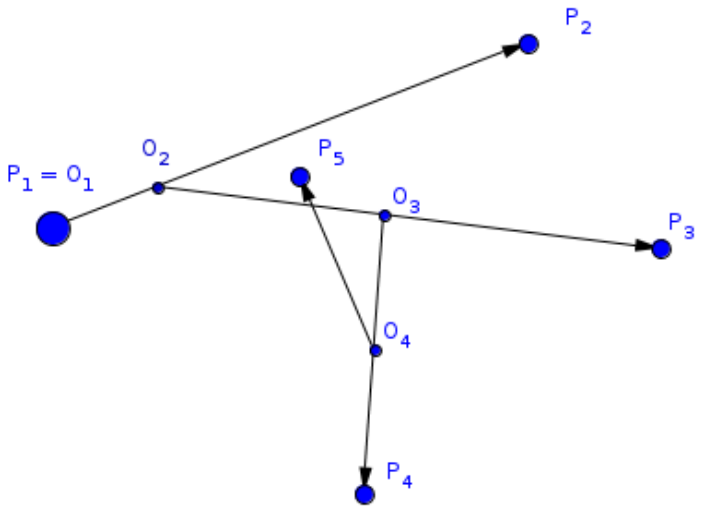


FIGURE 2.3 – Coordonnées de Jacobi

Le système de coordonnées s'exprime alors par

$$r_i^J = \begin{cases} r_i - \dfrac{1}{\mu_{i-1}} \displaystyle\sum_{j=1}^{i-1} m_j r_j & \text{si } \; i = 2...n \\ 0 & \text{si } \; i = 1. \end{cases}$$

Comme pour les coordonnées héliocentriques, considérer la quantité de mouvement comme moment associé ne donnera pas une transformation canonique. Les moments pour avoir une transformation canonique sont déterminés dans [23] et sont les suivants

$$p_i^J = \begin{cases} \dfrac{\mu_{i-1}}{\mu_i} m_i \dot{r}_i^J & \text{si } \; i = 2...n \\ 0 & \text{si } \; i = 1. \end{cases}$$

Les moments associés correspondent donc aux quantités de mouvement dans le repère de Jacobi multipliées par $\frac{\mu_{i-1}}{\mu_i}$.

Nous pouvons développer l'expression précédente

$$\begin{aligned} \frac{\mu_{i-1}}{\mu_i} m_i \dot{r}_i^J &= \frac{\mu_{i-1}}{\mu_i} m_i \frac{d}{dt} \left(r_i - \frac{1}{\mu_{i-1}} \sum_{j=1}^{i-1} m_j r_j \right) \\ &= \frac{\mu_{i-1}}{\mu_i} m_i \dot{r}_i - \frac{m_i}{\mu_i} \sum_{j=1}^{i-1} m_j \dot{r}_j \\ &= \frac{\mu_{i-1}}{\mu_i} \tilde{p}_i - \frac{m_i}{\mu_i} \sum_{j=1}^{i-1} \tilde{p}_j. \end{aligned}$$

Nous avons donc

$$p_i^J = \begin{cases} \dfrac{\mu_{i-1}}{\mu_i} \tilde{p}_i - \dfrac{m_i}{\mu_i} \displaystyle\sum_{j=1}^{i-1} \tilde{p}_j & \text{si } \; i = 2...n \\ 0 & \text{si } \; i = 1. \end{cases}$$

2.2 Intégrateurs symplectiques

Maintenant que nous avons introduit les différents jeux de coordonnées utilisés par Swifter, nous allons détailler la manière dont les équations du mouvement sont intégrées. Nous allons

commencer par présenter les intégrateurs symplectiques de manière générale pour ensuite préciser ceux présents dans Swifter.

2.2.1 Définition

Un intégrateur numérique permet de résoudre des équations différentielles ou, plus généralement, des équations aux dérivées partielles. Dans le cadre de ce mémoire, nous travaillons sur des systèmes évoluant dans le temps. En d'autres termes, les équations impliquent des dérivées temporelles. Afin de les résoudre, le programme discrétise le temps, c'est-à-dire, il résout les équations pour certains temps fixés. En effet, résoudre ces équations en tout temps n'est physiquement pas possible. Il fait donc évoluer le système pas à pas. Chaque solution en un temps donné est calculée sur base des solutions calculées précédemment. Mathématiquement, nous avons donc une fonction d'évolution de la forme

$$x_{k+1} = \Phi(x_k, x_{k-1}, ..., x_2, x_1, x_0), \tag{2.2.1}$$

où x_i est l'état du système au temps t_i et x_0 est l'état initial du système.

Par ailleurs, nos systèmes sont hamiltoniens. En d'autres mots, leur évolution est décrite par les équations d'Hamilton :

$$\begin{cases} \dfrac{dq_i}{dt} &= \dfrac{\partial \mathcal{H}}{\partial p_i} \\ \dfrac{dp_i}{dt} &= -\dfrac{\partial \mathcal{H}}{\partial q_i} \end{cases} \qquad i = 1...n, \tag{2.2.2}$$

où $\mathcal{H}$ est l'hamiltonien du système, q_i les coordonnées généralisées et p_i les moments associés à chaque q_i respectivement. Notons que puisque nous n'avons pas de forces de liaisons, les coordonnées généralisées sont simplement les coordonnées des corps dans un repère choisi. Remarquons ici toute l'importance d'avoir des transformations de coordonnées canoniques afin que le système reste hamiltonien et permette la suite des développements ci-dessous.

Dans notre cas, le programme n'utilise que l'état du système au temps précédent. L'équation (2.2.1) devient donc

$$(q_{k+1}, p_{k+1}) = \Phi(q_k, p_k),$$

où $q_i = \begin{pmatrix} q_{1,i} \\ q_{2,i} \\ \vdots \\ q_{n,i} \end{pmatrix}$ et $p_i = \begin{pmatrix} p_{1,i} \\ p_{2,i} \\ \vdots \\ p_{n,i} \end{pmatrix}$, avec $q_{j,i}$ et $p_{j,i}$, respectivement, la coordonnée généralisée du corps j et le moment associé au corps j au temps i .

De plus, dans ce mémoire, les systèmes traités sont conservatifs. En d'autres termes, la théorie nous dit que l'hamiltonien sera conservé au cours du temps, c'est-à-dire

$$\frac{d\mathcal{H}}{dt} = 0.$$

C'est pour ce type de systèmes que les intégrateurs symplectiques ont été développés (cfr [7]).

Un intégrateur sera dit symplectique si et seulement si

$$A^T J A = J,$$

où J est la matrice antisymétrique définie comme

$$J = \begin{pmatrix} 0 & I_n \\ -I_n & 0 \end{pmatrix},$$

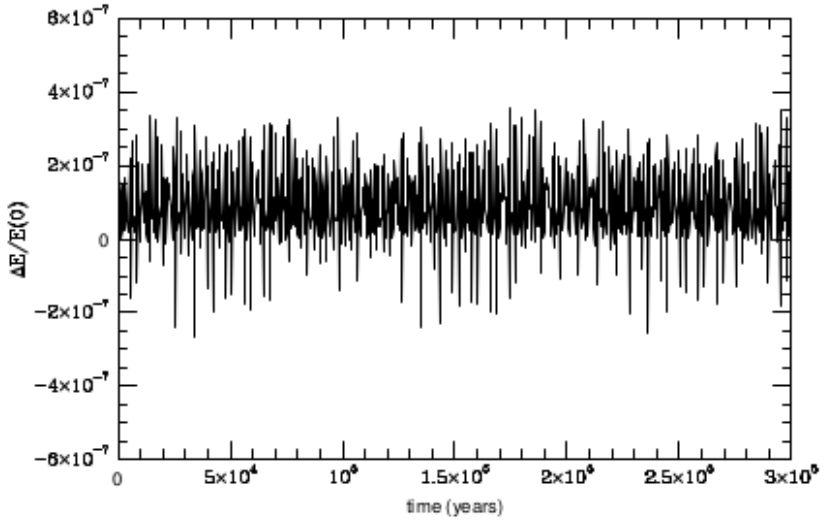


FIGURE 2.4 – Illustration de la conservation de l'hamiltonien lors d'une intégration avec un intégrateur symplectique (Source : [15])

où I_n est la matrice identité $n \times n$ et A est définie par composante comme

$$A_{ij} = \frac{\partial X_i}{\partial x_j},$$

avec $x = (p_k, q_k)$, $X = (p_{k+1}, q_{k+1})$, x_i est la composante i du vecteur x et X_j la composante j du vecteur X. Notons cependant que comme les intégrateurs fournissent des approximations numériques, ceux-ci ne gardent pas $\mathcal{H}$ exactement constant. Néanmoins, les intégrateurs symplectiques gardent l'erreur $\frac{\mathcal{H}_k - \mathcal{H}_0}{\mathcal{H}_0}$ bornée, où $\mathcal{H}_0$ est l'hamiltonien au temps initial et $\mathcal{H}_k$ est l'hamiltonien au temps k. Une telle conservation est représentée à la figure 2.4. Nous observons bien que l'erreur sur l'hamiltonien est bornée et maintenue faible.

2.2.2 Familles d'intégrateurs symplectiques

Nous allons à présent détailler la construction des intégrateurs symplectiques. La méthode que nous présentons [1] utilise les équations d'Hamilton (2.2.2), ainsi que les parenthèses de Poisson entre des quantités f et g :

$$\{f, g\} = \sum_{i=1}^{n} \frac{\partial f}{\partial q_i}\frac{\partial g}{\partial p_i} - \frac{\partial g}{\partial q_i}\frac{\partial f}{\partial q_i}. \tag{2.2.3}$$

Grâce aux équations d'Hamilton (2.2.2), nous allons pouvoir étudier l'évolution de n'importe quelle quantité f dépendant des coordonnées généralisées et des moments associés. Pour ce faire, nous allons travailler sur la variation de cette quantité au cours du temps, $\frac{df}{dt}$. Par le principe des dérivées en chaîne, nous exprimons la dérivée de f par rapport au temps comme suit

$$\frac{df}{dt} = \sum_{i=1}^{n} \frac{\partial f}{\partial q_i}\frac{\partial q_i}{\partial t} + \frac{\partial f}{\partial p_i}\frac{\partial p_i}{\partial t}.$$

Utilisons à présent les équations d'Hamilton décrites en (2.2.2) en considérant que p_i et q_i ne dépendent que du temps. Nous trouvons alors la relation suivante

$$\frac{df}{dt} = \sum_{i=1}^{n} \frac{\partial f}{\partial q_i}\frac{\partial \mathcal{H}}{p_i} - \frac{\partial f}{\partial p_i}\frac{\partial \mathcal{H}}{q_i},$$

1. Une deuxième méthode fortement répandue est de passer par des fonctions δ de Dirac et peut être trouvée dans [50].

qui se réécrit, en utilisant (2.2.3),

$$\frac{df}{dt} = \{f, \mathcal{H}\}. \tag{2.2.4}$$

Supposons à présent qu'il existe un opérateur linéaire T associé à l'hamiltonien tel que $Tf = \{f, \mathcal{H}\}$. Nous trouvons alors, en utilisant la relation précédente, que

$$\frac{df}{dt} = Tf.$$

Cette équation différentielle est une équation élémentaire dont la solution est donnée par

$$f(t) = e^{tT} f(0),$$

où $f(0)$ est la valeur de la quantité f au début de l'évolution.

Dans l'idée de simplifier cette expression, les intégrateurs symplectiques utilisent une décomposition de l'hamiltonien en morceaux qui peuvent être résolus séparément. Souvent, les intégrateurs symplectiques découpent l'hamiltonien en parties qui peuvent être résolues analytiquement, bien que cela ne soit pas obligatoire. Notons que si la résolution n'est pas exacte, la précision doit être supérieure à celle de l'intégration symplectique.

Dans le cas d'une décomposition de l'hamiltonien en deux parties, nous aurons

$$\mathcal{H} = \mathcal{H}_A + \mathcal{H}_B$$

et nous pouvons alors réécrire (2.2.4) comme

$$\frac{df}{dt} = \{f, \mathcal{H}_A + \mathcal{H}_B\}.$$

En appliquant la linéarité des parenthèses de Poisson, nous trouvons

$$\frac{df}{dt} = \{f, \mathcal{H}_A\} + \{f, \mathcal{H}_B\}.$$

Comme précédemment, nous définissons deux opérateurs linéaires T_A et T_B tels que $T_A f = \{f, \mathcal{H}_A\}$ et $T_B f = \{f, \mathcal{H}_B\}$ et trouvons

$$\frac{df}{dt} = T_A f + T_B f.$$

Si nous utilisons la définition de la somme d'opérateurs linéaires, nous avons

$$\frac{df}{dt} = (T_A + T_B)(f),$$

ce qui, comme précédemment, nous mène à une fonction exponentielle. Ainsi, f peut être calculé comme

$$f(t) = e^{t(T_A+T_B)} f(0). \tag{2.2.5}$$

Notons que nous avons effectué le raisonnement pour une quantité f quelconque dépendant des positions et des moments associés. En pratique, dans ce travail, cette quantité f sera les coordonnées généralisées et les moments associés.

Par ailleurs, l'évolution (2.2.5) n'est pas calculable aisément. Les intégrateurs symplectiques ont donc pour but de l'approximer au mieux. Cependant, l'approximation n'est pas directement effectuée sur cette expression. En effet, si nous approximons ce terme sans tenir compte de t, nous aurons des erreurs importantes pour des t élevés. Les intégrateurs vont donc, comme précisé

dans la section 2.2.2, réaliser une évolution pas à pas. Ils utilisent alors un pas d'intégration τ et $f(t)$ sera approximé par :

$$f(t) = e^{t(T_A+T_B)}f(0) = e^{\frac{t}{\tau}\tau(T_A+T_B)}f(0) \approx \Psi^{\left(\frac{t}{\tau}\right)}(f(0)), \tag{2.2.6}$$

où $\Psi(x)$ est une approximation de $e^{\tau(T_A+T_B)}x$.

Nous avons alors la relation de récurrence :

$$\begin{cases} f(t_{k+1}) = \Psi(f(t_k)) \\ f(t_0) = f_0 \end{cases} \quad k \in \mathbb{N}, \tag{2.2.7}$$

où f_0 est l'état initial du système et $t_{k+1} = t_k + \tau$. Et nous appliquons cette récurrence jusqu'à atteindre le temps t souhaité.

Les intégrateurs symplectiques approximent $e^{\tau(T_A+T_B)}$ par un produit de fonctions exponentielles. Le but est de trouver un produit d'exponentielles $e^{\tau_1 T_A}$ et $e^{\tau_2 T_B}$ de façon à approximer au mieux l'exponentielle $e^{\tau(T_A+T_B)}$. Il ne restera alors plus qu'à faire évoluer le système sous $\mathcal{H}_A$ ou sous $\mathcal{H}_B$ pour $e^{\tau_1 T_A}$ et $e^{\tau_2 T_B}$ respectivement.

Il existe plusieurs approximations possibles de ce terme par des produits débouchant sur différentes familles d'intégrateurs symplectiques. Nous présenterons ici les deux familles les plus utilisées : les intégrateurs de J. Laskar [24] et ceux de H. Yoshida [51]. Afin d'approximer $e^{\tau(T_A+T_B)}$ par un produit de fonctions exponentielles, ces deux familles d'intégrateurs utilisent la formule de Baker-Campbell-Haussdorff ([51]) :

$$e^A e^B = \exp\left\{A + B + \frac{1}{2}[A,B] + \frac{1}{12}([A,[A,B]] - [B,[A,B]]) + ...\right\}, \tag{2.2.8}$$

où $[A,B] = AB - BA$.

Famille d'intégrateurs de J. Laskar

J. Laskar et P. Robutel ont étudié dans [24] une décomposition de l'hamiltonien particulière. En effet, ils ont imposé que

$$\mathcal{H}_B = \epsilon \mathcal{H}_A,$$

avec

$$\epsilon << 1. \tag{2.2.9}$$

De tels hamiltoniens sont fréquents en théorie planétaire. Ainsi, ils ont négligé tous les termes d'ordre $\mathcal{O}(\epsilon^n)$ avec $n \geqslant 2$ dans les développements effectués sur base de la formule (2.2.8) et ont alors approximé $e^{\tau(\mathcal{H}_A+\mathcal{H}_B)}$ par quatre formules symétriques différentes :

$$\begin{array}{ll}
SABA_{2n} & e^{c_1\tau T_A}e^{d_1\tau T_B}e^{c_2\tau T_A}e^{d_2\tau T_B}...e^{d_n\tau T_B}e^{c_{n+1}\tau T_A}e^{d_n\tau T_B}...e^{d_2\tau T_B}e^{c_2\tau T_A}e^{d_1\tau T_B}e^{c_1\tau T_A} \\
SABA_{2n+1} & e^{c_1\tau T_A}e^{d_1\tau T_B}e^{c_2\tau T_A}e^{d_2\tau T_B}...e^{c_{n+1}\tau T_A}e^{d_{n+1}\tau T_B}e^{c_{n+1}\tau T_A}...e^{d_2\tau T_B}e^{c_2\tau T_A}e^{d_1\tau T_B}e^{c_1\tau T_A} \\
SBAB_{2n} & e^{d_1\tau T_B}e^{c_2\tau T_A}e^{d_2\tau T_B}e^{c_3\tau T_A}...e^{c_{n+1}\tau T_A}e^{d_{n+1}\tau T_B}e^{d_{n+1}\tau T_A}...e^{c_3\tau T_A}e^{d_2\tau T_B}e^{c_2\tau T_A}e^{d_1\tau T_B} \\
SBAB_{2n+1} & e^{d_1\tau T_B}e^{c_2\tau T_A}e^{d_2\tau T_B}e^{c_3\tau T_A}...e^{d_{n+1}\tau T_B}e^{c_{n+2}\tau T_A}e^{d_{n+1}\tau T_B}...e^{c_3\tau T_A}e^{d_2\tau T_B}e^{c_2\tau T_A}e^{d_1\tau T_B}
\end{array} \tag{2.2.10}$$

Ils ont ensuite, en développant les produits d'exponentielles au moyen de la formule (2.2.8) et en négligeant les termes d'ordre $\mathcal{O}(\epsilon^n)$ avec $n \geqslant 2$, trouvé les valeurs des coefficients c_i et d_i tels que les intégrateurs $SABA_i$ et $SBAB_i$ aient une erreur locale d'ordre $\mathcal{O}(\tau^{2i+1}\epsilon)$. Notons que l'erreur locale mentionnée ci-dessus est valable si nous considérons les termes proportionnels à ϵ^2 négligeables. L'ordre réel de l'erreur locale sera $\mathcal{O}(max(\tau^{2i+1}\epsilon, \tau^3\epsilon^2))$ comme le précise [24].

De plus, le nombre de pas nécessaires afin d'effectuer l'intégration totale étant proportionnel à $\frac{1}{\tau}$, l'erreur globale de l'intégrateur sera de l'ordre de l'erreur locale divisée par τ comme le

précise [10]. En effet, comme nous pouvons le constater à l'équation (2.2.6), si nous désirons effectuer une intégration pour un temps t_{tot}, nous appellerons l'intégrateur $\frac{t_{tot}}{\tau}$ fois et nous aurons donc une erreur globale de $\frac{t_{tot}}{\tau} \times$ *erreur locale*. De plus, t_{tot} étant une constante lors de l'intégration, l'ordre de l'erreur globale sera $\frac{1}{\tau} \times$ *ordre local*.

A noter que, par convention, nous appellerons l'ordre d'un intégrateur l'ordre de l'erreur globale de cet intégrateur $SABA_i$ est donc d'ordre simplifié $\mathcal{O}(\tau^{2i}\epsilon)$ et d'ordre réel $\mathcal{O}(max(\tau^{2i}\epsilon, \tau^2\epsilon^2)))$. De plus, nous adopterons la convention de notation :

$$\mathcal{O}(\tau^m) \stackrel{not}{\equiv} m.$$

Famille d'intégrateurs de H. Yoshida

H. Yoshida, pour sa part, a développé dans [51] des intégrateurs symplectiques sur base de l'algorithme Leapfrog qui consiste à utiliser l'approximation :

$$e^{\tau(T_A+T_B)} \approx e^{\frac{1}{2}\tau T_A} e^{\tau T_B} e^{\frac{1}{2}\tau T_A} \stackrel{not}{\equiv} Y_2(\tau),$$

qui est d'ordre 2 (notons que le calcul de l'ordre de cet intégrateur est aisé en utilisant la formule (2.2.8)). De plus, en comparant Y_2 aux intégrateurs de J. Laskar, nous constatons que Y_2 correspond à $SABA_2$ et $SBAB_2$.

Sur base de cela, H. Yoshida a construit ses intégrateurs de la manière récursive suivante :

$$Y_{2n+2}(\tau) = Y_{2n}(z_{1,2n}\tau)Y_{2n}(z_{0,2n}\tau)Y_{2n}(z_{1,2n}\tau), \qquad n \in \mathbb{N}. \tag{2.2.11}$$

Grâce à la formule (2.2.8), Yoshida a déterminé les coefficients $z_{0,2n}$ et $z_{1,2n}$ de manière à ce que Y_{2n+2} soit d'ordre $2n+2$. Il a alors trouvé

$$\begin{cases} z_{0,2n} = \dfrac{-2^{\frac{1}{2n+1}}}{2 - 2^{\frac{1}{2n+1}}} \\ z_{1,2n} = \dfrac{1}{2 - 2^{\frac{1}{2n+1}}}. \end{cases}$$

Comparaison des deux familles

La famille d'intégrateurs de J. Laskar permet d'obtenir des intégrateurs d'ordre $2i$ (si les termes en ϵ^2 sont négligeables) en un nombre d'évolutions par pas d'intégration de $2i+1$. En effet, en nous intéressant aux formules (2.2.10), nous constatons que $SABA_i$ nécessite $i+1$ évolutions sous $\mathcal{H}_A$ et i évolutions sous $\mathcal{H}_B$. Et pour $SBAB_i$, les rôles sont inversés et nous avons $i+1$ évolutions sous $\mathcal{H}_B$ et i évolutions sous $\mathcal{H}_A$. Dans les deux cas, le total donne bien $2i+1$ évolutions. Augmenter l'ordre des intégrateurs de J. Laskar demandera donc un nombre d'évolutions croissant de manière linéaire.

La famille d'intégrateurs de H. Yoshida, quant à elle, nécessite 3^i évolutions par pas d'intégration pour obtenir une intégration d'ordre $2i$. Nous pouvons aisément déduire le nombre d'opérations de la formule (2.2.11). Obtenir un ordre de précision élevé avec un intégrateur de H. Yoshida demandera donc un nombre d'évolutions croissant exponentiellement.

Dans un but d'efficacité de calcul, il nous apparaît donc préférable d'utiliser les intégrateurs de J. Laskar quand cela est possible et de se rabattre sur ceux de H. Yoshida dans le cas contraire. En d'autres termes, si la relation (2.2.9) est respectée, l'utilisation de la famille de J. Laskar a notre préférence.

Notons également que la famille d'intégrateurs de Yoshida demande des évolutions avec des pas de temps négatifs ce qui est moins stable numériquement comme le fait remarquer J. Laskar dans [24].

2.2.3 Décomposition en trois morceaux

Nous allons à présent nous intéresser aux intégrateurs présents dans Swifter. Ceux-ci utilisent l'intégrateur $SBAB_1$ ou $SABA_1$. Il vont donc approximer la quantité $f(t_{k+1})$ de l'équation (2.2.7) par

$$f(t_{k+1}) = \exp\left(\frac{\tau}{2}T_A\right)\exp\left(\tau T_B\right)\exp\left(\frac{\tau}{2}T_A\right)f(t_k), \qquad k \in \mathbb{N}, \tag{2.2.12}$$

où T_A et T_B sont les opérateurs linéaires associés respectivement à $\mathcal{H}_A$ et $\mathcal{H}_B$, les deux parties de l'hamiltonien.

L'équation (2.2.12) revient à faire évoluer f d'un demi-pas sous l'influence de $\mathcal{H}_\mathcal{A}$, ensuite d'un pas complet sous l'influence de $\mathcal{H}_\mathcal{B}$ et enfin à nouveau d'un demi-pas sous l'influence de $\mathcal{H}_\mathcal{A}$. Rappelons que cet intégrateur est d'ordre $\epsilon\tau^2$, avec ϵ le facteur de proportion entre les deux morceaux de notre hamiltonien. Notons que dans la séparation en deux morceaux les rôles de T_A et T_B peuvent être échangés. En effet, $SABA_1$ et $SBAB_1$ ont les mêmes coefficients c_i et d_i.

Par ailleurs, nous avons considéré jusqu'ici une décomposition de l'hamiltonien en deux parties mais Swifter utilise également des décompositions de l'hamiltonien en trois parties. Nous avons alors,

$$\mathcal{H} = \mathcal{H}_A + \mathcal{H}_B + \mathcal{H}_C.$$

L'évolution se fait alors au moyen du schéma explicité par (2.2.12) généralisé à trois parties, à savoir

$$f(t_{k+1}) = \exp\left(\frac{\tau}{2}T_A\right)\exp\left(\frac{\tau}{2}T_B\right)\exp\left(\tau T_C\right)\exp\left(\frac{\tau}{2}T_B\right)\exp\left(\frac{\tau}{2}T_A\right)f(t_k), \qquad k \in \mathbb{N}, \tag{2.2.13}$$

où T_A, T_B et T_C sont les opérateurs linéaires associés à $\mathcal{H}_A$, $\mathcal{H}_B$ et $\mathcal{H}_C$ respectivement.

Cet intégrateur consiste simplement à appliquer successivement l'approximation $SBAB_1$ à deux morceaux. En effet, nous l'appliquons une première fois à

$$e^{\tau(T_A+(T_B+T_C))}$$

et trouvons l'approximation

$$e^{\tau(T_A+T_B+T_C)} \approx \exp\left(\frac{\tau}{2}T_A\right)\exp\left(\tau(T_B+T_C)\right)\exp\left(\frac{\tau}{2}T_A\right).$$

Nous l'appliquons alors une deuxième fois et obtenons

$$e^{\tau(T_A+T_B+T_C)} \approx \exp\left(\frac{\tau}{2}T_A\right)\exp\left(\frac{\tau}{2}T_B\right)\exp\left(\tau T_C\right)\exp\left(\frac{\tau}{2}T_B\right)\exp\left(\frac{\tau}{2}T_A\right).$$

Dans un tel cas, l'intégrateur va donc d'abord faire avancer la quantité à évoluer d'un demi-pas par rapport à $\mathcal{H}_C$, ensuite d'un demi-pas par rapport à $\mathcal{H}_B$, puis d'un pas complet par rapport à $\mathcal{H}_A$, par après à nouveau d'un demi-pas par rapport à $\mathcal{H}_B$ et pour terminer d'un demi-pas par rapport à $\mathcal{H}_C$.

Notons que dans ce cas-ci l'intégrateur aura une erreur locale d'ordre $\mathcal{O}((\epsilon_1+\epsilon_2)\tau^3)$, avec $\epsilon_1 = \frac{\mathcal{H}_A}{\mathcal{H}_B+\mathcal{H}_C}$ et $\epsilon_2 = \frac{\mathcal{H}_C}{\mathcal{H}_B}$. En effet, nous appliquons deux fois un intégrateur avec une erreur locale d'ordre $\mathcal{O}(\epsilon\tau^3)$. Lors de la première application, nous aurons une erreur locale proportionnelle à $\epsilon_1\tau^3$ et la deuxième conduira à une erreur locale proportionnelle à $\epsilon_2\tau^3$. L'erreur locale de cet intégrateur sera donc proportionnelle à $(\epsilon_1+\epsilon_2)\tau^3$ et l'erreur globale sera proportionnelle à

$$(\epsilon_1+\epsilon_2)\frac{\tau^3}{\tau} = (\epsilon_1+\epsilon_2)\tau^2. \tag{2.2.14}$$

Notons également que pour avoir une bonne intégration, nous avons besoin des deux conditions

$$\epsilon_1 << 1 \quad \text{et} \quad \epsilon_2 << 1. \tag{2.2.15}$$

Nous allons à présent décrire les différents intégrateurs symplectiques présents dans Swifter qui correspondent à différents découpages de l'hamiltonien. Notons que, comme précédemment, le corps numéroté 1 sera l'étoile du système.

2.2.4 WHM

Le premier intégrateur que nous allons présenter est le mapping de Wisdom-Holman ([50]). Dans celui-ci, l'hamiltonien est divisé en deux parties et l'évolution se fait donc au moyen de la formule (2.2.12) où T_A et T_B sont les opérateurs linéaires associés à chaque partie de l'hamiltonien comme explicité dans la section 2.2.2. Pour l'intégrateur WHM, l'hamiltonien est décomposé de la manière suivante :

$$\mathcal{H} = \mathcal{H}^J_{Kep} + \mathcal{H}^J_{Int},$$

avec

$$\mathcal{H}^J_{Kep} = \sum_{i=2}^{n} \left(\frac{\left\| p_i^J \right\|^2}{2\tilde{m}_i} - \frac{G m_i m_1}{\| r_i^J \|} \right)$$

et

$$\mathcal{H}^J_{Int} = \sum_{i=2}^{n} G m_i m_1 \left(\frac{1}{\| r_i^J \|} - \frac{1}{\| r_{1i} \|} \right) - \sum_{1<i<j\leqslant n} \frac{G m_i m_j}{\| r_{ij} \|}.$$

où $\tilde{m}_i = \frac{\mu_{i-1}}{\mu_i} m_i$ avec μ_k défini à l'équation (2.1.2).

Notons que $\mathcal{H}^J_{Kep}$ est la partie de l'hamiltonien associée au mouvement képlérien des corps du système (interaction entre les corps et l'étoile) et $\mathcal{H}^J_{Int}$ correspond aux interactions entre les différents corps. Cette séparation est rendue possible par le passage aux coordonnées de Jacobi présentées à la section 2.1.4. En effet, comme explicité par [50], l'hamiltonien exprimé au moyen des coordonnées héliocentriques ou barycentriques n'est pas décomposable en une partie relative au mouvement képlérien et une partie relative aux interactions entre les différents corps, d'où l'introduction des coordonnées de Jacobi. En effet, la résolution du mouvement képlérien ayant été largement traitée dans la littérature, y recourir apparaît naturel. L'article [50] donne plus de précisions concernant cette décomposition.

Avec cette décomposition, la contrainte (2.2.9) sera vérifiée tant qu'il n'y a pas de rencontres proches entre les corps. En effet, en cas de rencontres proches, la partie $\frac{G m_i m_j}{\| r_{ij} \|}$ peut devenir extrêmement grande (puisque petit dénominateur) et la contrainte (2.2.9) n'est alors plus vérifiée. Afin de traiter ce problème, l'algorithme RMVS a été introduit dans le package (section 2.2.5).

WHM consiste ainsi en les schémas d'intégration suivants :

$$f(t_{k+1}) = \exp\left(\frac{\tau}{2} T^J_{int}\right) \exp\left(\tau T^J_{Kep}\right) \exp\left(\frac{\tau}{2} T^J_{int}\right) f(t_k), \quad k \in \mathbb{N}. \tag{2.2.16}$$

ou

$$f(t_{k+1}) = \exp\left(\frac{\tau}{2} T^J_{Kep}\right) \exp\left(\tau T^J_{Int}\right) \exp\left(\frac{\tau}{2} T^J_{Kep}\right) f(t_k), \quad k \in \mathbb{N}. \tag{2.2.17}$$

suivant l'utilisation de $SBAB_1$ ou $SABA_1$ respectivement.

Nous allons maintenant expliciter succinctement comment les évolutions sous chaque morceau de l'hamiltonien sont réalisées. Tout d'abord, l'évolution due à $\mathcal{H}^J_{Kep}$ revient à calculer les mouvements képlériens des corps. Pour un hamiltonien de la forme

$$\mathcal{H} = \frac{\| mv \|^2}{2m} - \frac{\nu m}{\| r \|}, \tag{2.2.18}$$

l'évolution peut être calculée au moyen des formules

$$\begin{cases} r(t_{k+1}) &= f(\nu)r(t_k) + g(\nu)v(t_k) \\ v(t_{k+1}) &= \dot{f}(\nu)r(t_k) + \dot{g}(\nu)v(t_k) \end{cases} \quad k \in \mathbb{N}, \tag{2.2.19}$$

où f et g sont les fonctions de Gauss. Plus d'informations concernant la détermination de f et g peuvent être trouvées dans le livre de J. Danby ([13]). Cette méthode permet de déterminer efficacement les nouvelles positions ainsi que les nouvelles vitesses.

Dans notre cas, l'hamiltonien est de la forme

$$\mathcal{H}^J_{Kep} = \sum_{i=2}^{n} \left(\frac{\left\| p_i^J \right\|^2}{2\tilde{m}_i} - \frac{Gm_i m_1}{\| r_i^J \|} \right),$$

avec $p_i^J = \tilde{m}_i v_i^J$ comme explicité à la section 2.1.4. La formule (2.2.19) peut alors être appliquée en utilisant

$$\nu = G\frac{m_i}{\tilde{m}_i} m_1 = G\frac{\mu_i}{\mu_{i-1}} m_1.$$

Notons que cette évolution peut être effectuée séparément pour chaque corps car chaque terme de la somme ne dépend que d'un unique corps. En effet, si nous posons

$$\mathcal{H}^J_{Kep,i} = \frac{\left\| p_i^J \right\|^2}{2\tilde{m}_i} - \frac{Gm_i m_1}{\| r_i^J \|}, \quad i = 2...n,$$

nous avons

$$\frac{\partial \mathcal{H}^J_{Kep}}{\partial r_i^J} = \frac{\partial \mathcal{H}^J_{Kep,i}}{\partial r_i^J} \text{ et } \frac{\partial \mathcal{H}^J_{Kep}}{\partial p_i^J} = \frac{\partial \mathcal{H}^J_{Kep,i}}{\partial p_i^J}, \quad i = 2...n,$$

car $\mathcal{H}^J_{Kep,i}$ est indépendant de p_j^J et r_j^J pour tous les corps en dehors du corps i. En terme d'équations d'Hamilton (2.2.2), nous avons alors

$$\begin{cases} \dfrac{dr_i^J}{dt} &= \dfrac{\partial \mathcal{H}^J_{Kep}}{\partial p_i^J} = \dfrac{\partial \mathcal{H}^J_{Kep,i}}{\partial p_i^J} \\ \dfrac{dp_i^J}{dt} &= -\dfrac{\partial \mathcal{H}^J_{Kep}}{\partial r_i^J} = -\dfrac{\partial \mathcal{H}^J_{Kep,i}}{\partial r_i^J} \end{cases} \quad i = 2...n,$$

Faire évoluer le système sous $\mathcal{H}_{Kep}$ est donc équivalent à faire évoluer chaque particule i sous $\mathcal{H}_{Kep,i}$ indépendamment. Nous pouvons donc appliquer la formule (2.2.19) à chaque particule séparément en utilisant

$$\nu = G\frac{m_i}{\tilde{m}_i} m_1.$$

Ensuite, grâce à la reformulation de l'hamiltonien au moyen des coordonnées de Jacobi, la partie relative à l'interaction entre les corps (i.e. $\mathcal{H}^J_{Int}$) n'est fonction que de la position des corps. L'évolution due à celle-ci peut, de ce fait, être calculée analytiquement. Ce calcul analytique s'effectue en passant par les coordonnées héliocentriques. Plus d'informations sur la méthode utilisée peuvent être trouvées dans la section 2.3.7.

2.2.5 RMVS

L'intégrateur RMVS, présenté dans [16], est une amélioration de l'intégrateur WHM décrit à la section précédente, qui permet de traiter les rencontres proches entre les planètes et les petits corps (astéroïdes dans [16]). En effet, en cas de rencontre, la partie $\mathcal{H}^J_{int}$ va augmenter et la condition (2.2.9) sera violée comme précisé précédemment. Cette rencontre sera traitée différemment suivant deux cas.

Si le petit corps se trouve dans la sphère de Hill de la planète, le code va simplement changer les coordonnées et au lieu de considérer l'étoile comme premier corps des coordonnées de Jacobi, il va considérer la planète comme premier corps. En effet, la sphère de Hill représente l'espace autour de la planète où l'interaction avec la planète est supérieure à celle avec l'étoile. Dans ce cas-là, en changeant les coordonnées, nous retrouvons une structure similaire de l'hamiltonien et de ses décompositions mais avec la propriété désirée (2.2.9). Notons que ce changement de coordonnées fonctionne car le petit corps est de masse négligeable en comparaison avec la masse de l'étoile et celle de la planète. Cette technique ne peut être appliquée pour les rencontres entre planètes.

Cependant ce changement de repère ne fonctionnera pas si le petit corps est proche de la planète mais pas dans la sphère de Hill de celle-ci. Dans ce cas-là, afin d'avoir une erreur locale et donc globale acceptable malgré l'augmentation de ϵ, le code va diminuer le pas d'intégration. Nous obtenons alors le schéma d'intégration suivant en cas de rencontre :

$$f(t_{k+1}) = \left[\exp\left(\frac{\tau}{2n}T^J_{int}\right)\exp\left(\frac{\tau}{n}T^J_{Kep}\right)\exp\left(\frac{\tau}{2n}T^J_{int}\right)\right]^n f(t_k), \qquad k \in \mathbb{N},$$

où n est un nombre naturel fixé préalablement. Nous réduisons donc le pas d'intégration lors de rencontres pour plus de précision. En effet, l'erreur locale étant proportionnelle à $\epsilon\tau^3$, si ϵ augmente, il suffit de diminuer le pas afin de garder l'erreur locale acceptable. Cela aura un coût en temps de calcul mais si les rencontres ne sont pas trop fréquentes, le temps total n'en sera pas trop affecté. Cependant, un intégrateur réduisant toujours le pas de la même façon lors d'une rencontre n'est pas optimal. Cette raison ainsi que l'absence de solution en cas de rencontre entre planètes ont poussé le développement du logiciel SyMBA (cfr section 2.2.7).

Les deux intégrateurs que nous venons de décrire n'étant pas modifiés lors de notre implémentation pour les systèmes binaires, nous en avons donné les grandes idées par complétude mais nous ne les détaillerons pas plus dans ce manuscrit. Plus d'informations peuvent être trouvées concernant WHM dans l'article [50] et dans l'article [16] concernant RMVS. Nous allons à présent détailler les deux intégrateurs que nous allons modifier pour les appliquer aux systèmes binaires.

2.2.6 Helio

L'intégrateur Helio décompose l'hamiltonien, exprimé en coordonnées héliocentriques canoniques, en trois parties et utilise alors l'équation (2.2.13) pour réaliser l'évolution. Notons que l'algorithme SyMBA qui nous intéresse est une modification de cet algorithme permettant les rencontres proches (cfr section suivante). Pour modifier SyMBA, nous devrons donc modifier également l'intégrateur Helio.

La décomposition de $\mathcal{H}$, l'hamiltonien en coordonnées héliocentriques canoniques, pour cet intégrateur est la suivante :

$$\mathcal{H} = \mathcal{H}^{HC}_{Int} + \mathcal{H}^{HC}_{Kep} + \mathcal{H}^{HC}_{Jump},$$

avec

$$\mathcal{H}^{HC}_{Kep} = \sum_{i=2}^{n}\left(\frac{\left\|p_i^{HC}\right\|^2}{2m_i} - \frac{Gm_1m_i}{\left\|r_i^{HC}\right\|}\right),$$

$$\mathcal{H}^{HC}_{Int} = -\sum_{i=2}^{n-1}\sum_{i<j\leqslant n}\frac{Gm_im_j}{\|r_{ij}\|} \tag{2.2.20}$$

et

$$\mathcal{H}^{HC}_{Jump} = \frac{1}{2m_1}\left\|\sum_{i=2}^{n} p_i^{HC}\right\|^2, \tag{2.2.21}$$

où $\|r_{ij}\| = \left\|r_i^{HC} - r_j^{HC}\right\|$.

Notons que le terme $\frac{\left\|p_1^{HC}\right\|^2}{2m_1}$ a été négligé dans l'hamiltonien comme le suggère [10]. En effet, ce terme fait évoluer le centre de masse du système suivant une trajectoire linéaire (la première coordonnée du repère héliocentrique canonique correspond à la position du centre de masse du système comme montré à la section 2.1.3). Afin de montrer cette propriété, nous calculons le mouvement induit par ce terme sur base des équations d'Hamilton :

$$\begin{cases} \dfrac{dr_{1,c}^{HC}}{dt} = \dfrac{\partial \mathcal{H}}{\partial p_{1,c}^{HC}} = \dfrac{\partial}{\partial p_{1,c}^{HC}} \left(\dfrac{\left\|p_1^{HC}\right\|^2}{2m_1} \right) \\ \dfrac{dp_{1,c}^{HC}}{dt} = -\dfrac{\partial \mathcal{H}}{\partial r_{1,c}^{HC}} = -\dfrac{\partial}{\partial r_{1,c}^{HC}} \left(\dfrac{\left\|p_1^{HC}\right\|^2}{2m_1} \right) \end{cases} \quad c = 1, 2, 3, \tag{2.2.22}$$

avec $r_1^{HC} = (r_{1,1}^{HC}, r_{1,2}^{HC}, r_{1,3}^{HC})$ et $p_1^{HC} = (p_{1,1}^{HC}, p_{1,2}^{HC}, p_{1,3}^{HC})$. c représente les directions x, y et z. Notons que cette notation pour les composantes avec l'indice c sera utilisée dans toute la suite du manuscrit. De plus, $\frac{\left\|p_{1,c}^{HC}\right\|^2}{2m_1}$ étant indépendant de $r_{1,c}^{HC}$, nous avons

$$\frac{\partial}{\partial r_{1,c}^{HC}} \left(\frac{\left\|p_1^{HC}\right\|^2}{2m_1} \right) = 0 \ , \quad c = 1, 2, 3.$$

D'autre part,

$$\begin{aligned} \frac{\partial}{\partial p_{1,c}^{HC}} \left(\frac{\left\|p_1^{HC}\right\|^2}{2m_1} \right) &= \frac{\partial}{\partial p_{1,c}^{HC}} \left(\frac{\sum_{a=1}^{3} \left(p_{1,a}^{HC}\right)^2}{2m_1} \right) \\ &= \frac{1}{2m_1} \sum_{a=1}^{3} \left(\frac{\partial}{\partial p_{1,c}^{HC}} \left(p_{1,a}^{HC}\right)^2 \right) \\ &= \frac{1}{2m_1} \sum_{a=1}^{3} \left(2p_{1,a}^{HC} \frac{\partial p_{1,a}^{HC}}{\partial p_{1,c}^{HC}} \right) \\ &= \frac{1}{2m_1} \sum_{a=1}^{3} \left(2p_{1,a}^{HC} \delta_{ac} \right) \\ &= \frac{p_{1,c}^{HC}}{m_1}. \end{aligned}$$

Le système (2.2.22) devient alors

$$\begin{cases} \dfrac{dr_{1,c}^{HC}}{dt} = \dfrac{p_{1,c}^{HC}}{m_1} \\ \dfrac{dp_{1,c}^{HC}}{dt} = 0 \end{cases} \quad c = 1, 2, 3$$

et s'exprime vectoriellement comme :

$$\begin{cases} \dfrac{dr_1^{HC}}{dt} = \dfrac{p_1^{HC}}{m_1} \\ \dfrac{dp_1^{HC}}{dt} = 0. \end{cases}$$

La solution de ce système est aisément calculable. En effet, par la deuxième équation p_1^{HC} est constant et la première équation se réduit donc à une dérivée égale à une constante. La solution du système est alors donnée par

$$\begin{cases} r_1^{HC}(t) = \dfrac{p_1^{HC}(0)}{m_1} t + r_1^{HC}(0) \\ p_1^{HC}(t) = p_1^{HC}(0). \end{cases}$$

Le centre de masse du système se déplacera donc bien comme une particule libre avec un mouvement linéaire uniforme. Or ce qui nous intéresse réellement est le mouvement des planètes par rapport à l'étoile afin d'étudier la dynamique du système et non le mouvement du centre de masse. De plus, comme nous venons de le voir, le terme $\frac{\left\|p_1^{HC}\right\|^2}{2m_1}$ influence le centre de masse mais n'influence nullement les autres corps. Il sera donc négligé.

Revenons à présent à l'évolution de notre système. En utilisant la décomposition de l'hamiltonien explicitée ci-dessus et la formule (2.2.13), nous aurons l'équation d'évolution :

$$f(t_{k+1}) = \exp\left(\frac{\tau}{2} T_{Jump}^{HC}\right) \exp\left(\frac{\tau}{2} T_{Int}^{HC}\right) \exp\left(\tau T_{Kep}^{HC}\right) \exp\left(\frac{\tau}{2} T_{Int}^{HC}\right) \exp\left(\frac{\tau}{2} T_{Jump}^{HC}\right) f(t_k), \qquad k \in \mathbb{N}.$$

Cette équation correspond à faire évoluer le système d'un demi-pas suivant $\mathcal{H}_{Jump}^{HC}$, ensuite d'un demi-pas suivant $\mathcal{H}_{Int}^{HC}$, suivi d'une évolution d'un pas complet suivant $\mathcal{H}_{Kep}^{HC}$, puis à nouveau d'un demi-pas suivant $\mathcal{H}_{Int}^{HC}$ et enfin d'un demi-pas suivant $\mathcal{H}_{Jump}^{HC}$. Nous allons donc à présent détailler l'évolution sous chacune des parties de l'hamiltonien.

Evolution sous $\mathcal{H}_{Kep}^{HC}$

Tout d'abord, l'évolution sous $\mathcal{H}_{Kep}^{HC}$ se calcule au moyen des fonctions de Gauss comme décrit dans la section 2.2.4. En effet, cette partie de l'hamiltonien correspond au mouvement képlérien du système en coordonnées héliocentriques canoniques. Il a donc une forme similaire à celle de l'équation (2.2.18).

En effet, en utilisant la propriété énoncée à la section 2.1.3 selon laquelle les moments associés dans le repère héliocentrique canonique sont égaux aux moments dans le repère barycentrique qui sont eux-même égaux à la quantité de mouvement dans le repère barycentrique, nous avons

$$p_{i,c}^{HC} = m_i v_{i,c}^{B}\ , \qquad i = 2, ...n, \qquad c = 1, 2, 3. \tag{2.2.23}$$

$\mathcal{H}_{Kep}^{HC}$ se réécrit alors

$$\mathcal{H}_{Kep}^{HC} = \sum_{i=2}^{n} \left(\frac{\left\|m_i v_i^B\right\|^2}{2m_i} - \frac{G m_1 m_i}{\left\|r_i^{HC}\right\|} \right).$$

Cet hamiltonien est donc bien de la forme explicitée à l'équation (2.2.18). L'évolution sous $\mathcal{H}_{Kep}^{HC}$ se fera donc au moyen des équations (2.2.19) en coordonnées héliocentriques canoniques avec $\nu = m_1$. Nous avons alors

$$\begin{cases} r_i^{HC}(t_{k+1}) &= f(\nu) r_i^{HC}(t_k) + g(\nu) v_i^B(t_k) \\ v_i^B(t_{k+1}) &= \dot{f}(\nu) r_i^{HC}(t_k) + \dot{g}(\nu) v_i^B(t_k) \end{cases} \qquad i = 2...n, \qquad k \in \mathbb{N}. \tag{2.2.24}$$

Notons que comme pour l'algorithme WHM, nous pouvons effectuer cette formule sur chaque corps indépendamment car chaque terme dans la somme de $\mathcal{H}_{Kep}^{HC}$ ne dépend que d'un corps. En effet, en utilisant un raisonnement similaire à celui de la section 2.2.4, nous pouvons montrer qu'évoluer le système sous $\mathcal{H}_{Kep}$ est équivalent à faire évoluer chaque corps indépendamment sous

$$\mathcal{H}_{Kep,i}^{HC} = \frac{\left\|p_i^{HC}\right\|^2}{2\tilde{m}_i} - \frac{G m_i m_1}{\left\|r_i^{HC}\right\|}, \qquad i = 2...n.$$

Evolution sous $\mathcal{H}_{Int}^{HC}$

Intéressons-nous à présent à l'évolution du système sous $\mathcal{H}_{Int}^{HC}$. Cette évolution s'exprime par les équations d'Hamilton (2.2.2). Nous avons alors :

$$\begin{cases} \dfrac{dr_{i,c}^{HC}}{dt} &= \dfrac{\partial \mathcal{H}_{Int}^{HC}}{\partial p_{i,c}^{HC}} \\ \dfrac{dp_{i,c}^{HC}}{dt} &= -\dfrac{\partial \mathcal{H}_{Int}^{HC}}{\partial r_{i,c}^{HC}} \end{cases} \qquad i = 2...n, \quad c = 1,2,3, \tag{2.2.25}$$

où c correspond aux positions x, y et z des corps. Pour rappel, $r_i^{HC} = (r_{i,1}^{HC}, r_{i,2}^{HC}, r_{i,3}^{HC})$ et $p_i^{HC} = (p_{i,1}^{HC}, p_{i,2}^{HC}, p_{i,3}^{HC})$. Comme $\mathcal{H}_{Int}^{HC}$ ne dépend d'aucun $p_{i,c}^{HC}$ (nous pouvons le constater à l'équation (2.2.20)), nous avons $\frac{\partial \mathcal{H}_{Int}^{HC}}{\partial p_{i,c}^{HC}} = 0$ pour tous les corps et toutes les directions. Il nous reste à calculer :

$$\begin{aligned}
-\frac{\partial \mathcal{H}_{Int}^{HC}}{\partial r_{i,c}^{HC}} &= -\frac{\partial}{\partial r_{i,c}^{HC}} \left(-\sum_{k=2}^{n-1} \sum_{k<j\leqslant n} \frac{G m_k m_j}{\|r_{kj}\|} \right) \\
&= \sum_{k=2}^{n-1} \sum_{k<j\leqslant n} \left[\frac{\partial}{\partial r_{i,c}^{HC}} \left(\frac{G m_k m_j}{\|r_{kj}\|} \right) \right] \\
&= \sum_{k=2}^{n-1} \sum_{k<j\leqslant n} \left[G m_k m_j \frac{\partial}{\partial r_{i,c}^{HC}} \left(\frac{1}{\left\| r_k^{HC} - r_j^{HC} \right\|} \right) \right] \\
&= \sum_{k=2}^{n-1} \sum_{k<j\leqslant n} \left[G m_k m_j \frac{\partial}{\partial r_{i,c}^{HC}} \left(\left(\sum_{a=1}^{3} \left(r_{k,a}^{HC} - r_{j,a}^{HC} \right)^2 \right)^{-\frac{1}{2}} \right) \right] \\
&= \sum_{k=2}^{n-1} \sum_{k<j\leqslant n} \left[G m_k m_j \left(-\frac{1}{2} \right) \left(\sum_{a=1}^{3} \left(r_{k,a}^{HC} - r_{j,a}^{HC} \right)^2 \right)^{-\frac{3}{2}} \frac{\partial}{\partial r_{i,c}^{HC}} \left(\sum_{a=1}^{3} \left(r_{k,a}^{HC} - r_{j,a}^{HC} \right)^2 \right) \right] \\
&= \sum_{k=2}^{n-1} \sum_{k<j\leqslant n} \left[G m_k m_j \left(-\frac{1}{2} \right) \frac{1}{\left\| r_k^{HC} - r_j^{HC} \right\|^3} \sum_{a=1}^{3} \left(2 \left(r_{k,a}^{HC} - r_{j,a}^{HC} \right) \frac{\partial}{\partial r_{i,c}^{HC}} \left(r_{k,a}^{HC} - r_{j,a}^{HC} \right) \right) \right] \\
&= \sum_{k=2}^{n-1} \sum_{k<j\leqslant n} \left[G m_k m_j \left(-\frac{1}{2} \right) \frac{1}{\left\| r_k^{HC} - r_j^{HC} \right\|^3} \sum_{a=1}^{3} \left(2 \left(r_{k,a}^{HC} - r_{j,a}^{HC} \right) \left(\delta_{ik}\delta_{ca} - \delta_{ij}\delta_{ca} \right) \right) \right] \\
&= \sum_{k=2}^{n-1} \sum_{k<j\leqslant n} \left[-\frac{G m_k m_j}{\left\| r_k^{HC} - r_j^{HC} \right\|^3} \left(r_{k,c}^{HC} - r_{j,c}^{HC} \right) \left(\delta_{ik} - \delta_{ij} \right) \right] \\
&= \sum_{2\leqslant k<j\leqslant n} \left[-\frac{G m_k m_j}{\left\| r_k^{HC} - r_j^{HC} \right\|^3} \left(r_{k,c}^{HC} - r_{j,c}^{HC} \right) \delta_{ik} \right] \\
&\qquad + \sum_{2\leqslant k<j\leqslant n} \left[-\frac{G m_k m_j}{\left\| r_k^{HC} - r_j^{HC} \right\|^3} \left(r_{k,c}^{HC} - r_{j,c}^{HC} \right) \left(-\delta_{ij} \right) \right] \\
&= \sum_{i<j\leqslant n} \left[-\frac{G m_i m_j}{\left\| r_i^{HC} - r_j^{HC} \right\|^3} \left(r_{i,c}^{HC} - r_{j,c}^{HC} \right) \right] + \sum_{2\leqslant k<i} \left[\frac{G m_k m_i}{\left\| r_k^{HC} - r_i^{HC} \right\|^3} \left(r_{k,c}^{HC} - r_{i,c}^{HC} \right) \right]
\end{aligned}$$

$$= \sum_{i<j\leqslant n} \left[-\frac{Gm_i m_j}{\left\| r_i^{HC} - r_j^{HC} \right\|^3} \left(r_{i,c}^{HC} - r_{j,c}^{HC} \right) \right] + \sum_{2\leqslant k<i} \left[-\frac{Gm_k m_i}{\| r_i^{HC} - r_k^{HC} \|^3} \left(r_{i,c}^{HC} - r_{k,c}^{HC} \right) \right]$$

$$= -\sum_{j=2, j\neq i}^{n} \left[\frac{Gm_i m_j}{\left\| r_i^{HC} - r_j^{HC} \right\|^3} \left(r_{i,c}^{HC} - r_{j,c}^{HC} \right) \right].$$

Sur base des dérivées de l'hamiltonien, le système d'équations (2.2.25) devient

$$\begin{cases} \frac{dr_{i,c}^{HC}}{dt} &= 0 \\ \frac{dp_{i,c}^{HC}}{dt} &= -\sum\limits_{j=2, j\neq i}^{n} \left[\frac{Gm_i m_j}{\| r_i^{HC} - r_j^{HC} \|^3} \left(r_{i,c}^{HC} - r_{j,c}^{HC} \right) \right] \end{cases} \qquad i = 2...n, \quad c = 1, 2, 3. \tag{2.2.26}$$

De la première égalité, nous déduisons que la position reste constante pour tous les corps. Le membre de droite de la deuxième égalité est donc une constante. Le système se réduit alors à un système d'équations différentielles élémentaires dont la solution est aisément calculable :

$$\begin{cases} r_{i,c}^{HC}(t_{k+1}) &= r_{i,c}^{HC}(t_k) \\ p_{i,c}^{HC}(t_{k+1}) &= p_{i,c}^{HC}(t_k) - \tau \sum\limits_{j=2, j\neq i}^{n} \left[\frac{Gm_i m_j}{\| r_i^{HC}(t_k) - r_j^{HC}(t_k) \|^3} \left(r_{i,c}^{HC}(t_k) - r_{j,c}^{HC}(t_k) \right) \right] \end{cases} \begin{array}{l} c = 1, 2, 3, \\ i = 2...n, \\ k \in \mathbb{N}, \end{array} \tag{2.2.27}$$

où $r_{i,c}^{HC}(t)$ est la coordonnée c de la position du corps i en coordonnées héliocentriques canoniques au temps t et $p_{i,c}^{HC}(t)$ est la coordonnée c du moment associé du corps i en coordonnées héliocentriques canoniques au temps t.

Cependant, le code Helio va faire évoluer les vitesses plutôt que les moments. En utilisant l'équation (2.2.23) et en divisant les équations relatives aux moments par m_i dans le système (2.2.27), nous trouvons les équations d'évolutions

$$\begin{cases} r_{i,c}^{HC}(t_{k+1}) &= r_{i,c}^{HC}(t_k) \\ v_{i,c}^{B}(t_{k+1}) &= v_{i,c}^{B}(t_k) - \tau \sum\limits_{j=2, j\neq i}^{n} \left[\frac{Gm_j}{\| r_i^{HC}(t_k) - r_j^{HC}(t_k) \|^3} \left(r_{i,c}^{HC}(t_k) - r_{j,c}^{HC}(t_k) \right) \right] \end{cases} \begin{array}{l} c = 1, 2, 3, \\ i = 2...n, \\ k \in \mathbb{N}. \end{array} \tag{2.2.28}$$

Remarquons que $\sum\limits_{j=2, j\neq i}^{n} \left[\frac{Gm_j}{\| r_i^{HC}(t_k) - r_j^{HC}(t_k) \|^3} \left(r_{i,c}^{HC}(t_k) - r_{j,c}^{HC}(t_k) \right) \right]$ ne modifie que les vitesses barycentriques. Il s'agit donc d'une accélération barycentrique due à $\mathcal{H}_{Int}^{HC}$.

Evolution sous $\mathcal{H}_{Jump}^{HC}$

Il ne nous reste plus qu'à déterminer l'évolution sous $\mathcal{H}_{Jump}^{HC}$. Les équations d'Hamilton (2.2.2) deviennent sous $\mathcal{H}_{Jump}^{HC}$:

$$\begin{cases} \frac{dr_{i,c}^{HC}}{dt} &= \frac{\partial \mathcal{H}_{Jump}^{HC}}{\partial p_{i,c}^{HC}} \\ \frac{dp_{i,c}^{HC}}{dt} &= -\frac{\partial \mathcal{H}_{Jump}^{HC}}{\partial r_{i,c}^{HC}} \end{cases} \qquad i = 2...n, \quad c = 1, 2, 3. \tag{2.2.29}$$

$\mathcal{H}_{Jump}^{HC}$ ne dépendant d'aucune position comme nous pouvons le constater dans l'équation (2.2.21), nous avons

$$\frac{\partial \mathcal{H}_{Jump}^{HC}}{\partial r_{i,c}^{HC}} = 0 \quad i = 2...n, \quad c = 1, 2, 3.$$

Il nous reste alors à calculer :

$$\begin{aligned}
\frac{\partial \mathcal{H}_{Jump}^{HC}}{\partial p_{i,c}^{HC}} &= \frac{\partial}{\partial p_{i,c}^{HC}} \left[\frac{1}{2m_1} \left\| \sum_{j=2}^{n} p_j^{HC} \right\|^2 \right] \\
&= \frac{\partial}{\partial p_{i,c}^{HC}} \left[\frac{1}{2m_1} \sum_{a=1}^{3} \left[\sum_{j=2}^{n} p_{j,a}^{HC} \right]^2 \right] \\
&= \frac{1}{2m_1} \sum_{a=1}^{3} \left(\frac{\partial}{\partial p_{i,c}^{HC}} \left[\sum_{j=2}^{n} p_{j,a}^{HC} \right]^2 \right) \\
&= \frac{1}{2m_1} \sum_{a=1}^{3} \left(2 \left[\sum_{j=2}^{n} p_{j,a}^{HC} \right] \frac{\partial}{\partial p_{i,c}^{HC}} \left[\sum_{k=2}^{n} p_{k,a}^{HC} \right] \right) \\
&= \frac{1}{2m_1} \sum_{a=1}^{3} \left(2 \left[\sum_{j=2}^{n} p_{j,a}^{HC} \right] \sum_{k=2}^{n} \frac{\partial p_{k,a}^{HC}}{\partial p_{i,c}^{HC}} \right) \\
&= \frac{1}{2m_1} \sum_{a=1}^{3} \left(2 \left[\sum_{j=2}^{n} p_{j,a}^{HC} \right] \sum_{k=2}^{n} [\delta_{ik}\delta_{ac}] \right) \\
&= \frac{1}{m_1} \sum_{a=1}^{3} \left(\left[\sum_{j=2}^{n} p_{j,a}^{HC} \right] \delta_{ac} \right) \\
&= \frac{1}{m_1} \left[\sum_{j=2}^{n} p_{j,c}^{HC} \right]. \qquad (2.2.30)
\end{aligned}$$

Sur base des dérivées de l'hamiltonien, le système d'équations (2.2.29) devient :

$$\begin{cases} \frac{dr_{i,c}^{HC}}{dt} &= \frac{1}{m_1} \left[\sum_{j=2}^{n} p_{j,c}^{HC} \right] \\ \frac{dp_{i,c}^{HC}}{dt} &= 0 \end{cases} \qquad i = 2...n, \quad c = 1, 2, 3.$$

Par la deuxième égalité, nous déduisons que $p_{i,c}^{HC}$ est constant pour tous les corps suivant tous les axes. La première égalité se réduit alors à une dérivée égale à une constante et se résout aisément. Nous obtenons alors les équations d'évolutions :

$$\begin{cases} r_{i,c}^{HC}(t_{k+1}) &= r_{i,c}^{HC}(t_k) + \tau \frac{1}{m_1} \left[\sum_{j=2}^{n} p_{j,c}^{HC}(t_k) \right] \\ p_{i,c}^{HC}(t_{k+1}) &= p_{i,c}^{HC}(t_k) \end{cases} \qquad i = 2...n, \quad c = 1, 2, 3, \quad k \in \mathbb{N}.$$

Et si nous utilisons l'équation (2.2.23), nous avons les équations d'évolution exprimées en terme de vitesses :

$$\begin{cases} r_{i,c}^{HC}(t_{k+1}) &= r_{i,c}^{HC}(t_k) + \tau \frac{1}{m_1} \left[\sum_{j=2}^{n} m_j v_{j,c}^{B}(t_k) \right] \\ v_{i,c}^{B}(t_{k+1}) &= v_{i,c}^{B}(t_k) \end{cases} \qquad i = 2...n, \quad c = 1, 2, 3, \quad k \in \mathbb{N}. \quad (2.2.31)$$

En utilisant l'équation d'évolution (2.2.6) et les évolutions analytiques sous les différentes parties de l'hamiltonien, nous sommes capable de faire évoluer le système tout entier. Néanmoins, un problème va survenir lors de la rencontre entre deux corps. Dans une telle configuration, $\mathcal{H}_{Int}^{HC}$ peut devenir arbitrairement grand. En effet, $\mathcal{H}_{Int}^{HC}$ a ses termes proportionnels à l'inverse de la distance entre les corps comme nous pouvons le constater à l'équation (2.2.20). Les conditions (2.2.15) seront alors violées. Ce problème a poussé le développement de l'intégrateur SyMBA que nous présentons dans la prochaine section.

2.2.7 SyMBA

L'intégrateur SyMBA est à l'intégrateur Helio ce que l'intégrateur RMVS est à l'intégrateur WHM. Il est donc une amélioration de l'intégrateur Helio permettant de gérer les rencontres proches entre les corps. Notons que, contrairement à l'intégrateur RMVS, celui-ci parvient à gérer toutes les rencontres proches, même celles entre planètes.

En cas de rencontre proche, le terme qui pose souci est le terme $\mathcal{H}_{Int}^{HC}$. En effet, dans ce cas-là, celui-ci possède certains termes qui vont devenir déraisonnablement grands et vont donc entraîner la violation des conditions (2.2.15) de sorte que $\mathcal{H}_{Int}^{HC}$ va avoir un ordre de grandeur comparable ou supérieur à celui de $\mathcal{H}_{Kep}^{HC}$. Afin de garder l'erreur globale d'ordre $\epsilon\tau^2$ petite, en cas de rencontre, le pas sera alors réduit afin de compenser l'augmentation de ϵ. Néanmoins cette réduction se fera en gardant une intégration symplectique comme le précise [15]. L'hamiltonien sera alors décomposé en plusieurs termes et les différentes parties seront intégrées selon un pas optimal.

Dans un premier temps, $\mathcal{H}$ sera divisé en :

$$\mathcal{H} = \mathcal{H}_{Jump}^{HC} + \mathcal{H}_{IntNR}^{HC} + (\mathcal{H}_{IntR}^{HC} + \mathcal{H}_{Kep}^{HC}),$$

où $\mathcal{H}_{IntR}^{HC}$ est la partie de $\mathcal{H}_{Int}^{HC}$ comprenant les corps ayant une rencontre proche et $\mathcal{H}_{IntNR}^{HC} = \mathcal{H}_{Int}^{HC} - \mathcal{H}_{IntR}^{HC}$. Mathématiquement ces termes s'expriment :

$$\mathcal{H}_{IntR}^{HC} = -\sum_{i=2}^{n-1} \sum_{i<j\leqslant n} \left(\frac{Gm_i m_j}{\|r_{ij}\|} H(\|r_{ij}\|) \right)$$

et

$$\begin{aligned} \mathcal{H}_{IntNR}^{HC} = \mathcal{H}_{Int}^{HC} - \mathcal{H}_{IntR}^{HC} &= -\sum_{i=2}^{n-1} \sum_{i<j\leqslant n} \frac{Gm_i m_j}{\|r_{ij}\|} + \sum_{i=2}^{n-1} \sum_{i<j\leqslant n} \left(\frac{Gm_i m_j}{\|r_{ij}\|} H(\|r_{ij}\|) \right) \\ &= -\sum_{i=2}^{n-1} \sum_{i<j\leqslant n} \left(\frac{Gm_i m_j}{\|r_{ij}\|} (1 - H(\|r_{ij}\|)) \right) \end{aligned}$$

avec

$$H(r) = \begin{cases} 0 & \text{si } r > r_{crit} \\ 1 & \text{sinon} \end{cases} \tag{2.2.32}$$

où r_{crit} est la distance en dessous de laquelle les corps seront considérés comme ayant une rencontre proche.

L'évolution sous $\mathcal{H}$ se fera alors en utilisant l'équation (2.2.13) et nous aurons :

$$\begin{aligned} f(t_{k+1}) = \exp\left(\frac{\tau}{2} T_{Jump}^{HC}\right) \exp\left(\frac{\tau}{2} T_{IntNR}^{HC}\right) \exp\left(\tau \left[T_{Kep}^{HC} + T_{IntR}^{HC}\right]\right) \\ \exp\left(\frac{\tau}{2} T_{IntNR}^{HC}\right) \exp\left(\frac{\tau}{2} T_{Jump}^{HC}\right) f(t_k), \quad k \in \mathbb{N}. \end{aligned} \tag{2.2.33}$$

Il nous reste à décrire l'évolution sous les différentes parties de l'hamiltonien. $\mathcal{H}_{Jump}^{HC}$ n'ayant pas été modifié, son évolution peut être effectuée en utilisant la formule (2.2.31) de la section 2.2.6.

Evolution sous $\mathcal{H}_{IntNR}^{HC}$

L'évolution sous $\mathcal{H}_{IntNR}^{HC}$ est également aisée. En effet, sa dérivée par rapport à $r_{i,c}^{HC}$ est similaire à celle de $\mathcal{H}_{Int}^{HC}$. Plus précisément, nous avons

$$-\frac{\partial \mathcal{H}_{IntNR}^{HC}}{\partial r_{i,c}^{HC}} = -\frac{\partial}{\partial r_{i,c}^{HC}} \left(-\sum_{k=2}^{n-1} \sum_{k<j\leqslant n} \frac{Gm_k m_j}{\|r_{kj}\|} H(\|r_{kj}\|) \right) = \sum_{k=2}^{n-1} \sum_{k<j\leqslant n} \left[\frac{\partial}{\partial r_{i,c}^{HC}} \left(\frac{Gm_k m_j}{\|r_{kj}\|} \right) H(\|r_{kj}\|) \right].$$

Notons que nous avons utilisé ici le fait que $H'(r) = 0$ pour tout r différent de r_{crit}. Ensuite, en utilisant le même raisonnement que celui effectué à la section 2.2.6, nous avons

$$-\frac{\partial \mathcal{H}_{IntNR}^{HC}}{\partial r_{i,c}^{HC}} = -\sum_{j=2,j\neq i}^{n} \left[\frac{G m_i m_j}{\left\| r_i^{HC} - r_j^{HC} \right\|^3} \left(r_{i,c}^{HC} - r_{j,c}^{HC} \right) H(\|r_{kj}\|) \right].$$

Par ailleurs, $\mathcal{H}_{IntNR}^{HC}$ est indépendant des moments. Nous avons donc

$$\frac{\partial \mathcal{H}_{IntNR}^{HC}}{\partial p_{i,c}^{HC}} = 0.$$

Les équations d'évolution pour $\mathcal{H}_{IntNR}^{HC}$ sont donc similaires à celles pour $\mathcal{H}_{Int}^{HC}$ (2.2.26) :

$$\begin{cases} \frac{dr_{i,c}^{HC}}{dt} &= 0 \\ \frac{dp_{i,c}^{HC}}{dt} &= -\sum\limits_{j=2,j\neq i}^{n} \left[\frac{G m_i m_j}{\left\| r_i^{HC} - r_j^{HC} \right\|^3} \left(r_{i,c}^{HC} - r_{j,c}^{HC} \right) H(\|r_{ij}\|) \right] \end{cases} \quad i = 2...n, \quad c = 1,2,3.$$

Par un raisonnement analogue à celui de la section 2.2.6, nous trouvons les équations d'évolution :

$$\begin{cases} r_{i,c}^{HC}(t_{k+1}) &= r_{i,c}^{HC}(t_k) \\ v_{i,c}^{B}(t_{k+1}) &= v_{i,c}^{B}(t_k) - \tau \sum\limits_{j=2,j\neq i}^{n} \left[\frac{G m_j}{\left\| r_i^{HC}(t_k) - r_j^{HC}(t_k) \right\|^3} \left(r_{i,c}^{HC}(t_k) - r_{j,c}^{HC}(t_k) \right) H(\|r_{ij}\|) \right] \end{cases} \begin{array}{l} c = 1,2,3, \\ i = 2...n, \\ k \in \mathbb{N}, \end{array}$$

qui sont donc similaires aux équations (2.2.28).

Evolution sous $\mathcal{H}_{IntR}^{HC} + \mathcal{H}_{Kep}^{HC}$

Il nous reste à discuter l'évolution sous $\mathcal{H}_{IntR}^{HC} + \mathcal{H}_{Kep}^{HC}$. Comme expliqué dans [15], cette expression sera partitionnée suivant l'ordre de grandeur de ses termes. Le code adaptera alors le pas d'intégration en fonction de l'ordre de grandeur des parties de $\mathcal{H}_{IntR}^{HC}$ afin de garder l'erreur globale d'ordre $\epsilon\tau^2$. Ainsi si ϵ est multiplié par α, le pas sera divisé par un facteur d'au moins $\alpha^{\frac{1}{2}}$ afin de garder l'erreur globale de l'ordre de $\epsilon\tau^2$.

Mathématiquement, $\mathcal{H}_{IntR}^{HC} + \mathcal{H}_{Kep}^{HC}$ sera donc reformulé comme

$$\mathcal{H}_{Kep}^{HC} + \sum_{i=0}^{\infty} \mathcal{H}_{IntR,i}^{HC},$$

de sorte que $\sum\limits_{i=0}^{\infty} \mathcal{H}_{IntR,i}^{HC} = \mathcal{H}_{IntR}^{HC}$ et $\mathcal{H}_{IntR,i}^{HC}$ est d'un ordre de grandeur inférieur à celui de $\mathcal{H}_{IntR,i+1}^{HC}$.

Lorsque nous appliquons l'intégrateur $SABA_1$ à cette somme en réduisant le pas de manière successive, nous avons

$$\begin{aligned} f(t_{k+1}) = &\exp\left(\tau \left[T_{Kep}^{HC} + \sum_{i=0}^{\infty} T_{IntR,i}^{HC} \right] \right) f(t_k) \\ \approx &\exp\left(\frac{\tau}{2} T_{IntR,0}^{HC} \right) \exp\left(\tau \left[T_{Kep}^{HC} + \sum_{i=1}^{\infty} T_{IntR,i}^{HC} \right] \right) \exp\left(\frac{\tau}{2} T_{IntR,0}^{HC} \right) f(t_k) \end{aligned}$$

$$
\begin{aligned}
&\approx \exp\left(\frac{\tau}{2}T^{HC}_{IntR,0}\right)\left\{\exp\left(\frac{\tau}{2n_1}T^{HC}_{IntR,1}\right)\exp\left(\frac{\tau}{n_1}\left[T^{HC}_{Kep}+\sum_{i=2}^{\infty}T^{HC}_{IntR,i}\right]\right)\exp\left(\frac{\tau}{2n_1}T^{HC}_{IntR,1}\right)\right\}^{n_1}\\
&\qquad \exp\left(\frac{\tau}{2}T^{HC}_{IntR,0}\right)f(t_k)\\
&\approx \exp\left(\frac{\tau}{2}T^{HC}_{IntR,0}\right)\left\{\exp\left(\frac{\tau}{2n_1}T^{HC}_{IntR,1}\right)\left\{\exp\left(\frac{\tau}{2n_1n_2}T^{HC}_{IntR,2}\right)\right.\right.\\
&\qquad \left.\left.\exp\left(\frac{\tau}{n_1n_2}\left[T^{HC}_{Kep}+\sum_{i=3}^{\infty}T^{HC}_{IntR,i}\right]\right)\exp\left(\frac{\tau}{2n_2n_1}T^{HC}_{IntR,2}\right)\right\}^{n_2}\exp\left(\frac{\tau}{2n_1}T^{HC}_{IntR,1}\right)\right\}^{n_1}\\
&\qquad \exp\left(\frac{\tau}{2}T^{HC}_{IntR,0}\right)f(t_k)\\
&\ \vdots
\end{aligned}
$$

où $k \in \mathbb{N}$, T^{HC}_{Kep} et $T^{HC}_{IntR,i}$ sont les opérateurs linéaires associés à $\mathcal{H}^{HC}_{Kep}$ et $\mathcal{H}^{HC}_{IntR,i}$ respectivement. Notons que, dans SyMBA, le pas sera toujours réduit de la même manière. Nous aurons donc

$$n_1 = n_2 = n_i = n, \quad i \in \mathbb{N}_0.$$

Par ailleurs, ce processus de séparation continue, en théorie, à l'infini suivant tous les ordres de grandeur possibles. Néanmoins, $\mathcal{H}^{HC}_{IntR}$ étant une somme avec un nombre fini de termes, nous aurons une séparation avec un nombre fini de termes et donc un nombre d'ordres de grandeur différents fini. De plus, ces ordres de grandeur ne peuvent devenir infiniment grands. En effet, si deux corps sont trop proches, le programme considère qu'une collision se produit et ces corps ne seront pas intégrés mais fusionnés.

La question est maintenant de voir comment partitionner notre expression en gardant la structure symplectique de l'intégrateur malgré le pas variable. En pratique, le code, plutôt que de partitionner $\mathcal{H}^{HC}_{IntR}$, va partitionner $-\frac{\partial \mathcal{H}^{HC}_{IntR}}{\partial r_{j,c}}$ afin de rendre la résolution analytique plus aisée. Cette dérivée est aisément déductible de la dérivée de $\mathcal{H}^{HC}_{Int}$ (section 2.2.6). En effet,

$$
\begin{aligned}
-\frac{\partial \mathcal{H}^{HC}_{IntR}}{\partial r^{HC}_{i,c}} &= -\frac{\partial}{\partial r^{HC}_{i,c}}\left(-\sum_{k=2}^{n-1}\sum_{k<j\leqslant n}\frac{Gm_km_j}{\|r_{kj}\|}\left(1-H(\|r_{kj})\|\right)\right)\\
&= \sum_{k=2}^{n-1}\sum_{k<j\leqslant n}\left[\frac{\partial}{\partial r^{HC}_{i,c}}\left(\frac{Gm_km_j}{\|r_{kj}\|}\right)\left(1-H(\|r_{kj}\|)\right)\right]\\
&= -\sum_{j=2,j\neq i}^{n}\left[\frac{Gm_im_j}{\|r_{ij}\|^3}\left(r^{HC}_{i,c}-r^{HC}_{j,c}\right)\left(1-H(\|r_{ij}\|)\right)\right].
\end{aligned}
$$

Celle-ci sera reformulée

$$-\frac{\partial \mathcal{H}^{HC}_{IntR}}{\partial r^{HC}_{i,c}} = \sum_{l=0}^{\infty}\left(-\sum_{j=2,j\neq i}^{n}\left[\frac{Gm_im_j}{\|r_{ij}\|^3}\left(r^{HC}_{i,c}-r^{HC}_{j,c}\right)f_l(\|r_{ij}\|)\left(1-H(\|r_{ij}\|)\right)\right]\right),$$

avec f_l une fonction de partition définie comme

$$f_l(\|r_{ij}\|) = \begin{cases} \bar{f}_l(\|r_{ij}\|) - \bar{f}_{l-1}(\|r_{ij}\|) & \text{si } \ i \in \mathbb{N}_0 \\ \bar{f}_0(\|r_{ij}\|) & \text{si } \ i = 0, \end{cases} \tag{2.2.34}$$

avec

$$\bar{f}_l(\|r_{ij}\|) = \begin{cases} 1 & \text{si } \ \|r_{ij}\| \geqslant R_l \\ g\left(\dfrac{R_l - \|r_{ij}\|}{R_l - R_{l+1}}\right) & \text{si } R_{l+1} \leqslant \|r_{ij}\| < R_l \\ 0 & \text{si } \ \|r_{ij}\| < R_{l+1}. \end{cases}$$

En d'autres termes, nous aurons

$$-\frac{\partial \mathcal{H}^{HC}_{IntR,l}}{\partial r^{HC}_{i,c}} = -\sum_{j=2,j\neq i}^{n}\left[\frac{Gm_im_j}{\|r_{ij}\|^3}\left(r^{HC}_{i,c}-r^{HC}_{j,c}\right)f_l(\|r_{ij}\|)\left(1-H(\|r_{ij}\|)\right)\right] \qquad l\in\mathbb{N}. \qquad (2.2.35)$$

Il ne reste donc plus qu'à déterminer la fonction g. Comme le précise l'article [15], la fonction $\bar{f}_l$ doit être de classe $\mathcal{C}^4$ afin que les calculs sur l'erreur restent cohérents. En effet, comme montré dans [15], la formule d'erreur développée sur base de la relation (2.2.8) fait apparaître des parenthèses de Poisson qui vont demander des différentiations successives de l'hamiltonien jusqu'à l'ordre 4 pour les termes dominants de l'erreur. Donc, afin de garder une erreur de l'ordre de $\epsilon\tau^2$, M. Duncan, H. Levison & M. H. Lee imposent que les différentes parties de $-\frac{\partial \mathcal{H}^{HC}_{IntR}}{\partial r_{j,c}}$ soient de classe $\mathcal{C}^4$ ([15]). Cela entraîne que f_l et $\bar{f}_l$ doivent également être de classe $\mathcal{C}^4$.

Néanmoins depuis la parution de l'article [15], K. P. Rauch et M. Holman ont proposé dans [39] d'utiliser une fonction $\bar{f}_l$ de classe $\mathcal{C}^\infty$ afin de gagner en stabilité lors de l'intégration. C'est cette dernière fonction qui est actuellement utilisée dans SyMBA. Notons que $\bar{f}_l$ sera de classe $\mathcal{C}^\infty$ si g satisfait les propriétés suivantes :

- $g(0) = 0$
- $g(1) = 1$
- $g^{(l)}(0) = g^{(l)}(1) = 0 \qquad \forall l \in \mathbb{N}.$

La fonction proposée par [39] et implémentée dans SyMBA est la suivante :

$$g(r) = \frac{1}{2}\left[1 + tanh\left(\frac{2r-1}{r(1-r)}\right)\right].$$

Elle possède bien les propriétés désirées.

Intéressons-nous à présent à l'évolution analytique sous $\mathcal{H}^{HC}_{IntR,l}$. Cette évolution s'exprime en utilisant les équations d'Hamilton (2.2.2) :

$$\begin{cases} \dfrac{dr^{HC}_{i,c}}{dt} &= \dfrac{\partial \mathcal{H}^{HC}_{IntR,l}}{\partial p^{HC}_{i,c}} \\ \dfrac{dp^{HC}_{i,c}}{dt} &= -\dfrac{\partial \mathcal{H}^{HC}_{IntR,l}}{\partial r^{HC}_{i,c}} \end{cases} \qquad i = 2...n, \quad c = 1,2,3, \quad l \in \mathbb{N}.$$

Or, $\mathcal{H}^{HC}_{IntR}$ ne dépendant d'aucun moment associé, sa restriction $\mathcal{H}^{HC}_{IntR,l}$ ne peut dépendre d'un moment associé. Nous avons donc $\frac{\partial \mathcal{H}^{HC}_{IntR,l}}{\partial p^{HC}_{i,c}} = 0$. En utilisant cette dérivée, les équations d'Hamilton (2.2.2) et la dérivée (2.2.35), nous obtenons les équations d'évolution :

$$\begin{cases} \frac{dr^{HC}_{i,c}}{dt} &= 0 \\ \frac{dp^{HC}_{i,c}}{dt} &= -\sum\limits_{j=2,j\neq i}^{n}\left[\frac{Gm_im_j}{\|r_{ij}\|^3}\left(r^{HC}_{i,c}-r^{HC}_{j,c}\right)f_l(\|r_{ij}\|)\left(1-H(\|r_{ij}\|)\right)\right] \end{cases} \begin{array}{l} c = 1,2,3, \\ i = 2...n, \\ l \in \mathbb{N}. \end{array}$$

A nouveau, nous pouvons résoudre le système aisément :

$$\begin{cases} r^{HC}_{i,c}(t_{k+1}) &= r^{HC}_{i,c}(t_k) \\ p^{HC}_{i,c}(t_{k+1}) &= p^{HC}_{i,c}(t_k) - \tau\sum\limits_{j=2,j\neq i}^{n}\left[\frac{Gm_im_j}{\|r_{ij}(t_k)\|^3}\left(r^{HC}_{i,c}(t_k)-r^{HC}_{j,c}(t_k)\right)\right. \\ & \qquad \left. f_l(\|r_{ij}(t_k)\|)\left(1-H(\|r_{ij}\|)\right)\right] \end{cases} \begin{array}{l} c = 1,2,3, \\ i = 2...n, \\ k,l \in \mathbb{N}. \end{array}$$

Comme précédemment, en utilisant l'équation (2.2.23), nous trouvons

$$\begin{cases} r_{i,c}^{HC}(t_{k+1}) &= r_{i,c}^{HC}(t_k) \\ v_{i,c}^{B}(t_{k+1}) &= v_{i,c}^{B}(t_k) - \tau \sum\limits_{j=2,j\neq i}^{n} \left[\frac{Gm_j}{\|r_{ij}(t_k)\|^3} \left(r_{i,c}^{HC}(t_k) - r_{j,c}^{HC}(t_k) \right) \right. \\ & \qquad \left. f_l(\|r_{ij}(t_k)\|) \left(1 - H(\|r_{ij}\|)\right)\right] \end{cases} \begin{array}{l} c = 1,2,3, \\ i = 2...n, \\ k,l \in \mathbb{N}. \end{array}$$

En pratique, le code utilisera une reformulation de cette évolution en terme de $\bar{f}_l$. Pour $\mathcal{H}_{IntR,0}^{HC}$, l'évolution sera donnée par

$$\begin{cases} r_{i,c}^{HC}(t_{k+1}) &= r_{i,c}^{HC}(t_k) \\ v_{i,c}^{B}(t_{k+1}) &= v_{i,c}^{B}(t_k) - \tau \sum\limits_{j=2,j\neq i}^{n} \left[\frac{Gm_j}{\|r_{ij}(t_k)\|^3} \left(r_{i,c}^{HC}(t_k) - r_{j,c}^{HC}(t_k) \right) \right. \\ & \qquad \left. \bar{f}_0(\|r_{ij}(t_k)\|) \left(1 - H(\|r_{ij}\|)\right)\right] \end{cases} \begin{array}{l} c = 1,2,3, \\ i = 2...n, \\ k \in \mathbb{N} \end{array}$$

et, pour $l \in \mathbb{N}_0$, nous aurons

$$\begin{cases} r_{i,c}^{HC}(t_{k+1}) &= r_{i,c}^{HC}(t_k) \\ v_{i,c}^{B}(t_{k+1}) &= v_{i,c}^{B}(t_k) - \tau \sum\limits_{j=2,j\neq i}^{n} \left[\frac{Gm_j}{\|r_{ij}(t_k)\|^3} \left(r_{i,c}^{HC}(t_k) - r_{j,c}^{HC}(t_k) \right) \right. \\ & \qquad \left. \left[\bar{f}_l(\|r_{ij}(t_k)\|) - \bar{f}_{l-1}(\|r_{ij}(t_k)\|) \right] \left(1 - H(\|r_{ij}\|)\right)\right] \end{cases} \begin{array}{l} c = 1,2,3, \\ i = 2...n, \\ k,l \in \mathbb{N}, \end{array}$$

qui se réécrit

$$\begin{cases} r_{i,c}^{HC}(t_{k+1}) &= r_{i,c}^{HC}(t_k) \\ v_{i,c}^{B}(t_{k+1}) &= v_{i,c}^{B}(t_k) - \tau \sum\limits_{j=2,j\neq i}^{n} \left[\frac{Gm_j}{\|r_{ij}(t_k)\|^3} \left(r_{i,c}^{HC}(t_k) - r_{j,c}^{HC}(t_k) \right) \bar{f}_l(\|r_{ij}(t_k)\|) \right. \\ & \quad \left. \left(1 - H(\|r_{ij}\|)\right)\right] + \tau \sum\limits_{j=2,j\neq i}^{n} \left[\frac{Gm_j}{\|r_{ij}(t_k)\|^3} \left(r_{i,c}^{HC}(t_k) - r_{j,c}^{HC}(t_k) \right) \right. \\ & \quad \left. \bar{f}_{l-1}(\|r_{ij}(t_k)\|) \left(1 - H(\|r_{ij}\|)\right)\right] \end{cases} \begin{array}{l} c = 1,2,3, \\ i = 2...n, \\ k,l \in \mathbb{N}. \end{array} \tag{2.2.36}$$

Nous allons à présent discuter l'évolution sous $\mathcal{H}_{Kep}^{HC}$. Celle-ci se fera, en théorie, en réduisant le pas au maximum.

Néanmoins, il n'est pas obligatoire de réduire ce pas au maximum pour toutes les parties de $\mathcal{H}_{Kep}^{HC}$. En effet, si une particule l n'apparaît plus dans $\sum\limits_{n=k}^{\infty} \mathcal{H}_{IntR,i}^{HC}$, elle ne sera plus mise à jour lors des intégrations des différents $\mathcal{H}_{IntR,i}^{HC}$ restants. Il n'est donc plus utile de réduire le pas avant d'intégrer cette particule pour $\mathcal{H}_{Kep,l}^{HC} = \left(\frac{\left\| p_l^{HC} \right\|^2}{2m_l} - \frac{Gm_1 m_l}{\left\| r_l^{HC} \right\|} \right)$. Continuer la réduction du pas pour ce corps entraînerait uniquement de multiples intégrations sous $\mathcal{H}_{Kep,l}^{HC}$ sans aucune influence de $\mathcal{H}_{IntR,i}^{HC}$ avec $i > k-1$ (car la particule l n'apparaît plus dans cette partie de l'hamiltonien). Le code va donc directement intégrer $\mathcal{H}_{Kep,l}^{HC}$ au moyen des formules d'évolution (2.2.24) sans réduire davantage le pas. Notons que cette procédure est correcte car l'évolution de chaque corps l sous $\mathcal{H}_{Kep}^{HC}$ peut être effectuée indépendamment en utilisant $\mathcal{H}_{Kep,l}^{HC}$, comme nous l'avons explicité à la section 2.2.6.

Toutes les informations concernant SyMBA ainsi qu'une discussion concernant le choix de n et des bornes de séparation peuvent être trouvées dans l'article [15].

2.3 Code

A présent, nous allons décrire les différentes fonctions intervenant dans l'intégrateur SyMBA qui, pour rappel, est l'intégrateur que nous modifierons par la suite. Pour ce faire, nous analysons les dossiers qui constituent l'essentiel du code, à savoir Anal, Coord, CPP_test, Discard, Helio, IO, MVS, RMVS, Symba7, Orbel et Util.

Notons que le programme est construit avec la constante gravitationnelle G égale à 1. Les unités des masses sont donc modifiées dans l'initialisation du code afin d'avoir cette propriété.

2.3.1 Anal

Ce dossier permet de calculer l'énergie du système [1], l'intégrale de Jacobi d'une particule et le moment angulaire du système. Notons que ces différentes grandeurs peuvent être utilisées pour avaliser les résultats renvoyés par le code. En effet, lorsque aucune dissipation n'est présente, l'énergie du système ainsi que le moment angulaire total doivent être conservés tout au long de l'intégration. Nous allons maintenant détailler les différentes formules utilisées pour calculer ces quantités.

Tout d'abord, l'énergie est calculée au moyen de la formule suivante :

$$E = K + P,$$

où K est l'énergie cinétique et P est l'énergie potentielle du système. Ces deux quantités sont calculées comme suit :

$$K = \sum_{i=1}^{n} \frac{m_i \left\| v_i^B \right\|^2}{2}$$

et

$$P = -\sum_{1 \leqslant i < j \leqslant n} \frac{G m_i m_j}{\|r_{ij}\|} + P_A,$$

où P_A est l'énergie potentielle due à l'aplatissement du corps central et ne sera pas considérée dans la suite de ce travail.

Ensuite, concernant le moment angulaire total, également appelé moment cinétique total, sa définition est ([48])

$$L = \sum_{i=1}^{n} r_i^B \wedge m_i v_i^B.$$

Rappelons que le moment angulaire total constitue une intégrale première dans le cas d'un système de particules soumises à la troisième loi de Newton ([26]).

Pour terminer, discutons l'intégrale de Jacobi. Celle-ci, également appelée constante de Jacobi, est une intégrale première du mouvement circulaire restreint à trois corps ([18]). Cette quantité ne nous intéresse donc pas pour la suite de notre travail puisque nous étudions le problème complet à n corps.

A présent, passons en revue les fonctions présentes dans ce dossier :

- *anal_energy.f* : renvoie les coordonnées du moment angulaire total ainsi que l'énergie potentielle, cinétique et totale du système.
- *anal_energy_write.f* : écrit l'énergie totale et les coordonnées du moment angulaire total du système dans un fichier.

1. Concernant l'énergie potentielle, le code permet d'également tenir compte de l'énergie due à l'aplatissement du corps central

- *anal_jacobi.f* : calcule l'intégrale de Jacobi d'une particule.
- *anal_jacobi_write.f* : écrit, dans un fichier, l'intégrale de Jacobi pour les particules non massives du système.

2.3.2 Coord

Ce dossier contient toutes les fonctions permettant de faire des changements de coordonnées entre les différents systèmes de coordonnées décrits à la section 2.1.

- *coord_b2h.f* : permet de passer des coordonnées barycentriques aux coordonnées héliocentriques.
- *coord_b2j.f* : permet de passer des coordonnées barycentriques aux coordonnées de Jacobi.
- *coord_h2b.f* : permet de passer des coordonnées héliocentriques canoniques aux coordonnées barycentriques.
- *coord_j2b.f* : permet de passer des coordonnées de Jacobi aux coordonnées barycentriques.
- *coord_j2h.f* : permet de passer des coordonnées de Jacobi aux coordonnées héliocentriques.
- *coord_vb2h.f* : permet de transformer des vitesses dans un repère barycentrique en vitesses dans un repère héliocentrique.
- *coord_vh2b.f* : permet de transformer des vitesses dans un repère héliocentrique en vitesses dans un repère barycentrique.
- *coord_vh2vj.f* : permet de transformer des vitesses dans un repère héliocentrique en vitesses dans un repère de Jacobi.

Notons que toutes les sous-routines travaillant sur un repère héliocentrique en entrée ou en sortie considèrent que l'étoile est le premier corps du système. Par ailleurs, toutes ces sous-routines possèdent également une version dédiée aux particules tests, c'est-à-dire particules de masse négligeable. Ces particules non massives sont très utiles pour les théories de formation avec disque de planétésimaux (par exemple coord_b2h_tp.f).

2.3.3 CPP_test

Ce dossier contient toutes les fonctions permettant de connaître les informations importantes sur les compilateurs de notre ordinateur afin de pouvoir effectuer la compilation de SyMBA :

- *open_test1.f* : permet de tester si le compilateur utilise "access = 'append'" lors de l'ouverture d'un fichier. Si oui, dans les makefiles il faut utiliser l'option "-D_OPEN_POSITION" pour la compilation.
- *open_test2.f* : permet de tester si le compilateur utilise "position = 'append'" lors de l'ouverture d'un fichier. Si oui, dans les makefiles il faut utiliser l'option "-U_OPEN_POSITION" pour la compilation.
- *recur_test1.f* : permet de tester si le compilateur accepte les fonctions récursives sans l'entête "recursive". Si oui, dans le makefile, il faut utiliser l'option "-U_RECUR_SUB" pour la compilation.
- *recur_test2.f* : permet de tester si le compilateur a besoin de l'entête "recursive" pour les fonctions de ce type. Si oui, dans le makefile, il faut utiliser l'option "-D_RECUR_SUB" pour la compilation.
- *underscore_testc1.c* : permet de voir si une fonction en C sans underscore à la fin de son nom peut être appelée avec le compilateur fortran de notre ordinateur.

- *underscore_testc2.c* : permet de voir si une fonction en C avec un underscore à la fin de son nom peut être appelée avec le compilateur fortran de notre ordinateur.
- *underscore_testc3.c* : permet de voir si une fonction en C avec deux underscores à la fin de son nom peut être appelée avec le compilateur fortran de notre ordinateur.
- *underscore_testf.f* : fait appel aux trois fonctions précédentes afin de faire le test en pratique sur l'appel de fonctions en C avec le compilateur fortran de notre ordinateur.

Une fois les différents tests effectués, l'en-tête pour les makefiles à utiliser est semblable à :

```
set SWIFT_DIR=/home/arnaud/Documents/SYMBA/
set FORTRAN=gfortran
set FFLAGS='-O -c'
set PRECOMP=/usr/bin/cpp
set CPPFLAGS='-traditional -D_OPEN_POSITION -D_RECUR_SUB -U_FXDR_AVAIL'.
```

Détaillons à présent ce que représentent les différents lignes de cette entête. Tout d'abord, la première ligne correspond à l'emplacement du dossier SyMBA dans l'ordinateur. Ensuite, la deuxième ligne correspond au compilateur fortran utilisé. Pour ce qui est de la troisième ligne, celle-ci correspond aux options de compilation souhaitées. La quatrième ligne contient, quant à elle, l'emplacement du précompilateur. Enfin, la cinquième ligne correspond aux différentes options déterminées par les fonctions décrites ci-dessus.

2.3.4 Discard

Ce dossier contient les fonctions permettant de repérer et de traiter certains cas particuliers pouvant survenir durant le mouvement des corps célestes. En particulier, sont visés celui où un corps serait éjecté du système et celui où deux corps entreraient en collision. Dans les deux cas, un corps sera retiré du système. Le dossier contient les fonctions suivantes :

- *discard.f* : teste les corps pour savoir si nous sommes dans un cas particulier nécessitant l'abandon d'un corps. Si tel est le cas, le programme abandonne ce corps.
- *discard_massive.f* : teste les corps massifs afin de savoir s'ils doivent être abandonnés ou si deux corps se rencontrent et fusionnent. Pour ce faire, la fonction utilise une matrice de rencontre.
- *discard_massive5.f* : cette fonction est une révision de la fonction précédente. Elle effectue la même tâche en se détachant de l'utilisation d'une matrice de rencontre.
- *discard_mass_merge.f* : permet d'effectuer la fusion de deux corps massifs au moyen d'une matrice de rencontre.
- *discard_mass_merge5.f* : cette fonction est une révision de la fonction précédente permettant de se détacher de l'utilisation d'une matrice de rencontre.
- *discard_mass_peri.f* : teste les corps massifs afin de savoir s'ils doivent être abandonnés car il ont une distance périhélique trop faible. Ce cas représente une rencontre entre une planète et l'étoile. Une telle rencontre entraîne la disparition de la planète.
- *discard_mass_reorder.f* : permet de retirer un corps massif du système. Cette fonction fait appel à une matrice de rencontre.
- *discard_mass_reorder5.f* : cette fonction est une révision de la fonction précédente sans matrice de rencontre.

- *discard_peri.f* : teste les particules tests afin de savoir si elles ont une rencontre avec un corps massif au moyen de la distance périhélique. Si tel est le cas, la particule test est abandonnée.
- *discard_pl.f* : teste les particules tests afin de savoir si elles ont une rencontre avec un corps massif sans utiliser la distance périhélique. Si tel est le cas, la particule test est abandonnée.
- *discard_pl_close.f* : teste si une planète et une particule test se rencontrent ou vont se rencontrer au prochain pas d'intégration au moyen d'interpolation linéaire.
- *discard_sun.f* : teste les particules tests pour savoir si certaines doivent être abandonnées du système.

2.3.5 Helio

Ce dossier contient l'intégrateur symplectique héliocentrique canonique détaillé à la section 2.2.6. Les différentes fonctions utilisées pour cet intégrateur sont :

- *helio_drift.f* : fait évoluer le système par rapport à $\mathcal{H}_{Kep}^{HC}$ au moyen de l'équation (2.2.24).
- *getacch.f* : calcule $-\frac{1}{m_i}\frac{\partial \mathcal{H}_{Int}^{HC}}{\partial r_i^{HC}}$ pour tous les corps massifs, c'est-à-dire l'accélération barycentrique induite par $\mathcal{H}_{Int}^{HC}$.
- *helio_lindrift.f* : fait évoluer les particules massives du système par rapport à $\mathcal{H}_{Jump}^{HC}$ au moyen de l'équation (2.2.31).
- *helio_step.f* : effectue (2.2.13) pour tous les corps du système.
- *helio_step_pl.f* : effectue (2.2.13) sur les particules massives du système.

Pour terminer, notons que comme pour les fonctions contenues dans le dossier Coord, les sous-routines présentées ci-dessus possèdent une version dédiée aux particules tests (par exemple *helio_getacch_tp.f*).

2.3.6 IO

Ce dossier contient les fonctions permettant d'importer les données provenant des fichiers d'entrée et de renvoyer les informations générées par le programme. Nous allons commencer par détailler les fichiers param.in, pl.in et tp.in qui sont les fichiers d'entrée du programme. Notons que des informations complémentaires à cette présentation peuvent être trouvées dans les fichiers README associés au code.

Param.in

Ce fichier est de la forme suivante :

```
t0, tstop, dt
dtout, dtdump
L1 L2 L3 L4 L5 L6
rmin, rmax, rmaxu, qmin, lclose
binary_outputfile
status_flag_for_open_statements.
```

Détaillons chacun de ces éléments.

- Première ligne :

  - t0 : le temps initial de l'intégration.
  - tstop : le temps final de l'intégration.
  - dt : le pas d'intégration.

- Deuxième ligne :

  - dtout : le pas de temps entre chaque output dans le fichier de sortie.
  - dtdump : le pas de temps entre chaque sauvegarde de l'état du système. Celles-ci seront effectuées au sein des trois fichiers suivants :
    * dump_pl.dat
    * dump_tp.dat
    * dump_param.dat

    Ces sauvegardes permettent de relancer le code en cas d'erreur lors de l'exécution du code.

- Troisième ligne : l'ensemble des éléments de cette ligne sont des booléens. Ceux-ci ne peuvent prendre que deux valeurs qui sont T pour True (vrai) et F pour False (faux).

  - L1 : est égal à T si le fichier pl.in contient les valeurs J2 et J4 qui sont utilisées pour les calculs liés à l'aplatissement du corps central.
  - L2 : est égal à T si le code doit tenir compte des rencontres entre les corps et de l'éjection possible d'un corps du système. Notons que si cette valeur est mise à F, la troisième ligne du fichier param.in doit être omise.
  - L3 : est égal à T si l'utilisateur souhaite que le programme calcule l'intégrale de Jacobi et la renvoie dans jacobi.out.
  - L4 : est égal à T si l'utilisateur souhaite que le programme calcule l'énergie et le moment angulaire total du système et les renvoie dans energy.out.
  - L5 : est égal à T si l'utilisateur souhaite renvoyer les éléments orbitaux des différentes particules dans des fichiers de sortie binaires. Ces éléments orbitaux sont les suivants :
    * demi-grand axe de l'orbite
    * excentricité de l'orbite
    * inclinaison de l'orbite
    * longitude du noeud ascendant
    * argument du périhélie
    * anomalie moyenne.
  - L6 : donne la même information que L5. Cependant, si L6 est mis à T, les informations écrites pour L5 sont renvoyées sous forme d'entiers. Notons que L5 et L6 ne peuvent être tous les deux mis à la valeur T.

- Quatrième ligne :

  - rmin : distance minimale acceptée entre un corps et le corps central. Si la distance entre le corps et le corps central est plus petite que cette valeur, les deux corps sont considérés comme se croisant. Notons que si cette valeur est mise à -1, le programme ne teste pas cette distance.
  - rmax : distance maximale acceptée entre un corps et le corps central. Notons que si cette valeur est égale à -1, le programme ne teste pas cette distance.
  - rmaxu : distance maximale acceptée entre un corps et le système. Le programme va alors évaluer la distance de chaque corps par rapport au centre de masse du système.
  - qmin : distance minimale acceptée pour la distance périhélique d'un corps. Notons que si cette valeur est égale à -1, le programme ne teste pas cette distance.

  - lclose : est un booléen. Si cette valeur est à T, le programme teste si les particules se retrouvent trop proches des particules massives. Dans ce cas, il faut rajouter la distance minimale acceptée entre les particules massives et les particules non massives dans pl.in. Si cette distance est violée, les particules seront considérées comme se croisant.

- Cinquième ligne :

  - binary_outputfile : contient le nom du fichier dans lequel les éléments orbitaux seront renvoyés si T5 ou T6 est mis à T.

- Sixième ligne :

  - status_flag_for_open_statements : définit le statut des fichiers d'output. Il y a trois statuts possibles :
    * new : les fichiers d'output n'existent pas encore.
    * append : les fichiers d'output existent et il faut ajouter les valeurs à la suite de celles existantes.
    * unknow : l'existence des fichiers est inconnue. Le programme supprime donc les fichiers s'ils existent. Ensuite, il crée de nouveaux fichiers pour écrire les valeurs de sortie.

Voici un exemple de fichier param.in :

```
0.0d0 1.d6 0.1d0
1.d3 1.d3
F T F F T F
-1. 1000. -1. -1. T
bin.dat
unknown
```

Pl.in

Ce fichier d'entrée contient les informations relatives aux particules massives du système (i.e. les planètes). Il est de la forme suivante :

```
n
m1, (j2r2), (j4r4)
x1,y1,z1
vx1, vy1, vz1
m (rhil) (rpl)
x,y,z
vx, vy, vz
...
m (rhil) (rpl)
x,y,z
vx, vy, vz
```

Notons que les valeurs entre parenthèses sont des valeurs optionnelles telles que détaillées ci-dessous. Présentons chacun de ces éléments.

- Première ligne :

  - n : nombre de particules massives comprises dans le fichier.

- Deuxième ligne :

  - m1 : la masse du corps central mise à l'échelle de façon à avoir $G = 1$.
  - j2r2 : valeur optionnelle qui n'est encodée que si L1 est mis à la valeur T dans param.in. Dans ce cas-là, cette valeur est égale à J2 multiplié par le rayon du corps central au carré. Notons que J2 est l'harmonique zonale d'ordre 2 du corps central et est une quantité propre à un corps massif.
  - j4r4 : valeur optionnelle qui n'est encodée que si L1 est mis à la valeur T dans param.in. Dans ce cas-là, cette valeur est égale à J4 multiplié par le rayon du corps central à la puissance quatre. Notons que J4 est l'harmonique zonale d'ordre 4 du corps central et est une quantité propre à un corps massif.

- Troisième ligne :
  - x1, y1, z1 : position du corps massif dans un repère héliocentrique. Le premier corps étant le corps central, ces valeurs sont nulles.
- Quatrième ligne :
  - vx1, vy1, vz1 : vitesse du corps central dans un repère héliocentrique. Ces valeurs sont supposées nulles pour le programme.
- Le reste du fichier se compose de n-1 triplets de lignes similaires. Chaque triplet correspond à un nouveau corps dans le système et est de la forme
  - première ligne :
    * m : masse du corps massif mise à l'échelle de façon à avoir $G = 1$.
    * rhil : cette valeur n'est initialisée que dans le cas de l'utilisation de l'algorithme SyMBA et correspond au rayon de la sphère de Hill du corps massif.
    * rpl : cette valeur n'est initialisée que si lclose a la valeur T dans param.in. Celle-ci correspond à la distance minimale acceptée entre le corps massif référencé dans ce triplet et les particules non massives du système.
  - deuxième ligne :
    * x, y, z : position du corps massif dans un repère héliocentrique.
  - troisième ligne :
    * vx, vy, vz : vitesse du corps massif dans un repère héliocentrique.

Voici un exemple de fichier pl.in :

```
5
2.95913981972262D-004
0.d0  0.d0  0.d0
0.d0  0.d0  0.d0
2.82536284454158D-007   0.3686  4.77213e-4
5.39946000000000        0.00000000000000D+000   0.00000000000000D+000
0.00000000000000D+000   7.40690072992482D-003   1.29274805105691D-008
8.45976589289600D-008   0.4065  4.03281e-4
8.89911000000000        0.00000000000000D+000   0.00000000000000D+000
0.00000000000000D+000   5.76757485366185D-003   2.01326342104077D-008
1.29203266113083D-008   0.2845  1.75136e-4
11.6488350000000        0.00000000000000D+000   0.00000000000000D+000
0.00000000000000D+000   5.04048922618387D-003   8.79731329082991D-009
1.52436803773432D-008   0.3664  1.68619e-4
14.1285870000000        0.00000000000000D+000   0.00000000000000D+000
0.00000000000000D+000   4.57684576492736D-003   1.59762055908530D-008
```

Notons que dans ce cas-ci, lclose et L1 sont égaux à F ce qui explique l'absence des quantités optionnelles.

Tp.in

Ce fichier contient toutes les informations relatives aux particules non massives du système. Il est construit de la manière suivante :

```
n
x, y, z
vx, vy, vz
i1, i2, ..., ik
j1, j2, ..., jl
...
x, y, z
vx, vy, vz
i1, i2, ..., ik
j1, j2, ..., jl
```

Détaillons chacune de ces valeurs.

- Première ligne :
  - n : nombre de particules dans le fichier.
- La suite du fichier est une succession de quadruplets de lignes. Chaque quadruplet correspond aux paramètres d'une particule non massive et est de la forme suivante :
  - première ligne :
    - x, y, z : position dans un repère héliocentrique de la particule.
  - deuxième ligne :
    - vx, vy, vz : vitesse dans un repère héliocentrique de la particule.
  - troisième et quatrième lignes : informations relatives à l'état de la particule. Toutes ces informations sont détaillées dans README.stat.

Fonctions

Nous pouvons maintenant décrire les différentes fonctions présentes dans ce dossier :

- *io_discard_mass.f* : écrit l'information relative à un corps massif retiré du système dans un fichier formaté de sortie.
- *io_discard_merge.f* : écrit l'information relative à la fusion de deux corps massifs dans un fichier formaté de sortie.
- *io_discard_write.f* : vérifie si, depuis son dernier appel, de nouvelles particules ont été retirées du système. Si tel est le cas, la fonction écrit l'information relative au rejet des particules en question dans un fichier formaté de sortie.
- *io_dump_param.f* : crée le fichier formaté de sauvegarde dump_param.in.
- *io_dump_pl.f* : crée le fichier formaté de sauvegarde dump_pl.in sans les rayons des sphères de Hill.
- *io_dump_pl_symba.f* : crée le fichier formaté de sauvegarde dump_pl.in avec les rayons des sphères de Hill.
- *io_dump_tp.f* : crée le fichier formaté de sauvegarde dump_tp.in.
- *io_energy_write.f* : effectue les écritures requises si L4 a la valeur T dans le fichier formaté param.in.

- *io_getns.f* : détermine le nombre de paramètres présents aux troisième et quatrième lignes des différentes particules dans tp.in.

- *io_init_param.f* : initialise les paramètres manipulés par le programme provenant de param.in.

- *io_init_pl.f* : initialise les paramètres manipulés par le programme provenant de pl.in sans les rayons des sphères de Hill.

- *io_init_pl_symba.f* : initialise les paramètres manipulés par le programme provenant de pl.in avec les rayons des sphères de Hill.

- *io_init_tp.f* : initialise les paramètres manipulés par le programme provenant de tp.in.

- *io_jacobi_write.f* : effectue les écritures requises si L3 a la valeur T dans le fichier param.in. Cette fonction va donc écrire l'intégrale de Jacobi dans un fichier formaté nommé jacobi.out.

- *io_open.f* : permet d'ouvrir les fichiers en respectant les caractéristiques du compilateur décrites dans la section 2.3.3.

- *io_openfxdr.f* : révision de la fonction précédente utilisant la bibliothèque fxdr [36].

- *io_read_hdr.f* : lit les informations contenues au début des fichiers binaires de sortie du programme en utilisant la bibliothèque fxdr.

- *io_read_line.f* : permet de lire une ligne des fichiers binaires de sortie contenant les éléments orbitaux des corps en utilisant la bibliothèque fxdr.

- *io_read_mass.f* : permet de lire les masses contenues dans un fichier binaire de sortie.

- *io_write_frame.f* : écrit les informations relatives aux particules dans un fichier binaire de sortie.

- *io_write_hdr.f* : écrit le temps, le nombre de corps massifs présents et le nombre de particules test actives dans un fichier binaire de sortie.

- *io_write_line.f* : permet d'écrire les éléments orbitaux relatifs à un corps dans un fichier binaire de sortie.

- *io_write_mass.f* : écrit les masses des corps dans un fichier binaire de sortie.

Notons que les fichiers formatés le sont au format ASCII. De plus, les fonctions travaillant sur les fichiers binaires décrites ci-dessus utilisent la bibliothèque fxdr ([36]) mais possèdent un équivalent se passant de celle-ci (par exemple io_read_line_r.f) en utilisant à la place des fonctions de fxdr, les fonctions read et write de fortran. L'avantage d'utiliser la bibliothèque fxdr est qu'elle assure une grande portabilité des fichiers binaires créés. En effet, cette bibliothèque contient des fonctions permettant de réaliser des lectures et des écritures de fichiers binaires suivant des formats ne dépendant pas de l'ordinateur. Elle permet donc de créer des fichiers qui sont interprétables par n'importe quel ordinateur. Cependant pour obtenir cette portabilité, les valeurs manipulées perdent en précision car elles sont stockées au moyen d'entiers. Enfin, notons que la méthode utilisée dépend des valeurs de L5 et L6 contenues dans le fichier param.in. Si L5 a T pour valeur, ce sont les fonctions de base qui seront utilisées. Si, par contre, il s'agit de L6 qui a T pour valeur, le programme utilisera la bibliothèque fxdr.

2.3.7 MVS

Ce dossier contient les fonctions associées à l'intégrateur détaillé à la section 2.2.4. Ces fonctions sont les suivantes :

- drift : permet d'effectuer l'évolution due à l'équation (2.2.18).
- getacch : permet de calculer l'accélération des corps dans le système due à l'équation (2.2.4).
- kickvh : permet d'effectuer l'évolution due à à l'équation (2.2.4).
- step : permet d'effectuer l'évolution totale telle que décrite par les formules (2.2.16) et (2.2.17).

Nous allons maintenant décrire les différents sous-dossiers et les fonctions qui les composent.

Drift

Ce sous-dossier effectue l'évolution due à la partie $\mathcal{H}^J_{Kep}$. Ce dossier permet donc d'appliquer l'équation (2.2.19). Les différentes fonctions qui composent ce dossier sont :

- *drift.f* : effectue l'évolution due à (2.2.18) pour tout le système.
- *drift_dan.f* : calcule l'équation (2.2.19).
- *drift_one.f* : effectue l'évolution due à (2.2.18) pour une seule particule.
- *drift_tp.f* : effectue l'évolution due à (2.2.18) pour les particules non massives.

Getacch

Ce sous-dossier calcule l'évolution due à $\mathcal{H}^J_{Int}$. $\mathcal{H}^J_{Int}$ ne dépendant que des positions, il n'influencera que les moments des corps. En effet, comme $\mathcal{H}^J_{Int}$ ne dépend que des positions, les équations d'Hamilton (2.2.2) se réécrivent :

$$\begin{cases} \dfrac{dq_i}{dt} &= \dfrac{\partial \mathcal{H}^J_{Int}}{\partial p_i} = 0 \\ \dfrac{dp_i}{dt} &= -\dfrac{\partial \mathcal{H}^J_{Int}}{\partial q_i} \end{cases} \qquad i = 1...n.$$

Nous déduisons de la première équation que les positions resteront constantes sous $\mathcal{H}^J_{Int}$. Il n'y a donc que les moments qui seront modifiés. Les moments étant associés aux vitesses (suivant les formules de la section 2.1), le code considérera les vitesses comme modifiées. Afin de réaliser l'évolution sous $\mathcal{H}^J_{Int}$, il va donc calculer l'accélération induite par cette partie de l'hamiltonien. De plus, dans un but d'efficacité de calcul, cette accélération sera calculée en coordonnées héliocentriques. Le code va donc convertir les coordonnées de Jacobi en coordonnées héliocentriques, calculer l'accélération, faire évoluer les vitesses et reconvertir les vitesses en coordonnées de Jacobi. L'accélération en coordonnées héliocentriques sera alors calculée par les formules :

$$a_i^H = \left(a_i^H\right)_0 + \left(a_i^H\right)_1 + \left(a_i^H\right)_2 + \left(a_i^H\right)_3,$$

avec :

$$\left(a_i^H\right)_0 = -\sum_{j=2}^{n-1} m_j \frac{r_j^H}{\left\|r_j^H\right\|^3},$$

$$\left(a_i^H\right)_1 = m_0 \left(\frac{r_i^J}{\left\|r_i^J\right\|^3} - \frac{r_i^H}{\left\|r_i^H\right\|^3}\right),$$

$$\left(a_i^H\right)_2 = \sum_{j=2}^{i} \frac{m_j m_0}{\sum_{k=0}^{j-1} m_k} \frac{r_j^J}{\left\|r_j^J\right\|^3}$$

et

$$\left(a_i^H\right)_3 = \sum_{1\leq j<n, j\neq i} \frac{m_i r_{ji}}{\|r_{ji}\|^3}.$$

Plus d'informations concernant les différents calculs effectués pour arriver à de tels résultats sont disponibles dans le document kick.ps présent dans le sous-dossier getacch. Ces calculs étant associés à l'intégrateur WHM, nous ne les détaillerons pas dans ce manuscrit.

Les différentes fonctions présentes dans le sous-dossier getacch sont :

- *getacch.f* : calcule l'accélération des particules massives en coordonnées héliocentriques.
- *getacch_ah0.f* : calcule $\left(a_i^H\right)_0$ de chaque particule.
- *getacch_ah1.f* : calcule $\left(a_i^H\right)_1$ de chaque particule.
- *getacch_ah2.f* : calcule $\left(a_i^H\right)_2$ de chaque particule.
- *getacch_ah3.f* : calcule $\left(a_i^H\right)_3$ de chaque particule massive.
- *getacch_ah3_tp.f* : calcule $\left(a_i^H\right)_3$ de chaque particule non massive.
- : *getacch_ir3.f* : calcule $\frac{1}{\|r_i\|}$ et $\frac{1}{\|r_i\|^3}$ pour un ensemble de particules en coordonnées héliocentriques.
- *getacch_tp.f* : calcule l'accélération des particules non massives en coordonnées héliocentriques.

Kickvh

Ce sous-dossier contient les fonctions permettant d'effectuer l'évolution due à une accélération. Il permet donc de calculer les formules de la forme :

$$v_i(t_{k+1}) = a(t_k)\tau + v_i(t_k). \tag{2.3.1}$$

Cette formule sera utilisée pour l'évolution sous chaque hamiltonien ne dépendant que des positions.

Deux fonctions composent ce sous-dossier :

- *kickvh.f* : applique la formule (2.3.1) pour les particules massives.
- *kickvh_tp.f* : applique la formule (2.3.1) pour les particules non massives.

Step

Ce sous-dossier contient les fonctions permettant d'effectuer l'évolution telle que décrite aux équations (2.2.16) et (2.2.17). Les fonctions qu'il contient sont les suivantes :

- *step_dkd.f* : effectue (2.2.17) sur toutes les particules.
- *step_dkd_pl.f* : effectue (2.2.17) sur les particules massives.
- *step_dkd_tp.f* : effectue (2.2.17) sur les particules non massives.

- *step_kdk.f* : effectue (2.2.16) sur toutes les particules.
- *step_kdk_pl.f* : effectue (2.2.16) sur les particules massives.
- *step_kdk_tp.f* : effectue (2.2.16) sur les particules non massives.

2.3.8 RMVS

Ce dossier contient les fonctions associées à l'intégrateur symplectique RMVS décrit à la section 2.2.5. Cependant, cet intégrateur étant moins efficace que l'intégrateur SyMBA, une seule fonction de ce dossier nous sera utile :

- *rmvs_chk_ind.f* : teste si des rencontres se produisent entre les différents corps traités par l'intégrateur.

2.3.9 SyMBA

Ce dossier contient les fonctions de l'intégrateur SyMBA présenté à la section 2.2.7. Ce dossier contient les fonctions suivantes :

- *swift_symba7.f* : programme principal contenant l'intégrateur SyMBA et contenu dans le dossier main.
- *symba7_chk.f* : teste si des corps se rencontrent au cours de l'intégration.
- *symba7_getacch.f* : calcule $-\frac{1}{m_j}\frac{\partial \mathcal{H}^{HC}_{IntNR}}{\partial r_{j,c}}$ qui correspond à l'accélération induite par $\frac{\partial \mathcal{H}^{HC}_{IntNR}}{\partial r_{j,c}}$ et sert donc à effectuer l'évolution sous cette quantité au moyen de kickvh.f.
- *symba7_helio_drift.f* : fait évoluer le système par rapport à $\mathcal{H}^{HC}_{Kep}$ pour les corps dont l'intégration par rapport à $\mathcal{H}^{HC}_{Int}$ a déjà été réalisée. Comme précisé à la section 2.2.7, les corps qui ne sont plus influencés par l'intégration sous $\sum_{i=l}^{\infty} \mathcal{H}^{HC}_{IntR,i}$, peuvent évoluer directement sous $\mathcal{H}^{HC}_{Kep}$.
- *symba7_helio_getacch.f* : calcule $-\frac{1}{m_j}\frac{\partial \mathcal{H}^{HC}_{Int}}{\partial r_{j,c}}$ pour toutes les particules quand il n'y a pas de rencontres. Cette quantité correspond à l'accélération induite par $\frac{\partial \mathcal{H}^{HC}_{Int}}{\partial r_{j,c}}$ et permet donc de faire évoluer les corps sous cette quantité.
- *symba7_kick.f* : calcule

$$v^B_{i,c}(t_{k+1}) = v^B_{i,c}(t_k) + sgn \times \tau \sum_{j=2, j\neq i}^{n} \left[\frac{G m_j}{\|r_{ij}(t_k)\|^3} \left(r^{HC}_{i,c}(t_k) - r^{HC}_{j,c}(t_k) \right) \bar{f}_l(\|r_{ij}(t_k)\|) \right]$$

  pour tous les corps avec en paramètre l'indice l et le nombre sgn. Ce dernier peut être égal à 1 ou -1. Afin de réaliser l'évolution (2.2.36), le code va alors appliquer deux fois cette fonction. Une première fois en utilisant les paramètres $l = m$ et $sgn = -1$ et avec $l = m - 1$ et $sgn = 1$ la deuxième fois.
- *symba7_merge.f* : teste pour voir s'il y a des rencontres et fusionne les corps se rencontrant.
- *symba7_nbodm.f* : renvoie l'indice du dernier objet massif présent dans le système.
- *symba7_step_helio.f* : effectue l'intégration Helio décrite par l'équation (2.2.6).
- *symba7_step_interp.f* : applique l'intégrateur SyMBA décrit à la section 2.2.7. Cette fonction calcule donc l'évolution de la formule (2.2.33).
- *symba7_step_pl.f* : teste s'il y a des corps proches et choisit l'intégrateur approprié à la situation, à savoir SyMBA en cas de rencontres et Helio sinon.
- *symba7_step_recur.f* : évolue le système sous $\mathcal{H}^{HC}_{IntR} + \mathcal{H}^{HC}_{Kep}$ dans le cas où il y a des rencontres.

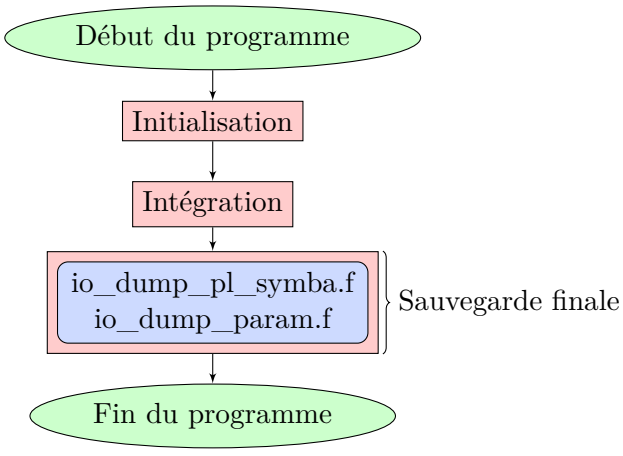
2.3.10 Orbel et Util

Ces dossiers contiennent un ensemble d'outils divers et variés. Nous explicitons uniquement les fonctions utilisées par SyMBA. Celles-ci sont les suivantes :

- *orbel_scget.f* : calcule le cosinus et le sinus d'une valeur entrée.
- *orbel_xv2aeq.f* : calcule l'excentricité, la distance périhélique et le demi-grand axe de l'orbite d'un corps sur base de sa position et de sa vitesse dans un repère cartésien.
- *orbel_xv2el.f* : calcule l'anomalie moyenne et les caractéristiques de l'orbite d'un corps sur base de sa position et de sa vitesse dans un repère cartésien. Les caractéristiques de l'orbite sont son type, son demi-grand axe, son inclinaison, son excentricité, l'argument de son périhélie, la longitude de son nœud ascendant et son anomalie moyenne.
- *util_exit.f* : met fin au programme en précisant si le programme s'est terminé normalement ou s'il s'est arrêté à cause d'une erreur.
- *util_hills.f* : calcule le carré du rayon des sphères de Hill des différents corps manipulés par l'intégrateur.
- *util_hills1.f* : calcule le rayon des sphères de Hill des différents corps manipulés par l'intégrateur.
- *util_mass_peri.f* : évalue si des corps sont au périhélie et si tel est le cas, calcule la distance périhélique.
- *util_version.f* : affiche à l'écran la version de l'intégrateur utilisée.

2.4 Organigramme

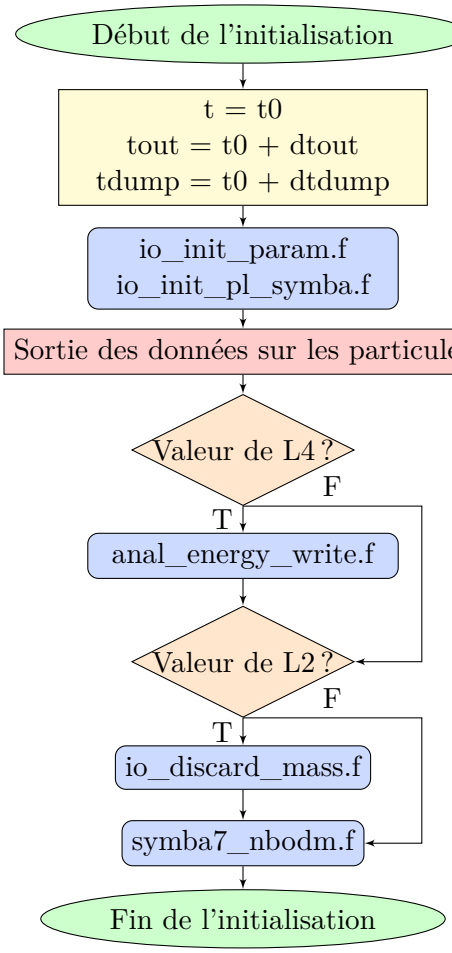
Nous allons à présent réaliser un organigramme du code SyMBA afin de mettre en lumière l'articulation des sous-routines décrites précédemment. Nous pouvons tout d'abord obsrver que le programme se découpe en trois phases disctinctes :



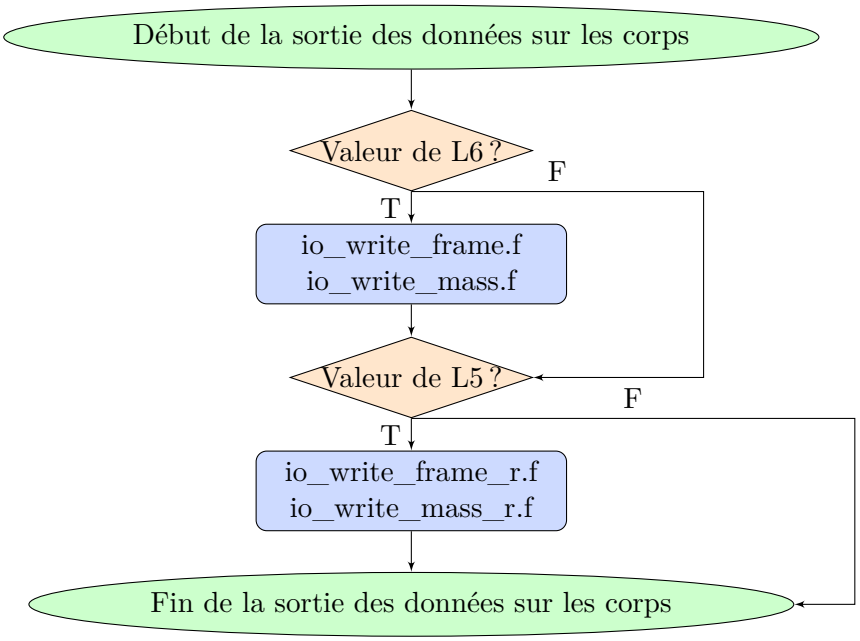
Notons que la phase de sauvegarde ne comprend que deux sous-routines examinées à la section 2.3.6. Les deux autres parties sont plus complexes et font l'objet des diagrammes suivants.

2.4.1 Initialisation

Schématiquement, la partie initialisation se résume comme suit :

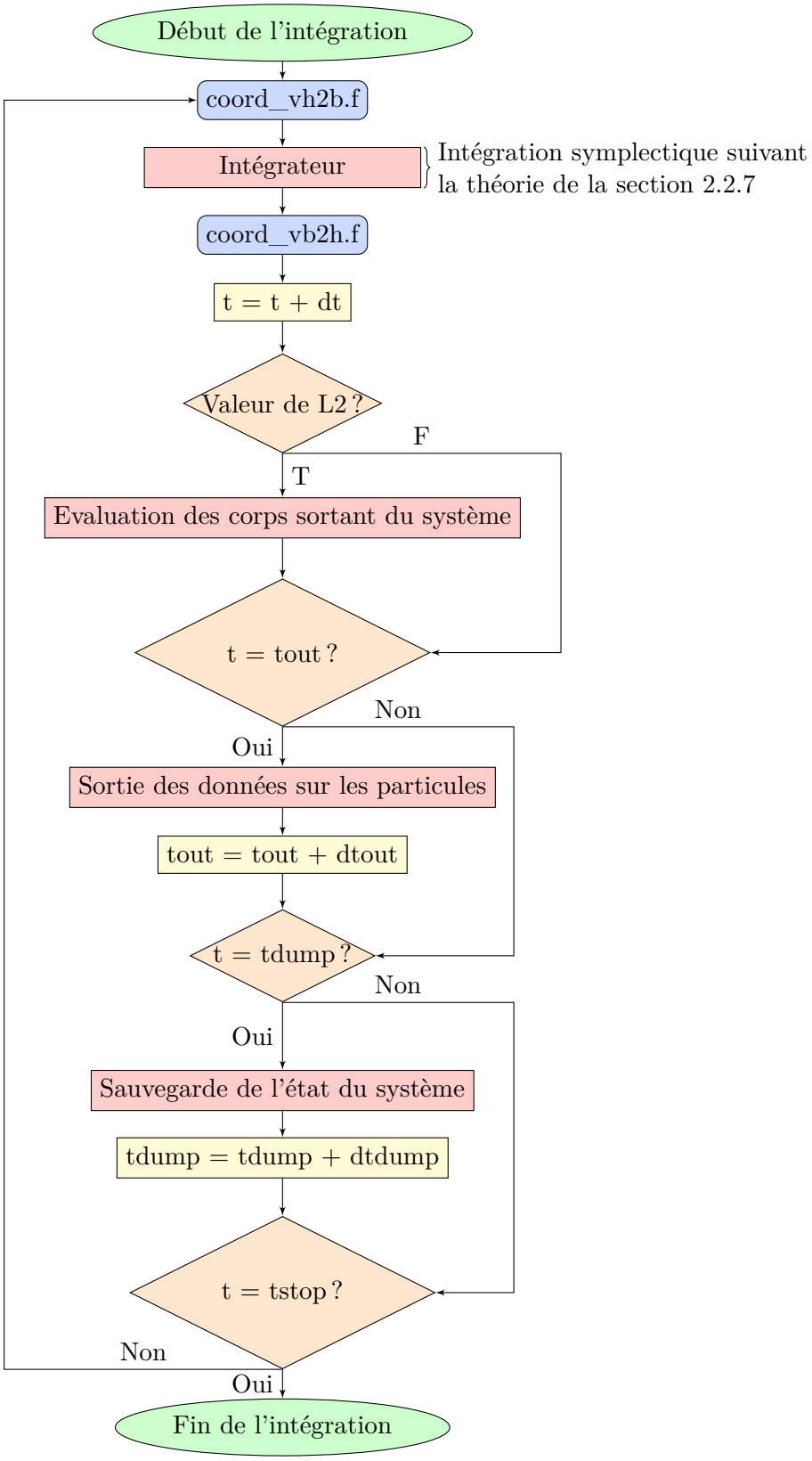


L'étape "Sortie des données sur les corps" consiste en :



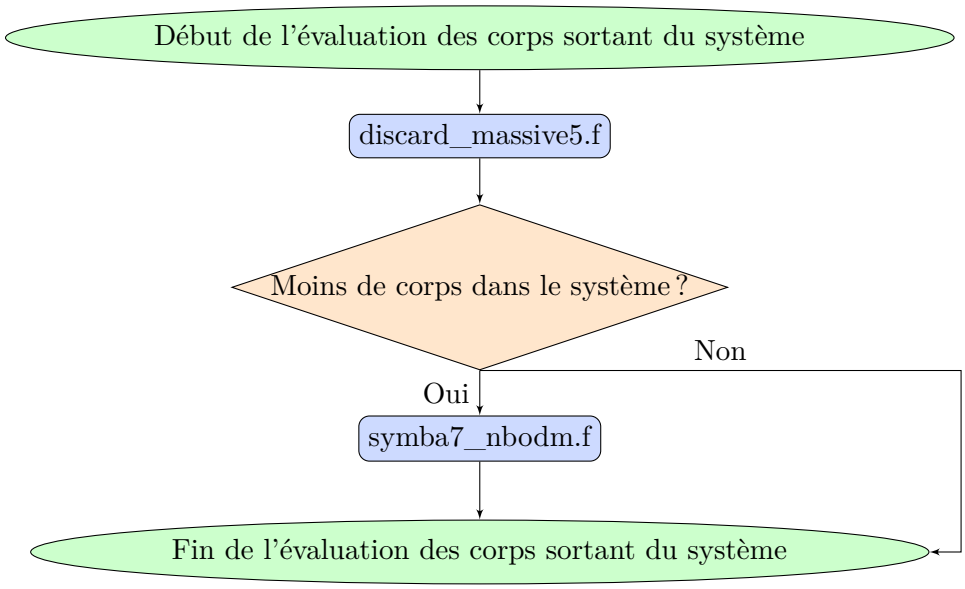
2.4.2 Intégration

Nous présentons à présent l'organigramme correspondant à la phase d'intégration du système.

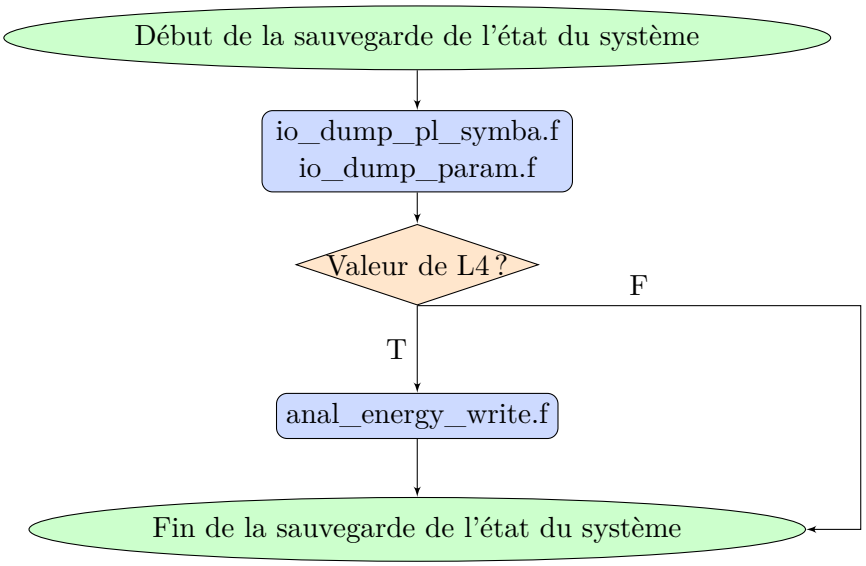


Remarquons sur ce graphe la transformation des vitesses en vitesses barycentriques afin d'être dans le repère héliocentrique canonique. Les étapes intermédiaires (en rouge sur le diagramme) sont décrites ci-après, à l'exception de la sortie des données sur les particules qui a déjà été discutée.

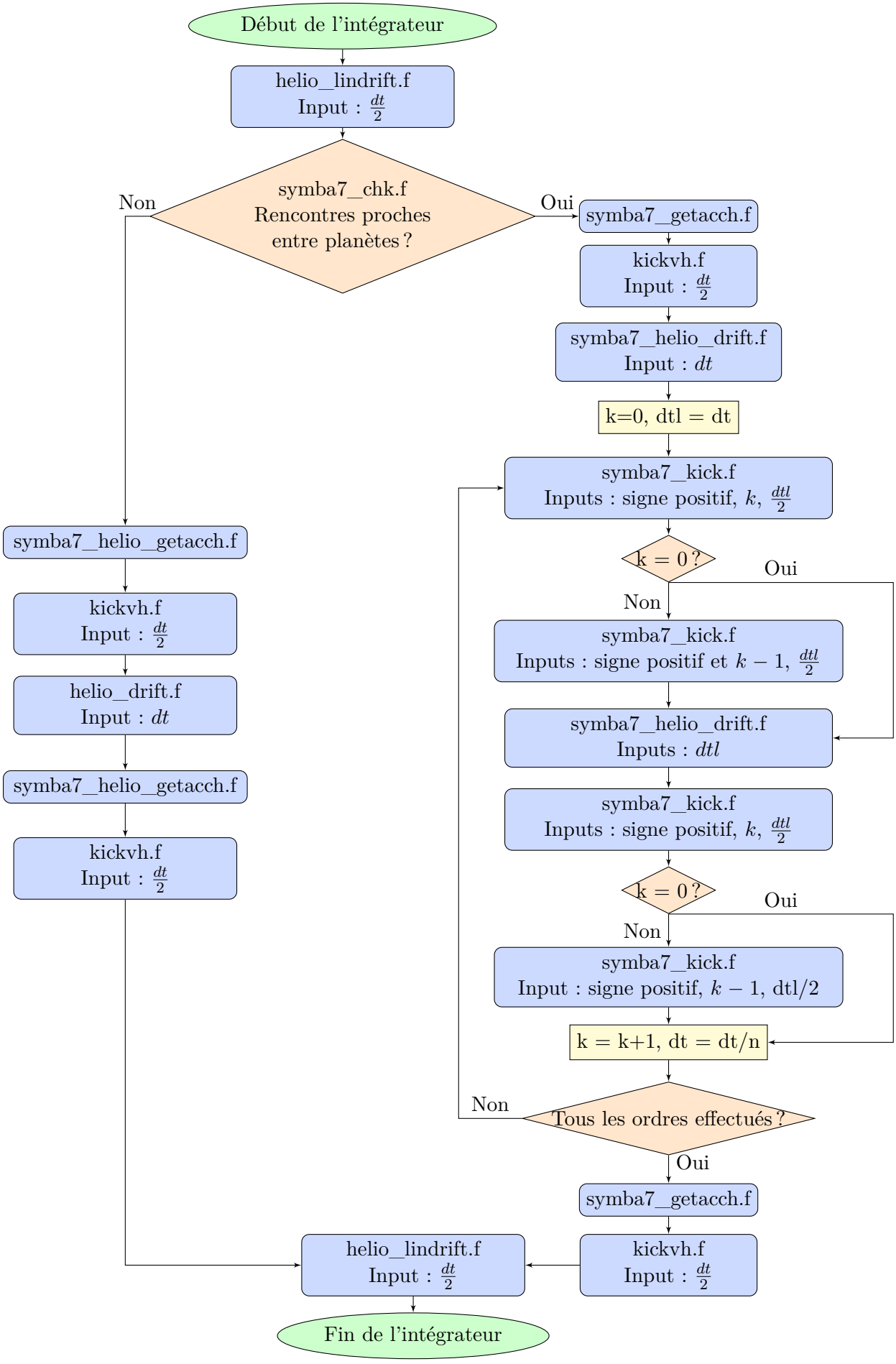
Evaluation des corps sortant du système



Sauvegarde de l'état du système



Intégrateur



Chapitre 3

Algorithme pour les systèmes binaires

Dans ce chapitre nous allons présenter les modifications que nous avons effectuées dans SyMBA afin de l'adapter aux systèmes binaires.

Ces changements sont nécessaires. En effet, l'ajout d'un deuxième corps stellaire dans SyMBA peut entraîner la violation des contraintes (2.2.15). Afin d'illustrer cette violation, nous avons étudié un système constitué du soleil et des quatre planètes géantes du système solaire (Jupiter, Saturne, Uranus et Neptune) avec et sans compagnon stellaire. Les conditions initiales utilisées pour l'intégration sont listées à la table 3.1.

Notons que toutes les conditions initiales dans ce manuscrit seront données en éléments orbitaux. Ceux-ci sont : le demi-grand axe (a), l'excentricité (e), l'inclinaison (i), la longitude du noeud ascendant (Ω), l'argument du périastre (ω) et l'anomalie moyenne (M). Plus d'informations concernant ces éléments peuvent être trouvées dans [28]. De plus, les masses des corps stellaires seront toujours données en masse solaire (MS), celles des planètes seront données en masse de Jupiter (MJ), les distances seront référencées en unité astronomique (UA [1]) et les angles seront exprimés en degré. En outre, la valeur 10^{-7} sera utilisée en lieu et place de la valeur zéro afin d'éviter les problèmes numériques.

Nous avons comparé les valeurs $\mathcal{H}^{HC}_{Kep}$, $\mathcal{H}^{HC}_{Jump}$, $\mathcal{H}^{HC}_{Int}$, ainsi que ϵ_1 et ϵ_2, pour le système de la table 3.1, avec et sans le compagnon stellaire (corps 2), afin d'évaluer les contraintes (2.2.15). Ces valeurs sont représentées aux figures 3.1 et 3.2. Nous pouvons remarquer que, lors de l'ajout

Corps	Masse	a (UA)	e	i (°)	Ω (°)	ω (°)	M (°)
1	1 MS	Etoile centrale					
2	1 MS	160	0.25	10^{-7}	50	45	10^{-7}
3	1 MJ	5.2033	0.0484	1.305	100.556	275.066	10
4	0.299 MJ	9.555	0.0542	2.484	113.715	338.717	10^{-7}
5	0.0457 MJ	19.189	0.0473	0.77	73.99	96.541	18
6	0.054 MJ	30.104	0.00859	29.58	131.722	273.25	130

TABLE 3.1 – Conditions initiales du système solaire restreint aux quatre planètes géantes avec un compagnon stellaire (Sources : [49])

1. 1 UA ≈ 149 597 870 km et correspond approximativement à la distance entre la Terre et le Soleil

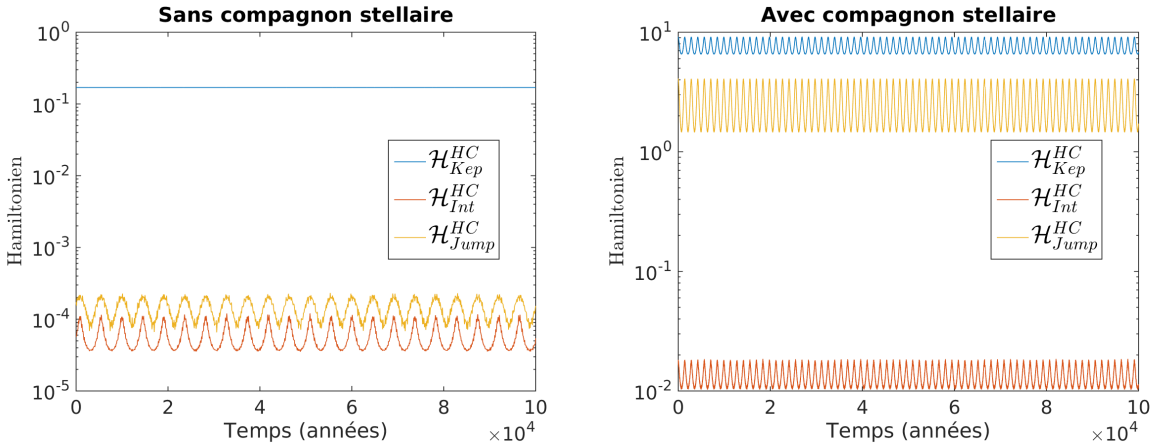


FIGURE 3.1 – Comparaison des valeurs des différentes parties de $\mathcal{H}^{HC}$ au cours de l'intégration du système de la table 3.1

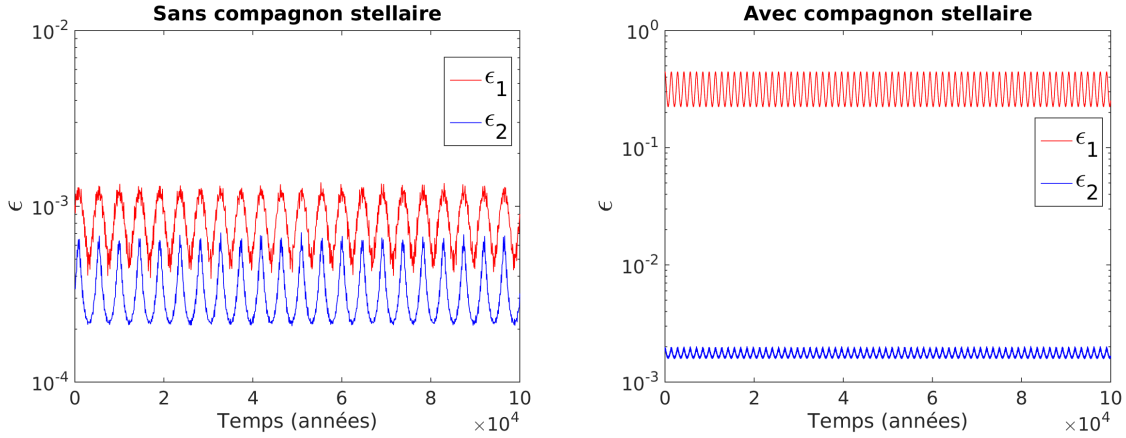


FIGURE 3.2 – Comparaison des valeurs de ϵ_1 et ϵ_2 au cours de l'intégration du système de la table 3.1 avec SyMBA

d'un deuxième corps stellaire, les conditions (2.2.15) peuvent être violées. En effet, dans ce cas-ci, ϵ_1 et ϵ_2, définis, conformément aux formules de la section 2.2.3, par

$$\epsilon_1 = \frac{\mathcal{H}^{BE}_{Jump}}{\mathcal{H}^{BE}_{Int} + \mathcal{H}^{BE}_{Kep}} \quad \text{et} \quad \epsilon_2 = \frac{\mathcal{H}^{BE}_{Int}}{\mathcal{H}^{BE}_{Kep}}$$

vont être multipliés par un facteur 10^3 et 10 respectivement. Cette augmentation est due à l'augmentation de $\mathcal{H}^{HC}_{Int}$ et $\mathcal{H}^{HC}_{Jump}$ d'un facteur 10^2 et 10^4 respectivement, alors que $\mathcal{H}^{HC}_{Kep}$ augmente d'un facteur 10. Notons que ces facteurs sont des ordres de grandeur. En effet, le but est de quantifier approximativement l'augmentation non négligeable de ϵ_1 et ϵ_2 en cas d'ajout d'un compagnon binaire.

Cette augmentation va donc entraîner la violation des contraintes (2.2.15). L'erreur (2.2.14) va alors augmenter. Afin de garder cette erreur au moins aussi bonne lors de l'ajout du compagnon stellaire, il faudrait diviser le pas par $\sqrt{max(\alpha_1, \alpha_2)}$, où α_1 et α_2 sont les facteurs d'augmentation de ϵ_1 et ϵ_2 respectivement. En effet si $\tilde{\epsilon}_1 = \alpha_1\epsilon_1$, $\tilde{\epsilon}_2 = \alpha_2\epsilon_2$ et $\bar{\tau} = \frac{\tau}{\sqrt{max(\alpha_1,\alpha_2)}}$, l'erreur (2.2.14) devient

$$\begin{aligned}(\bar{\epsilon_1} + \bar{\epsilon_2})\bar{\tau}^2 &= (\alpha_1\epsilon_1 + \alpha_2\epsilon_2)\left[\frac{\tau}{\sqrt{max(\alpha_1, \alpha_2)}}\right]^2 \\ &= \left(\frac{\alpha_1}{max(\alpha_1, \alpha_2)}\epsilon_1 + \frac{\alpha_2}{max(\alpha_1, \alpha_2)}\epsilon_2\right)\tau^2\end{aligned}$$

$$\leqslant (\epsilon_1 + \epsilon_2)\tau^2.$$

Dans le cas illustré aux figures 3.1 et 3.2, nous avons

$$\sqrt{max(\alpha_1, \alpha_2)} \approx \sqrt{max(10, 10^3)} = \sqrt{10^3} \approx 31.$$

Le pas devrait donc être réduit approximativement d'un facteur 31 pour obtenir une intégration aussi bonne lors de l'ajout d'un compagnon binaire. Or réduire le pas d'un tel facteur entraîne l'augmentation du temps d'intégration d'un facteur équivalent. De plus, SyMBA a déjà, dans certains cas, un temps d'intégration de l'ordre de l'heure ou de la journée. Le multiplier par un facteur 30 n'est donc pas envisageable. Afin de solutionner ce problème, nous allons utiliser un nouveau jeu de coordonnées dans le but de partitionner l'hamiltonien en respectant les contraintes (2.2.15). Ce jeu de coordonnées provient de [10] et sera explicité dans la suite de ce chapitre (section 3.1). Nous allons ensuite décrire le nouvel intégrateur construit sur base de ces coordonnées (sections 3.2 et 3.3). Enfin nous terminerons par présenter l'ensemble des tests de validation qui ont été effectués (section 3.4).

3.1 Coordonnées

Tout d'abord, nommons les deux étoiles de la binaire A et B. De plus, dans ce manuscrit nous ne souhaitons étudier que le cas des planètes de type S. Nous considérons donc qu'il y a $n-2$ planètes de type S en orbite autour de l'étoile A et qu'il n'y a pas de planète en orbite autour de l'étoile B ou du système constitué des deux corps stellaires. Notons qu'afin de coller au code, nous allons numéroter les planètes de 3 à n dans ce chapitre. En effet, les corps numéro 1 et 2 seront l'étoile A et l'étoile B respectivement.

Nous allons définir un nouveau jeu de coordonnées afin d'obtenir une division de l'hamiltonien respectant les contraintes (2.2.15) pour les planètes de type S. Notons qu'un autre système de coordonnées existe pour les planètes de type P et est également présenté dans [10]. De plus, le jeu de coordonnées présenté dans cette section n'est valable que pour les compagnons stellaires éloignés. Il ne peut être appliqué dans le cas d'un compagnon stellaire proche où les planètes pourraient être attirées par cette seconde masse.

Avant de présenter les nouvelles coordonnées, nous allons définir les deux quantités

$$m_{sys} = m_A + m_B + \sum_{i=3}^{n} m_i$$

G	constante gravitationnelle
m_i	masse du corps i
$\tilde{p}_i$	moment cinétique du corps i dans un référentiel inertiel
p_i^{BE}	moment associé du corps i dans un repère binaire éloigné
$\tilde{p}_i^H$	moment cinétique du corps i dans un repère héliocentrique
r_i	position du corps i dans un référentiel inertiel
r_i^{BE}	position du corps i dans un repère binaire éloigné
r_{ij}	distance entre le corps i et le corps j
v_i^{BE}	vitesse du corps i dans un repère binaire éloigné
v_i^H	vitesse du corps i dans un repère héliocentrique

TABLE 3.2 – Notations utilisées dans ce chapitre

et

$$m_{red} = m_A + \sum_{i=3}^{n} m_i,$$

qui sont respectivement la masse totale du système et la masse du système "réduit", sans le compagnon stellaire. L'ensemble des notations utilisées dans ce chapitre est résumé à la table 3.2.

Positions

Nous définissons le système de coordonnées du repère que nous appellerons repère binaire éloignée. Dans ce repère, les positions sont exprimées en fonction des positions héliocentriques par

$$\begin{cases} r_A^{BE} &= \frac{m_A r_A^H + m_B r_B^H + \sum_{j=3}^{n} m_j r_j^H}{m_{sys}} \\ r_i^{BE} &= r_i^H - r_A^H \quad i = 3...n \\ r_B^{BE} &= r_B^H - \frac{m_A r_A^H + \sum_{j=1}^{n} m_j r_j^H}{m_{red}}. \end{cases} \tag{3.1.1}$$

Chaque planète est donc repérée en prenant l'étoile A comme origine du repère, l'étoile B est repérée par rapport au centre de masse du système restreint à l'étoile A et aux planètes et r_A^{BE} est le centre de masse du système.

De plus, comme SyMBA considère l'étoile A comme le corps central pour le repère héliocentrique, nous avons $r_A^H = 0$. Le système (3.1.1) devient alors

$$\begin{cases} r_A^{BE} &= \frac{m_B r_B^H + \sum_{j=3}^{n} m_j r_j^H}{m_{sys}} \\ r_i^{BE} &= r_i^H \quad i = 3...n \\ r_B^{BE} &= r_B^H - \frac{\sum_{j=3}^{n} m_j r_j^H}{m_{red}}. \end{cases} \tag{3.1.2}$$

Afin d'effectuer les sorties du code, nous devons être capable de passer des coordonnées binaire éloignée aux coordonnées héliocentriques. Nous inversons donc le système (3.1.2). Cette inversion est aisée car $r_A^H = 0$ et $r_i^{BE} = r_i^H$. Nous avons alors la relation

$$\begin{cases} r_A^H &= 0 \\ r_i^H &= r_i^{BE} \quad i = 3...n \\ r_B^H &= r_B^{BE} + \frac{\sum_{j=3}^{n} m_j r_j^{BE}}{m_{red}}. \end{cases} \tag{3.1.3}$$

Nous sommes maintenant capable de passer d'un jeu de coordonnées à l'autre. Ces transformations sur les positions ont été implémentées dans les fonctions *coord_posh2bin.f* et *coord_posbin2h.f* pour les systèmes (3.1.2) et (3.1.3) respectivement. Les codes de ces deux fonctions sont disponibles à l'Annexe A.1.

Moments associés et vitesses

Nous allons à présent nous intéresser aux moments associés à ces positions. Ceux-ci sont déterminés dans l'annexe de l'article [11] au moyen de fonctions génératrices. Ils sont définis par

$$\begin{cases} p_A^{BE} &= \tilde{p}_A^H + \tilde{p}_B^H + \sum\limits_{j=3}^{n} \tilde{p}_j^H \\ p_i^{BE} &= \tilde{p}_i^H - m_i \left(\frac{\tilde{p}_A^H + \sum\limits_{j=3}^{n} \tilde{p}_j^H}{m_{red}} \right) \quad i = 3...n \\ p_B^{BE} &= \tilde{p}_B^H - m_B \left(\frac{\tilde{p}_A^H + \tilde{p}_B^H + \sum\limits_{j=3}^{n} \tilde{p}_j^H}{m_{sys}} \right). \end{cases}$$

Notons que le recours aux fonctions génératrices permet d'avoir une transformation de coordonnées symplectiques. Comme explicité précédemment, la symplecticité du changement de coordonnées est indispensable à l'utilisation des intégrateurs symplectiques.

Comme le corps central du repère héliocentrique est l'étoile A, nous avons $\tilde{p}_A^H = m_A v_A^H = 0$. Le système se réécrit alors

$$\begin{cases} p_A^{BE} &= \tilde{p}_B^H + \sum\limits_{j=3}^{n} \tilde{p}_j^H \\ p_i^{BE} &= \tilde{p}_i^H - m_i \left(\frac{\sum\limits_{j=3}^{n} \tilde{p}_j^H}{m_{red}} \right) \quad i = 3...n \\ p_B^{BE} &= \tilde{p}_B^H - m_B \left(\frac{\tilde{p}_B^H + \sum\limits_{j=3}^{n} \tilde{p}_j^H}{m_{sys}} \right). \end{cases} \tag{3.1.4}$$

De plus, afin d'être cohérent avec SyMBA, nous allons faire évoluer les vitesses et non les moments associés. Pour ce faire, nous introduisons les vitesses

$$\begin{cases} p_A^{BE} &= m_a v_A^{BE} \\ p_i^{BE} &= m_i v_i^{BE} \quad i = 3...n \\ p_B^{BE} &= \mu_{BE} v_B^{BE}, \end{cases} \tag{3.1.5}$$

où μ_{BE} est défini dans [10] par

$$\mu_{BE} = \frac{m_B m_{red}}{m_{sys}}. \tag{3.1.6}$$

Le choix de ces vitesses est motivé par la forme de la partie képlérienne de l'hamiltonien exprimé dans ces nouvelles coordonnées qui sera explicitée plus loin dans ce chapitre.

Le système (3.1.4) se réécrit en terme de vitesses :

$$\begin{cases} m_A v_A^{BE} & = \tilde{p}_B^H + \sum\limits_{j=3}^{n} \tilde{p}_j^H \\ m_i v_i^{BE} & = \tilde{p}_i^H - m_i \left(\dfrac{\sum\limits_{j=3}^{n} \tilde{p}_j^H}{m_{red}} \right) \quad i = 3...n \\ \mu_{BE} v_B^{BE} & = \tilde{p}_B^H - m_B \left(\dfrac{\tilde{p}_B^H + \sum\limits_{j=3}^{n} \tilde{p}_j^H}{m_{sys}} \right). \end{cases}$$

Si nous isolons les vitesses dans le repère binaire éloignée, nous avons

$$\begin{cases} v_A^{BE} & = \dfrac{m_B v_B^H + \sum\limits_{j=3}^{n} m_j v_j^H}{m_A} \\ v_i^{BE} & = v_i^H - \dfrac{\sum\limits_{j=3}^{n} m_j v_j^H}{m_{red}} \quad i = 3...n \\ v_B^{BE} & = \frac{m_{sys}}{m_{red}} \left(v_B^H - \dfrac{m_B v_B^H + \sum\limits_{j=3}^{n} m_j v_j^H}{m_{sys}} \right). \end{cases} \tag{3.1.7}$$

Notons que nous avons utilisé la définition des moments cinétiques :

$$\tilde{p}_i^H = m_i v_i^H \quad i = A, B, 3, ..., n,$$

afin de simplifier l'expression précédente.

Nous pouvons également simplifier l'expression de la vitesse v_B^{BE} du système (3.1.7) :

$$\begin{aligned} v_B^{BE} &= \frac{m_{sys}}{m_{red}} \left(v_B^H - \frac{m_B v_B^H + \sum\limits_{j=3}^{n} m_j v_j^H}{m_{sys}} \right) \\ &= \frac{m_{sys}}{m_{red}} \left(\frac{m_{sys} v_B^H - m_B v_B^H - \sum\limits_{j=3}^{n} m_j v_j^H}{m_{sys}} \right) \\ &= \frac{m_{sys}}{m_{red}} \left(\frac{(m_{sys} - m_B) v_B^H - \sum\limits_{j=3}^{n} m_j v_j^H}{m_{sys}} \right) \\ &= \frac{m_{sys}}{m_{red}} \left(\frac{m_{red} v_B^H - \sum\limits_{j=3}^{n} m_j v_j^H}{m_{sys}} \right) \\ &= v_B^H - \frac{\sum\limits_{j=3}^{n} m_j v_j^H}{m_{red}}. \end{aligned}$$

Le système (3.1.7) se réécrit alors :

$$\begin{cases} v_A^{BE} &= \dfrac{m_B v_B^H + \sum\limits_{j=3}^{n} m_j v_j^H}{m_A} \\ v_i^{BE} &= v_i^H - \dfrac{\sum\limits_{j=3}^{n} m_j v_j^H}{m_{red}} \quad i = 3...n \\ v_B^{BE} &= v_B^H - \dfrac{\sum\limits_{j=3}^{n} m_j v_j^H}{m_{red}}. \end{cases} \tag{3.1.8}$$

Les vitesses des planètes et de l'étoile B peuvent alors être vues comme les vitesses barycentriques par rapport au centre de masse du système restreint aux planètes et à l'étoile A. Cette affirmation est vraie car $v_A^H = 0$, d'où $\dfrac{\sum\limits_{j=1}^{n} m_j v_j^{HC}}{m_{red}} = \dfrac{m_A v_A^H + \sum\limits_{j=1}^{n} m_j v_j^{HC}}{m_{red}}$.

Comme pour les positions, il nous reste maintenant à inverser la transformation des vitesses (3.1.8) afin de retourner dans le repère héliocentrique à la fin de l'intégration et effectuer les sorties du programme. Pour ce faire, nous calculons

$$\begin{aligned} \sum_{k=3}^{n} m_k v_k^{BE} &= \sum_{k=3}^{n} m_k \left[v_k^H - \left(\frac{\sum\limits_{j=3}^{n} m_j v_j^H}{m_{red}} \right) \right] \\ &= \sum_{k=3}^{n} m_k v_k^H - \sum_{k=3}^{n} m_k \left(\frac{\sum\limits_{j=3}^{n} m_j v_j^H}{m_{red}} \right) \\ &= \sum_{k=3}^{n} m_k v_k^H \left(1 - \frac{\sum\limits_{j=3}^{n} m_j}{m_{red}} \right) \\ &= \sum_{k=3}^{n} m_k v_k^H \frac{m_{red} - \sum\limits_{j=3}^{n} m_j}{m_{red}} \\ &= \sum_{k=3}^{n} m_k v_k^H \frac{m_A}{m_{red}}. \end{aligned}$$

Nous avons donc

$$\frac{\sum\limits_{j=3}^{n} m_j v_j^H}{m_{red}} = \frac{\sum\limits_{j=3}^{n} m_j v_j^{BE}}{m_A}.$$

Sur base de cette égalité et du fait que $v_A^H = 0$ (car A est le corps central du système dans le repère héliocentrique), nous pouvons inverser la relation (3.1.8). Nous avons alors :

$$\begin{cases} v_A^H &= 0 \\ v_i^H &= v_i^{BE} + \dfrac{\sum\limits_{j=3}^{n} m_j v_j^{BE}}{m_A} \quad i = 3...n \\ v_B^H &= v_B^{BE} + \dfrac{\sum\limits_{j=3}^{n} m_j v_j^{BE}}{m_A}. \end{cases} \tag{3.1.9}$$

Les fonctions *coord_vh2bin.f* et *coord_vbin2h.f* calculent les transformations (3.1.8) et (3.1.9) respectivement. Les codes de ces fonctions sont inclus dans l'Annexe A.1.

Grâce à ces nouvelles coordonnées, nous allons pouvoir décomposer l'hamiltonien de manière à respecter les contraintes (2.2.15), comme nous le montrons à la section suivante.

3.2 Intégrateur

La nouvelle décomposition de l'hamiltonien, en coordonnées binaire éloignée, est alors donnée par

$$\mathcal{H} = \mathcal{H}^{BE}_{Kep} + \mathcal{H}^{BE}_{Int} + \mathcal{H}^{BE}_{Jump},$$

avec

$$\mathcal{H}^{BE}_{Kep} = \frac{\left\|p^{BE}_B\right\|^2}{2\mu_{BE}} - \frac{Gm_{sys}\mu_{BE}}{\|r^{BE}_B\|} + \sum_{i=3}^{n}\left(\frac{\left\|p^{BE}_i\right\|^2}{2m_i} - \frac{Gm_Am_B}{\|r^{BE}_i\|}\right),$$

$$\begin{aligned}\mathcal{H}^{BE}_{int} = &-\sum_{i=3}^{n-1}\sum_{i<j\leqslant n}\frac{Gm_im_j}{\|r_{ij}\|} + Gm_Am_B\left(\frac{1}{\|r^{BE}_B\|} - \frac{1}{\|r^{BE}_B + s\|}\right)\\ &+ Gm_B\sum_{i=3}^{n}m_i\left(\frac{1}{\|r^{BE}_B\|} - \frac{1}{\|r^{BE}_B - r^{BE}_i + s\|}\right)\end{aligned}$$

et

$$\mathcal{H}^{BE}_{Jump} = \frac{\left\|\sum_{i=3}^{n} p^{BE}_i\right\|^2}{2m_A}, \tag{3.2.1}$$

où

$$s = \frac{\sum_{i=3}^{n} m_i r^{BE}_i}{m_{red}}\ ,\ r_{ij} = r^{BE}_i - r^{BE}_j.$$

Remarquons que l'intégration de cet hamiltonien peut poser problème pour des planètes ayant des rencontres proches. A nouveau, $\mathcal{H}^{BE}_{Int}$ est susceptible d'augmenter drastiquement. Nous allons donc discuter dans la suite des évolutions avec et sans rencontres proches.

Notons que cette nouvelle découpe permet bien de régler les soucis dus à l'ajout du compagnon stellaire. En effet, si nous considérons à nouveau le système de la table 3.1, lorsque nous traçons les quantités ϵ_1, ϵ_2, $\mathcal{H}^{BE}_{Kep}$, $\mathcal{H}^{BE}_{Int}$ et $\mathcal{H}^{BE}_{Jump}$, nous retrouvons bien des ordres de grandeur similaires à ceux du système sans compagnon binaire présenté à la section 2.2.6. Nous pouvons le constater en comparant la figure 3.3 à la partie sans compagnon stellaire des figures 3.1 et 3.2.

De plus, remarquons que r^{BE}_A et p^{BE}_A n'apparaissent dans aucune partie de l'hamiltonien. Ces deux quantités vont donc rester constantes lors de l'intégration. Elle ne seront donc jamais discutées dans la suite de ce manuscrit.

3.2.1 Sans rencontres proches

Nous commençons par détailler l'évolution quand aucune rencontre proche n'est présente. Comme pour l'algorithme Helio, nous avons utilisé la formule d'évolution (2.2.13) avec $\mathcal{H}_A = \mathcal{H}^{BE}_{Kep}$, $\mathcal{H}_B = \mathcal{H}^{BE}_{Int}$ et $\mathcal{H}_C = \mathcal{H}^{BE}_{Jump}$. Nous trouvons alors l'équation d'évolution

$$f(t_{k+1}) = \exp\left(\frac{\tau}{2}T^{BE}_{Jump}\right)\exp\left(\frac{\tau}{2}T^{BE}_{Int}\right)\exp\left(\tau T^{BE}_{Kep}\right)\exp\left(\frac{\tau}{2}T^{BE}_{Int}\right)\exp\left(\frac{\tau}{2}T^{BE}_{Jump}\right)f(t_k), \qquad k\in\mathbb{N},$$

où T^{BE}_{Jump}, T^{BE}_{Int} et T^{BE}_{Kep} sont les opérateurs linéaires associés à $\mathcal{H}^{BE}_{Jump}$, $\mathcal{H}^{BE}_{Int}$ et $\mathcal{H}^{BE}_{Kep}$ respectivement. Cette évolution revient donc à d'abord faire évoluer le système d'un demi-pas par

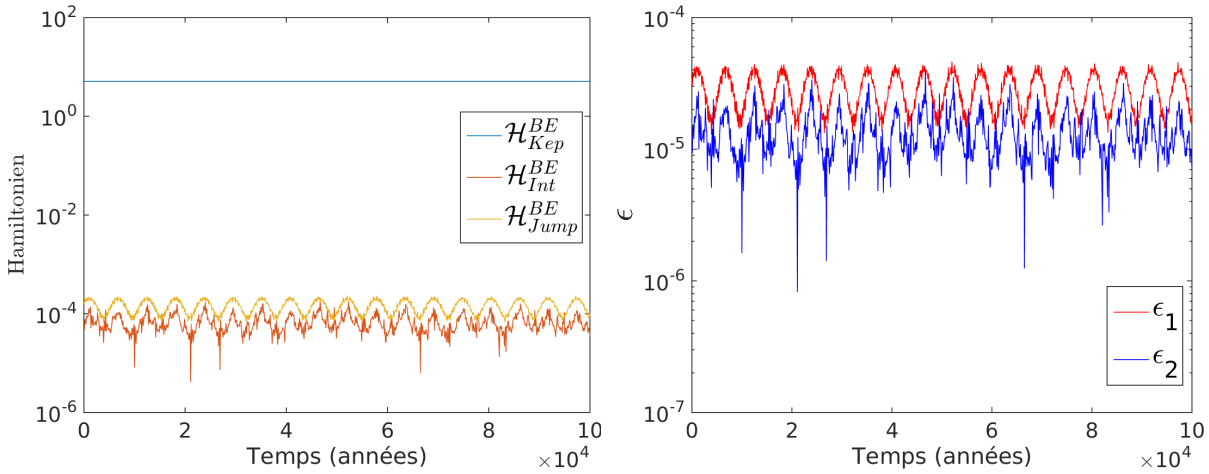


FIGURE 3.3 – les différentes parties de l'hamiltonien ainsi que ϵ_1 et ϵ_2 dans les coordonnées binaire éloignée pour le système décrit à la table 3.1

rapport à $\mathcal{H}^{BE}_{Jump}$, ensuite d'un demi-pas par rapport à $\mathcal{H}^{BE}_{Int}$, puis d'un pas complet par rapport à $\mathcal{H}^{BE}_{Kep}$, par après à nouveau d'un demi-pas par rapport à $\mathcal{H}^{BE}_{Int}$ et pour terminer d'un demi-pas par rapport à $\mathcal{H}^{BE}_{Jump}$.

Nous allons à présent détailler l'évolution sous les différentes parties de l'hamiltonien.

Evolutions sous $\mathcal{H}^{BE}_{Kep}$

Nous commençons par détailler l'évolution sous $\mathcal{H}^{BE}_{Kep}$. Comme pour les autres algorithmes, l'intégration sous cette partie de l'hamiltonien peut être effectuée indépendamment pour chaque corps. En effet, si nous posons

$$\mathcal{H}^{BE}_{Kep,i} = \frac{\left\|p^{BE}_i\right\|^2}{2m_i} - \frac{Gm_A m_B}{\left\|r^{BE}_i\right\|}, \qquad i = 3...n \tag{3.2.2}$$

et

$$\mathcal{H}^{BE}_{Kep,B} = \frac{\left\|p^{BE}_B\right\|^2}{2\mu_{BE}} - \frac{Gm_{sys}\mu_{BE}}{\left\|r^{BE}_B\right\|}, \tag{3.2.3}$$

nous avons

$$\frac{\partial \mathcal{H}^{BE}_{Kep}}{\partial r^{BE}_i} = \frac{\partial \mathcal{H}^{BE}_{Kep,i}}{\partial r^{BE}_i} \text{ et } \frac{\partial \mathcal{H}^{BE}_{Kep}}{\partial p^{BE}_i} = \frac{\partial \mathcal{H}^{BE}_{Kep,i}}{\partial p^{BE}_i}, \qquad i = B, 3, ..., n$$

car $\mathcal{H}^{BE}_{Kep,i}$ est indépendant de p^{BE}_j et r^{BE}_j pour tous les corps en dehors du corps i.

Les équations d'Hamilton (2.2.2) pour $\mathcal{H}^{BE}_{Kep}$ se réécrivent alors

$$\begin{cases} \dfrac{dr^J_i}{dt} &= \dfrac{\partial \mathcal{H}^J_{Kep}}{\partial p^J_i} = \dfrac{\partial \mathcal{H}^J_{Kep,i}}{\partial p^J_i} \\ \dfrac{dp^J_i}{dt} &= -\dfrac{\partial \mathcal{H}^J_{Kep}}{\partial r^J_i} = -\dfrac{\partial \mathcal{H}^J_{Kep,i}}{\partial r^J_i} \end{cases} \qquad i = B, 3, ..., n.$$

Chaque corps i peut donc être évolué indépendamment sous $\mathcal{H}^{BE}_{Kep,i}$ afin de faire évoluer le système complet sous $\mathcal{H}^{BE}_{Kep}$. Nous allons donc détailler l'évolution sous $\mathcal{H}^{BE}_{Kep,i}$.

Tout d'abord, pour les planètes, en utilisant la définition des vitesses (3.1.5), l'équation (3.2.2) devient

$$\mathcal{H}^{BE}_{Kep,i} = \frac{\left\| m_i v_i^{BE} \right\|^2}{2m_i} - \frac{G m_A m_B}{\| r_i^{BE} \|}, \quad i = 3...n,$$

qui est de la forme explicitée à l'équation (2.2.18). L'évolution se fera donc au moyen de formule (2.2.19) avec $\nu = m_A$ et est donnée par

$$\begin{cases} r_i^{BE}(t_{k+1}) &= f(\nu) r_i^{BE}(t_k) + g(\nu) v_i^{BE}(t_k) \\ v_i^{BE}(t_{k+1}) &= \dot{f}(\nu) r_i^{BE}(t_k) + \dot{g}(\nu) v_i^{BE}(t_k) \end{cases} \quad i = 3...n, \quad k \in \mathbb{N}, \tag{3.2.4}$$

avec $\nu = m_1$.

Intéressons-nous à présent à l'évolution sous $\mathcal{H}^{BE}_{Kep,i}$ pour le compagnon stellaire (i.e. $i = B$). Grâce à la définition de v_B^{BE} (3.1.5), l'équation (3.2.3) se réécrit

$$\mathcal{H}^{BE}_{Kep,B} = \frac{\left\| \mu_{BE} v_B^{BE} \right\|^2}{2\mu_{BE}} - \frac{G m_{sys} \mu_{BE}}{\| r_B^{BE} \|}$$

et est de la forme prescrite à l'équation (2.2.18) avec $m = \mu_{BE}$ et $\nu = m_{sys}$. L'évolution sous $\mathcal{H}^{BE}_{Kep,B}$ se fera donc au moyen des équations (2.2.19) avec $\nu = m_{sys}$. Nous avons alors :

$$\begin{cases} r_B^{BE}(t_{k+1}) &= f(\nu) r_B^{BE}(t_k) + g(\nu) v_B^{BE}(t_k) \\ v_B^{BE}(t_{k+1}) &= \dot{f}(\nu) r_B^{BE}(t_k) + \dot{g}(\nu) v_B^{BE}(t_k) \end{cases} \quad k \in \mathbb{N}, \tag{3.2.5}$$

avec $\nu = m_{sys}$. Le choix de définir la vitesse v_B^{BE} tel qu'explicité à l'équation (3.1.5) apparaît clairement ici. En effet, une autre définition de la vitesse aurait rendu l'utilisation des équations (2.2.19) bien moins aisée, voire même impossible.

Evolution sous $\mathcal{H}^{BE}_{Int}$

Nous allons à présent nous intéresser à l'évolution sous $\mathcal{H}^{BE}_{Int}$. Pour ce faire, rappelons les notations (section 2.2.6) :

$$r_i^{BE} = (r_{i,1}^{BE}, r_{i,2}^{BE}, r_{i,3}^{BE}) \text{ et } p_i^{BE} = (p_{i,1}^{BE}, p_{i,2}^{BE}, p_{i,3}^{BE}), \quad i = B, 3, ..., n,$$

où $1, 2$ et 3 représentent les coordonnées x, y et z des corps.

Nous pouvons alors exprimer l'évolution sous $\mathcal{H}^{BE}_{Int}$ au moyen des équations d'Hamilton (2.2.2) :

$$\begin{cases} \dfrac{dr_{i,c}^{BE}}{dt} &= \dfrac{\partial \mathcal{H}^{BE}_{Int}}{\partial p_{i,c}^{BE}} \\ \dfrac{dp_{i,c}^{BE}}{dt} &= -\dfrac{\partial \mathcal{H}^{BE}_{Int}}{\partial r_{i,c}^{BE}} \end{cases} \quad i = B, 3, ..., n, \quad c = 1, 2, 3. \tag{3.2.6}$$

Comme $\mathcal{H}^{BE}_{Int}$ est indépendant de tous les moments associés, nous avons $\frac{\partial \mathcal{H}^{BE}_{Int}}{\partial p_{i,c}^{BE}} = 0$ pour tous les corps. Il nous reste donc à calculer $\frac{\partial \mathcal{H}^{BE}_{Int}}{\partial r_{i,c}^{BE}}$. Notons que ce calcul sera différent pour le compagnon stellaire et pour les planètes. Nous commençons par calculer cette dérivée pour les planètes.

Nous avons

$$
\begin{aligned}
-\frac{\partial \mathcal{H}_{Int}^{BE}}{\partial r_{i,c}^{BE}} = & -\frac{\partial}{\partial r_{i,c}^{BE}}\left[-\sum_{k=3}^{n-1}\sum_{k<j\leqslant n}\frac{Gm_km_j}{\|r_{kj}\|}+Gm_Am_B\left(\frac{1}{\|r_B^{BE}\|}-\frac{1}{\|r_B^{BE}+s\|}\right)\right.\\
& \left.+Gm_B\sum_{k=3}^{n}m_k\left(\frac{1}{\|r_B^{BE}\|}-\frac{1}{\|r_B^{BE}-r_k^{BE}+s\|}\right)\right]\\
= & -\frac{\partial}{\partial r_{i,c}^{BE}}\left[-\sum_{k=3}^{n-1}\sum_{k<j\leqslant n}\frac{Gm_km_j}{\|r_{kj}\|}\right]-\frac{\partial}{\partial r_{i,c}^{BE}}\left[Gm_Am_B\left(\frac{1}{\|r_B^{BE}\|}-\frac{1}{\|r_B^{BE}+s\|}\right)\right]\\
& -\frac{\partial}{\partial r_{i,c}^{BE}}\left[Gm_B\sum_{k=3}^{n}m_k\left(\frac{1}{\|r_B^{BE}\|}-\frac{1}{\|r_B^{BE}-r_k^{BE}+s\|}\right)\right].
\end{aligned}
\tag{3.2.7}
$$

Nous allons calculer ces différentes dérivées de (3.2.7) séparément. Pour la première dérivée, un calcul identique à celui de la section 2.2.6 permet de trouver

$$
-\frac{\partial}{\partial r_{i,c}^{BE}}\left(-\sum_{k=3}^{n-1}\sum_{k<j\leqslant n}\frac{Gm_km_j}{\|r_{kj}\|}\right)=-\sum_{j=3,j\neq i}^{n}\left[\frac{Gm_im_j}{\left\|r_i^{BE}-r_j^{BE}\right\|^3}\left(r_{i,c}^{BE}-r_{j,c}^{BE}\right)\right].
$$

Ensuite, nous calculons la deuxième dérivée de (3.2.7) :

$$
\begin{aligned}
& -\frac{\partial}{\partial r_{i,c}^{BE}}\left[Gm_Am_B\left(\frac{1}{\|r_B^{BE}\|}-\frac{1}{\|r_B^{BE}+s\|}\right)\right]\\
& \quad = -Gm_Am_B\left(\frac{\partial}{\partial r_{i,c}^{BE}}\left[\frac{1}{\|r_B^{BE}\|}\right]-\frac{\partial}{\partial r_{i,c}^{BE}}\left[\frac{1}{\|r_B^{BE}+s\|}\right]\right)\\
& \quad = Gm_Am_B\left(\frac{\partial}{\partial r_{i,c}^{BE}}\left[\left(\sum_{a=1}^{3}\left(r_{B,a}^{BE}+s_a\right)^2\right)^{-\frac{1}{2}}\right]\right)\\
& \quad = Gm_Am_B\left(-\frac{1}{2}\right)\left[\left(\sum_{a=1}^{3}\left(r_{B,a}^{BE}+s_a\right)^2\right)^{-\frac{3}{2}}\right]\left[\frac{\partial}{\partial r_{i,c}^{BE}}\left(\sum_{a=1}^{3}\left(r_{B,a}^{BE}+s_a\right)^2\right)\right]\\
& \quad = -\frac{Gm_Am_B}{2\left\|r_B^{BE}+s\right\|^3}\left[2\sum_{a=1}^{3}\left[\left(r_{B,a}^{BE}+s_a\right)\frac{\partial}{\partial r_{i,c}^{BE}}\left(r_{B,a}^{BE}+s_a\right)\right]\right]\\
& \quad = -\frac{Gm_Am_B}{2\left\|r_B^{BE}+s\right\|^3}\left[2\sum_{a=1}^{3}\left[\left(r_{B,a}^{BE}+s_a\right)\frac{\partial}{\partial r_{i,c}^{BE}}\left(r_{B,a}^{BE}+\frac{\sum_{k=3}^{n}m_kr_{k,a}^{BE}}{m_{red}}\right)\right]\right]\\
& \quad = \frac{-Gm_Am_B}{\left\|r_B^{BE}+s\right\|^3}\left[\sum_{a=1}^{3}\left[\left(r_{B,a}^{BE}+s_a\right)\left(0+\frac{\sum_{k=3}^{n}m_k\delta_{ik}\delta_{ac}}{m_{red}}\right)\right]\right]\\
& \quad = -\frac{Gm_Am_B}{\left\|r_B^{BE}+s\right\|^3}\left[\sum_{a=1}^{3}\left[\left(r_{B,a}^{BE}+s_a\right)\frac{m_i\delta_{ac}}{m_{red}}\right]\right]\\
& \quad = -\frac{Gm_Am_B}{\left\|r_B^{BE}+s\right\|^3}\left[\left(r_{B,c}^{BE}+s_c\right)\frac{m_i}{m_{red}}\right]\\
& \quad = -\frac{Gm_Am_Bm_i}{m_{red}}\left[\frac{r_{B,c}^{BE}+s_c}{\left\|r_B^{BE}+s\right\|^3}\right].
\end{aligned}
$$

Notons qu'au cours de ce calcul, nous avons utilisé la définition par composante de s :

$$s_c = \frac{\sum_{i=3}^{n} m_i r_{i,c}^{BE}}{m_{red}} \qquad c = 1, 2, 3. \tag{3.2.8}$$

Nous allons à présent calculer la dernière dérivée de (3.2.7) :

$$\begin{aligned}
&- \frac{\partial}{\partial r_{i,c}^{BE}} \left[G m_B \sum_{k=3}^{n} m_k \left(\frac{1}{\|r_B^{BE}\|} - \frac{1}{\|r_B^{BE} - r_k^{BE} + s\|} \right) \right] \\
&\quad = -G m_B \sum_{k=3}^{n} \left[m_k \left(\frac{\partial}{\partial r_{i,c}^{BE}} \left[\frac{1}{\|r_B^{BE}\|} \right] - \frac{\partial}{\partial r_{i,c}^{BE}} \left[\frac{1}{\|r_B^{BE} - r_k^{BE} + s\|} \right] \right) \right] \\
&\quad = G m_B \sum_{k=3}^{n} \left[m_k \frac{\partial}{\partial r_{i,c}^{BE}} \left(\left[\sum_{a=1}^{3} \left(r_{B,a}^{BE} - r_{k,a}^{BE} + s_a \right)^2 \right]^{-\frac{1}{2}} \right) \right] \\
&\quad = G m_B \sum_{k=3}^{n} \left[-\frac{m_k}{2} \left[\sum_{a=1}^{3} \left(r_{B,a}^{BE} - r_{k,a}^{BE} + s_a \right)^2 \right]^{-\frac{3}{2}} \frac{\partial}{\partial r_{i,c}^{BE}} \left(\left[\sum_{a=1}^{3} \left(r_{B,a}^{BE} - r_{k,a}^{BE} + s_a \right)^2 \right]^{-\frac{1}{2}} \right) \right] \\
&\quad = -G m_B \sum_{k=3}^{n} \left[\frac{m_k}{2 \left\| r_B^{BE} - r_k^{BE} + s \right\|^3} \sum_{a=1}^{3} \left(\frac{\partial}{\partial r_{i,c}^{BE}} \left[\left(r_{B,a}^{BE} - r_{k,a}^{BE} + s_a \right)^2 \right] \right) \right] \\
&\quad = -G m_B \sum_{k=3}^{n} \left[\frac{m_k}{2 \left\| r_B^{BE} - r_k^{BE} + s \right\|^3} \sum_{a=1}^{3} \left[2 \left(r_{B,a}^{BE} - r_{k,a}^{BE} + s_a \right) \frac{\partial}{\partial r_{i,c}^{BE}} \left(r_{B,a}^{BE} - r_{k,a}^{BE} + s_a \right) \right] \right] \\
&\quad = -G m_B \sum_{k=3}^{n} \left[\frac{m_k}{\left\| r_B^{BE} - r_k^{BE} + s \right\|^3} \sum_{a=1}^{3} \left[\left(r_{B,a}^{BE} - r_{k,a}^{BE} + s_a \right) \right. \right. \\
&\qquad \left. \left. \frac{\partial}{\partial r_{i,c}^{BE}} \left(r_{B,a}^{BE} - r_{k,a}^{BE} + \frac{\sum_{l=3}^{n} m_i r_{l,a}^{BE}}{m_{red}} \right) \right] \right] \\
&\quad = -G m_B \sum_{k=3}^{n} \left[\frac{m_k}{\left\| r_B^{BE} - r_k^{BE} + s \right\|^3} \sum_{a=1}^{3} \left[\left(r_{B,a}^{BE} - r_{k,a}^{BE} + s_a \right) \right. \right. \\
&\qquad \left. \left. \left(0 - \delta_{ik}\delta_{ac} + \frac{\sum_{l=3}^{n} m_l \delta_{il} \delta_{ac}}{m_{red}} \right) \right] \right] \\
&\quad = -G m_B \sum_{k=3}^{n} \left[\frac{m_k}{\left\| r_B^{BE} - r_k^{BE} + s \right\|^3} \sum_{a=1}^{3} \left[\left(r_{B,a}^{BE} - r_{k,a}^{BE} + s_a \right) \delta_{ac} \left(-\delta_{ik} + \frac{m_i}{m_{red}} \right) \right] \right] \\
&\quad = -G m_B \sum_{k=3}^{n} \left[\frac{m_k}{\left\| r_B^{BE} - r_k^{BE} + s \right\|^3} \left(r_{B,c}^{BE} - r_{k,c}^{BE} + s_c \right) \left(-\delta_{ik} + \frac{m_i}{m_{red}} \right) \right] \\
&\quad = -G m_B \sum_{k=3}^{n} \left[\frac{m_k m_i \left(r_{B,c}^{BE} - r_{k,c}^{BE} + s_c \right)}{m_{red} \left\| r_B^{BE} - r_k^{BE} + s \right\|^3} - \delta_{ik} \frac{m_k \left(r_{B,c}^{BE} - r_{k,c}^{BE} + s_c \right)}{\left\| r_B^{BE} - r_k^{BE} + s \right\|^3} \right] \\
&\quad = -G m_B \sum_{k=3}^{n} \left[\frac{m_k m_i \left(r_{B,c}^{BE} - r_{k,c}^{BE} + s_c \right)}{m_{red} \left\| r_B^{BE} - r_k^{BE} + s \right\|^3} \right] + G m_B \sum_{k=3}^{n} \left[\delta_{ik} \frac{m_k \left(r_{B,c}^{BE} - r_{k,c}^{BE} + s_c \right)}{\left\| r_B^{BE} - r_k^{BE} + s \right\|^3} \right] \\
&\quad = -G m_B \sum_{k=3}^{n} \left[\frac{m_k m_i \left(r_{B,c}^{BE} - r_{k,c}^{BE} + s_c \right)}{m_{red} \left\| r_B^{BE} - r_k^{BE} + s \right\|^3} \right] + G m_B \frac{m_i \left(r_{B,c}^{BE} - r_{i,c}^{BE} + s_c \right)}{\left\| r_B^{BE} - r_i^{BE} + s \right\|^3}.
\end{aligned}$$

En rassemblant les différentes parties, la dérivée (3.2.7) s'écrit

$$-\frac{\partial \mathcal{H}_{Int}^{BE}}{\partial r_{i,c}^{BE}} = -\sum_{j=3,j\neq i}^{n}\left[\frac{Gm_im_j}{\|r_{ij}\|^3}\left(r_{i,c}^{BE}-r_{j,c}^{BE}\right)\right]-\frac{Gm_Am_Bm_i}{m_{red}}\left[\frac{r_{B,c}^{BE}+s_c}{\|r_B^{BE}+s\|^3}\right]$$
$$-Gm_B\sum_{k=3}^{n}\left[\frac{m_km_i\left(r_{B,c}^{BE}-r_{k,c}^{BE}+s_c\right)}{m_{red}\|r_B^{BE}-r_k^{BE}+s\|^3}\right]+Gm_B\frac{m_i\left(r_{B,c}^{BE}-r_{i,c}^{BE}+s_c\right)}{\|r_B^{BE}-r_i^{BE}+s\|^3}.$$

Puisque nous avons $\frac{\partial \mathcal{H}_{Int}^{BE}}{\partial p_{i,c}^{BE}}=0$, les équations d'Hamilton (3.2.6) pour les planètes deviennent :

$$\begin{cases}\frac{dr_{i,c}^{BE}}{dt} &= 0\\ \frac{dp_{i,c}^{BE}}{dt} &= -\sum\limits_{j=3,j\neq i}^{n}\left[\frac{Gm_im_j}{\|r_{ij}\|^3}\left(r_{i,c}^{BE}-r_{j,c}^{BE}\right)\right]-\frac{Gm_Am_Bm_i}{m_{red}}\left[\frac{r_{B,c}^{BE}+s_c}{\|r_B^{BE}+s\|^3}\right]\\ &\quad -Gm_B\sum\limits_{k=3}^{n}\left[\frac{m_km_i\left(r_{B,c}^{BE}-r_{k,c}^{BE}+s_c\right)}{m_{red}\|r_B^{BE}-r_k^{BE}+s\|^3}\right]+Gm_B\frac{m_i\left(r_{B,c}^{BE}-r_{i,c}^{BE}+s_c\right)}{\|r_B^{BE}-r_i^{BE}+s\|^3} \quad i=3,...,n, \quad c=1,2,3.\end{cases} \tag{3.2.9}$$

Ces équations d'évolution dépendent des positions du compagnon stellaire. Nous avons donc besoin de ses équations d'évolution afin de résoudre ce système.

Intéressons-nous donc à l'évolution du compagnon stellaire sous $\mathcal{H}_{Int}^{BE}$. Comme $\mathcal{H}_{Int}^{BE}$ ne dépend pas du moment associé du compagnon stellaire, nous avons $\frac{\partial \mathcal{H}_{Int}^{BE}}{\partial p_{B,c}^{BE}}=0$. Il nous reste donc à calculer $\frac{\partial \mathcal{H}_{Int}^{BE}}{\partial r_{B,c}^{BE}}$. Comme pour les planètes nous allons calculer cette dérivée en plusieurs morceaux :

$$-\frac{\partial \mathcal{H}_{Int}^{BE}}{\partial r_{B,c}^{BE}} = -\frac{\partial}{\partial r_{B,c}^{BE}}\left[-\sum_{k=3}^{n-1}\sum_{k<j\leqslant n}\frac{Gm_km_j}{\|r_{kj}\|}\right]-\frac{\partial}{\partial r_{B,c}^{BE}}\left[Gm_Am_B\left(\frac{1}{\|r_B^{BE}\|}-\frac{1}{\|r_B^{BE}+s\|}\right)\right]$$
$$-\frac{\partial}{\partial r_{B,c}^{BE}}\left[Gm_B\sum_{k=3}^{n}m_k\left(\frac{1}{\|r_B^{BE}\|}-\frac{1}{\|r_B^{BE}-r_k^{BE}+s\|}\right)\right]. \tag{3.2.10}$$

La première dérivée ne dépend pas du compagnon stellaire. Nous avons donc

$$-\frac{\partial}{\partial r_{B,c}^{BE}}\left[-\sum_{k=3}^{n-1}\sum_{k<j\leqslant n}\frac{Gm_km_j}{\|r_{kj}\|}\right]=0.$$

Pour deuxième dérivée, nous obtenons :

$$\begin{aligned}-\frac{\partial}{\partial r_{B,c}^{BE}}&\left[Gm_Am_B\left(\frac{1}{\|r_B^{BE}\|}-\frac{1}{\|r_B^{BE}+s\|}\right)\right]\\ &= -Gm_Am_B\left(\frac{\partial}{\partial r_{B,c}^{BE}}\left[\frac{1}{\|r_B^{BE}\|}\right]-\frac{\partial}{\partial r_{B,c}^{BE}}\left[\frac{1}{\|r_B^{BE}+s\|}\right]\right)\\ &= -Gm_Am_B\left(\frac{\partial}{\partial r_{B,c}^{BE}}\left[\left(\sum_{a=1}^{3}\left[r_{B,a}^{BE}\right]^2\right)^{-\frac{1}{2}}\right]-\frac{\partial}{\partial r_{B,c}^{BE}}\left[\left(\sum_{a=1}^{3}\left[r_{B,a}^{BE}-s_a\right]^2\right)^{-\frac{1}{2}}\right]\right)\\ &= -Gm_Am_B\left(-\frac{1}{2}\left(\sum_{a=1}^{3}\left[r_{B,a}^{BE}\right]^2\right)^{-\frac{3}{2}}\frac{\partial}{\partial r_{B,c}^{BE}}\left[\sum_{a=1}^{3}\left[r_{B,a}^{BE}\right]^2\right]\right.\\ &\qquad \left.-\left(-\frac{1}{2}\right)\left(\sum_{a=1}^{3}\left[r_{B,a}^{BE}-s_a\right]^2\right)^{-\frac{3}{2}}\frac{\partial}{\partial r_{B,c}^{BE}}\left(\sum_{a=1}^{3}\left[r_{B,a}^{BE}-s_a\right]^2\right)\right)\end{aligned}$$

$$
\begin{aligned}
&= Gm_Am_B\left(\frac{1}{2}\frac{1}{\left\|r_B^{BE}\right\|^3}\sum_{a=1}^{3}\left[\frac{\partial}{\partial r_{B,c}^{BE}}\left(\left[r_{B,a}^{BE}\right]^2\right)\right]\right.\\
&\qquad\left.-\frac{1}{2}\frac{1}{\left\|r_B^{BE}-s\right\|^3}\sum_{a=1}^{3}\left[\frac{\partial}{\partial r_{B,c}^{BE}}\left(\left[r_{B,a}^{BE}-s_a\right]^2\right)\right]\right)\\
&= Gm_Am_B\left(\frac{1}{2}\frac{1}{\left\|r_B^{BE}\right\|^3}\sum_{a=1}^{3}\left(2r_{B,a}^{BE}\frac{\partial}{\partial r_{B,c}^{BE}}\left[r_{B,a}^{BE}\right]\right)\right.\\
&\qquad\left.-\frac{1}{2}\frac{1}{\left\|r_B^{BE}-s\right\|^3}\sum_{a=1}^{3}\left[2\left(r_{B,a}^{BE}-s_a\right)\frac{\partial}{\partial r_{B,c}^{BE}}\left(r_{B,a}^{BE}-s_a\right)\right]\right)\\
&= Gm_Am_B\left(\frac{1}{\left\|r_B^{BE}\right\|^3}\sum_{a=1}^{3}\left(r_{B,a}^{BE}\delta_{ac}\right)\right.\\
&\qquad\left.-\frac{1}{\left\|r_B^{BE}-s\right\|^3}\sum_{a=1}^{3}\left[\left(r_{B,a}^{BE}-s_a\right)\frac{\partial}{\partial r_{B,c}^{BE}}\left(r_{B,a}^{BE}-\frac{\sum_{i=3}^{n}m_ir_{i,a}^{BE}}{m_{red}}\right)\right]\right)\\
&= Gm_Am_B\left(\frac{1}{\left\|r_B^{BE}\right\|^3}\sum_{a=1}^{3}\left(r_{B,a}^{BE}\delta_{ac}\right)-\frac{1}{\left\|r_B^{BE}-s\right\|^3}\sum_{a=1}^{3}\left[\left(r_{B,a}^{BE}-s_a\right)\delta_{ac}\right]\right)\\
&= Gm_Am_B\left(\frac{r_{B,c}^{BE}}{\left\|r_B^{BE}\right\|^3}-\frac{r_{B,c}^{BE}-s_c}{\left\|r_B^{BE}-s\right\|^3}\right).
\end{aligned}
$$

Il nous reste la dernière dérivée de (3.2.10) à calculer :

$$
\begin{aligned}
&-\frac{\partial}{\partial r_{B,c}^{BE}}\left[Gm_B\sum_{k=3}^{n}m_k\left(\frac{1}{\left\|r_B^{BE}\right\|}-\frac{1}{\left\|r_B^{BE}-r_k^{BE}+s\right\|}\right)\right]\\
&\quad= -Gm_B\sum_{k=3}^{n}m_k\left(\frac{\partial}{\partial r_{B,c}^{BE}}\left[\frac{1}{\left\|r_B^{BE}\right\|}\right]-\frac{\partial}{\partial r_{B,c}^{BE}}\left[\frac{1}{\left\|r_B^{BE}-r_k^{BE}+s\right\|}\right]\right)\\
&\quad= -Gm_B\sum_{k=3}^{n}m_k\left(\frac{\partial}{\partial r_{B,c}^{BE}}\left[\left(\sum_{a=1}^{3}\left[r_{B,a}^{BE}\right]^2\right)^{-\frac{1}{2}}\right]-\frac{\partial}{\partial r_{B,c}^{BE}}\left[\left(\sum_{a=1}^{3}\left[r_{B,a}^{BE}-r_{k,a}^{BE}-s_a\right]^2\right)^{-\frac{1}{2}}\right]\right)\\
&\quad= -Gm_B\sum_{k=3}^{n}m_k\left(-\frac{1}{2}\left(\sum_{a=1}^{3}\left[r_{B,a}^{BE}\right]^2\right)^{-\frac{3}{2}}\frac{\partial}{\partial r_{B,c}^{BE}}\left[\sum_{a=1}^{3}\left[r_{B,a}^{BE}\right]^2\right]\right.\\
&\qquad\left.-\left(-\frac{1}{2}\right)\left(\sum_{a=1}^{3}\left[r_{B,a}^{BE}-r_{k,a}^{BE}-s_a\right]^2\right)^{-\frac{3}{2}}\frac{\partial}{\partial r_{B,c}^{BE}}\left(\sum_{a=1}^{3}\left[r_{B,a}^{BE}-r_{k,a}^{BE}-s_a\right]^2\right)\right)\\
&\quad= Gm_B\sum_{k=3}^{n}m_k\left(\frac{1}{2}\frac{1}{\left\|r_B^{BE}\right\|^3}\sum_{a=1}^{3}\frac{\partial}{\partial r_{B,c}^{BE}}\left[\left[r_{B,a}^{BE}\right]^2\right]\right.\\
&\qquad\left.-\left(\frac{1}{2}\right)\frac{1}{\left\|r_B^{BE}-r_k^{BE}-s\right\|^3}\sum_{a=1}^{3}\frac{\partial}{\partial r_{B,c}^{BE}}\left(\left[r_{B,a}^{BE}-r_{k,a}^{BE}-s_a\right]^2\right)\right)\\
&\quad= Gm_B\sum_{k=3}^{n}m_k\left(\frac{1}{2}\frac{1}{\left\|r_B^{BE}\right\|^3}\sum_{a=1}^{3}\left(2r_{B,a}^{BE}\frac{\partial}{\partial r_{B,c}^{BE}}\left[r_{B,a}^{BE}\right]\right)\right.\\
&\qquad\left.-\frac{1}{2}\frac{1}{\left\|r_B^{BE}-r_k^{BE}-s\right\|^3}\sum_{a=1}^{3}\left[2\left(r_{B,a}^{BE}-r_{k,a}^{BE}-s_a\right)\frac{\partial}{\partial r_{B,c}^{BE}}\left(r_{B,a}^{BE}-r_{k,a}^{BE}-s_a\right)\right]\right)
\end{aligned}
$$

$$
\begin{aligned}
&= Gm_B \sum_{k=3}^{n} m_k \left(\frac{1}{\left\|r_B^{BE}\right\|^3} \sum_{a=1}^{3} \left(r_{B,a}^{BE} \delta_{ac} \right) \right. \\
&\qquad \left. - \frac{1}{\left\|r_B^{BE} - r_k^{BE} - s\right\|^3} \sum_{a=1}^{3} \left[\left(r_{B,a}^{BE} - r_{k,a}^{BE} - s_a \right) \frac{\partial}{\partial r_{B,c}^{BE}} \left(r_{B,a}^{BE} - r_{k,a}^{BE} - \frac{\sum_{i=3}^{n} m_i r_{i,a}^{BE}}{m_{red}} \right) \right] \right) \\
&= Gm_B \sum_{k=3}^{n} m_k \left(\frac{1}{\left\|r_B^{BE}\right\|^3} \sum_{a=1}^{3} \left(r_{B,a}^{BE} \delta_{ac} \right) - \frac{1}{\left\|r_B^{BE} - r_k^{BE} - s\right\|^3} \sum_{a=1}^{3} \left[\left(r_{B,a}^{BE} - r_{k,a}^{BE} - s_a \right) \delta_{ac} \right] \right) \\
&= Gm_B \sum_{k=3}^{n} m_k \left(\frac{r_{B,c}^{BE}}{\left\|r_B^{BE}\right\|^3} - \frac{r_{B,c}^{BE} - r_{k,c}^{BE} - s_c}{\left\|r_B^{BE} - r_k^{BE} - s\right\|^3} \right).
\end{aligned}
$$

La dérivée (3.2.10) devient alors :

$$
-\frac{\partial \mathcal{H}_{Int}^{BE}}{\partial r_{B,c}^{BE}} = Gm_A m_B \left(\frac{r_{B,c}^{BE}}{\left\|r_B^{BE}\right\|^3} - \frac{r_{B,c}^{BE} - s_c}{\left\|r_B^{BE} - s\right\|^3} \right) + Gm_B \sum_{k=3}^{n} m_k \left(\frac{r_{B,c}^{BE}}{\left\|r_B^{BE}\right\|^3} - \frac{r_{B,c}^{BE} - r_{k,c}^{BE} - s_c}{\left\|r_B^{BE} - r_k^{BE} - s\right\|^3} \right).
$$

Comme nous avons également que $\frac{\partial \mathcal{H}_{Int}^{BE}}{\partial p_{B,c}^{BE}} = 0$ comme précisé précédemment, les équations d'Hamilton (3.2.6) deviennent, pour le compagnon stellaire,

$$
\begin{cases}
\frac{dr_{B,c}^{BE}}{dt} &= 0 \\
\frac{dp_{B,c}^{BE}}{dt} &= Gm_A m_B \left(\frac{r_{B,c}^{BE}}{\left\|r_B^{BE}\right\|^3} - \frac{r_{B,c}^{BE} - s_c}{\left\|r_B^{BE} - s\right\|^3} \right) + Gm_B \sum_{k=3}^{n} m_k \left(\frac{r_{B,c}^{BE}}{\left\|r_B^{BE}\right\|^3} - \frac{r_{B,c}^{BE} - r_{k,c}^{BE} - s_c}{\left\|r_B^{BE} - r_k^{BE} - s\right\|^3} \right)
\end{cases} \qquad c = 1, 2, 3. \tag{3.2.11}
$$

En rassemblant les systèmes (3.2.9) et (3.2.11), nous avons les équations d'évolution du système sous $\mathcal{H}_{Int}^{BE}$:

$$
\begin{cases}
\frac{dr_{i,c}^{BE}}{dt} &= 0 \\
\frac{dr_{B,c}^{BE}}{dt} &= 0 \\
\frac{dp_{i,c}^{BE}}{dt} &= -\sum_{j=3, j\neq i}^{n} \left[\frac{Gm_i m_j}{\left\|r_{ij}\right\|^3} \left(r_{i,c}^{BE} - r_{j,c}^{BE} \right) \right] - \frac{Gm_A m_B m_i}{m_{red}} \left[\frac{r_{B,c}^{BE} + s_c}{\left\|r_B^{BE} + s\right\|^3} \right] \\
&\quad -Gm_B \sum_{k=3}^{n} \left[\frac{m_k m_i \left(r_{B,c}^{BE} - r_{k,c}^{BE} + s_c \right)}{m_{red} \left\|r_B^{BE} - r_k^{BE} + s\right\|^3} \right] + Gm_B \frac{m_i \left(r_{B,c}^{BE} - r_{i,c}^{BE} + s_c \right)}{\left\|r_B^{BE} - r_i^{BE} + s\right\|^3} \\
\frac{dp_{B,c}^{BE}}{dt} &= Gm_A m_B \left(\frac{r_{B,c}^{BE}}{\left\|r_B^{BE}\right\|^3} - \frac{r_{B,c}^{BE} - s_c}{\left\|r_B^{BE} - s\right\|^3} \right) + Gm_B \sum_{k=3}^{n} m_k \left(\frac{r_{B,c}^{BE}}{\left\|r_B^{BE}\right\|^3} - \frac{r_{B,c}^{BE} - r_{k,c}^{BE} - s_c}{\left\|r_B^{BE} - r_k^{BE} - s\right\|^3} \right)
\end{cases} \qquad \begin{array}{l} c = 1, 2, 3, \\ i = 3...n. \end{array}
$$

Nous déduisons des deux premières égalités que les positions sont constantes sous $\mathcal{H}_{Int}^{BE}$ pour toutes les planètes et le compagnon planétaire. Les deux égalités suivantes se réduisent donc à une dérivée égale à une constante car s ne dépend que des positions des planètes (cfr (3.2.8)) et est donc constant sous $\mathcal{H}_{Int}^{BE}$. Le système se résout donc aisément et nous trouvons les équations

d'évolution :

$$\begin{cases} r_{i,c}^{BE}(t_{k+1}) &= r_{i,c}^{BE}(t_k) \\ r_{B,c}^{BE}(t_{k+1}) &= r_{B,c}^{BE}(t_k) \\ p_{i,c}^{BE}(t_{k+1}) &= p_{i,c}^{BE}(t_k) - \tau \left[\sum\limits_{j=3, j\neq i}^{n} \left[\frac{Gm_i m_j}{\|r_{ij}(t_k)\|^3} \left(r_{i,c}^{BE}(t_k) - r_{j,c}^{BE}(t_k) \right) \right] \right. \\ & \quad + \frac{Gm_A m_B m_i}{m_{red}} \left[\frac{r_{B,c}^{BE}(t_k) + s_c(t_k)}{\left\| r_B^{BE}(t_k) + s(t_k) \right\|^3} \right] & c = 1,2,3, \\ & \quad + Gm_B \sum\limits_{k=3}^{n} \left[\frac{m_k m_i \left(r_{B,c}^{BE}(t_k) - r_{k,c}^{BE}(t_k) + s_c(t_k) \right)}{m_{red} \left\| r_B^{BE}(t_k) - r_k^{BE}(t_k) + s(t_k) \right\|^3} \right] & i = 3...n, \\ & \quad \left. - Gm_B \frac{m_i \left(r_{B,c}^{BE}(t_k) - r_{i,c}^{BE}(t_k) + s_c(t_k) \right)}{\left\| r_B^{BE}(t_k) - r_i^{BE}(t_k) + s(t_k) \right\|^3} \right] & k \in \mathbb{N}. \\ p_{B,c}^{BE}(t_{k+1}) &= p_{B,c}^{BE}(t_k) + \tau \left[Gm_A m_B \left(\frac{r_{B,c}^{BE}(t_k)}{\left\| r_B^{BE}(t_k) \right\|^3} - \frac{r_{B,c}^{BE}(t_k) - s_c(t_k)}{\left\| r_B^{BE}(t_k) - s(t_k) \right\|^3} \right) \right. \\ & \quad \left. + Gm_B \sum\limits_{k=3}^{n} m_k \left(\frac{r_{B,c}^{BE}(t_k)}{\left\| r_B^{BE}(t_k) \right\|^3} - \frac{r_{B,c}^{BE}(t_k) - r_{k,c}^{BE}(t_k) - s_c(t_k)}{\left\| r_B^{BE}(t_k) - r_k^{BE}(t_k) - s(t_k) \right\|^3} \right) \right] \end{cases}$$

Nous pouvons ré-exprimer ce système pour les vitesses. Pour ce faire nous divisons l'égalité d'évolution des p_i^{BE} par m_i et celle d'évolution de p_B^{BE} par μ_{BE}. Nous utilisons ensuite les définitions des vitesses (3.1.5) et de μ_{BE}. Nous trouvons alors :

$$\begin{cases} r_{i,c}^{BE}(t_{k+1}) &= r_{i,c}^{BE}(t_k) \\ r_{B,c}^{BE}(t_{k+1}) &= r_{B,c}^{BE}(t_k) \\ v_{i,c}^{BE}(t_{k+1}) &= v_{i,c}^{BE}(t_k) - \tau \left[\sum\limits_{j=3, j\neq i}^{n} \left[\frac{Gm_j}{\|r_{ij}(t_k)\|^3} \left(r_{i,c}^{BE}(t_k) - r_{j,c}^{BE}(t_k) \right) \right] \right. \\ & \quad + \frac{Gm_A m_B}{m_{red}} \left[\frac{r_{B,c}^{BE}(t_k) + s_c(t_k)}{\left\| r_B^{BE}(t_k) + s(t_k) \right\|^3} \right] & c = 1,2,3, \\ & \quad + Gm_B \sum\limits_{k=3}^{n} \left[\frac{m_k \left(r_{B,c}^{BE}(t_k) - r_{k,c}^{BE}(t_k) + s_c(t_k) \right)}{m_{red} \left\| r_B^{BE}(t_k) - r_k^{BE}(t_k) + s(t_k) \right\|^3} \right] & i = 3...n, \\ & \quad \left. - Gm_B \frac{\left(r_{B,c}^{BE}(t_k) - r_{i,c}^{BE}(t_k) + s_c(t_k) \right)}{\left\| r_B^{BE}(t_k) - r_i^{BE}(t_k) + s(t_k) \right\|^3} \right] & k \in \mathbb{N}. \\ v_{B,c}^{BE}(t_{k+1}) &= v_{B,c}^{BE}(t_k) + \tau \frac{m_{sys}}{m_{red}} \left[Gm_A \left(\frac{r_{B,c}^{BE}(t_k)}{\left\| r_B^{BE}(t_k) \right\|^3} - \frac{r_{B,c}^{BE}(t_k) - s_c(t_k)}{\left\| r_B^{BE}(t_k) - s(t_k) \right\|^3} \right) \right. \\ & \quad \left. + G \sum\limits_{k=3}^{n} m_k \left(\frac{r_{B,c}^{BE}(t_k)}{\left\| r_B^{BE}(t_k) \right\|^3} - \frac{r_{B,c}^{BE}(t_k) - r_{k,c}^{BE}(t_k) - s_c(t_k)}{\left\| r_B^{BE}(t_k) - r_k^{BE}(t_k) - s(t_k) \right\|^3} \right) \right] \end{cases} \tag{3.2.12}$$

Il nous reste donc plus que l'évolution sous $\mathcal{H}_{Jump}^{BE}$ à déterminer.

Evolution sous $\mathcal{H}_{Jump}^{BE}$

Comme $\mathcal{H}_{Jump}^{BE}$ ne dépend ni des positions, ni des moments associés du compagnon stellaire, celui-ci ne sera pas affecté par l'évolution sous cette partie de l'hamiltonien. Nous ne regardons donc que l'évolution pour les planètes. Celle-ci se déduit également au moyen des équations

d'Hamilton (2.2.2). Nous avons alors :

$$\begin{cases} \frac{dr_{i,c}^{BE}}{dt} &= \frac{\partial \mathcal{H}_{Jump}^{BE}}{\partial p_{i,c}^{BE}} \\ \frac{dp_{i,c}^{BE}}{dt} &= -\frac{\partial \mathcal{H}_{Jump}^{BE}}{\partial r_{i,c}^{BE}} \end{cases} \qquad i = 3, ..., n, \quad c = 1, 2, 3. \tag{3.2.13}$$

Comme $\mathcal{H}_{Jump}^{BE}$ ne dépend que des moments associés, nous avons $-\frac{\partial \mathcal{H}_{Jump}^{BE}}{\partial r_{i,c}^{BE}} = 0$ pour toutes les planètes. De plus, la dérivée $\frac{\partial \mathcal{H}_{Jump}^{BE}}{\partial p_{i,c}^{BE}}$ se calcule d'une façon analogue à la dérivée $\frac{\partial \mathcal{H}_{Jump}^{HC}}{\partial p_{i,c}^{HC}}$ développée à la section 2.2.6 car $\mathcal{H}_{Jump}^{HC}$ et $\mathcal{H}_{Jump}^{BE}$ sont similaires. Nous trouvons alors :

$$\frac{\partial \mathcal{H}_{Jump}^{BE}}{\partial p_{i,c}^{BE}} = \frac{1}{m_1} \left[\sum_{j=3}^{n} p_{j,c}^{BE} \right] \qquad i = 3...n.$$

Les équations d'Hamilton (3.2.13) se réécrivent donc :

$$\begin{cases} \frac{dr_{i,c}^{BE}}{dt} &= \frac{1}{m_1} \left[\sum_{j=3}^{n} p_{j,c}^{BE} \right] \\ \frac{dp_{i,c}^{BE}}{dt} &= 0 \end{cases} \qquad i = 3, ..., n, \quad c = 1, 2, 3.$$

La deuxième égalité nous apprend que les moments sont constants lors de l'évolution sous $\mathcal{H}_{Jump}^{BE}$. La première égalité se réduit donc à une dérivée égale à une constante et se résout donc aisément. Nous trouvons alors les équations d'évolution :

$$\begin{cases} r_{i,c}^{BE}(t_{k+1}) &= r_{i,c}^{BE}(t_k) + \tau \frac{1}{m_1} \left[\sum_{j=3}^{n} p_{j,c}^{BE}(t_k) \right] \\ p_{i,c}^{BE}(t_{k+1}) &= p_{i,c}^{BE}(t_k) \end{cases} \qquad i = 3, ..., n, \quad c = 1, 2, 3, \quad k \in \mathbb{N}.$$

Par ailleurs, en divisant la deuxième égalité par m_i et en utilisant la définition des vitesses (3.1.5), ces équations se réécrivent en terme de vitesses et positions :

$$\begin{cases} r_{i,c}^{BE}(t_{k+1}) &= r_{i,c}^{BE}(t_k) + \tau \frac{1}{m_1} \left[\sum_{j=3}^{n} m_j v_{j,c}^{BE}(t_k) \right] \\ v_{i,c}^{BE}(t_{k+1}) &= v_{i,c}^{BE}(t_k) \end{cases} \qquad i = 3, ..., n, \quad c = 1, 2, 3, \quad k \in \mathbb{N}. \tag{3.2.14}$$

Nous sommes à présent capable d'intégrer l'évolution d'un système binaire lorsqu'il n'y a pas de rencontres proches. Nous allons à présent nous intéresser aux rencontres proches.

3.2.2 Avec rencontres proches

Cette section présente la modification de notre algorithme en cas de rencontres proches. En effet, en cas de rencontres proches, la distance entre les corps est petite. Comme $\mathcal{H}_{Int}^{BE}$ possède des termes proportionnels à l'inverse de la distance entre les corps, cette partie de l'hamiltonien peut augmenter drastiquement. Cela entraînera la violation des contraintes (2.2.15). Le problème étant identique à celui rencontré pour l'algorithme Helio, il sera solutionné de la même manière. Nous allons donc adapter la méthode décrite à la section 2.2.7 à notre problème.

L'hamiltonien est donc divisé sous le même schéma que (2.2.7) :

$$\mathcal{H} = \mathcal{H}_{Jump}^{BE} + \mathcal{H}_{IntNR}^{BE} + (\mathcal{H}_{IntR}^{BE} + \mathcal{H}_{Kep}^{BE}),$$

où $\mathcal{H}_{IntR}^{BE}$ contient les termes de $\mathcal{H}_{Int}$ proportionnels à l'inverse de la distance pour les corps se rencontrant et $\mathcal{H}_{InNR}^{BE}$ contient le reste des termes de $\mathcal{H}_{Int}$. Nous avons donc $\mathcal{H}_{IntNR}^{BE} = \mathcal{H}_{Int}^{BE} - \mathcal{H}_{IntR}^{BE}$.

Mathématiquement, nous pouvons utiliser la fonction H définie à l'équation (2.2.32). Nous avons alors :

$$\mathcal{H}_{IntR}^{BE} = -\sum_{i=3}^{n-1}\sum_{i<j\leqslant n}\left(\frac{Gm_im_j}{\|r_{ij}\|}H(\|r_{ij}\|)\right)$$

et

$$\begin{aligned}
\mathcal{H}_{IntNR}^{BE} &= \mathcal{H}_{Int}^{BE} - \mathcal{H}_{IntR}^{BE} \\
&= -\sum_{i=3}^{n-1}\sum_{i<j\leqslant n}\frac{Gm_im_j}{\|r_{ij}\|} + Gm_Am_B\left(\frac{1}{\|r_B^{BE}\|} - \frac{1}{\|r_B^{BE}+s\|}\right) \\
&\quad + Gm_B\sum_{i=3}^{n} m_i\left(\frac{1}{\|r_B^{BE}\|} - \frac{1}{\|r_B^{BE}-r_i^{BE}+s\|}\right) + \sum_{i=3}^{n-1}\sum_{i<j\leqslant n}\left(\frac{Gm_im_j}{\|r_{ij}\|}H(\|r_{ij}\|)\right) \\
&= -\sum_{i=3}^{n-1}\sum_{i<j\leqslant n}\left(\frac{Gm_im_j}{\|r_{ij}\|}(1-H(\|r_{ij}\|))\right) + Gm_Am_B\left(\frac{1}{\|r_B^{BE}\|} - \frac{1}{\|r_B^{BE}+s\|}\right) \\
&\quad + Gm_B\sum_{i=3}^{n} m_i\left(\frac{1}{\|r_B^{BE}\|} - \frac{1}{\|r_B^{BE}-r_i^{BE}+s\|}\right).
\end{aligned}$$

Nous utilisons ensuite l'équation d'évolution (2.2.33) utilisée par SyMBA et appliquée à cette décomposition. Nous avons alors l'équation d'évolution :

$$\begin{aligned}
f(t_{k+1}) = \exp\left(\frac{\tau}{2}T_{Jump}^{BE}\right)\exp\left(\frac{\tau}{2}T_{IntNR}^{BE}\right)\exp\left(\tau\left[T_{Kep}^{BE}+T_{IntR}^{BE}\right]\right) \\
\exp\left(\frac{\tau}{2}T_{IntNR}^{BE}\right)\exp\left(\frac{\tau}{2}T_{Jump}^{BE}\right)f(t_k), \qquad k\in\mathbb{N},
\end{aligned}$$

où T_{Jump}^{BE}, T_{IntNR}^{BE}, T_{Kep}^{BE} et T_{IntR}^{BE} sont les opérateurs linéaires associés à $\mathcal{H}_{Jump}^{BE}$, $\mathcal{H}_{IntNR}^{BE}$, $\mathcal{H}_{Kep}^{BE}$ et $\mathcal{H}_{IntR}^{BE}$ respectivement.

Il nous reste alors à déterminer l'évolution sous les différentes parties de l'hamiltonien. L'évolution sous $\mathcal{H}_{Jump}^{BE}$ a déjà été déterminée dans le cas sans rencontres proches. L'évolution sous cette partie de l'hamiltonien sera donc réalisée, comme précédemment, au moyen des équations (3.2.14).

Evolution sous $\mathcal{H}_{IntNR}^{BE}$

Intéressons-nous à présent à l'évolution sous $\mathcal{H}_{IntNR}^{BE}$. Comme précédemment, nous utilisons les équations d'Hamilton (2.2.2) afin d'obtenir les équations d'évolution :

$$\begin{cases}
\dfrac{dr_{i,c}^{BE}}{dt} &= \dfrac{\partial\mathcal{H}_{IntNR}^{BE}}{\partial p_{i,c}^{BE}} \\
\dfrac{dp_{i,c}^{BE}}{dt} &= -\dfrac{\partial\mathcal{H}_{IntNR}^{BE}}{\partial r_{i,c}^{BE}}
\end{cases} \qquad i = B,3,...,n, \quad c = 1,2,3.$$

Puisque $\mathcal{H}_{IntNR}^{BE}$ est indépendant des moments associés, nous avons $\frac{\partial\mathcal{H}_{IntNR}^{BE}}{\partial p_{i,c}^{BE}} = 0$ pour toutes les planètes et le compagnon stellaire. Nous devons donc calculer $-\frac{\partial\mathcal{H}_{IntNR}^{BE}}{\partial r_{i,c}^{BE}}$. Nous allons réaliser ce

calcul séparément pour les planètes et l'étoile B.

Pour les planètes, nous trouvons

$$\begin{aligned}
-\frac{\partial \mathcal{H}^{BE}_{IntNR}}{\partial r^{BE}_{i,c}} = & -\frac{\partial}{\partial r^{BE}_{i,c}}\left[-\sum_{k=3}^{n-1}\sum_{k<j\leqslant n}\left(\frac{Gm_km_j}{\|r_{kj}\|}(1-H(\|r_{kj}\|))\right)+Gm_Am_B\left(\frac{1}{\|r^{BE}_B\|}-\frac{1}{\|r^{BE}_B+s\|}\right)\right.\\
& \left.+Gm_B\sum_{k=3}^{n}m_k\left(\frac{1}{\|r^{BE}_B\|}-\frac{1}{\|r^{BE}_B-r^{BE}_k+s\|}\right)\right]\\
= & \frac{\partial}{\partial r^{BE}_{i,c}}\left[\sum_{k=3}^{n-1}\sum_{k<j\leqslant n}\left(\frac{Gm_km_j}{\|r_{kj}\|}(1-H(\|r_{kj}\|))\right)\right]+\frac{\partial}{\partial r^{BE}_{k,c}}\left[Gm_Am_B\left(\frac{1}{\|r^{BE}_B\|}\right.\right.\\
& \left.\left.-\frac{1}{\|r^{BE}_B+s\|}\right)\right]+\frac{\partial}{\partial r^{BE}_{i,c}}\left[Gm_B\sum_{k=3}^{n}m_k\left(\frac{1}{\|r^{BE}_B\|}-\frac{1}{\|r^{BE}_B-r^{BE}_k+s\|}\right)\right] \qquad \begin{array}{l} i=3...n,\\ c=1,2,3.\end{array}
\end{aligned}$$

Nous calculons séparément les différentes parties de la dérivée. La première est similaire à la dérivée de $\mathcal{H}^{HC}_{IntNR}$. Le calcul est donc similaire à celui de la section 2.2.7. Nous avons alors :

$$\frac{\partial}{\partial r^{BE}_{i,c}}\left[\sum_{k=3}^{n-1}\sum_{k<j\leqslant n}\left(\frac{Gm_km_j}{\|r_{kj}\|}(1-H(\|r_{kj}\|))\right)\right]=-\sum_{j=3,j\neq i}^{n}\left[\frac{Gm_im_j}{\|r_{ij}\|^3}\left(r^{BE}_{i,c}-r^{BE}_{j,c}\right)H(\|r_{kj}\|)\right].$$

Les deux autres dérivées ont été calculées dans le cas sans rencontres proches. Nous avons donc :

$$\frac{\partial}{\partial r^{BE}_{i,c}}\left[Gm_Am_B\left(\frac{1}{\|r^{BE}_B\|}-\frac{1}{\|r^{BE}_B+s\|}\right)\right]=-\frac{Gm_Am_Bm_i}{m_{red}}\left[\frac{r^{BE}_{B,c}+s_c}{\|r^{BE}_B+s\|^3}\right]$$

et

$$\begin{aligned}
&\frac{\partial}{\partial r^{BE}_{i,c}}\left[+Gm_B\sum_{k=3}^{n}m_k\left(\frac{1}{\|r^{BE}_B\|}-\frac{1}{\|r^{BE}_B-r^{BE}_k+s\|}\right)\right]\\
&\qquad = -Gm_B\sum_{k=3}^{n}\left[\frac{m_km_i\left(r^{BE}_{B,c}-r^{BE}_{k,c}+s_c\right)}{m_{red}\|r^{BE}_B-r^{BE}_k+s\|^3}\right]+Gm_B\frac{m_i\left(r^{BE}_{B,c}-r^{BE}_{i,c}+s_c\right)}{\|r^{BE}_B-r^{BE}_i+s\|^3}.
\end{aligned}$$

La dérivée $-\frac{\partial \mathcal{H}^{BE}_{IntNR}}{\partial r^{BE}_{i,c}}$ devient donc :

$$\begin{aligned}
-\frac{\partial \mathcal{H}^{BE}_{IntNR}}{\partial r^{BE}_{i,c}} = & -\sum_{j=3,j\neq i}^{n}\left[\frac{Gm_im_j}{\|r_{ij}\|^3}\left(r^{BE}_{i,c}-r^{BE}_{j,c}\right)H(\|r_{kj}\|)\right]-\frac{Gm_Am_Bm_i}{m_{red}}\left[\frac{r^{BE}_{B,c}+s_c}{\|r^{BE}_B+s\|^3}\right]\\
& -Gm_B\sum_{k=3}^{n}\left[\frac{m_km_i\left(r^{BE}_{B,c}-r^{BE}_{k,c}+s_c\right)}{m_{red}\|r^{BE}_B-r^{BE}_k+s\|^3}\right]+Gm_B\frac{m_i\left(r^{BE}_{B,c}-r^{BE}_{i,c}+s_c\right)}{\|r^{BE}_B-r^{BE}_i+s\|^3}.
\end{aligned}$$

Les équations d'Hamilton (3.2.2) deviennent donc :

$$\left\{\begin{array}{ll}
\frac{dr^{BE}_{i,c}}{dt} = 0 & \\
\frac{dp^{BE}_{i,c}}{dt} = -\sum\limits_{j=3,j\neq i}^{n}\left[\frac{Gm_im_j}{\|r_{ij}\|^3}\left(r^{BE}_{i,c}-r^{BE}_{j,c}\right)H(\|r_{kj}\|)\right]-\frac{Gm_Am_Bm_i}{m_{red}}\left[\frac{r^{BE}_{B,c}+s_c}{\|r^{BE}_B+s\|^3}\right] & c=1,2,3,\\
\qquad -Gm_B\sum\limits_{k=3}^{n}\left[\frac{m_km_i\left(r^{BE}_{B,c}-r^{BE}_{k,c}+s_c\right)}{m_{red}\|r^{BE}_B-r^{BE}_k+s\|^3}\right]+Gm_B\frac{m_i\left(r^{BE}_{B,c}-r^{BE}_{i,c}+s_c\right)}{\|r^{BE}_B-r^{BE}_i+s\|^3} & i=3...n.
\end{array}\right. \tag{3.2.15}$$

Afin de résoudre ce système nous avons également besoin des équations (3.2.2) pour le compagnon stellaire. Comme mentionné précédemment, $\frac{\partial \mathcal{H}^{BE}_{IntNR}}{\partial p^{BE}_{B,c}}=0$ pour toutes les composantes. Il

nous reste donc à calculer $-\frac{\partial \mathcal{H}^{BE}_{IntNR}}{\partial r^{BE}_{B,c}}$. En utilisant le même développement que pour les planètes, nous trouvons :

$$
\begin{aligned}
&-\frac{\partial \mathcal{H}^{BE}_{IntNR}}{\partial r^{BE}_{B,c}} \\
&\qquad = \frac{\partial}{\partial r^{BE}_{B,c}}\left[\sum_{k=3}^{n-1}\sum_{k<j\leqslant n}\left(\frac{Gm_km_j}{\|r_{kj}\|}(1-H(\|r_{kj}\|))\right)\right] + \frac{\partial}{\partial r^{BE}_{B,c}}\left[Gm_Am_B\left(\frac{1}{\|r^{BE}_B\|}\right.\right. \\
&\qquad\quad \left.\left.-\frac{1}{\|r^{BE}_B+s\|}\right)\right] + \frac{\partial}{\partial r^{BE}_{B,c}}\left[Gm_B\sum_{k=3}^{n}m_k\left(\frac{1}{\|r^{BE}_B\|}-\frac{1}{\|r^{BE}_B-r^{BE}_k+s\|}\right)\right].
\end{aligned}
$$

La première dérivée ne dépendant pas du compagnon stellaire, nous avons :

$$
\frac{\partial}{\partial r^{BE}_{B,c}}\left[\sum_{k=3}^{n-1}\sum_{k<j\leqslant n}\left(\frac{Gm_km_j}{\|r_{kj}\|}(1-H(\|r_{kj}\|))\right)\right] = 0.
$$

Les deux autres dérivées ont également été calculées dans le cas sans rencontres proches. Nous avons donc :

$$
\frac{\partial}{\partial r^{BE}_{B,c}}\left[Gm_Am_B\left(\frac{1}{\|r^{BE}_B\|}-\frac{1}{\|r^{BE}_B+s\|}\right)\right] = Gm_Am_B\left(\frac{r^{BE}_{B,c}}{\|r^{BE}_B\|^3}-\frac{r^{BE}_{B,c}-s_c}{\|r^{BE}_B-s\|^3}\right)
$$

et

$$
\frac{\partial}{\partial r^{BE}_{B,c}}\left[Gm_B\sum_{k=3}^{n}m_k\left(\frac{1}{\|r^{BE}_B\|}-\frac{1}{\|r^{BE}_B-r^{BE}_k+s\|}\right)\right] = Gm_B\sum_{k=3}^{n}m_k\left(\frac{r^{BE}_{B,c}}{\|r^{BE}_B\|^3}-\frac{r^{BE}_{B,c}-r^{BE}_{k,c}-s_c}{\|r^{BE}_B-r^{BE}_k-s\|^3}\right).
$$

En rassemblant ces expressions, nous avons :

$$
-\frac{\partial \mathcal{H}^{BE}_{IntNR}}{\partial r^{BE}_{B,c}} = Gm_Am_B\left(\frac{r^{BE}_{B,c}}{\|r^{BE}_B\|^3}-\frac{r^{BE}_{B,c}-s_c}{\|r^{BE}_B-s\|^3}\right)+Gm_B\sum_{k=3}^{n}m_k\left(\frac{r^{BE}_{B,c}}{\|r^{BE}_B\|^3}-\frac{r^{BE}_{B,c}-r^{BE}_{k,c}-s_c}{\|r^{BE}_B-r^{BE}_k-s\|^3}\right).
$$

Les équations d'Hamilton (3.2.2) deviennent donc, pour le compagnon stellaire,

$$
\left\{\begin{array}{ll}
\frac{dr^{BE}_{B,c}}{dt} &= 0 \\
\frac{dp^{BE}_{B,c}}{dt} &= Gm_Am_B\left(\frac{r^{BE}_{B,c}}{\|r^{BE}_B\|^3}-\frac{r^{BE}_{B,c}-s_c}{\|r^{BE}_B-s\|^3}\right)+Gm_B\sum\limits_{k=3}^{n}m_k\left(\frac{r^{BE}_{B,c}}{\|r^{BE}_B\|^3}-\frac{r^{BE}_{B,c}-r^{BE}_{k,c}-s_c}{\|r^{BE}_B-r^{BE}_k-s\|^3}\right)
\end{array}\right. \quad c=1,2,3. \tag{3.2.16}
$$

En rassemblant les systèmes (3.2.15) et (3.2.16), nous obtenons donc le système d'évolution sous $\mathcal{H}^{BE}_{IntNR}$:

$$
\left\{\begin{array}{ll}
\frac{dr^{BE}_{i,c}}{dt} &= 0 \\
\frac{dr^{BE}_{B,c}}{dt} &= 0 \\
\frac{dp^{BE}_{i,c}}{dt} &= -\sum\limits_{j=3,j\neq i}^{n}\left[\frac{Gm_im_j}{\|r_{ij}\|^3}\left(r^{BE}_{i,c}-r^{BE}_{j,c}\right)H(\|r_{kj}\|)\right]-\frac{Gm_Am_Bm_i}{m_{red}}\left[\frac{r^{BE}_{B,c}+s_c}{\|r^{BE}_B+s\|^3}\right] \qquad c=1,2,3, \\
&\quad -Gm_B\sum\limits_{k=3}^{n}\left[\frac{m_km_i\left(r^{BE}_{B,c}-r^{BE}_{k,c}+s_c\right)}{m_{red}\|r^{BE}_B-r^{BE}_k+s\|^3}\right]+Gm_B\frac{m_i\left(r^{BE}_{B,c}-r^{BE}_{i,c}+s_c\right)}{\|r^{BE}_B-r^{BE}_i+s\|^3} \qquad i=3...n, \\
\frac{dp^{BE}_{B,c}}{dt} &= Gm_Am_B\left(\frac{r^{BE}_{B,c}}{\|r^{BE}_B\|^3}-\frac{r^{BE}_{B,c}-s_c}{\|r^{BE}_B-s\|^3}\right)+Gm_B\sum\limits_{k=3}^{n}m_k\left(\frac{r^{BE}_{B,c}}{\|r^{BE}_B\|^3}-\frac{r^{BE}_{B,c}-r^{BE}_{k,c}-s_c}{\|r^{BE}_B-r^{BE}_k-s\|^3}\right)
\end{array}\right.
$$

Nous déduisons des deux premières égalités que les positions restent constantes sous $\mathcal{H}^{BE}_{IntNR}$. Comme s ne dépend que des positions (3.2.8), les deux égalités suivantes se réduisent à une dérivée égale à une constante et se résolvent aisément. Nous avons donc les équations d'évolution :

$$
\begin{cases}
r_{i,c}^{BE}(t_{k+1}) &= r_{i,c}^{BE}(t_k) \\
r_{B,c}^{BE}(t_{k+1}) &= r_{B,c}^{BE}(t_k) \\
p_{i,c}^{BE}(t_{k+1}) &= p_{i,c}^{BE}(t_k) + \tau \left[- \sum\limits_{j=3, j\neq i}^{n} \left[\frac{G m_i m_j}{\|r_{ij}(t_k)\|^3} \left(r_{i,c}^{BE}(t_k) - r_{j,c}^{BE}(t_k) \right) H(\|r_{kj}(t_k)\|) \right] \right. \\
& \quad - \frac{G m_A m_B m_i}{m_{red}} \left[\frac{r_{B,c}^{BE}(t_k) + s_c(t_k)}{\left\| r_B^{BE}(t_k) + s(t_k) \right\|^3} \right] - G m_B \sum\limits_{k=3}^{n} \left[\frac{m_k m_i \left(r_{B,c}^{BE}(t_k) - r_{k,c}^{BE}(t_k) + s_c(t_k) \right)}{m_{red} \left\| r_B^{BE}(t_k) - r_k^{BE}(t_k) + s(t_k) \right\|^3} \right] \\
& \quad \left. + G m_B \frac{m_i \left(r_{B,c}^{BE}(t_k) - r_{i,c}^{BE}(t_k) + s_c(t_k) \right)}{\left\| r_B^{BE}(t_k) - r_i^{BE}(t_k) + s(t_k) \right\|^3} \right] \\
p_{B,c}^{BE}(t_{k+1}) &= p_{B,c}^{BE}(t_k) + \tau \left[G m_A m_B \left(\frac{r_{B,c}^{BE}(t_k)}{\left\| r_B^{BE}(t_k) \right\|^3} - \frac{r_{B,c}^{BE}(t_k) - s_c(t_k)}{\left\| r_B^{BE}(t_k) - s(t_k) \right\|^3} \right) \right. \\
& \quad \left. + G m_B \sum\limits_{k=3}^{n} m_k \left(\frac{r_{B,c}^{BE}(t_k)}{\left\| r_B^{BE}(t_k) \right\|^3} - \frac{r_{B,c}^{BE}(t_k) - r_{k,c}^{BE}(t_k) - s_c(t_k)}{\left\| r_B^{BE}(t_k) - r_k^{BE}(t_k) - s(t_k) \right\|^3} \right) \right]
\end{cases}
\quad \begin{array}{l} c = 1, 2, 3, \\ i = 3 \ldots n, \\ k \in \mathbb{N}. \end{array}
$$

En divisant la troisième égalité par m_i, la quatrième par μ_{BE} et en utilisant la définition des vitesses (3.1.5) et de μ_{BE} (3.1.6), nous pouvons ré-exprimer le système précédent en fonction des positions et des vitesses :

$$
\begin{cases}
r_{i,c}^{BE}(t_{k+1}) &= r_{i,c}^{BE}(t_k) \\
r_{B,c}^{BE}(t_{k+1}) &= r_{B,c}^{BE}(t_k) \\
v_{i,c}^{BE}(t_{k+1}) &= v_{i,c}^{BE}(t_k) + \tau \left[- \sum\limits_{j=3, j\neq i}^{n} \left[\frac{G m_j}{\|r_{ij}(t_k)\|^3} \left(r_{i,c}^{BE}(t_k) - r_{j,c}^{BE}(t_k) \right) H(\|r_{kj}(t_k)\|) \right] \right. \\
& \quad - \frac{G m_A m_B}{m_{red}} \left[\frac{r_{B,c}^{BE}(t_k) + s_c(t_k)}{\left\| r_B^{BE}(t_k) + s(t_k) \right\|^3} \right] - G m_B \sum\limits_{k=3}^{n} \left[\frac{m_k \left(r_{B,c}^{BE}(t_k) - r_{k,c}^{BE}(t_k) + s_c(t_k) \right)}{m_{red} \left\| r_B^{BE}(t_k) - r_k^{BE}(t_k) + s(t_k) \right\|^3} \right] \\
& \quad \left. + G m_B \frac{\left(r_{B,c}^{BE}(t_k) - r_{i,c}^{BE}(t_k) + s_c(t_k) \right)}{\left\| r_B^{BE}(t_k) - r_i^{BE}(t_k) + s(t_k) \right\|^3} \right] \\
v_{B,c}^{BE}(t_{k+1}) &= v_{B,c}^{BE}(t_k) + \tau \frac{m_{sys}}{m_{red}} \left[G m_A \left(\frac{r_{B,c}^{BE}(t_k)}{\left\| r_B^{BE}(t_k) \right\|^3} - \frac{r_{B,c}^{BE}(t_k) - s_c(t_k)}{\left\| r_B^{BE}(t_k) - s(t_k) \right\|^3} \right) \right. \\
& \quad \left. + G \sum\limits_{k=3}^{n} m_k \left(\frac{r_{B,c}^{BE}(t_k)}{\left\| r_B^{BE}(t_k) \right\|^3} - \frac{r_{B,c}^{BE}(t_k) - r_{k,c}^{BE}(t_k) - s_c(t_k)}{\left\| r_B^{BE}(t_k) - r_k^{BE}(t_k) - s(t_k) \right\|^3} \right) \right]
\end{cases}
\quad \begin{array}{l} c = 1, 2, 3, \\ i = 3 \ldots n, \\ k \in \mathbb{N}. \end{array}
\tag{3.2.17}
$$

Nous pouvons à présent étudier l'évolution du système sous $\mathcal{H}_{Kep}^{BE} + \mathcal{H}_{IntR}^{BE}$.

Evolution sous $\mathcal{H}_{Kep}^{BE} + \mathcal{H}_{IntR}^{BE}$

Cette évolution sera analogue à celle utilisée dans SyMBA et présentée à la section 2.2.7. L'algorithme va donc progressivement réduire le pas en fonction de l'ordre de grandeur des termes. Nous réécrivons donc $\mathcal{H}_{Kep}^{BE} + \mathcal{H}_{IntR}^{BE}$ comme

$$\mathcal{H}_{Kep}^{BE} + \sum_{i=0}^{\infty} \mathcal{H}_{IntR,i}^{BE},$$

de sorte que $\sum\limits_{i=0}^{\infty} \mathcal{H}_{IntR,i}^{BE} = \mathcal{H}_{IntR}^{BE}$ et $\mathcal{H}_{IntR,i}^{BE}$ est d'un ordre de grandeur inférieur à celui de $\mathcal{H}_{IntR,i+1}^{BE}$. Ainsi, nous utiliserons le même schéma d'intégration que dans SyMBA :

$$f(t_{k+1}) = \exp\left(\tau \left[T_{Kep}^{BE} + \sum_{i=0}^{\infty} T_{IntR,i}^{BE} \right] \right) f(t_k)$$

$$
\begin{aligned}
&\approx \exp\left(\frac{\tau}{2}T^{BE}_{IntR,0}\right)\exp\left(\tau\left[T^{BE}_{Kep}+\sum_{i=1}^{\infty}T^{BE}_{IntR,i}\right]\right)\exp\left(\frac{\tau}{2}T^{BE}_{IntR,0}\right)f(t_k)\\
&\approx \exp\left(\frac{\tau}{2}T^{BE}_{IntR,0}\right)\left\{\exp\left(\frac{\tau}{2n}T^{BE}_{IntR,1}\right)\exp\left(\frac{\tau}{n}\left[T^{BE}_{Kep}+\sum_{i=2}^{\infty}T^{BE}_{IntR,i}\right]\right)\exp\left(\frac{\tau}{2n}T^{BE}_{IntR,1}\right)\right\}^n\\
&\qquad \exp\left(\frac{\tau}{2}T^{BE}_{IntR,0}\right)f(t_k)\\
&\approx \exp\left(\frac{\tau}{2}T^{BE}_{IntR,0}\right)\left\{\exp\left(\frac{\tau}{2n}T^{BE}_{IntR,1}\right)\left\{\exp\left(\frac{\tau}{2n^2}T^{BE}_{IntR,2}\right)\exp\left(\frac{\tau}{n^2}\left[T^{BE}_{Kep}+\sum_{i=3}^{\infty}T^{BE}_{IntR,i}\right]\right)\right.\right.\\
&\qquad \left.\left.\exp\left(\frac{\tau}{2n^2}T^{BE}_{IntR,2}\right)\right\}^n\exp\left(\frac{\tau}{2n}T^{BE}_{IntR,1}\right)\right\}^n\exp\left(\frac{\tau}{2}T^{BE}_{IntR,0}\right)f(t_k)\\
&\vdots
\end{aligned}
$$

Comme expliqué dans la section 2.2.7, cette réduction du pas permettra de garder l'intégration satisfaisante en cas de rencontre proche et est en théorie infinie mais sera finie en pratique.

Comme pour SyMBA, la décomposition sera effectuée sur $-\frac{\partial \mathcal{H}^{BE}_{IntR}}{\partial r_{j,c}}$ plutôt que sur $\mathcal{H}^{BE}_{IntR}$. De plus, nous utiliserons la même fonction de partition f_l (2.2.34). Nous aurons donc :

$$-\frac{\partial \mathcal{H}^{BE}_{IntR}}{\partial r_{j,c}}=\sum_{l=0}^{\infty}\left[-\frac{\partial \mathcal{H}^{BE}_{IntR,l}}{\partial r^{BE}_{i,c}}\right],$$

avec

$$-\frac{\partial \mathcal{H}^{BE}_{IntR,l}}{\partial r^{BE}_{i,c}}=-\sum_{j=3,j\neq i}^{n}\left[\frac{Gm_im_j}{\|r_{ij}\|^3}\left(r^{BE}_{i,c}-r^{BE}_{j,c}\right)f_l(\|r_{ij}\|)\left(1-H(\|r_{ij}\|)\right)\right].$$

Nous devons à présent déterminer les équations d'évolution sous chaque partie $\mathcal{H}^{BE}_{IntR,l}$. Celles-ci sont déterminées au moyen des équations d'Hamilton (2.2.2) :

$$
\begin{cases}
\dfrac{dr^{BE}_{i,c}}{dt} &= \dfrac{\partial \mathcal{H}^{BE}_{IntR,l}}{\partial p^{BE}_{i,c}}\\[2ex]
\dfrac{dp^{BE}_{i,c}}{dt} &= -\dfrac{\partial \mathcal{H}^{BE}_{IntR,l}}{\partial r^{BE}_{i,c}}
\end{cases}
\qquad i=3,...,n, \quad c=1,2,3. \tag{3.2.18}
$$

Notons que comme $\mathcal{H}^{BE}_{IntR}$ est indépendant des positions et moments associés du compagnon stellaire, celui-ci sera invariant sous cette évolution et ne sera donc pas abordé. Par ailleurs, comme $\mathcal{H}^{BE}_{IntR}$ est indépendant des positions, $\mathcal{H}^{BE}_{IntR,l}$ l'est aussi pour tous les l. Le système (3.2.18) se réécrit donc :

$$
\begin{cases}
\frac{dr^{BE}_{i,c}}{dt} &= 0 & c=1,2,3,\\
\frac{dp^{BE}_{i,c}}{dt} &= -\sum\limits_{j=3,j\neq i}^{n}\left[\frac{Gm_im_j}{\|r_{ij}\|^3}\left(r^{BE}_{i,c}-r^{BE}_{j,c}\right)f_l(\|r_{ij}\|)\left(1-H(\|r_{ij}\|)\right)\right] & i=3...n,\\
& & l\in\mathbb{N}.
\end{cases}
$$

Nous déduisons de la première égalité que les positions sont constantes sous cette partie de l'hamiltonien. La deuxième égalité se réduit donc à une dérivée égale à une constante et se résout aisément :

$$
\begin{cases}
r^{BE}_{i,c}(t_{k+1}) &= r^{BE}_{i,c}(t_k) & \\
p^{BE}_{i,c}(t_{k+1}) &= p^{BE}_{i,c}(t_k)-\tau\sum\limits_{j=3,j\neq i}^{n}\left[\frac{Gm_im_j}{\|r_{ij}(t_k)\|^3}\left(r^{BE}_{i,c}(t_k)-r^{BE}_{j,c}(t_k)\right)\right. & c=1,2,3,\\
& \left. f_l(\|r_{ij}(t_k)\|)\left(1-H(\|r_{ij}(t_k)\|)\right)\right] & i=3...n,\\
& & k,l\in\mathbb{N}.
\end{cases}
$$

En divisant la deuxième égalité par m_i et en utilisant la définition de la vitesse (3.1.5), nous pouvons ré-exprimer le système en terme de vitesses et de positions :

$$\begin{cases} r_{i,c}^{BE}(t_{k+1}) &= r_{i,c}^{BE}(t_k) \\ v_{i,c}^{BE}(t_{k+1}) &= v_{i,c}^{BE}(t_k) - \tau \sum\limits_{j=3,j\neq i}^{n} \left[\frac{Gm_j}{\|r_{ij}(t_k)\|^3} \left(r_{i,c}^{BE}(t_k) - r_{j,c}^{BE}(t_k) \right) \right. \\ & \qquad \left. f_l(\|r_{ij}(t_k)\|) \left(1 - H(\|r_{ij}(t_k)\|)\right) \right] \end{cases} \begin{array}{l} c = 1,2,3, \\ i = 3...n, \\ k, l \in \mathbb{N}. \end{array}$$

Enfin nous pouvons ré-exprimer ces équations d'évolution en terme de $\bar{f}_l$ en utilisant la définition (2.2.34). Nous avons alors, sous $\mathcal{H}_{IntR,0}^{BE}$, les équations d'évolution :

$$\begin{cases} r_{i,c}^{BE}(t_{k+1}) &= r_{i,c}^{BE}(t_k) \\ v_{i,c}^{BE}(t_{k+1}) &= v_{i,c}^{BE}(t_k) - \tau \sum\limits_{j=3,j\neq i}^{n} \left[\frac{Gm_j}{\|r_{ij}(t_k)\|^3} \left(r_{i,c}^{BE}(t_k) - r_{j,c}^{BE}(t_k) \right) \right. \\ & \qquad \left. \bar{f}_0(\|r_{ij}(t_k)\|) \left(1 - H(\|r_{ij}(t_k)\|)\right) \right] \end{cases} \begin{array}{l} c = 1,2,3, \\ i = 3...n, \\ k \in \mathbb{N}. \end{array}$$

et sous $\mathcal{H}_{IntR,l}^{BE}$ avec $l \in \mathbb{N}_0$:

$$\begin{cases} r_{i,c}^{BE}(t_{k+1}) &= r_{i,c}^{BE}(t_k) \\ v_{i,c}^{BE}(t_{k+1}) &= v_{i,c}^{BE}(t_k) - \tau \sum\limits_{j=3,j\neq i}^{n} \left[\frac{Gm_j}{\|r_{ij}(t_k)\|^3} \left(r_{i,c}^{BE}(t_k) - r_{j,c}^{BE}(t_k) \right) \right. \\ & \qquad \left. \left[\bar{f}_l(\|r_{ij}(t_k)\|) - \bar{f}_{l-1}(\|r_{ij}(t_k)\|) \right] \left(1 - H(\|r_{ij}(t_k)\|)\right) \right] \end{cases} \begin{array}{l} c = 1,2,3, \\ i = 3...n, \\ k \in \mathbb{N}, \\ l \in \mathbb{N}_0. \end{array}$$

qui se réécrit :

$$\begin{cases} r_{i,c}^{BE}(t_{k+1}) &= r_{i,c}^{BE}(t_k) \\ v_{i,c}^{BE}(t_{k+1}) &= v_{i,c}^{BE}(t_k) - \tau \sum\limits_{j=3,j\neq i}^{n} \left[\frac{Gm_j}{\|r_{ij}(t_k)\|^3} \left(r_{i,c}^{BE}(t_k) - r_{j,c}^{BE}(t_k) \right) \right. \\ & \qquad \left. \bar{f}_l(\|r_{ij}(t_k)\|) \left(1 - H(\|r_{ij}(t_k)\|)\right) \right] + \tau \sum\limits_{j=3,j\neq i}^{n} \left[\frac{Gm_j}{\|r_{ij}(t_k)\|^3} \left(r_{i,c}^{BE}(t_k) - \right. \right. \\ & \qquad \left. \left. r_{j,c}^{BE}(t_k) \right) \bar{f}_{l-1}(\|r_{ij}(t_k)\|) \left(1 - H(\|r_{ij}(t_k)\|)\right) \right] \end{cases} \begin{array}{l} c = 1,2,3, \\ i = 3...n, \\ k \in \mathbb{N}, \\ l \in \mathbb{N}_0. \end{array}$$

Il ne nous reste que l'évolution sous $\mathcal{H}_{Kep}^{BE}$ à discuter. Comme explicité pour l'intégrateur SyMBA, nous effectuerons l'intégration sous $\mathcal{H}_{Kep}^{BE}$ pour un corps dès que celui-ci n'est plus présent dans la somme $\sum\limits_{i=k}^{\infty} \mathcal{H}_{IntR,i}^{BE}$. La raison est la même que pour SyMBA. De plus, notons que l'évolution sous $\mathcal{H}_{Kep}^{BE}$ sera effectuée au moyen des équations d'évolution trouvées dans le cas sans rencontres car $\mathcal{H}_{Kep}^{BE}$ n'a pas été modifié. Nous utiliserons donc le système (3.2.4) pour les planètes et le système (3.2.5) pour le compagnon stellaire.

3.3 Code

3.3.1 Fonctions

Sur base des développements précédents, nous avons implémenté un ensemble de fonctions qui forment l'intégrateur pour les systèmes binaires. Nous allons lister ici l'ensemble de ces fonctions ainsi que leur utilité. Notons qu'en dehors des fonctions de coordonnées, l'ensemble des autres fonctions est une adaptation des fonctions provenant de SyMBA et Helio. Afin de retrouver la fonction d'origine, il suffit de retirer le suffixe "_bin" des fonctions. Les fonctions sont listées par ordre alphabétique.

- *coord_posbin2h.f* : permet de passer des positions dans le repère binaire éloignée aux positions héliocentriques.
- *coord_posh2bin.f* : permet de passer des positions héliocentriques aux positions dans le repère binaire éloignée.
- *coord_vbin2h.f* : permet de passer des vitesses dans le repère binaire éloignée aux vitesses héliocentriques.
- *coord_vh2bin.f* : permet de passer des vitesses héliocentriques aux vitesses dans le repère binaire éloignée.
- *helio_drift_bin.f* : calcule l'évolution sous $\mathcal{H}^{BE}_{Kep}$ quand il n'y a pas de rencontres proches.
- *helio_lindrift_bin.f* : calcule l'évolution sous $\mathcal{H}^{BE}_{Jump}$.
- *symba7_getacch_bin.f* : calcule $-\frac{1}{m_i}\frac{\partial \mathcal{H}^{BE}_{IntNR}}{\partial r^{BE}_{i,c}}$ pour les planètes et $-\frac{1}{\mu_{BE}}\frac{\partial \mathcal{H}^{BE}_{IntNR}}{\partial r^{BE}_{B,c}}$ pour le compagnon stellaire. Ces quantités correspondent à l'accélération induite par $\mathcal{H}^{BE}_{IntNR}$ et permettra d'effectuer le système (3.2.17) au moyen de la fonction kickvh.f présentée à la section 2.3.7.
- *symba7_helio_drift_bin.f* : calcule l'évolution sous $\mathcal{H}_{Kep}$ en cas de rencontres proches.
- *symba7_helio_getacch_bin.f* : calcule $-\frac{1}{m_i}\frac{\partial \mathcal{H}^{BE}_{Int}}{\partial r^{BE}_{i,c}}$ pour les planètes et $-\frac{1}{\mu_{BE}}\frac{\partial \mathcal{H}^{BE}_{Int}}{\partial r^{BE}_{B,c}}$ pour le compagnon stellaire. Ces quantités correspondent à l'accélération induite par $\mathcal{H}^{BE}_{Int}$ et permettra d'effectuer le système (3.2.12) au moyen de la fonction kickvh.f présentée à la section 2.3.7.
- *symba7_step_helio_bin.f* : effectue l'intégration lorsqu'il n'y a pas de rencontres proches.
- *symba7_step_interp_bin.f* : effectue l'intégration en cas de rencontres proches.
- *symba7_step_pl_bin.f* : sélectionne l'intégrateur à utiliser.
- *symba7_step_recur_bin.f* effectue l'intégration sous $\mathcal{H}^{BE}_{IntR} + \mathcal{H}^{BE}_{Kep}$.

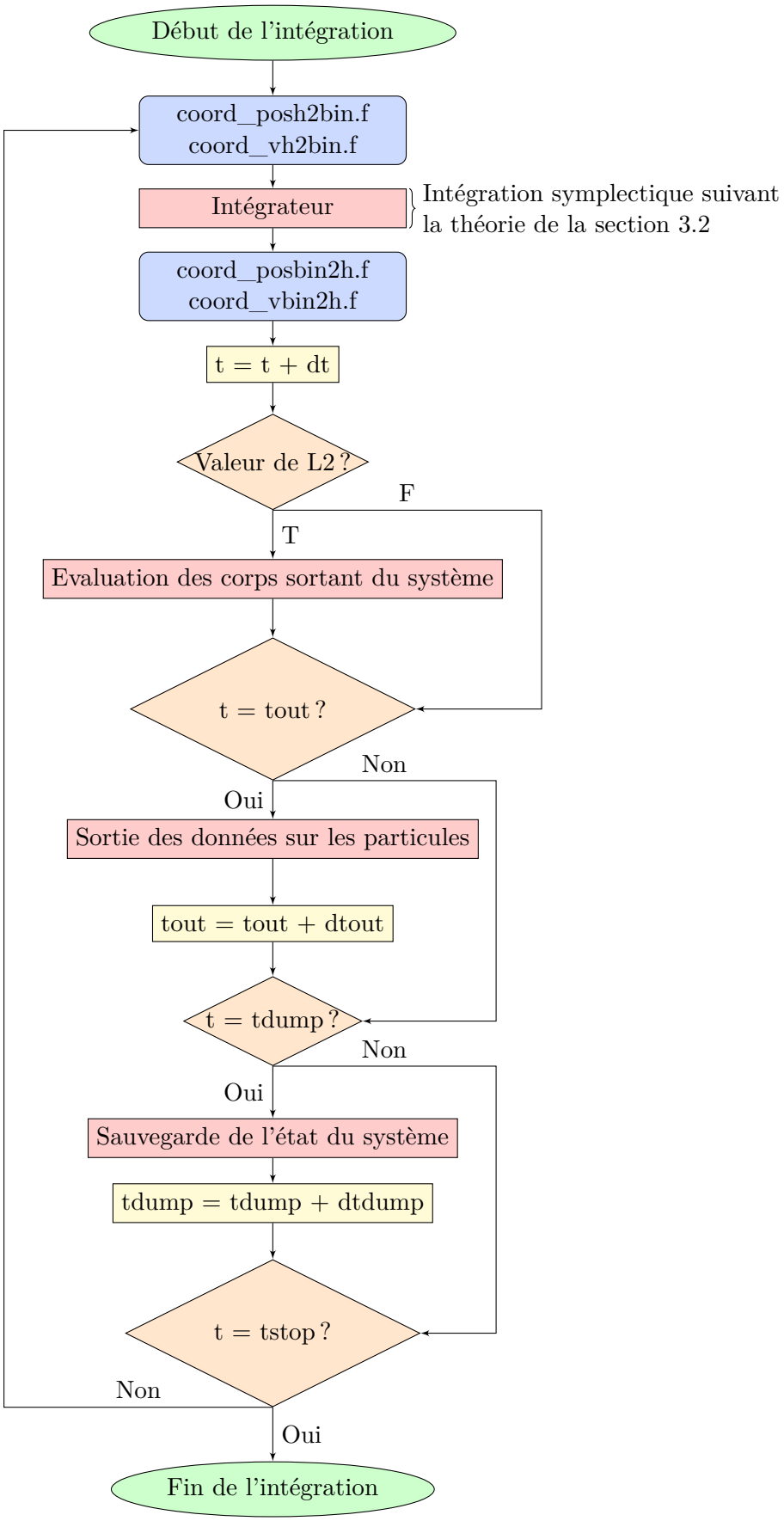
A titre d'exemple, les codes relatifs à la fonction *symba7_helio_getacch_bin.f* sont disponibles à l'Annexe A.2. Notons que la version originale est mise en parallèle.

3.3.2 Organigramme

Nous allons à présent discuter de l'organigramme du code pour les systèmes binaires. Celui-ci est similaire à celui présenté à la section 2.4. Seules deux parties de cet organigramme ont été modifiées. Il s'agit des diagrammes "Intégration" et "Intégrateur" de la sous-section 2.4.2.

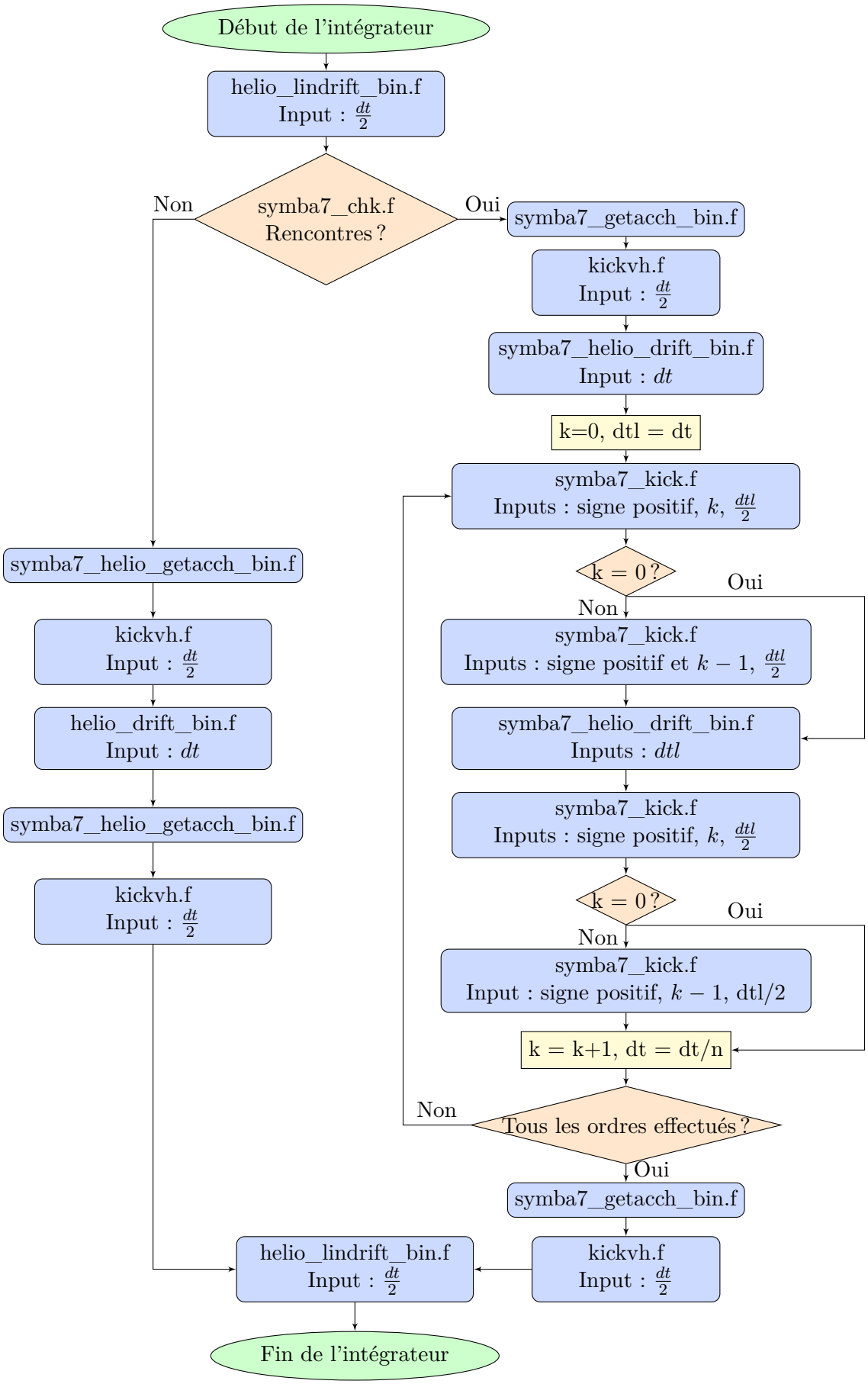
Intégration

Le nouveau diagramme "Intégration" est le suivant :



Intégrateur

Le diagramme "Intégrateur" est maintenant le suivant :



3.4 Validation

Nous allons à présent aborder les différents tests que nous avons réalisés afin de valider notre code pour les systèmes binaires. Dans un premier temps, nous avons testé l'ordre de notre schéma d'intégration sur le système de la table 3.1. Ensuite, nous avons sélectionné différents systèmes provenant de la littérature et avons étudié la conservation de l'hamiltonien et du moment angulaire total pour ces systèmes. Nous avons également comparé nos résultats avec ceux d'un autre code n-body développé dans le cadre du cours de *Chaos et déterminisme* [8]. Enfin, quand cela était possible, nous avons comparé nos résultats avec ceux de la littérature.

3.4.1 Ordre de l'intégrateur

Commençons pas vérifier l'ordre de notre intégrateur sur le système de la table 3.1. Notre schéma étant d'ordre global $\mathcal{O}(\epsilon\tau^2)$ avec $\epsilon = \epsilon_1 + \epsilon_2$ comme précisé à la section 2.2.3, si nous multiplions le pas d'intégration par α, l'ordre de l'erreur sera multipliée par α^2. En effet, nous avons alors $\tilde{\tau} = \alpha\tau$ et l'ordre de l'erreur devient $\mathcal{O}(\epsilon\tilde{\tau}^2) = \mathcal{O}(\epsilon(\alpha\tau)^2) = \mathcal{O}(\epsilon\alpha^2\tau^2)$. Cette propriété peut être observée numériquement. Pour ce faire, nous avons intégré le système de la table 3.1 pour différents pas d'intégration. Le résultat de ces intégrations est visible à la figure 3.4. Les trois pas de temps utilisés sont 0.4 ans, $\sqrt{10} \times 0.04$ ans et 0.04 ans. Nous avons donc utilisé $\alpha = \sqrt{10}$ et nous constatons bien qu'à chaque nouveau pas l'ordre de l'erreur est modifié d'un facteur $\alpha^2 = 10$.

Notons également que le pas de $\sqrt{10} \times 0.04$ ans est proche du pas utilisé dans le livre [10] dont est tiré le système de la table 3.1. Nous obtenons bien des résultats similaires pour la conservation de l'hamiltonien à ceux de ce livre, comme illustré à la figure 3.5.

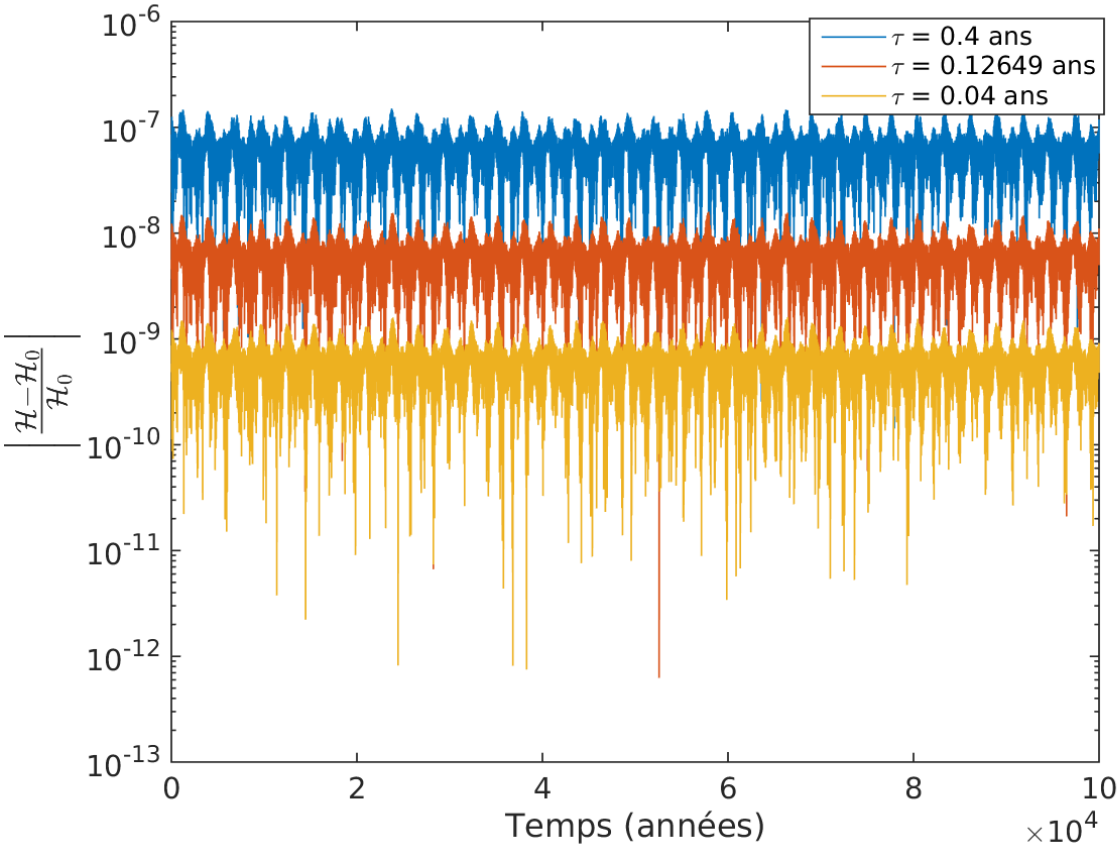


FIGURE 3.4 – Conservation de l'hamiltonien en fonction du pas

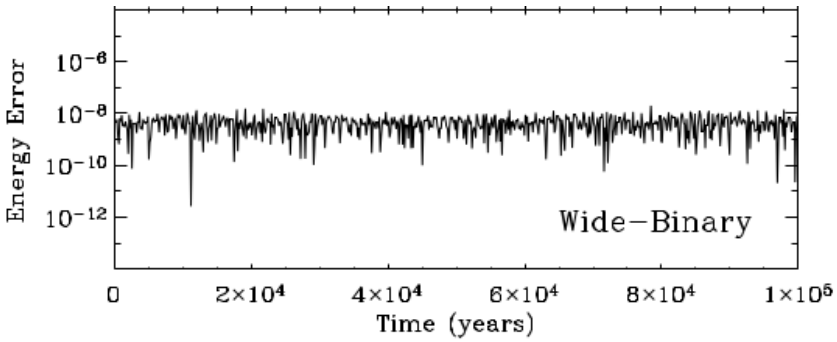


FIGURE 3.5 – Conservation de l'hamiltonien pour le système de la table 3.1. Figure issue de [10]

3.4.2 Comparaison avec la littérature

Nous avons étudié différents systèmes provenant de la littérature et avons, pour chacun, étudié la conservation de l'hamiltonien et du moment angulaire total. Nous avons effectué une comparaison entre 1) nos résultats, 2) ceux obtenus avec un autre code réalisé par nos soins pour le cours de chaos et déterminisme [42] et 3) ceux de la source d'où proviennent les systèmes lorsque cela était possible. Notre autre code consiste en un intégrateur pour le problème à n corps basé sur la famille d'intégrateurs de Yoshida présentée à la section 2.2.2. Une version de ce code a été développée et validée dans le cadre du cours [8]. Ce dernier n'étant contraint par aucune condition sur la taille de la perturbation, il est applicable aux systèmes binaires. Notons qu'il ne traite pas les collisions entre les corps et est moins efficace en temps de calcul. Plus d'informations concernant ce code et sa validation sont disponibles dans le travail [42].

Trois systèmes vont être discutés dans la suite : un système solaire restreint avec compagnon stellaire, un système de résonance de Kozai et le système binaire γ-Cephei.

Système solaire restreint avec compagnon stellaire

Commençons par nous intéresser au système de la table 3.1. Comme précisé précédemment, celui-ci est constitué du système solaire limité aux quatre planètes géantes auquel nous avons ajouté un compagnon stellaire. De plus, notons que nous avons utilisé un pas d'intégration de 10^{-2} ans.

Tout d'abord, Les conservations de l'hamiltonien et du moment angulaire total sont données à la figure 3.6. Notons que nous ne retrouvons pas le caractère constant de l'hamiltonien illustré à la figure 2.4, caractéristique des intégrateurs symplectiques. Cette différence provient du pas d'intégration. En effet, l'erreur due à l'intégrateur avec le pas choisi étant extrêmement faible, nous observons uniquement les erreurs dues à la précision numérique finie. En effet, l'utilisation de la double précision dans le programme entraîne une erreur machine à chaque pas d'intégration de l'ordre de 10^{-16}. De plus, nous utilisons un pas de 10^{-2} ans et nous intégrons jusqu'à un temps de 10^5 ans. Nous effectuons ainsi $\frac{10^5}{10^{-2}} = 10^7$ pas d'intégration. Nous avons donc une erreur due à la précision machine de l'ordre de $10^{-16} \times 10^7 = 10^{-9}$. Cette valeur est cohérente avec la figure 3.6 et l'erreur semble donc bien due à la précision machine. Par ailleurs lors de l'augmentation du pas, nous retrouvons bien l'allure caractéristique de l'erreur des intégrateurs symplectiques, comme nous l'avons observé à la figure 3.4.

Notons également que l'erreur relative sur le moment angulaire total est très faible. En effet, nous obtenons une erreur de l'ordre de 10^{-8} pour la composante z du moment angulaire total et de 10^{-9} pour les autres composantes, comme nous pouvons l'observer à la figure 3.6.

Nous avons également comparé les évolutions des éléments orbitaux des différents corps du système calculées avec le code présenté dans ce mémoire à celles calculées avec le code venant du

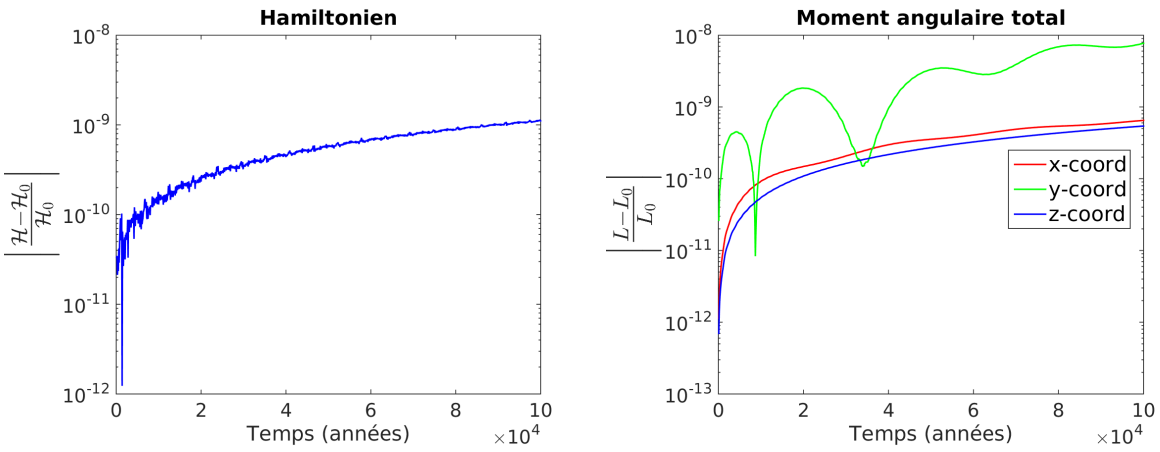


FIGURE 3.6 – Conservations des intégrales premières du mouvement pour le système de la table 3.1

travail de chaos [42]. Cette comparaison est effectuée aux figures 3.7, 3.8, 3.9, 3.10, 3.11 et 3.12, qui reprennent respectivement les demi-grands axes, excentricités, inclinaisons, longitudes des nœuds ascendants, arguments des périastres et anomalies moyennes pour les quatre planètes et le compagnon stellaire. Aucune différence n'est visible entre les évolutions des deux intégrateurs. Or, le système est extrêmement sensible, ce qui nous conforte dans la bonne précision de notre intégrateur.

Notons que le livre [10] dont provient le système ne parle pas des trajectoires des corps, nous ne pouvons donc pas utiliser ce livre comme point de comparaison supplémentaire.

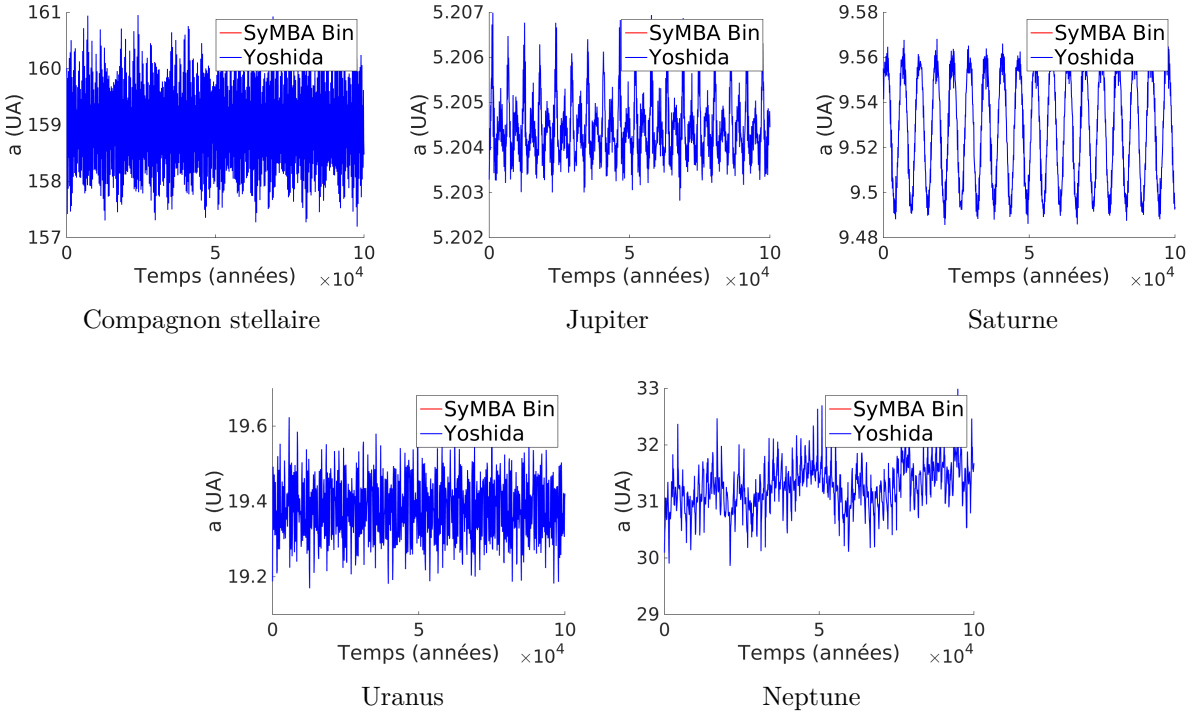


FIGURE 3.7 – Evolutions des Demi-grands axes du système de la table 3.1 obtenues avec les intégrateurs SyMBA binaire et Yoshida.

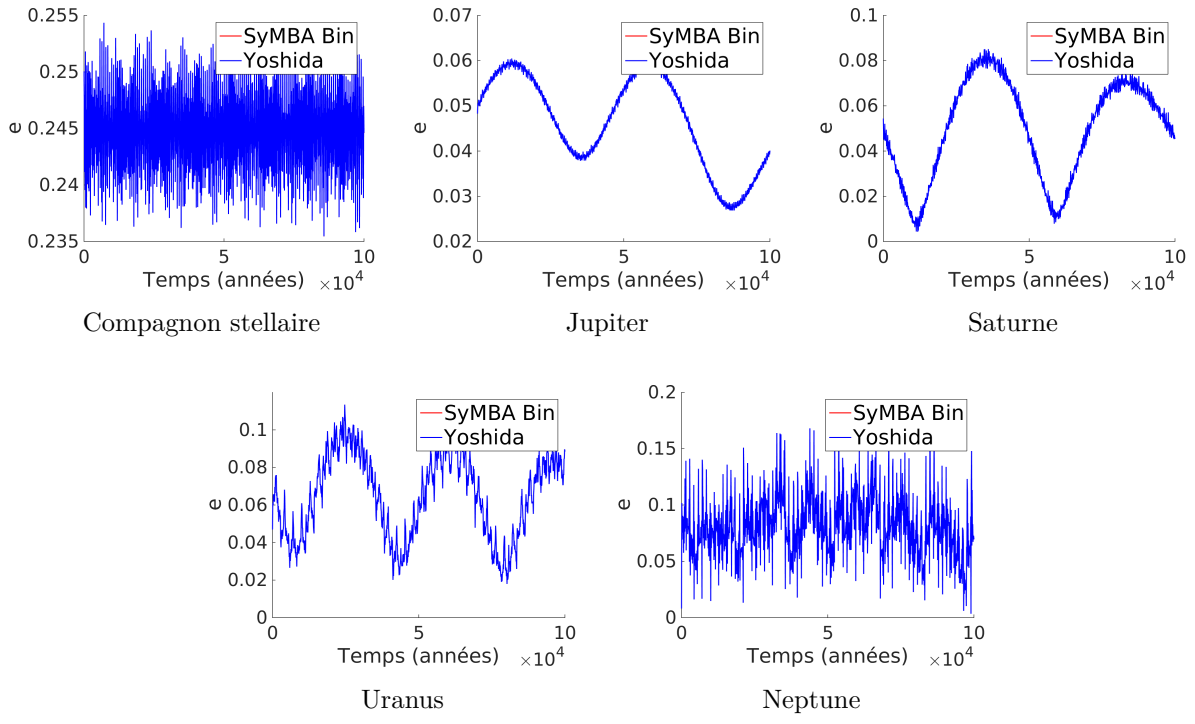


FIGURE 3.8 – Mêmes évolutions pour les excentricités

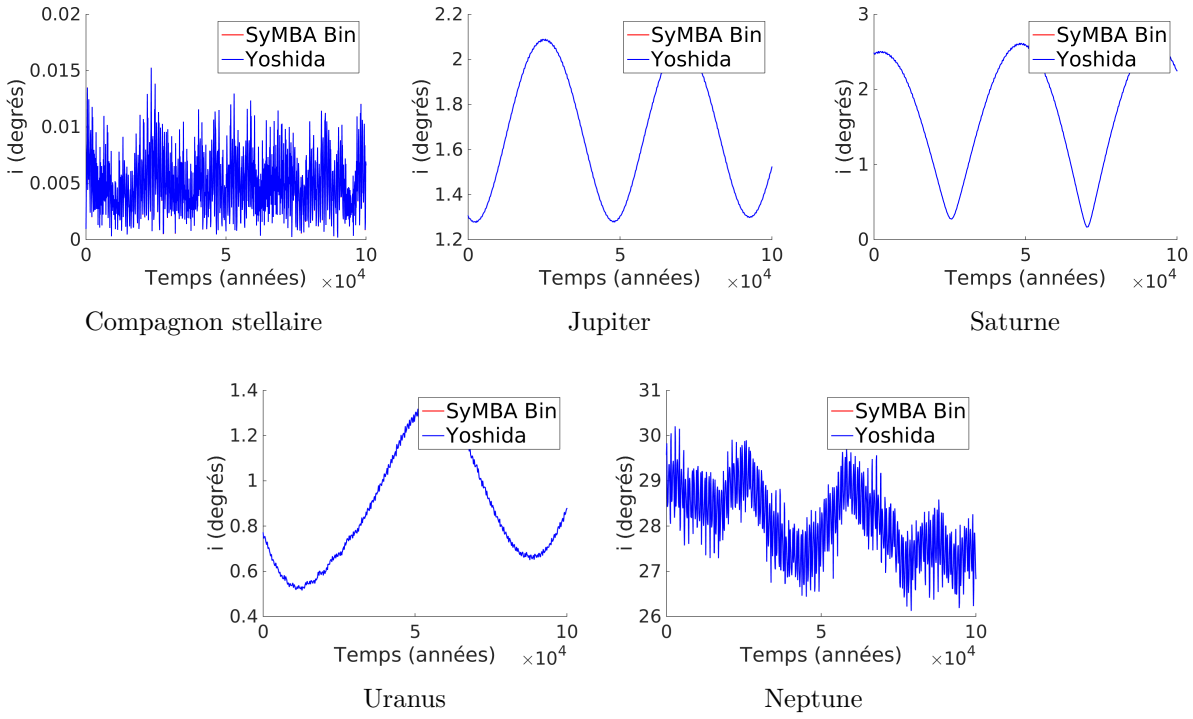


FIGURE 3.9 – Mêmes évolutions pour les inclinaisons

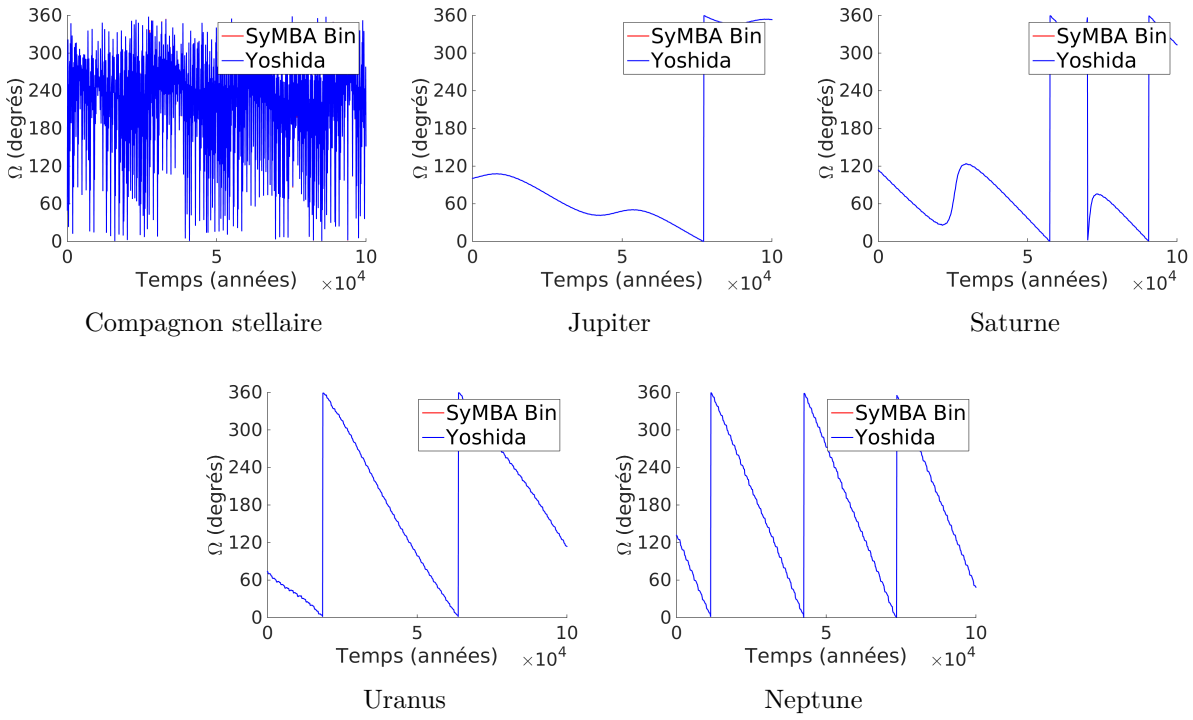


FIGURE 3.10 – Mêmes évolutions pour les longitudes des nœuds ascendants

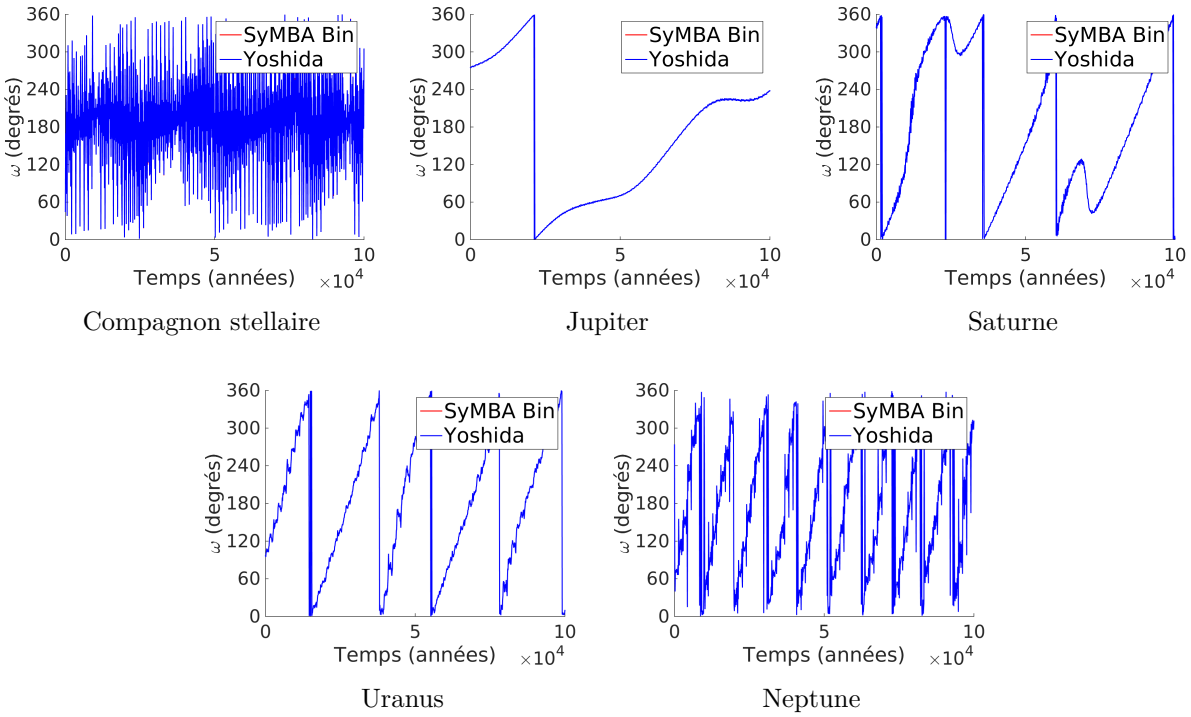


FIGURE 3.11 – Mêmes évolutions pour les arguments des périastres

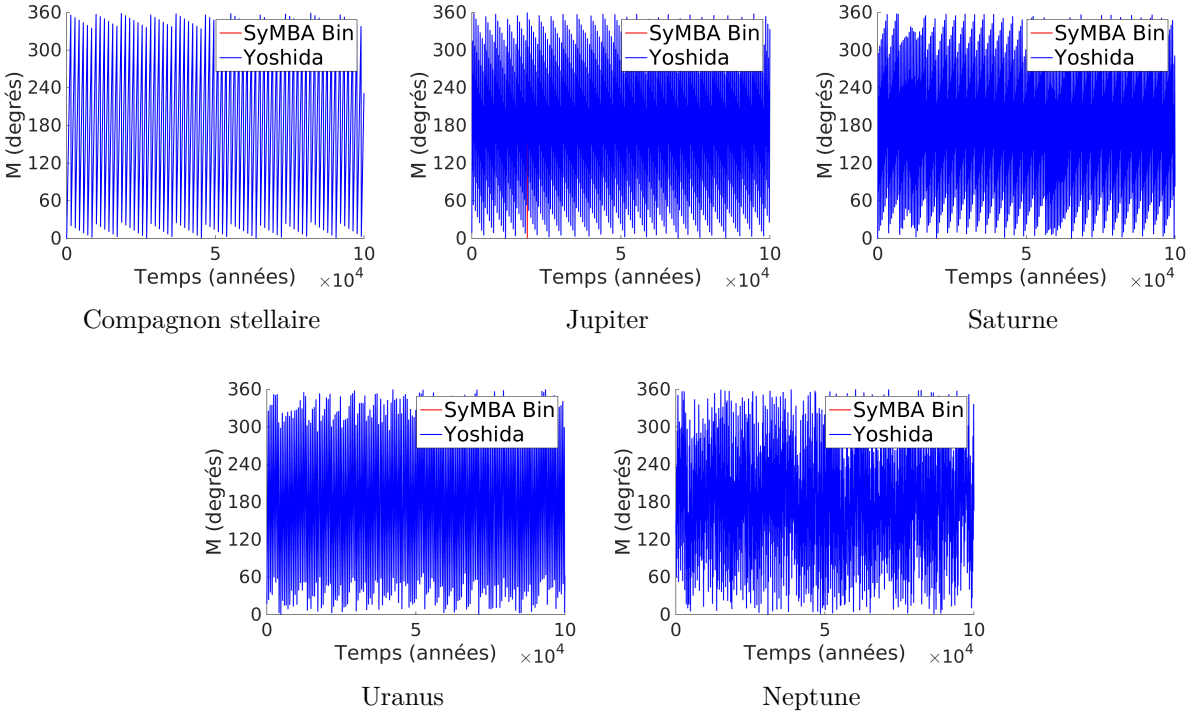


FIGURE 3.12 – Mêmes évolutions pour les anomalies moyennes

Corps	Masse	a (UA)	e	i (°)	Ω (°)	ω (°)	M (°)
1	1 MS	Etoile centrale					
2	0.5 MS	300	0.3	60	10^{-7}	10^{-7}	10^{-7}
3	0.054 MJ	30.07	0.0088	10^{-7}	131.72169	273.24966	130

TABLE 3.3 – Conditions initiales du système étudié dans [30]

Système binaire en résonance de Kozai

Nous allons à présent étudier un système décrit dans l'article [30] et constitué d'une étoile binaire et d'une planète. Notons que l'article ne renseigne qu'une partie des conditions initiales utilisées. En effet, celui-ci ne mentionne que les demi-grands axes, excentricités et inclinaisons utilisées. Néanmoins, les auteurs précisent que la planète est de type Neptune, nous avons donc utilisé les paramètres orbitaux de cette planète disponibles sur [49]. Ce système est donc constitué d'une étoile centrale (corps 1), d'un compagnon stellaire (corps 2) et d'une planète (corps 3). Les paramètres orbitaux initiaux que nous considérons ici sont présentés à la table 3.3. Ce système est un exemple typique du comportement chaotique que peut engendrer la résonance de Kozai dans les systèmes binaires, qui est présente lorsque le plan orbital de la planète est fortement incliné ($>\sim 40°$) par rapport au plan orbital du compagnon stellaire.

Lorsque nous intégrons l'évolution de ce système avec un pas de 10^{-2} ans, nous obtenons les conservations de l'hamiltonien et du moment angulaire total illustrées à la figure 3.13. Nous observons une erreur relative sur l'hamiltonien et le moment angulaire de l'ordre de 10^{-7}. Comme pour le système précédent, les erreurs provenant de l'arithmétique finie jouent un grand rôle sur l'erreur commise.

Nous avons ensuite comparé les évolutions des éléments orbitaux du compagnon stellaire et de la planète obtenues avec nos deux codes d'intégration. Ces résultats sont illustrés aux figures 3.14, 3.15, 3.16, 3.17, 3.18 et 3.19. Nous constatons sur ces figures que les deux codes donnent des résultats identiques.

Nous avons également comparé le graphique donné dans [30] pour l'évolution de l'excentricité de la planète (figure 3.20) à celui obtenu au moyen de notre code. Notons que l'allure générale est très semblable. La différence de période peut provenir de la différence sur les conditions initiales

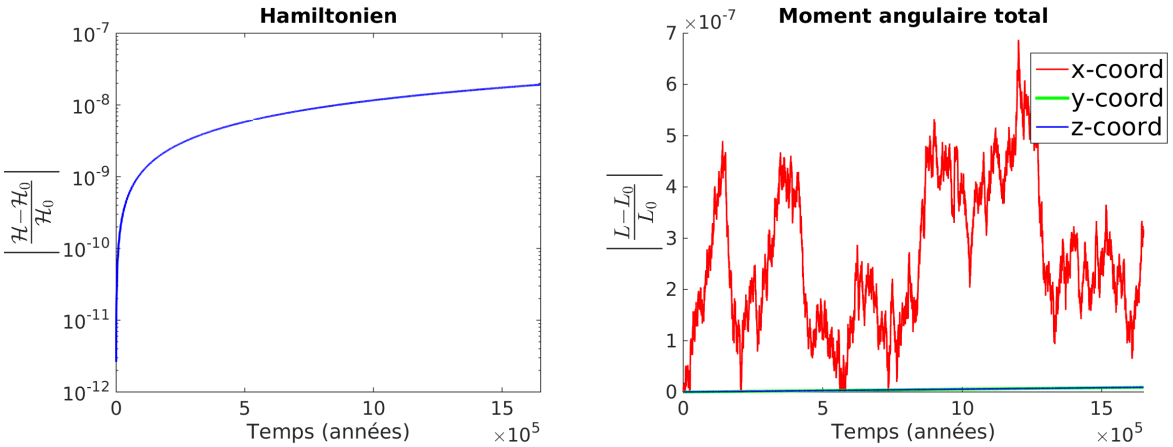
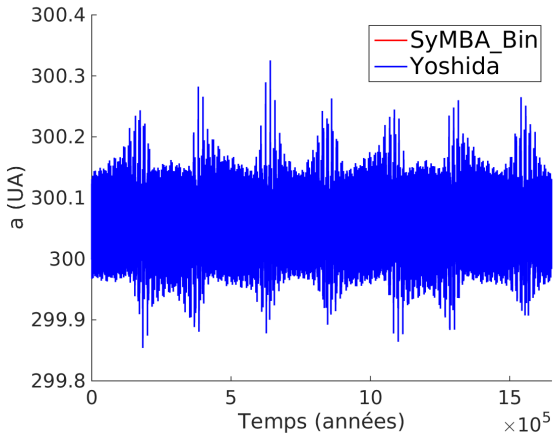
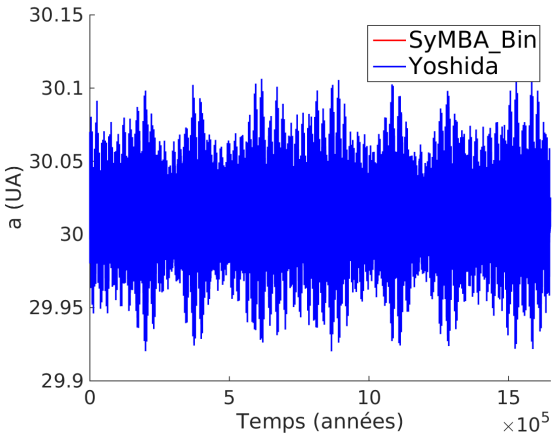


FIGURE 3.13 – Conservation des intégrales premières du mouvement pour le système de la table 3.3

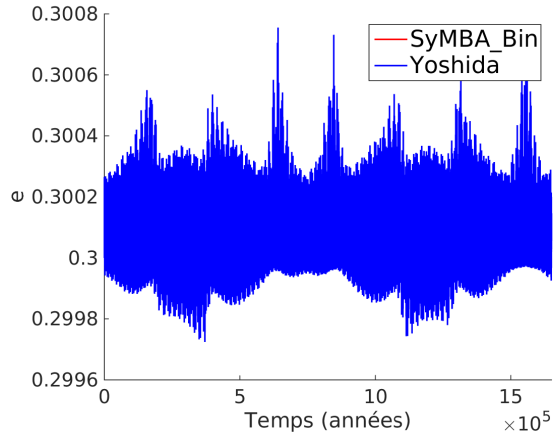


Compagnon stellaire

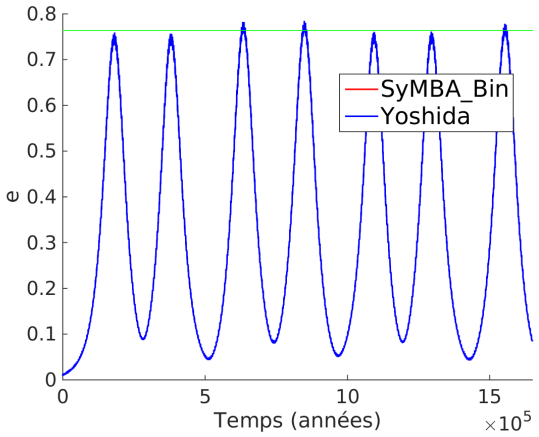


Planète

FIGURE 3.14 – Evolutions des demi-grands axes de la planète et du compagnon stellaire de la table 3.3 obtenus avec les intégrateurs SyMBA binaire et Yoshida.

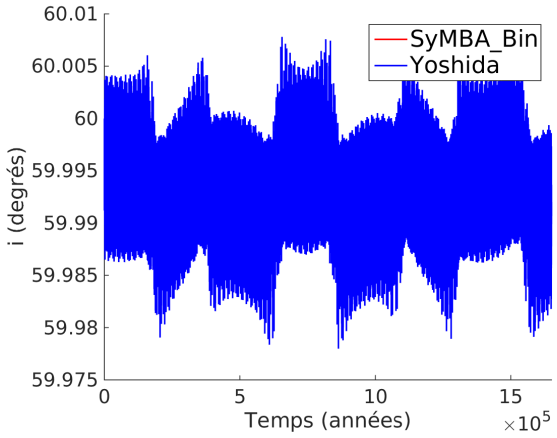


Compagnon stellaire

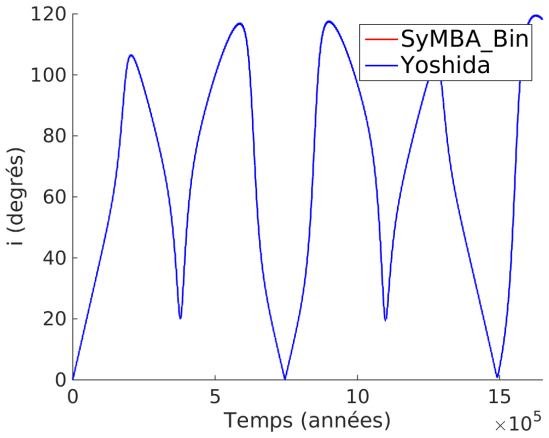


Planète

FIGURE 3.15 – Mêmes évolutions pour les excentricités

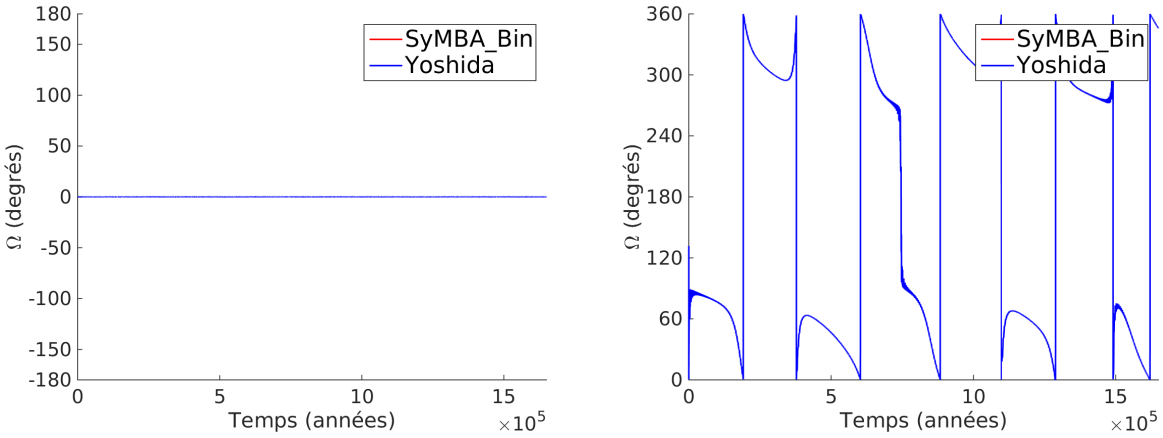


Compagnon stellaire



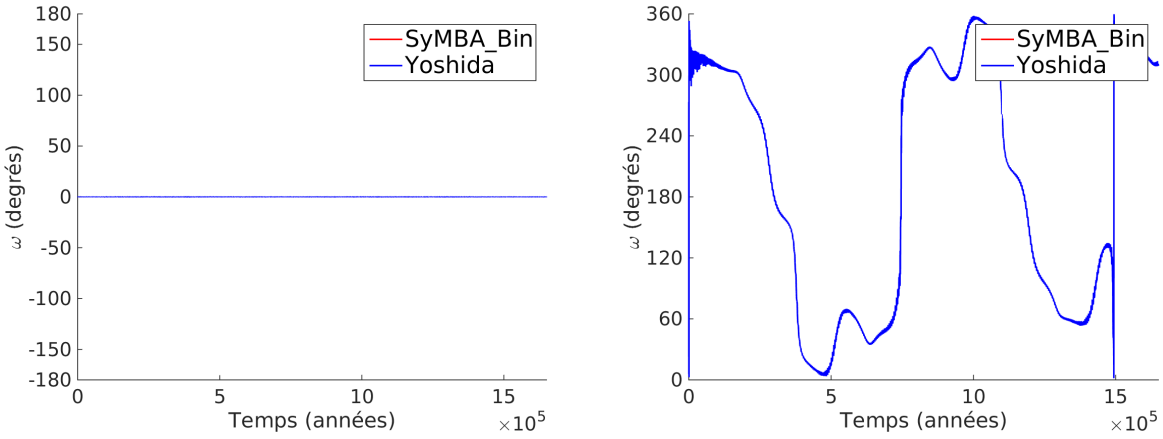
Planète

FIGURE 3.16 – Mêmes évolutions pour les inclinaisons



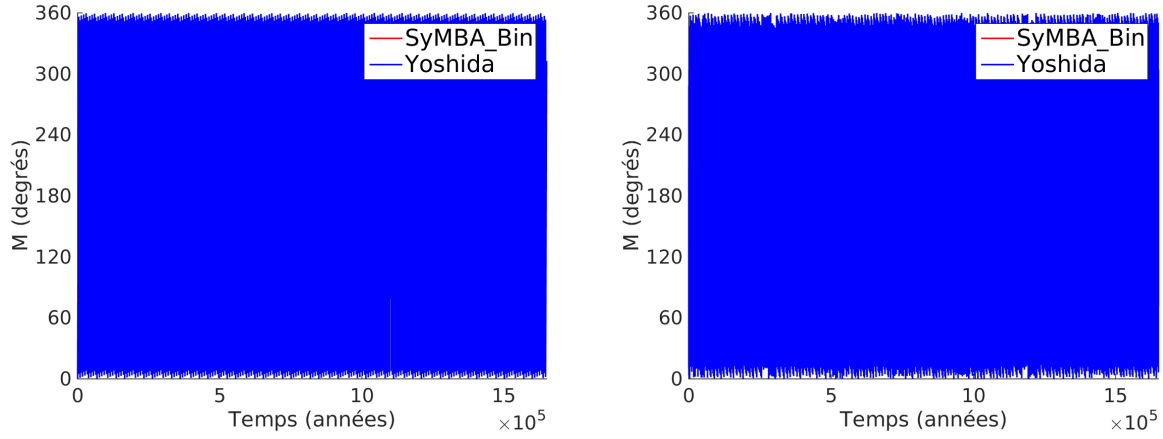
Compagnon stellaire Planète

FIGURE 3.17 – Mêmes évolutions pour les longitudes des nœuds ascendants



Compagnon stellaire Planète

FIGURE 3.18 – Mêmes évolutions pour les arguments des périastres



Compagnon stellaire Planète

FIGURE 3.19 – Mêmes évolutions pour les anomalies moyennes

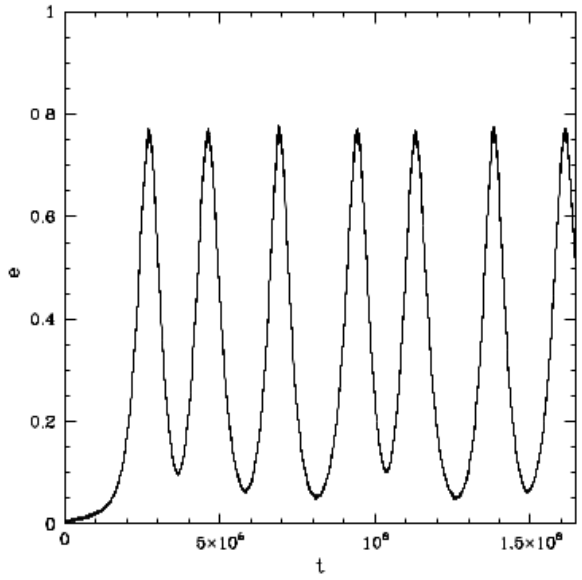


FIGURE 3.20 – Évolution de l'excentricité de la planète. Figure issue de [30].

induite par l'absence d'information sur certains paramètres dans [30]. En effet, l'article étudie la résonance de Kozai qui peut induire un comportement chaotique et est donc extrêmement sensible aux conditions initiales.

Remarquons que la ligne verte représentée à la figure 3.16, correspond à l'excentricité maximale prédite par la théorie de Kozai [22] qui vaut :

$$e_{max} = \sqrt{1 - \frac{5}{3} cos^2(i_0)},$$

où e_{max} est l'excentricité maximale atteinte et i_0 est l'inclinaison initiale du compagnon stellaire. Nous pouvons remarquer que notre évolution est en accord avec cette valeur. Les légers dépassements proviennent du fait que cette théorie est une approximation analytique (utilisant la troncature de polynômes de Legendre). De plus, cette théorie est développée pour un corps extérieur ayant une orbite circulaire (avec une excentricité égale à 0) ce qui n'est pas le cas de notre système.

Corps	Masse	a (UA)	e	i (°)	Ω (°)	ω (°)	M (°)
1	1.59 MS	Etoile centrale					
2	0.4 MS	21.5	0.45	10^{-7}	10^{-7}	160	353
3	1.7 MJ	2.13	0.17	10^{-7}	10^{-7}	74	104

TABLE 3.4 – Conditions initiales du système γ-Cephei (Source : [20])

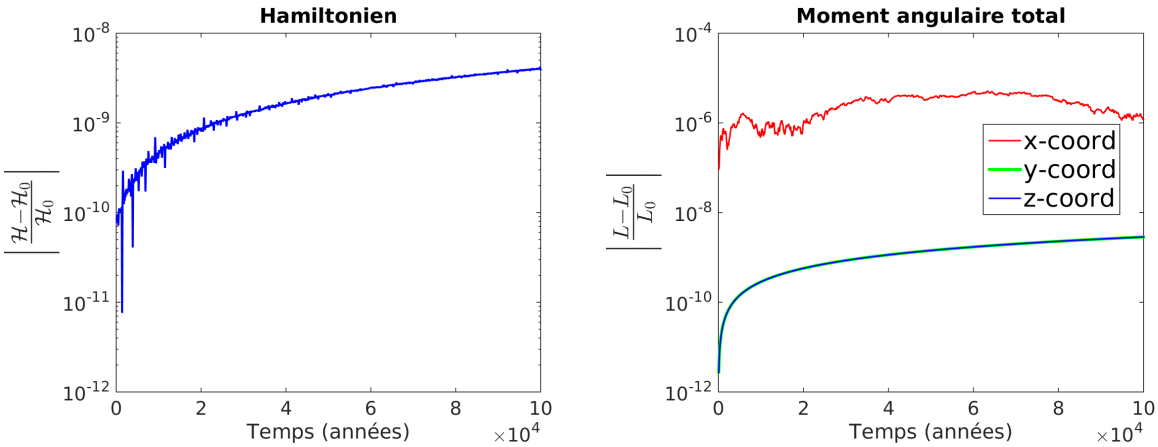


FIGURE 3.21 – Conservations des intégrales premières du mouvement pour le système de la table 3.4

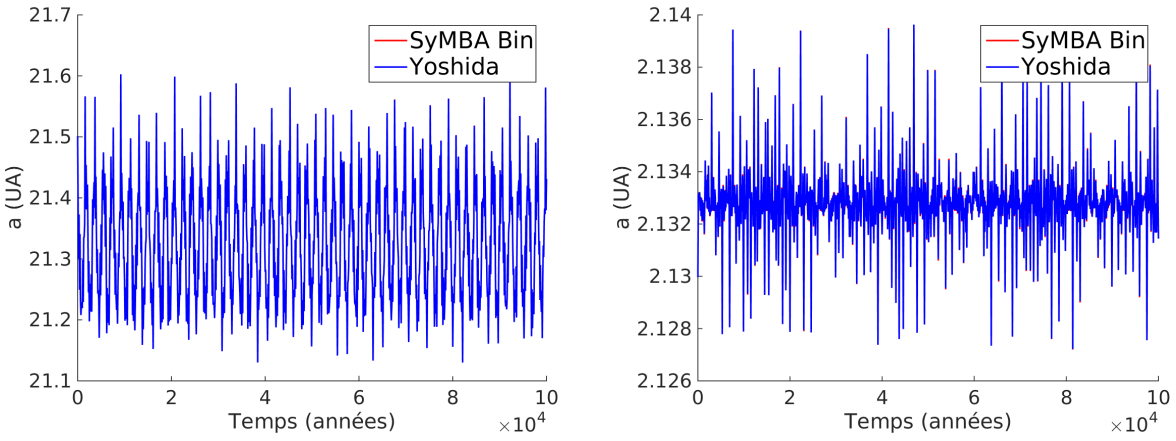
γ-Cephei

Nous allons étudier un dernier système : γ-Cephei. Ce système est composé d'une étoile binaire et d'une planète de type S. Sa dynamique a été étudiée dans [20]. Nous avons utilisé les éléments orbitaux référencés dans cet article. De plus, celui-ci fait varier l'excentricité et la masse du compagnon stellaire ainsi que l'inclinaison de la planète. Nous avons traité l'ensemble des cas présentés dans l'article afin de confronter nos résultats avec ceux de l'article. Dans tous les cas, nous avons bien obtenu des résultats similaires. Néanmoins, afin d'éviter d'alourdir ce manuscrit avec une multitude de graphiques, nous ne présenterons qu'une seule paramétrisation du système, reprise à la table 3.4. Notons que nous avons utilisé un pas de 10^{-3} ans. En effet, les corps étant plus proches de l'étoile centrale que dans les systèmes étudiés précédemment, ils nécessitent un pas plus faible pour conserver la même précision.

Les conservations de l'hamiltonien et du moment angulaire total sont illustrées à la figure 3.21. Nous constatons une erreur relative des intégrales premières du mouvement de l'ordre de 10^{-6} pour la composante x du moment angulaire total et de l'ordre de 10^{-9} pour les autres.

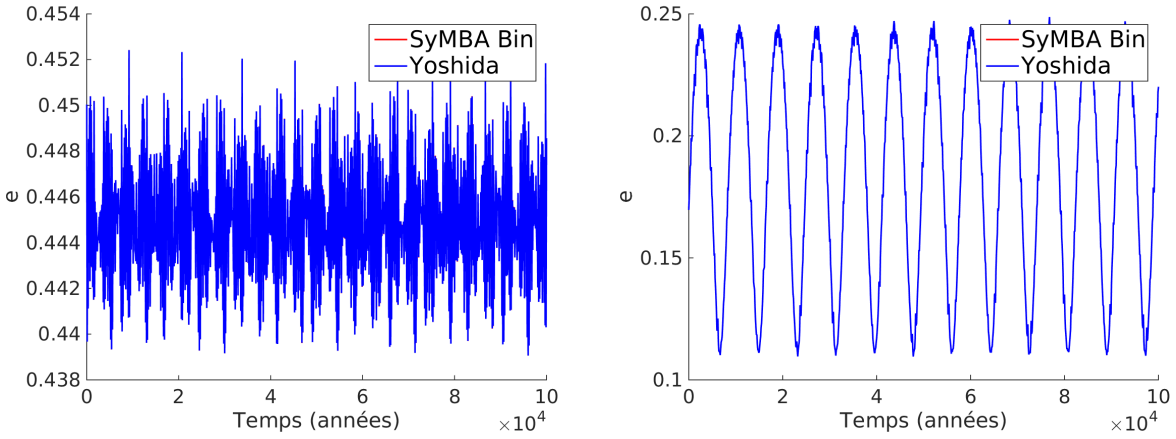
Nous avons ensuite comparé les évolutions des éléments orbitaux calculées avec notre code SyMBA pour les systèmes binaires à celles calculées avec le code provenant de [42].Les résultats sont illustrés aux figures 3.22, 3.23, 3.24, 3.25, 3.26 et 3.27. Nous pouvons constater que nous avons un léger décalage pour le demi-grand axe de la planète mais nous obtenons bien la même allure générale.

Notons que nous avons effectué l'évolution pour 10^5 ans. Si nous effectuons l'évolution sur un temps plus long, nous finirions par avoir un décalage entre les deux codes dû à l'accumulation des erreurs des schémas et des erreurs numériques. Néanmoins l'allure générale restera identique. Ce décalage est illustré aux figures 3.28, 3.29, 3.30, 3.31, 3.32 et 3.33.



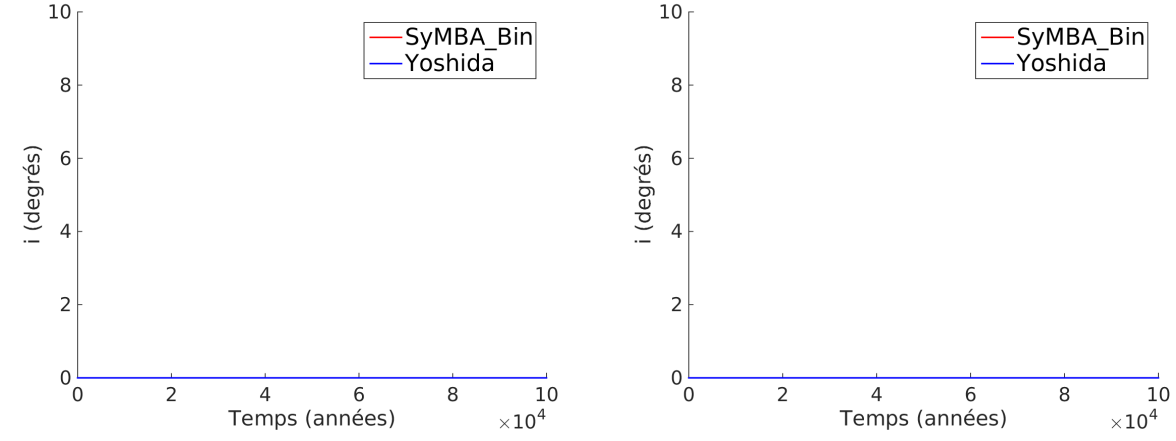
Compagnon stellaire Planète

FIGURE 3.22 – Evolutions des demi-grands axes de la planète et du compagnon stellaire de la table 3.4 obtenus avec les intégrateurs SyMBA binaire et Yoshida.



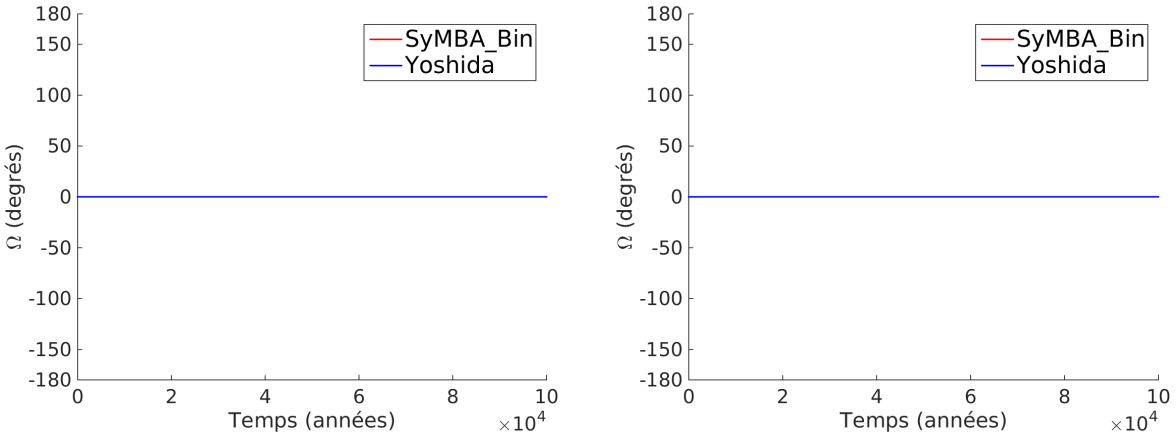
Compagnon stellaire Planète

FIGURE 3.23 – Mêmes évolutions pour les excentricités



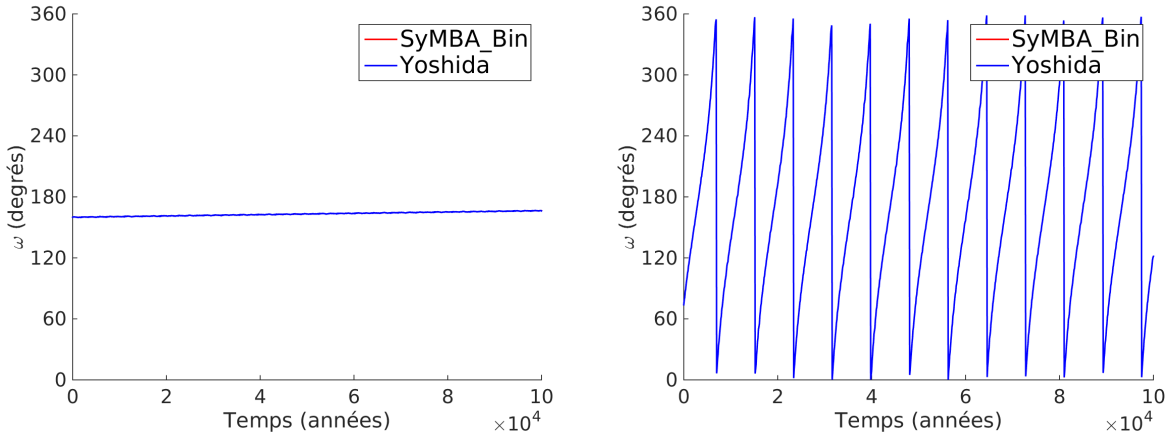
Compagnon stellaire Planète

FIGURE 3.24 – Mêmes évolutions pour les inclinaisons



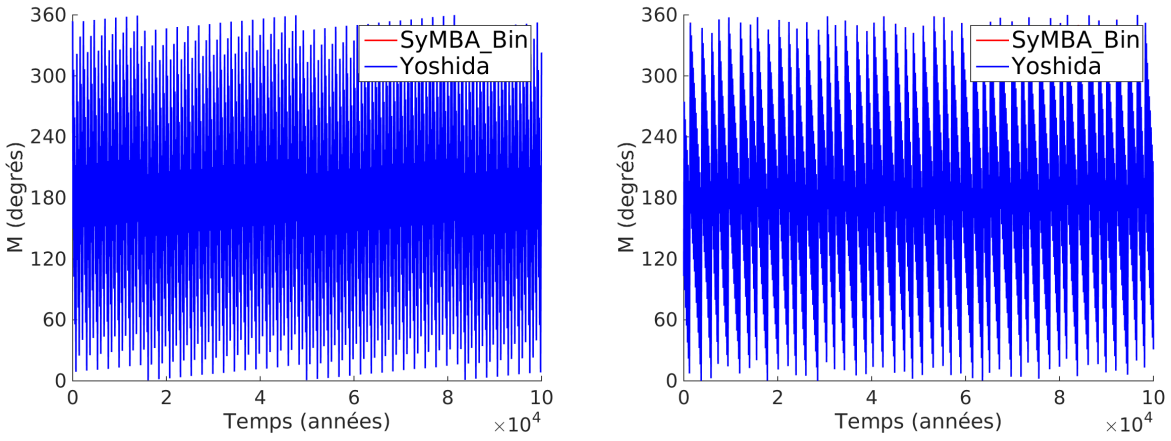
Compagnon stellaire Planète

FIGURE 3.25 – Mêmes évolutions pour les longitudes des nœuds ascendants



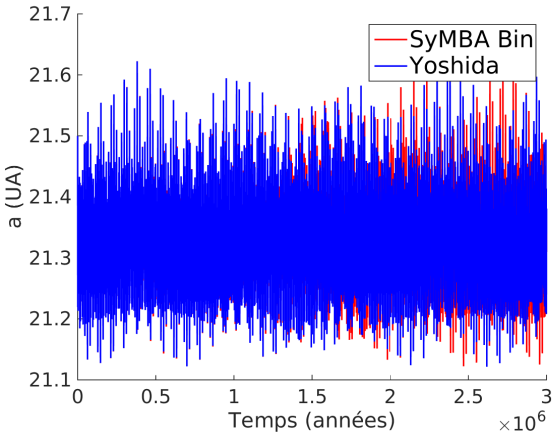
Compagnon stellaire Planète

FIGURE 3.26 – Mêmes évolutions pour les arguments des périastres

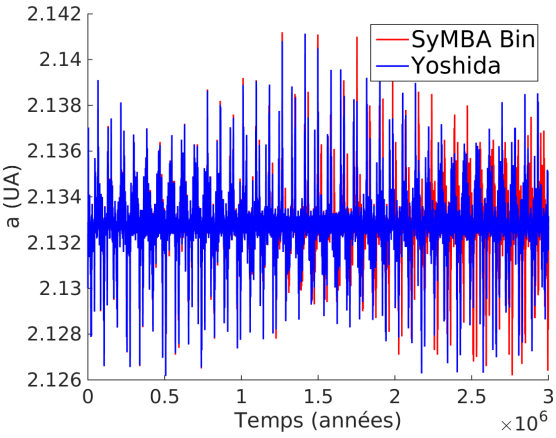


Compagnon stellaire Planète

FIGURE 3.27 – Mêmes évolutions pour les anomalies moyennes

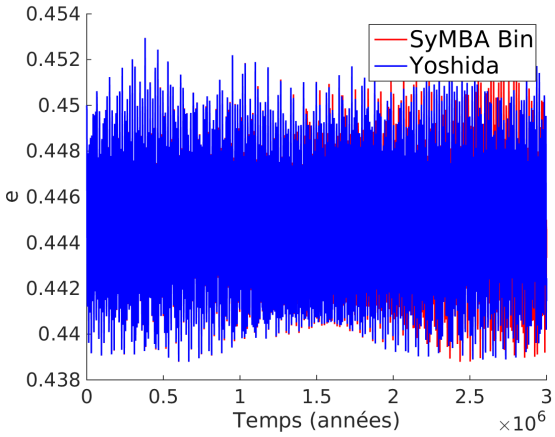


Compagnon stellaire

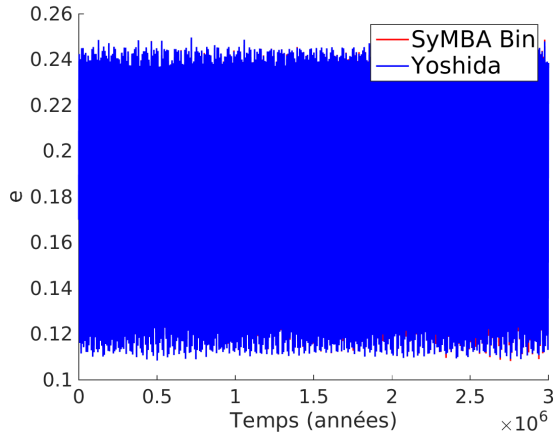


Planète

FIGURE 3.28 – Mêmes évolutions des demi-grands axes qu'à la figure 3.22 mais sur une période d'intégration plus longue.

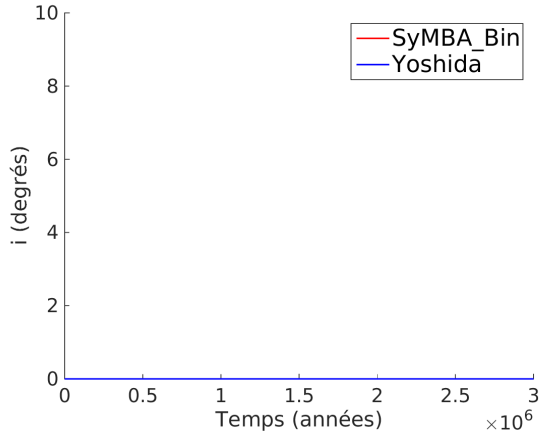


Compagnon stellaire

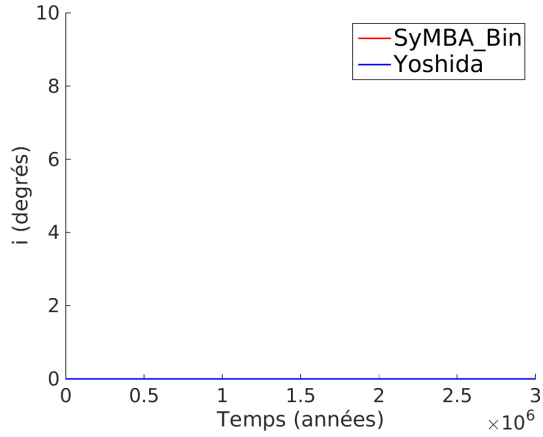


Planète

FIGURE 3.29 – Mêmes évolutions des excentricités qu'à la figure 3.23 mais avec une période d'intégration plus longue.

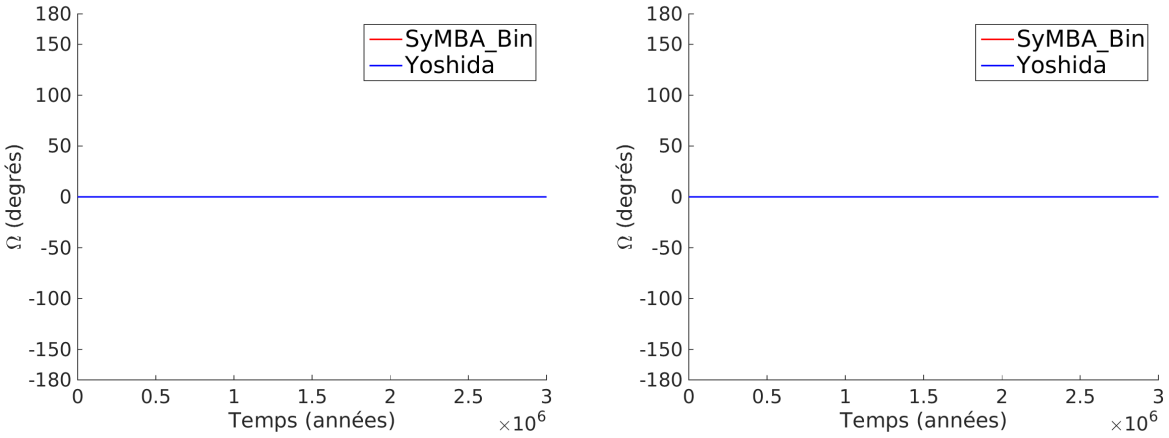


Compagnon stellaire



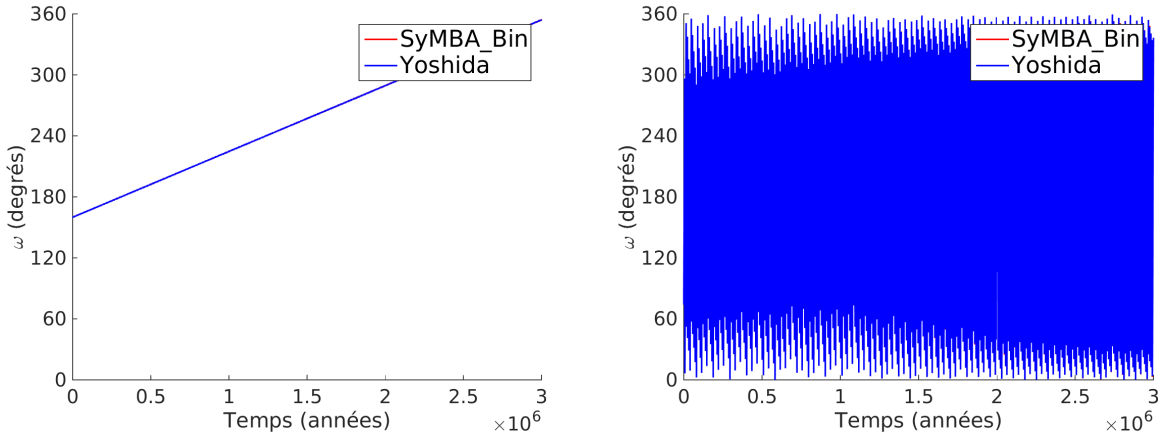
Planète

FIGURE 3.30 – Mêmes évolutions des inclinaisons qu'à la figure 3.24 mais avec une période d'intégration plus longue.



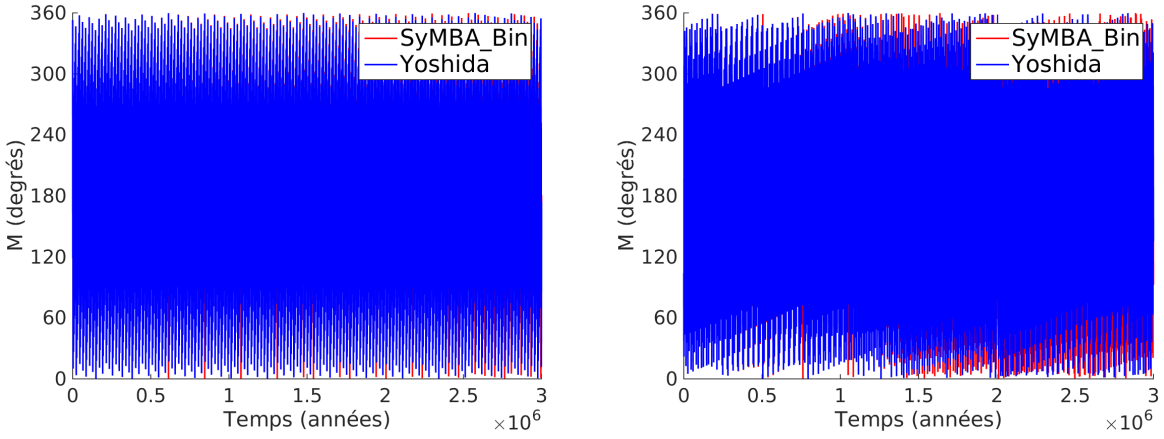
Compagnon stellaire Planète

FIGURE 3.31 – Mêmes évolutions des longitudes des nœuds ascendants qu'à la figure 3.25 mais avec une période d'intégration plus longue.



Compagnon stellaire Planète

FIGURE 3.32 – Mêmes évolutions des arguments des périastres qu'à la figure 3.26 mais avec une période d'intégration plus longue.



Compagnon stellaire Planète

FIGURE 3.33 – Mêmes évolutions des anomalies moyennes qu'à la figure 3.27 mais avec une période d'intégration plus longue.

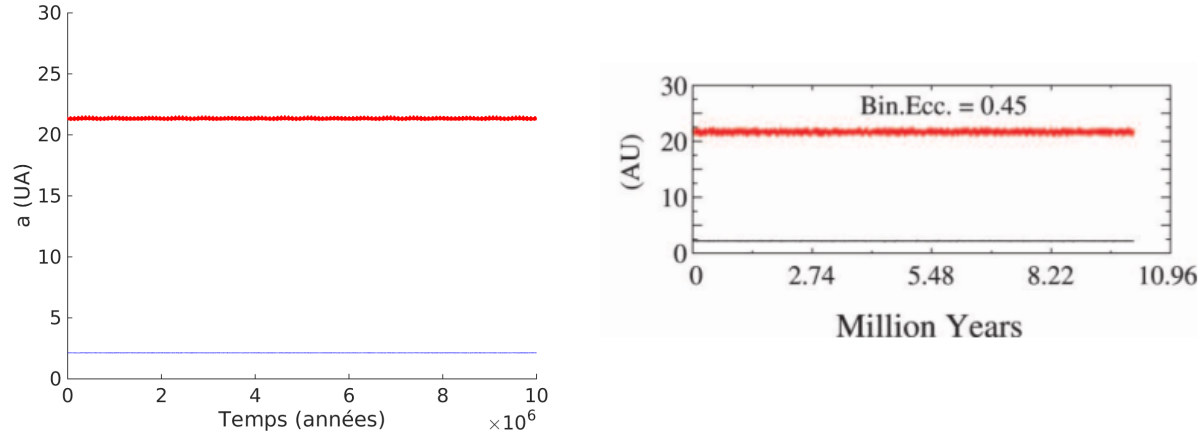


FIGURE 3.34 – Comparaison des évolutions des demi-grands axes obtenus avec notre code (à gauche) et tirées de l'article [20] (à droite)

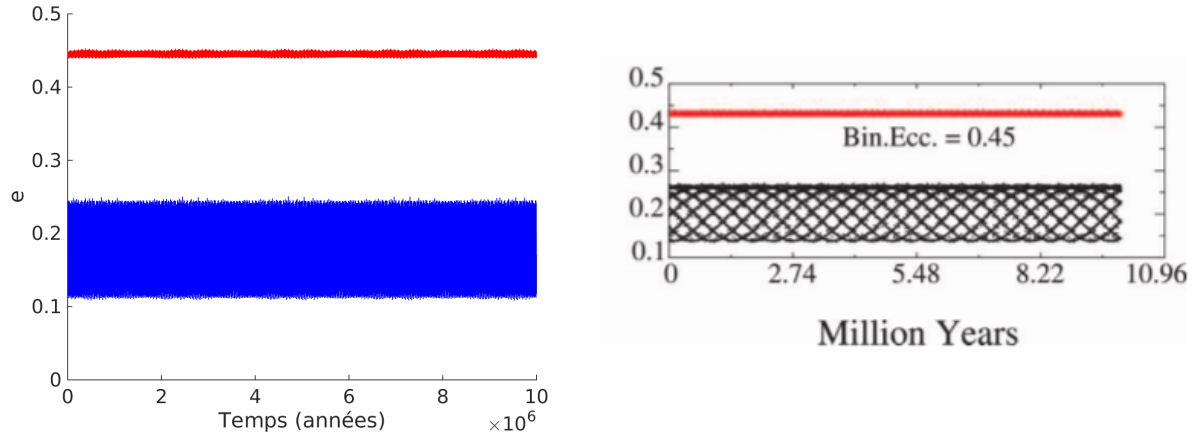


FIGURE 3.35 – Comparaison des évolutions des excentricités obtenus avec notre code (à gauche) et tirées de l'article [20] (à droite)

Nous avons également comparé nos résultats avec ceux de l'article [20]. Ces comparaisons des évolutions des demi-grands axes et excentricités sont données aux figures 3.34 et 3.35. Nous constatons bien une allure identique. Rappelons que les autres jeux de paramètres traités par l'article [20] ont été testés et ont donné des résultats similaires.

Les trois systèmes binaires détaillés à cette section nous ont permis de confronter nos évolutions numériques à celles présentes dans la littérature et attestent de la validité de notre code. Notons que le système de la table 3.1 possède, durant son évolution, des rencontres proches entre les planètes. Ces simulations valident donc également la partie du code traitant des rencontres proches.

Conclusion

Nous avons développé et validé un code décrivant les évolutions à long terme des systèmes binaires en tenant compte des rencontres proches. Ce code est un outil qui peut être utilisé dans de nombreuses applications. Il le sera tout d'abord dans le cadre de l'étude de la formation des systèmes planétaires lors du stage. Il peut également être utilisé pour étudier la stabilité des systèmes en y ajoutant un détecteur de chaos, similairement à celui présenté dans [42].

Néanmoins, afin d'effectuer ces études, il nous reste un cas particulier à résoudre. En effet, en cas de rencontres proches avec l'étoile centrale, il peut se produire les même problèmes que ceux mentionnés dans [14] pour SyMBA classique. Le code sera donc modifié afin d'également tenir compte de ce problème durant le stage.

Deux autres modifications pourraient également être implémentées. Tout d'abord l'implémentation du correcteur présenté dans [10] pourrait se révéler intéressant afin d'augmenter la précision du code. Notons que cela permettrait l'utilisation d'un pas légèrement supérieur. Néanmoins, cette modification n'est pas indispensable dans notre cas.

Ensuite, la modification du code pour les systèmes de type P pourrait être réalisé. Cette modification est détaillée dans [10]. Cependant, ces systèmes présentent des évolutions moins riches dynamiquement.

Deuxième partie

STAGE : Application à la formation des systèmes des planètes géantes

Introduction

Dans le cadre de la finalité approfondie, un stage de recherche a prolongé l'étude réalisée dans le mémoire. Celui-ci a été effectué au département de physique de l'Université de Thessalonique (Aristotle University of Thessaloniki) de février à mai 2017 sous la direction du Prof. K. Tsiganis.

Le but du stage est d'appliquer l'intégrateur créé dans le cadre du mémoire à la formation des systèmes planétaires dans les systèmes binaires. En effet, suite à la découverte continue de planètes dans les systèmes binaires ([34]), la question de la formation de celles-ci se pose. La pertinence de cette question est renforcée par l'hypothèse que la moitié des systèmes dans notre univers seraient multiples (binaires, triples ou plus) ([38]).

En particulier, nous nous intéressons à la phase finale du disque protoplanétaire caractérisée par la migration des planètes géantes (migration de type II) due aux interactions avec le disque de gaz. Nous supposons que les planètes géantes sont entièrement formées à ce stade du processus de formation. Nous avons analysé l'évolution dynamique de plus de 1000 systèmes de planètes géantes en migration autour de leur étoile-hôte et soumises à l'interaction gravitationnelle d'un compagnon stellaire. Cette étude s'inspire de celle de Libert & Tsiganis [29] à une seule étoile et la généralise aux étoiles binaires afin de mettre en évidence les similitudes et différences observées dans les systèmes planétaires formés. Les systèmes étudiés comprennent donc deux planètes en orbite autour d'une étoile munie d'un compagnon stellaire éloigné dont nous avons fait varier les paramètres initiaux afin d'étudier l'influence de celui-ci lors du processus de migration.

D'un point de vue pratique, la migration est implémentée selon le code de Sotiriadis et al. [47]. Ce code a été inséré dans notre intégrateur pour les systèmes binaires. Au cours du stage, nous avons également modifié notre algorithme afin de traiter correctement les rencontres proches entre les planètes et l'étoile centrale.

Au chapitre 4, nous détaillons les modifications effectuées à notre code en cas de rencontres proches avec l'étoile. La migration de type II et son implémentation dans le code font l'objet du chapitre 5. Enfin les simulations effectuées sont discutées au chapitre 6.

Chapitre 4

Rencontres proches avec le corps central

Dans le cadre du mémoire, nous avons solutionné les problèmes d'intégrations lors des rencontres proches entre les planètes. Comme le précisent Duncan & Levison [14], un problème survient également en cas de rencontre proche entre une planète et l'étoile centrale du système. En effet, dans ce cas, la vitesse de la planète augmente fortement, comme le prédisent les deuxième et troisième lois de Kepler. Cette vitesse élevée entraîne une augmentation de $\mathcal{H}^{BE}_{Jump}$ (défini à l'équation (3.2.1)). Cette augmentation va entraîner une violation de la contrainte (2.2.15). Puisque la quantité

$$\epsilon_1 = \frac{\mathcal{H}^{BE}_{Jump}}{\mathcal{H}^{BE}_{Int} + \mathcal{H}^{BE}_{Kep}}$$

va augmenter drastiquement et nous pouvons perdre la propriété :

$$\epsilon_1 << 1.$$

Dans le but d'étudier cette contrainte, nous avons intégré l'évolution du système présenté à la table 4.1 avec et sans le compagnon binaire (corps numéro 2), avec le code SyMBA et son adaptation aux systèmes binaires. Les valeurs des trois composantes de l'hamiltonien ainsi que celles de ϵ_1 et ϵ_2 sont présentées aux figures 4.1 et 4.2. Rappelons que

$$\epsilon_2 = \frac{\mathcal{H}^{BE}_{Int}}{\mathcal{H}^{BE}_{Kep}}$$

requiert la même condition que ϵ_1 (cfr. (2.2.15)). Notons que les paramètres initiaux du corps 3 ont été choisis de manière à ce qu'il passe à proximité du corps central. En effet, la distance minimale avec le corps central est la distance au péricentre :

$$r_{3,min} = a_3(1 - e_3) = 0.5(1 - 0.9) = 0.05 \text{ UA}.$$

Nous avons donc bien une rencontre proche entre le corps 3 et l'étoile centrale.

Nous pouvons constater sur les figures 4.1 et 4.2 que $\mathcal{H}^{HC}_{Jump}$ et $\mathcal{H}^{BE}_{Jump}$ ont des valeurs supérieures à celles que nous pouvions observer aux figures 3.1 et 3.3 respectivement. Cet accroissement entraîne l'augmentation de ϵ_1 illustrée aux figures 4.1 et 4.2 de l'ordre de 10^{-1} qui, dans le mémoire, était de l'ordre de 10^{-3} et 10^{-4} pour SyMBA et pour son adaptation aux systèmes binaires respectivement, en cas de non rencontre avec le corps central.

Cette augmentation de ϵ_1 en cas de rencontre entraînera potentiellement une plus grande erreur de l'intégrateur. En effet, comme explicité dans le mémoire, l'erreur est d'ordre $(\epsilon_1+\epsilon_2)\tau^2$. Une augmentation de ϵ_1 engendre donc une augmentation de l'erreur. Afin de quantifier cette erreur, nous avons étudié la conservation de l'hamiltonien lors de l'intégration du système de la

Corps	Masse	a (UA)	e	i (°)	Ω (°)	ω (°)	M (°)
1	1 MS	Etoile centrale					
2	1 MS	1000	10^{-5}	10^{-5}	10^{-5}	10^{-5}	10^{-5}
3	4 MJ	0.5	0.9	10^{-5}	10^{-5}	10^{-5}	10^{-5}
4	1 MJ	20	0.25	10^{-5}	10^{-5}	10^{-5}	10^{-5}

TABLE 4.1 – Conditions initiales d'un système dont un corps passe à proximité de l'étoile centrale

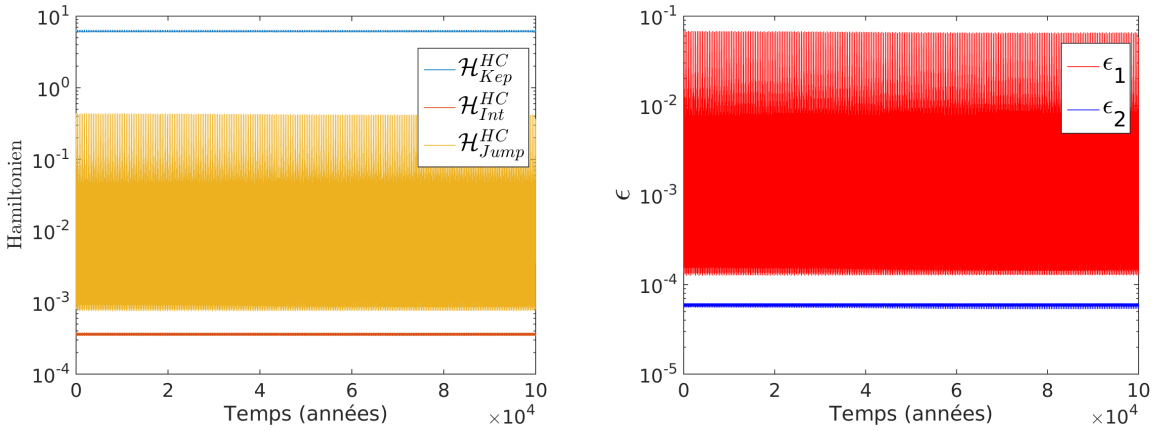


FIGURE 4.1 – Valeurs des composantes de l'hamiltonien ainsi que les valeurs de ϵ_1 et ϵ_2 pour le système de la table 4.1 sans le compagnon stellaire, intégré avec SyMBA

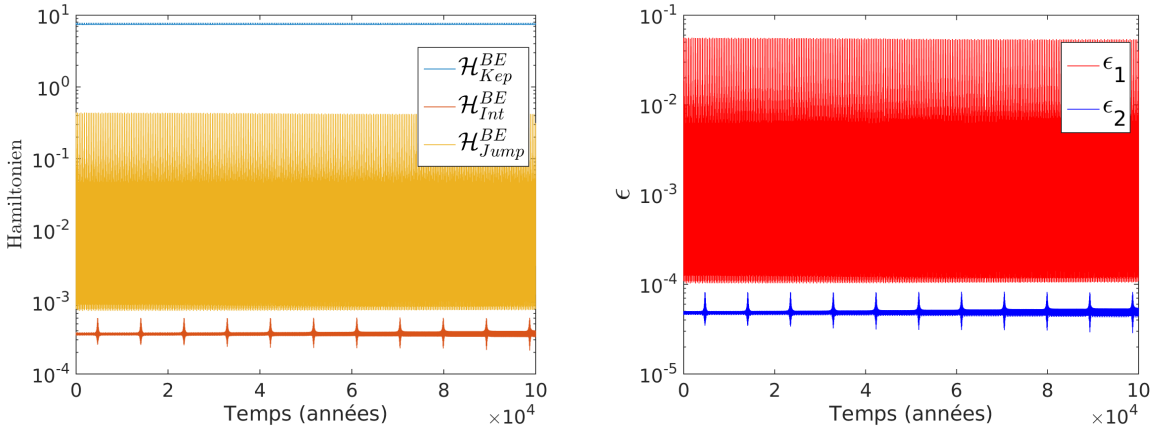


FIGURE 4.2 – Valeurs des composantes de l'hamiltonien ainsi que les valeurs de ϵ_1 et ϵ_2 pour le système de la table 4.1 avec notre code pour les systèmes binaires

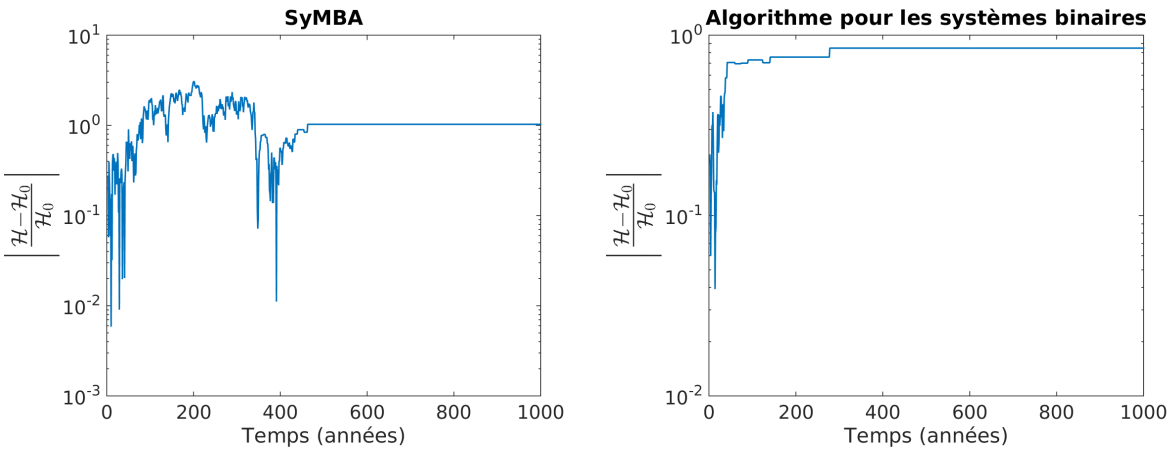


FIGURE 4.3 – Conservation de l'hamiltonien en cas de rencontre proche avec le corps central pour le système de la table 4.1 considéré sans compagnon stellaire (à gauche) et avec compagnon binaire (à droite)

table 4.1 avec un pas de 10^{-2} ans avec notre nouvel algorithme pour systèmes binaires et avec SyMBA sans le compagnon stellaire. Nous pouvons constater à la figure 4.3 que l'hamiltonien n'est plus conservé.

Afin de palier à ce problème, M. Duncan & H. Levison [14] ont proposé une solution pour SyMBA. Nous avons transposé cette solution dans notre algorithme pour les systèmes binaires. Notons que, comme pour l'algorithme Helio, une réduction du pas pourrait également résoudre le problème. Cependant cela entraînerait une augmentation considérable du temps d'intégration.

Nous allons donc, dans un premier temps, présenter la solution [14] implémentée dans SyMBA (section 4.1). Ensuite, nous présenterons son adaptation pour les systèmes binaires (section 4.2).

4.1 Systèmes contenant une seule étoile

Nous commençons par nous intéresser au cas sans étoile binaire. Dans cette section nous considérons donc le corps 1 comme le corps central (l'étoile) et les autres corps comme étant les planètes du système. Nous allons commencer par décrire les modifications apportées en cas de rencontre proche avec le corps central mais sans rencontres proches entre les planètes. Nous allons donc expliciter les changements apportés à l'algorithme Helio décrit à la section 2.2.6. Ensuite, nous discuterons du cas avec des planètes ayant des rencontres proches. Ces modifications sont issues de l'article de M. Duncan & H. Levison [14].

4.1.1 Algorithme sans rencontres proches entre les planètes

Nous commençons par rappeler la subdivision de l'hamiltonien dans l'algorithme Helio :

$$\mathcal{H} = \mathcal{H}_{Int}^{HC} + \mathcal{H}_{Kep}^{HC} + \mathcal{H}_{Jump}^{HC},$$

avec

$$\mathcal{H}_{Kep}^{HC} = \sum_{i=2}^{n} \left(\frac{\left\| p_i^{HC} \right\|^2}{2m_i} - \frac{Gm_1 m_i}{\left\| r_i^{HC} \right\|} \right),$$

$$\mathcal{H}_{Int}^{HC} = -\sum_{i=2}^{n-1} \sum_{i<j\leqslant n} \frac{Gm_i m_j}{\|r_{ij}\|}$$

et

$$\mathcal{H}_{Jump}^{HC} = \frac{1}{2m_1} \left\| \sum_{i=2}^{n} p_i^{HC} \right\|^2,$$

où $\|r_{ij}\| = \left\| r_i^{HC} - r_j^{HC} \right\|$.

Une possibilité pour résoudre le problème de rencontres proches avec le corps central serait de réduire le pas en partitionnant $\mathcal{H}_{Jump}^{HC}$ en fonction des ordres de grandeur de ses différents termes, comme cela a été effectué dans le cas des rencontres proches entre les planètes. Cependant cette technique demande une réduction du pas d'intégration. En cas de rencontre proche avec le corps central et entre les planètes simultanément, nous aurions une double réduction du pas ce qui rendrait l'algorithme extrêmement lent. Ce n'est donc pas la solution qui a été adoptée dans [14] et implémentée dans SyMBA. La solution proposée est d'introduire en partie $\mathcal{H}_{Jump}^{HC}$ dans $\mathcal{H}_{Kep}^{HC}$ de manière à réduire ϵ_1.

L'article [14] discute de cette inclusion de $\mathcal{H}_{Jump}^{HC}$ dans $\mathcal{H}_{Kep}^{HC}$. Afin d'effectuer celle-ci, nous utilisons une fonction de transition $F\left(\left\|r_2^{HC}\right\|^2, \left\|r_3^{HC}\right\|^2, ..., \left\|r_n^{HC}\right\|^2\right)$. Notons que la fonction de transition que nous exprimons ici est légèrement différente de celle présentée dans [14]. En effet, la fonction réellement implémentée dans le code SyMBA est légèrement différente de celle de l'article [14] pour une raison qui ne nous est pas connue. Nous avons contacté les auteurs de SyMBA pour avoir une précision concernant cette différence mais nous n'avons malheureusement pas eu de réponse.

La nouvelle décomposition de l'hamiltonien est donc :

$$\mathcal{H} = \tilde{\mathcal{H}}_{Kep}^{HC} + \mathcal{H}_{Int}^{HC} + \tilde{\mathcal{H}}_{Jump}^{HC},$$

où

$$\tilde{\mathcal{H}}_{Kep}^{HC} = \sum_{i=2}^{n} \left(\frac{\left\|p_i^{HC}\right\|^2}{2m_i} - \frac{Gm_1m_i}{\left\|r_i^{HC}\right\|} \right) + \frac{1}{2m_1} \left\| \sum_{i=2}^{n} p_i^{HC} \right\|^2 F\left(\left\|r_2^{HC}\right\|^2, \left\|r_3^{HC}\right\|^2, ..., \left\|r_n^{HC}\right\|^2\right)$$

et

$$\tilde{\mathcal{H}}_{Jump}^{HC} = \frac{1}{2m_1} \left\| \sum_{i=2}^{n} p_i^{HC} \right\|^2 \left[1 - F\left(\left\|r_2^{HC}\right\|^2, \left\|r_3^{HC}\right\|^2, ..., \left\|r_n^{HC}\right\|^2\right)\right],$$

Nous allons à présent discuter des propriétés de la fonction de transition F afin de déboucher sur sa définition. Comme le précisent [14] et [10], cette fonction doit posséder trois propriétés.

Tout d'abord, nous voulons qu'elle soit égale à 0 si aucun corps n'a de rencontre proche avec l'étoile. Nous avons alors $\tilde{\mathcal{H}}_{Kep}^{HC} = \mathcal{H}_{Kep}^{HC}$ et $\tilde{\mathcal{H}}_{Jump}^{HC} = \mathcal{H}_{Jump}^{HC}$ et nous retombons sur la décomposition présentée dans le mémoire pour l'algorithme Helio. En effet, lorsqu'il n'y a pas de rencontre proche avec le corps central, il n'y a aucun raison de changer l'algorithme.

Ensuite, lorsqu'un corps a une rencontre proche avec le corps central, nous voulons

$$F\left(\left\|r_2^{HC}\right\|^2, \left\|r_3^{HC}\right\|^2, ..., \left\|r_n^{HC}\right\|^2\right) = 1.$$

Nous avons alors $\tilde{\mathcal{H}}_{Kep}^{HC} = \mathcal{H}_{Kep}^{HC} + \mathcal{H}_{Jump}^{HC}$ et $\tilde{\mathcal{H}}_{Jump}^{HC} = 0$ et $\mathcal{H}_{Jump}^{HC}$ est entièrement introduit dans $\tilde{\mathcal{H}}_{Kep}^{HC}$.

Enfin, tout comme la fonction de partition utilisée pour SyMBA, nous avons besoin que cette fonction soit lisse pour assurer la stabilité du code comme le précise [14]. La transition de 0 à 1

doit donc se faire progressivement.

Sur base de ces propriétés, nous définissons F comme :

$$F\left(\left\|r_2^{HC}\right\|^2,\left\|r_3^{HC}\right\|^2,...,\left\|r_n^{HC}\right\|^2\right)=1-\prod_{i=2}^{n}\left[1-\tilde{f}\left(\left\|r_i^{HC}\right\|^2\right)\right], \tag{4.1.1}$$

avec

$$\tilde{f}\left(\left\|r_i^{HC}\right\|^2\right)=\begin{cases}1 & \text{si } \left\|r_i^{HC}\right\|^2<R_1^2\\ 10\left(\frac{R_2^2-\left\|r_i^{HC}\right\|^2}{R_2^2-R_1^2}\right)^3-15\left(\frac{R_2^2-\left\|r_i^{HC}\right\|^2}{R_2^2-R_1^2}\right)^4+6\left(\frac{R_2^2-\left\|r_i^{HC}\right\|^2}{R_2^2-R_1^2}\right)^5 & \text{si } R_1^2<\left\|r_i^{HC}\right\|^2<R_2^2\\ 0 & \text{si } \left\|r_i^{HC}\right\|^2>R_2^2,\end{cases} \tag{4.1.2}$$

où R_1 et R_2 sont respectivement les bornes inférieure et supérieure de l'intervalle de transition. Cet intervalle de transition se situe entre la zone au sein de laquelle la planète est considérée comme ayant une rencontre proche avec le corps central et la zone où elle n'est pas considérée comme tel.

Sur base de cette nouvelle décomposition de l'hamiltonien, l'évolution se fera au moyen de la formule (2.2.13). Nous aurons alors

$$f(t_{k+1})=\exp\left(\frac{\tau}{2}\tilde{T}_{Jump}^{HC}\right)\exp\left(\frac{\tau}{2}T_{Int}^{HC}\right)\exp\left(\tau\tilde{T}_{Kep}^{HC}\right)\exp\left(\frac{\tau}{2}T_{Int}^{HC}\right)\exp\left(\frac{\tau}{2}\tilde{T}_{Jump}^{HC}\right)f(t_k),\qquad k\in\mathbb{N},$$

où $\tilde{T}_{Kep}^{HC}$, T_{Int}^{HC} et $\tilde{T}_{Jump}^{HC}$ sont les opérateurs linéaires associés à $\tilde{\mathcal{H}}_{Kep}^{HC}$, $\mathcal{H}_{Int}^{HC}$ et $\tilde{\mathcal{H}}_{Jump}^{HC}$ respectivement. Cette formule revient à faire évoluer le système d'un demi-pas suivant $\tilde{\mathcal{H}}_{Jump}^{HC}$, ensuite d'un demi-pas suivant $\mathcal{H}_{Int}^{HC}$, suivi d'une évolution d'un pas complet suivant $\tilde{\mathcal{H}}_{Kep}^{HC}$, puis à nouveau d'un demi-pas suivant $\mathcal{H}_{Int}^{HC}$ et enfin d'un demi-pas suivant $\tilde{\mathcal{H}}_{Jump}^{HC}$. Nous allons à présent étudier l'évolution sous les différentes parties de l'hamiltonien. Tout d'abord, l'évolution sous $\mathcal{H}_{Int}^{HC}$ a déjà été traitée à la section 2.2.6.

Nous ne devons donc étudier que les évolutions sous $\tilde{\mathcal{H}}_{Kep}^{HC}$ et $\tilde{\mathcal{H}}_{Jump}^{HC}$.

Evolution sous $\tilde{\mathcal{H}}_{Kep}^{HC}$

Cette évolution se calcule au moyen des équations d'hamilton (2.2.2). Nous avons alors les équations d'évolution

$$\begin{cases}\dfrac{dr_{i,c}^{HC}}{dt} &= \dfrac{\partial\tilde{\mathcal{H}}_{Kep}^{HC}}{\partial p_{i,c}^{HC}}\\ \dfrac{dp_{i,c}^{HC}}{dt} &= -\dfrac{\partial\tilde{\mathcal{H}}_{Kep}^{HC}}{\partial r_{i,c}^{HC}}\end{cases}\qquad i=1...n,\quad c=1,2,3, \tag{4.1.3}$$

où c correspond aux coordonnées x, y et z comme explicité dans le mémoire.

Par ailleurs, $\tilde{\mathcal{H}}_{Kep}^{HC}$ étant indépendant de r_1^{HC} et p_1^{HC}, cette équation devient, pour $i=1$,

$$\begin{cases}\frac{dr_{1,c}^{HC}}{dt} &= 0\\ \frac{dp_{1,c}^{HC}}{dt} &= 0\end{cases}\qquad c=1,2,3.$$

r_1^{HC} et p_1^{HC} sont donc invariants sous $\tilde{\mathcal{H}}_{Kep}^{HC}$. Les équations d'évolution (4.1.3) deviennent alors

$$\begin{cases} \dfrac{dr_{i,c}^{HC}}{dt} &= \dfrac{\partial \tilde{\mathcal{H}}_{Kep}^{HC}}{\partial p_{i,c}^{HC}} \\ \dfrac{dp_{i,c}^{HC}}{dt} &= -\dfrac{\partial \tilde{\mathcal{H}}_{Kep}^{HC}}{\partial r_{i,c}^{HC}} \end{cases} \qquad i = 2...n, \quad c = 1,2,3. \tag{4.1.4}$$

Nous devons donc calculer $\frac{\partial \tilde{\mathcal{H}}_{Kep}^{HC}}{\partial p_{i,c}^{HC}}$ et $-\frac{\tilde{\mathcal{H}}_{Kep}^{HC}}{\partial r_{i,c}^{HC}}$. Commençons par

$$\begin{aligned} \frac{\partial \tilde{\mathcal{H}}_{Kep}^{HC}}{\partial p_{i,c}^{HC}} &= \frac{\partial}{\partial p_{i,c}^{HC}} \left(\sum_{j=2}^{n} \left(\frac{\left\| p_j^{HC} \right\|^2}{2m_j} - \frac{Gm_1 m_j}{\left\| r_j^{HC} \right\|} \right) + \frac{1}{2m_1} \left\| \sum_{j=2}^{n} p_j^{HC} \right\|^2 F\left(\left\| r_2^{HC} \right\|^2, \left\| r_3^{HC} \right\|^2, ..., \left\| r_n^{HC} \right\|^2 \right) \right) \\ &= \sum_{j=2}^{n} \left(\frac{\partial}{\partial p_{i,c}^{HC}} \left[\frac{\left\| p_j^{HC} \right\|^2}{2m_j} \right] \right) + \frac{\partial}{\partial p_{i,c}^{HC}} \left[\frac{1}{2m_1} \left\| \sum_{j=2}^{n} p_j^{HC} \right\|^2 \right] F\left(\left\| r_2^{HC} \right\|^2, \left\| r_3^{HC} \right\|^2, ..., \left\| r_n^{HC} \right\|^2 \right). \end{aligned} \tag{4.1.5}$$

La dérivée $\frac{\partial}{\partial p_{i,c}^{HC}} \left[\frac{1}{2m_1} \left\| \sum_{j=2}^{n} p_j^{HC} \right\|^2 \right]$ correspond à $\frac{\partial \mathcal{H}_{Jump}^{HC}}{\partial p_{i,c}^{HC}}$ et a été déterminée à l'équation (2.2.30). Cette dérivée vaut donc

$$\frac{\partial}{\partial p_{i,c}^{HC}} \left[\frac{1}{2m_1} \left\| \sum_{j=2}^{n} p_j^{HC} \right\|^2 \right] = \frac{1}{m_1} \left[\sum_{j=2}^{n} p_{j,c}^{HC} \right]. \tag{4.1.6}$$

Il nous reste donc la dérivée suivante à calculer :

$$\begin{aligned} \frac{\partial}{\partial p_{i,c}^{HC}} \left[\frac{\left\| p_j^{HC} \right\|^2}{2m_j} \right] &= \frac{1}{2m_j} \sum_{a=1}^{3} \left(\frac{\partial}{\partial p_{i,c}^{HC}} \left[\left(p_{j,a}^{HC} \right)^2 \right] \right) \\ &= \frac{1}{2m_j} \sum_{a=1}^{3} \left(2 p_{j,a}^{HC} \frac{\partial p_{j,a}^{HC}}{\partial p_{i,c}^{HC}} \right) \\ &= \frac{1}{2m_j} \sum_{a=1}^{3} \left(2 p_{j,a}^{HC} \delta_{ij} \delta_{ac} \right) \\ &= \frac{p_{j,c}^{HC}}{m_j} \delta_{ij}. \end{aligned}$$

La dérivée (4.1.5) devient alors

$$\begin{aligned} \frac{\partial \tilde{\mathcal{H}}_{Kep}^{HC}}{\partial p_{i,c}^{HC}} &= \sum_{j=2}^{n} \left(\frac{p_{j,c}^{HC}}{m_j} \delta_{ij} \right) + \frac{1}{m_1} \left[\sum_{j=2}^{n} p_{j,c}^{HC} \right] F\left(\left\| r_2^{HC} \right\|^2, \left\| r_3^{HC} \right\|^2, ..., \left\| r_n^{HC} \right\|^2 \right) \\ &= \frac{p_{i,c}^{HC}}{m_i} + \frac{1}{m_1} \left[\sum_{j=2}^{n} p_{j,c}^{HC} \right] F\left(\left\| r_2^{HC} \right\|^2, \left\| r_3^{HC} \right\|^2, ..., \left\| r_n^{HC} \right\|^2 \right). \end{aligned} \tag{4.1.7}$$

Nous allons à présent nous intéresser à la dérivée de la seconde équation de 4.1.4 :

$$-\frac{\partial \tilde{\mathcal{H}}_{Kep}^{HC}}{\partial r_{i,c}^{HC}} = -\frac{\partial}{\partial r_{i,c}^{HC}} \left[\sum_{j=2}^{n} \left(\frac{\left\| p_j^{HC} \right\|^2}{2m_j} - \frac{Gm_1 m_j}{\left\| r_j^{HC} \right\|} \right) + \frac{1}{2m_1} \left\| \sum_{j=2}^{n} p_j^{HC} \right\|^2 F\left(\left\| r_2^{HC} \right\|^2, \left\| r_3^{HC} \right\|^2, ..., \left\| r_n^{HC} \right\|^2 \right) \right]$$

$$= \sum_{j=2}^{n} \left[\frac{\partial}{\partial r_{i,c}^{HC}} \left(\frac{G m_1 m_j}{\left\| r_j^{HC} \right\|} \right) \right] - \frac{1}{2m_1} \left\| \sum_{j=2}^{n} p_j^{HC} \right\|^2 \frac{\partial}{\partial r_{i,c}^{HC}} \left[F\left(\left\| r_2^{HC} \right\|^2, \left\| r_3^{HC} \right\|^2, ..., \left\| r_n^{HC} \right\|^2 \right) \right]. \tag{4.1.8}$$

Calculons les dérivées de cette dernière expression séparément. Tout d'abord,

$$\begin{aligned}
\frac{\partial}{\partial r_{i,c}^{HC}} \left(\frac{G m_1 m_j}{\left\| r_j^{HC} \right\|} \right) &= G m_1 m_j \frac{\partial}{\partial r_{i,c}^{HC}} \left[\left(\sum_{a=1}^{3} \left(r_{j,a}^{HC} \right)^2 \right)^{-\frac{1}{2}} \right] \\
&= -\frac{G m_1 m_j}{2} \left(\sum_{a=1}^{3} \left(r_{j,a}^{HC} \right)^2 \right)^{-\frac{3}{2}} \frac{\partial}{\partial r_{i,c}^{HC}} \left[\sum_{a=1}^{3} \left(r_{j,a}^{HC} \right)^2 \right] \\
&= -\frac{G m_1 m_j}{2 \left\| r_j^{HC} \right\|^3} \sum_{a=1}^{3} \frac{\partial}{\partial r_{i,c}^{HC}} \left[\left(r_{j,a}^{HC} \right)^2 \right] \\
&= -\frac{G m_1 m_j}{2 \left\| r_j^{HC} \right\|^3} \sum_{a=1}^{3} \left(2 r_{j,a}^{HC} \frac{\partial r_{j,a}^{HC}}{\partial r_{i,c}^{HC}} \right) \\
&= -\frac{G m_1 m_j}{2 \left\| r_j^{HC} \right\|^3} \sum_{a=1}^{3} \left(2 r_{j,a}^{HC} \delta_{ij} \delta_{ac} \right) \\
&= -\frac{G m_1 m_j}{\left\| r_j^{HC} \right\|^3} r_{j,c}^{HC} \delta_{ij}.
\end{aligned}$$

Ensuite,

$$\begin{aligned}
&\frac{\partial}{\partial r_{i,c}^{HC}} \left[F\left(\left\| r_2^{HC} \right\|^2, \left\| r_3^{HC} \right\|^2, ..., \left\| r_n^{HC} \right\|^2 \right) \right] \\
&\quad = \sum_{j=1}^{n} \left(\frac{\partial \left\| r_j^{HC} \right\|^2}{\partial r_{i,c}^{HC}} \frac{\partial}{\partial \left\| r_j^{HC} \right\|^2} \left[F\left(\left\| r_2^{HC} \right\|^2, \left\| r_3^{HC} \right\|^2, ..., \left\| r_n^{HC} \right\|^2 \right) \right] \right) \\
&\quad = \sum_{j=1}^{n} \left(\left[\sum_{a=1}^{3} \frac{\partial}{\partial r_{i,c}^{HC}} \left(\left[r_{j,a}^{HC} \right]^2 \right) \right] \frac{\partial}{\partial \left\| r_j^{HC} \right\|^2} \left[F\left(\left\| r_2^{HC} \right\|^2, \left\| r_3^{HC} \right\|^2, ..., \left\| r_n^{HC} \right\|^2 \right) \right] \right) \\
&\quad = \sum_{j=1}^{n} \left(\left[\sum_{a=1}^{3} 2 r_{j,a}^{HC} \frac{\partial r_{j,a}^{HC}}{\partial r_{i,c}^{HC}} \right] \frac{\partial}{\partial \left\| r_j^{HC} \right\|^2} \left[F\left(\left\| r_2^{HC} \right\|^2, \left\| r_3^{HC} \right\|^2, ..., \left\| r_n^{HC} \right\|^2 \right) \right] \right) \\
&\quad = \sum_{j=1}^{n} \left(2 r_{j,a}^{HC} \left[\sum_{a=1}^{3} \delta_{ij} \delta_{ac} \right] \frac{\partial}{\partial \left\| r_j^{HC} \right\|^2} \left[F\left(\left\| r_2^{HC} \right\|^2, \left\| r_3^{HC} \right\|^2, ..., \left\| r_n^{HC} \right\|^2 \right) \right] \right) \\
&\quad = \sum_{j=1}^{n} \left(2 r_{j,c}^{HC} \delta_{ij} \frac{\partial}{\partial \left\| r_j^{HC} \right\|^2} \left[F\left(\left\| r_2^{HC} \right\|^2, \left\| r_3^{HC} \right\|^2, ..., \left\| r_n^{HC} \right\|^2 \right) \right] \right) \\
&\quad = 2 r_{i,c}^{HC} \frac{\partial}{\partial \left\| r_i^{HC} \right\|^2} \left[F\left(\left\| r_2^{HC} \right\|^2, \left\| r_3^{HC} \right\|^2, ..., \left\| r_n^{HC} \right\|^2 \right) \right].
\end{aligned}$$

Nous pouvons alors utiliser la définition de F donnée à l'équation (4.1.1) et nous trouvons

$$
\begin{aligned}
&\frac{\partial}{\partial \left\|r_i^{HC}\right\|^2}\left[F\left(\left\|r_2^{HC}\right\|^2,\left\|r_3^{HC}\right\|^2,...,\left\|r_n^{HC}\right\|^2\right)\right] \\
&\qquad = \frac{\partial}{\partial \left\|r_i^{HC}\right\|^2}\left[1-\prod_{j=2}^{n}\left[1-\tilde{f}\left(\left\|r_j^{HC}\right\|^2\right)\right]\right] \\
&\qquad = -\frac{\partial}{\partial \left\|r_i^{HC}\right\|^2}\left[\prod_{j=2}^{n}\left[1-\tilde{f}\left(\left\|r_j^{HC}\right\|^2\right)\right]\right] \\
&\qquad = -\frac{\partial}{\partial \left\|r_i^{HC}\right\|^2}\left[1-\tilde{f}\left(\left\|r_i^{HC}\right\|^2\right)\right]\prod_{j=2,j\neq i}^{n}\left[1-\tilde{f}\left(\left\|r_j^{HC}\right\|^2\right)\right] \\
&\qquad = \frac{\partial}{\partial \left\|r_i^{HC}\right\|^2}\left[\tilde{f}\left(\left\|r_i^{HC}\right\|^2\right)\right]\prod_{j=2,j\neq i}^{n}\left[1-\tilde{f}\left(\left\|r_j^{HC}\right\|^2\right)\right] \\
&\qquad = \frac{\partial\left[\tilde{f}\left(\left\|r_i^{HC}\right\|^2\right)\right]}{\partial \left\|r_i^{HC}\right\|^2}\prod_{j=2,j\neq i}^{n}\left[1-\tilde{f}\left(\left\|r_j^{HC}\right\|^2\right)\right], \qquad (4.1.9)
\end{aligned}
$$

où, en utilisant la définition de $\tilde{f}$ donnée à l'équation (4.1.2), nous avons

$$
\begin{aligned}
&\frac{\partial\left[\tilde{f}\left(\left\|r_i^{HC}\right\|^2\right)\right]}{\partial \left\|r_i^{HC}\right\|^2} \\
&\qquad = \begin{cases} -30\frac{\left(R_2^2-\left\|r_i^{HC}\right\|^2\right)^2}{\left(R_2^2-R_1^2\right)^3}+60\frac{\left(R_2^2-\left\|r_i^{HC}\right\|^2\right)^3}{\left(R_2^2-R_1^2\right)^4}-30\frac{\left(R_2^2-\left\|r_i^{HC}\right\|^2\right)^4}{\left(R_2^2-R_1^2\right)^5} & \text{si } R_1^2<\left\|r_i^{HC}\right\|^2<R_2^2 \\ 0 & \text{sinon.}\end{cases}
\end{aligned}
$$

En utilisant tous ces résultats, la dérivée (4.1.8) devient donc :

$$
\begin{aligned}
-\frac{\partial\tilde{\mathcal{H}}_{Kep}^{HC}}{\partial r_{i,c}^{HC}} &= -\sum_{j=2}^{n}\left[\frac{Gm_1m_j}{\left\|r_j^{HC}\right\|^3}r_{j,c}^{HC}\delta_{ij}\right] \\
&\quad -\frac{1}{2m_1}\left\|\sum_{j=2}^{n}p_j^{HC}\right\|^2 2r_{i,c}^{HC}\frac{\partial\left[\tilde{f}\left(\left\|r_i^{HC}\right\|^2\right)\right]}{\partial\left\|r_i^{HC}\right\|^2}\prod_{j=2,j\neq i}^{n}\left[1-\tilde{f}\left(\left\|r_j^{HC}\right\|^2\right)\right] \\
&= -\frac{Gm_1m_i}{\left\|r_i^{HC}\right\|^3}r_{i,c}^{HC}-\frac{r_{i,c}^{HC}}{m_1}\left\|\sum_{j=2}^{n}p_j^{HC}\right\|^2\frac{\partial\left[\tilde{f}\left(\left\|r_i^{HC}\right\|^2\right)\right]}{\partial\left\|r_i^{HC}\right\|^2}\prod_{j=2,j\neq i}^{n}\left[1-\tilde{f}\left(\left\|r_j^{HC}\right\|^2\right)\right]. \qquad (4.1.10)
\end{aligned}
$$

Les équations d'évolution (4.1.4) deviennent donc, en utilisant les dérivées (4.1.7) et (4.1.10),

$$
\begin{cases}
\frac{dr_{i,c}^{HC}}{dt} = \frac{p_{i,c}^{HC}}{m_i}+\frac{1}{m_1}\left[\sum_{j=2}^{n}p_{j,c}^{HC}\right]F\left(\left\|r_2^{HC}\right\|^2,\left\|r_3^{HC}\right\|^2,...,\left\|r_n^{HC}\right\|^2\right) & \\
\frac{dp_{i,c}^{HC}}{dt} = -\frac{Gm_1m_i}{\left\|r_i^{HC}\right\|^3}r_{i,c}^{HC}-\frac{r_{i,c}^{HC}}{m_1}\left\|\sum_{j=2}^{n}p_j^{HC}\right\|^2\frac{\partial\left[\tilde{f}\left(\left\|r_i^{HC}\right\|^2\right)\right]}{\partial\left\|r_i^{HC}\right\|^2}\prod_{j=2,j\neq i}^{n}\left[1-\tilde{f}\left(\left\|r_j^{HC}\right\|^2\right)\right] & \begin{matrix} i=2...n, \\ c=1,2,3.\end{matrix}
\end{cases}
$$

En utilisant la définition de p_i^{HC} détaillée à l'équation (2.2.23), ces équations d'évolution se

réécrivent

$$\begin{cases} \frac{dr_{i,c}^{HC}}{dt} & = \frac{m_i v_{i,c}^B}{m_i} + \frac{1}{m_1}\left[\sum_{j=2}^{n} m_j v_{j,c}^B\right] F\left(\left\|r_2^{HC}\right\|^2, \left\|r_3^{HC}\right\|^2, ..., \left\|r_n^{HC}\right\|^2\right) \\ \frac{d\left(m_i v_{i,c}^B\right)}{dt} & = -\frac{Gm_1 m_i}{\left\|r_i^{HC}\right\|^3} r_{i,c}^{HC} - \frac{r_{i,c}^{HC}}{m_1}\left\|\sum_{j=2}^{n} m_i v_{i,c}^B\right\|^2 \frac{\partial\left[\tilde{f}\left(\left\|r_i^{HC}\right\|^2\right)\right]}{\partial\left\|r_i^{HC}\right\|^2} \prod_{j=2, j\neq i}^{n}\left[1-\tilde{f}\left(\left\|r_j^{HC}\right\|^2\right)\right] \end{cases} \quad \begin{matrix} i = 2...n, \\ c = 1,2,3, \end{matrix}$$

qui se simplifient en

$$\begin{cases} \frac{dr_{i,c}^{HC}}{dt} & = v_{i,c}^B + \frac{1}{m_1}\left[\sum_{j=2}^{n} m_j v_{j,c}^B\right] F\left(\left\|r_2^{HC}\right\|^2, \left\|r_3^{HC}\right\|^2, ..., \left\|r_n^{HC}\right\|^2\right) \\ \frac{dv_{i,c}^B}{dt} & = -\frac{Gm_1}{\left\|r_i^{HC}\right\|^3} r_{i,c}^{HC} - \frac{r_{i,c}^{HC}}{m_1 m_i}\left\|\sum_{j=2}^{n} m_i v_{i,c}^B\right\|^2 \frac{\partial\left[\tilde{f}\left(\left\|r_i^{HC}\right\|^2\right)\right]}{\partial\left\|r_i^{HC}\right\|^2} \prod_{j=2, j\neq i}^{n}\left[1-\tilde{f}\left(\left\|r_j^{HC}\right\|^2\right)\right] \end{cases} \quad \begin{matrix} i = 2...n, \\ c = 1,2,3. \end{matrix} \tag{4.1.11}$$

Cependant, comme le précise [14], ces équations ne peuvent être résolues analytiquement. Elles seront donc résolues numériquement au moyen d'un intégrateur numérique utilisant les formules de Bulirsch-Stoer ([37]).

Notons que le recours à un intégrateur non symplectique dans ce cas très précis ne détruira pas la symplecticité de l'intégration. En effet, comme l'a montré J. E. Chambers dans [9], si l'erreur tolérée lors de l'intégration d'une partie de l'hamiltonien est suffisamment stricte, l'intégrateur global conservera sa symplecticité. En particulier, nous utilisons une erreur tolérée de 10^{-14} dans notre algorithme. Par ailleurs, l'algorithme de Bulirsch-Stoer sera détaillé à la fin de cette section. Nous allons à présent déterminer les équations d'évolution pour $\tilde{\mathcal{H}}_{Jump}^{HC}$.

Evolution sous $\tilde{\mathcal{H}}_{Jump}^{HC}$

Comme précédemment, nous utiliser les équations d'Hamilton (2.2.2) afin de déterminer les équations d'évolution sous $\tilde{\mathcal{H}}_{Jump}^{HC}$. Nous avons alors :

$$\begin{cases} \frac{dr_{i,c}^{HC}}{dt} & = \frac{\partial \tilde{\mathcal{H}}_{Jump}^{HC}}{\partial p_{i,c}^{HC}} \\ \frac{dp_{i,c}^{HC}}{dt} & = -\frac{\partial \tilde{\mathcal{H}}_{Jump}^{HC}}{\partial r_{i,c}^{HC}} \end{cases} \quad i = 1...n, \quad c = 1,2,3. \tag{4.1.12}$$

Comme pour $\tilde{\mathcal{H}}_{Kep}^{HC}$, $\tilde{\mathcal{H}}_{Jump}^{HC}$ est indépendant de r_1^{HC} et p_1^{HC}. En utilisant i = 1 dans l'équation précédente, nous trouvons donc

$$\begin{cases} \frac{dr_{1,c}^{HC}}{dt} & = 0 \\ \frac{dp_{1,c}^{HC}}{dt} & = 0 \end{cases} \quad c = 1,2,3$$

et r_1^{HC} et p_1^{HC} sont donc invariants sous $\tilde{\mathcal{H}}_{Jump}^{HC}$. Les équations d'évolution (4.1.12) deviennent alors

$$\begin{cases} \frac{dr_{i,c}^{HC}}{dt} & = \frac{\partial \tilde{\mathcal{H}}_{Jump}^{HC}}{\partial p_{i,c}^{HC}} \\ \frac{dp_{i,c}^{HC}}{dt} & = -\frac{\partial \tilde{\mathcal{H}}_{Jump}^{HC}}{\partial r_{i,c}^{HC}} \end{cases} \quad i = 2...n, \quad c = 1,2,3. \tag{4.1.13}$$

Nous devons donc calculer $\frac{\partial \tilde{\mathcal{H}}_{Jump}^{HC}}{\partial p_{i,c}^{HC}}$ et $-\frac{\partial \tilde{\mathcal{H}}_{Jump}^{HC}}{\partial r_{i,c}^{HC}}$. Commençons par le premier cité :

$$\frac{\partial \tilde{\mathcal{H}}_{Jump}^{HC}}{\partial p_{i,c}^{HC}} = \frac{\partial}{\partial p_{i,c}^{HC}}\left(\frac{1}{2m_1}\left\|\sum_{j=2}^{n} p_j^{HC}\right\|^2 \left[1 - F\left(\left\|r_2^{HC}\right\|^2, \left\|r_3^{HC}\right\|^2, ..., \left\|r_n^{HC}\right\|^2\right)\right]\right)$$

$$= \frac{\partial}{\partial p_{i,c}^{HC}} \left(\frac{1}{2m_1} \left\| \sum_{j=2}^{n} p_j^{HC} \right\|^2 \right) \left[1 - F\left(\left\| r_2^{HC} \right\|^2, \left\| r_3^{HC} \right\|^2, ..., \left\| r_n^{HC} \right\|^2 \right) \right].$$

La dérivée présente dans cette dernière égalité correspond à la dérivée $\frac{\partial \mathcal{H}_{Jump}^{HC}}{\partial p_{i,c}^{HC}}$ calculée à la section 2.2.6 et rappelée à l'équation (4.1.6). En utilisant cette expression, l'égalité précédente se réécrit

$$\frac{\partial \tilde{\mathcal{H}}_{Jump}^{HC}}{\partial p_{i,c}^{HC}} = \frac{1}{m_1} \left[\sum_{j=2}^{n} p_{j,c}^{HC} \right] \left[1 - F\left(\left\| r_2^{HC} \right\|^2, \left\| r_3^{HC} \right\|^2, ..., \left\| r_n^{HC} \right\|^2 \right) \right]. \tag{4.1.14}$$

Calculons à présent la deuxième dérivée de l'équation (4.1.13) :

$$\begin{aligned} -\frac{\partial \tilde{\mathcal{H}}_{Jump}^{HC}}{\partial r_{i,c}^{HC}} &= -\frac{\partial}{\partial r_{i,c}^{HC}} \left(\frac{1}{2m_1} \left\| \sum_{j=2}^{n} p_j^{HC} \right\|^2 \left[1 - F\left(\left\| r_2^{HC} \right\|^2, \left\| r_3^{HC} \right\|^2, ..., \left\| r_n^{HC} \right\|^2 \right) \right] \right) \\ &= -\frac{1}{2m_1} \left\| \sum_{j=2}^{n} p_j^{HC} \right\|^2 \frac{\partial}{\partial r_{i,c}^{HC}} \left[1 - F\left(\left\| r_2^{HC} \right\|^2, \left\| r_3^{HC} \right\|^2, ..., \left\| r_n^{HC} \right\|^2 \right) \right] \\ &= \frac{1}{2m_1} \left\| \sum_{j=2}^{n} p_j^{HC} \right\|^2 \frac{\partial}{\partial r_{i,c}^{HC}} \left[F\left(\left\| r_2^{HC} \right\|^2, \left\| r_3^{HC} \right\|^2, ..., \left\| r_n^{HC} \right\|^2 \right) \right]. \end{aligned}$$

Nous pouvons réécrire cette égalité en utilisant l'équation (4.1.9). Nous avons alors

$$\begin{aligned} -\frac{\partial \tilde{\mathcal{H}}_{Jump}^{HC}}{\partial r_{i,c}^{HC}} &= \frac{1}{2m_1} \left\| \sum_{j=2}^{n} p_j^{HC} \right\|^2 2r_{i,c}^{HC} \frac{\partial \left[\tilde{f}\left(\left\| r_i^{HC} \right\|^2 \right) \right]}{\partial \left\| r_i^{HC} \right\|^2} \prod_{j=2, j\neq i}^{n} \left[1 - \tilde{f}\left(\left\| r_j^{HC} \right\|^2 \right) \right] \\ &= \frac{r_{i,c}^{HC}}{m_1} \left\| \sum_{j=2}^{n} p_j^{HC} \right\|^2 \frac{\partial \left[\tilde{f}\left(\left\| r_i^{HC} \right\|^2 \right) \right]}{\partial \left\| r_i^{HC} \right\|^2} \prod_{j=2, j\neq i}^{n} \left[1 - \tilde{f}\left(\left\| r_j^{HC} \right\|^2 \right) \right]. \end{aligned} \tag{4.1.15}$$

En utilisant les équations (4.1.14) et (4.1.15), les équations d'évolution (4.1.13) se réécrivent :

$$\begin{cases} \frac{dr_{i,c}^{HC}}{dt} = \frac{1}{m_1} \left[\sum_{j=2}^{n} p_{j,c}^{HC} \right] \left[1 - F\left(\left\| r_2^{HC} \right\|^2, \left\| r_3^{HC} \right\|^2, ..., \left\| r_n^{HC} \right\|^2 \right) \right] & \\ \frac{dp_{i,c}^{HC}}{dt} = \frac{r_{i,c}^{HC}}{m_1} \left\| \sum_{j=2}^{n} p_j^{HC} \right\|^2 \frac{\partial \left[\tilde{f}\left(\left\| r_i^{HC} \right\|^2 \right) \right]}{\partial \left\| r_i^{HC} \right\|^2} \prod_{j=2, j\neq i}^{n} \left[1 - \tilde{f}\left(\left\| r_j^{HC} \right\|^2 \right) \right] & \begin{matrix} i = 2...n, \\ c = 1, 2, 3. \end{matrix} \end{cases}$$

En utilisant la définition des moments dans le repère héliocentrique canonique explicitée à l'équation (2.2.23), les équations d'évolution sous $\tilde{\mathcal{H}}_{Jump}^{HC}$ se réécrivent en terme de vitesses :

$$\begin{cases} \frac{dr_{i,c}^{HC}}{dt} = \frac{1}{m_1} \left[\sum_{j=2}^{n} m_j v_{j,c}^{B} \right] \left[1 - F\left(\left\| r_2^{HC} \right\|^2, \left\| r_3^{HC} \right\|^2, ..., \left\| r_n^{HC} \right\|^2 \right) \right] & \\ \frac{d\left(m_i v_{i,c}^{B} \right)}{dt} = \frac{r_{i,c}^{HC}}{m_1} \left\| \sum_{j=2}^{n} m_j v_j^{B} \right\|^2 \frac{\partial \left[\tilde{f}\left(\left\| r_i^{HC} \right\|^2 \right) \right]}{\partial \left\| r_i^{HC} \right\|^2} \prod_{j=2, j\neq i}^{n} \left[1 - \tilde{f}\left(\left\| r_j^{HC} \right\|^2 \right) \right] & \begin{matrix} i = 2...n, \\ c = 1, 2, 3. \end{matrix} \end{cases}$$

Ces dernières équations se réécrivent

$$\begin{cases} \frac{dr_{i,c}^{HC}}{dt} = \frac{1}{m_1} \left[\sum_{j=2}^{n} m_j v_{j,c}^{B} \right] \left[1 - F\left(\left\| r_2^{HC} \right\|^2, \left\| r_3^{HC} \right\|^2, ..., \left\| r_n^{HC} \right\|^2 \right) \right] & \\ \frac{dv_{i,c}^{B}}{dt} = \frac{r_{i,c}^{HC}}{m_1 m_i} \left\| \sum_{j=2}^{n} m_j v_j^{B} \right\|^2 \frac{\partial \left[\tilde{f}\left(\left\| r_i^{HC} \right\|^2 \right) \right]}{\partial \left\| r_i^{HC} \right\|^2} \prod_{j=2, j\neq i}^{n} \left[1 - \tilde{f}\left(\left\| r_j^{HC} \right\|^2 \right) \right] & \begin{matrix} i = 2...n, \\ c = 1, 2, 3. \end{matrix} \end{cases} \tag{4.1.16}$$

Comme pour l'évolution sous $\tilde{\mathcal{H}}_{Kep}^{HC}$, ces équations d'évolution ne peuvent être résolues analytiquement comme le précisent [14] et [10]. Celles-ci seront donc également résolues en utilisant les formules de Bulirsch-Stoer. Nous allons à présent décrire cette technique d'intégration.

Technique d'intégration de Bulirsch-Stoer

Cette technique est décrite dans [37]. Elle consiste en un intégrateur numérique amélioré par une extrapolation polynomiale. Le but est donc de résoudre un système d'équations de la forme

$$\begin{cases} \frac{dx(t)}{dt} &= f(x(t),t) \\ x(t_0) &= x_0, \end{cases}$$

avec, dans notre cas, $x = (r,p)^T$, où $r = (r_1, r_2 ... r_n)^T$ sont les positions des corps et $p = (p_1, p_2 ... p_n)^T$ les moments associés, $f(x(t),t) = \left(\frac{\partial \mathcal{H}}{\partial p}, -\frac{\partial \mathcal{H}}{\partial r}\right)^T$, t_0 le temps initial et $x_0 = (r(0), p(0))^T$ les paramètres initiaux des corps. Le but de la technique de Bulirsch-Stoer est de trouver aussi précisément que possible la valeur de $x(t_0 + \tau)$.

Commençons par présenter l'intégrateur numérique. Tout d'abord, nous choisissons un pas d'intégration h comme étant une fraction de τ

$$h = \frac{\tau}{\tilde{N}}.$$

Comme expliqué à la section 2.2.2, nous allons faire évoluer le système pas à pas. En d'autres termes, nous allons approximer les solutions $x(t_0 + h), x(t_0 + 2h), x(t_0 + 3h), ..., x(t_0 + \tilde{N}h)$ sur base des solutions calculées précédemment. Cette idée s'exprime mathématiquement, en utilisant la notation x_k pour l'approximation de $x(t_0 + kh)$, comme

$$\begin{cases} x_1 &= \Phi_1(x_0) \\ x_2 &= \Phi_2(x_0, x_1) \\ x_3 &= \Phi_3(x_0, x_1, x_2) \\ &\vdots \\ x_{\tilde{N}} &= \Phi_{\tilde{N}}(x_0, x_1, x_2 ... x_{\tilde{N}-1}). \end{cases} \tag{4.1.17}$$

L'intégration numérique du système consiste donc à déterminer les fonctions d'évolution Φ_i, $i \in \{1 ... \tilde{N}\}$. Notons que la théorie de Bulirsch-Stoer pourrait être appliquée avec n'importe quel intégrateur numérique. Dans notre cas nous utilisons l'intégrateur du point-milieu modifié.

Le système (4.1.17) s'exprime, pour la technique du point-milieu décrite dans [37], comme

$$\begin{cases} x_1 &= x_0 + hf(x_0, t_0) \\ x_{m+1} &= x_{m-1} + 2hf(x_m, t_0 + mh) \quad m \in \{1 ... \tilde{N} - 1\}. \end{cases} \tag{4.1.18}$$

La différence entre la méthode du point milieu et sa version modifiée se situe dans l'approximation finale de $x(t_0 + \tau)$. En effet, dans la version modifiée, au lieu d'approximer $x(t_0 + \tau)$ par $x_{\tilde{N}}$, nous raffinons cette approximation par la formule :

$$x(t_0 + \tau) \approx \tilde{x}_{\tilde{N}} \frac{1}{2} \left[x_{\tilde{N}} + x_{\tilde{N}-1} + hf(x_n, t_0 + \tau) \right]. \tag{4.1.19}$$

Comme le précise [37], le gros avantage de cette technique réside dans la forme de son erreur. En effet, celle-ci est composée de puissances paires du pas d'intégration. Mathématiquement, cela s'exprime par une erreur de la forme

$$erreur = \sum_{i=1}^{\infty} a_i h^{2i}.$$

Cette propriété permettra une meilleure extrapolation par la suite.

La technique de Bulirsch-Stoer consiste, dans un premier temps, à appliquer la méthode du point milieu modifiée avec un ensemble de pas. Nous obtenons alors avec une série de pas d'intégration : $h_1, h_2, ..., h_l$ et une série d'approximations : $\hat{x}_{h_1}, \hat{x}_{h_2}, ..., \hat{x}_{h_l}$ où $\hat{x}_h$ est l'approximation de $x(t_0 + \tau)$ trouvée au moyen de la méthode du point milieu modifiée avec un pas h. De plus, comme l'erreur de la technique du point milieu modifiée ne contient que des puissances d'ordre pair du pas, chaque solution $\hat{x}_{h_i}$ peut être considérée comme étant dépendante du pas au carré. Nous pouvons alors considérer les points $((h_1)^2, \hat{x}_{h_1}), ((h_2)^2, \hat{x}_{h_2}), ..., ((h_l)^2, \hat{x}_{h_l})$.

Nous effectuons, dans un deuxième temps, une extrapolation en $h^2 = 0$ sur base de ces points afin de trouver une bonne approximation de $x(t_0 + \tau)$. Cette extrapolation en $h^2 = 0$ est appelée extrapolation de Richardson ([40]). Dans notre cas, l'extrapolation utilisée sera une extrapolation polynomiale déterminée au moyen de l'algorithme de Neuville. Cette technique est décrite dans [37] pour un point quelconque. Ces formules ont donc été adaptées afin d'effectuer l'extrapolation en $h^2 = 0$.

Nous allons à présent expliciter les formules de Neuville pour $h^2 = 0$. Pour ce faire, nous introduisons une nouvelle notation, $\bar{x}_{i,i+1,...,i+m}$, qui est la valeur de l'extrapolation en utilisant les points $((h_i)^2, \hat{x}_{h_i})$, $((h_{i+1})^2, \hat{x}_{h_{i+1}})$, ..., $((h_{i+m})^2, \hat{x}_{h_{i+m}})$. L'extrapolation sera alors calculée récursivement par

$$\begin{cases} \tilde{e}_{1,i} & = \hat{x}_{h_i} \\ \bar{e}_{1,i} & = \hat{x}_{h_i} \\ \bar{x}_i & = \hat{x}_{h_i} \\ \tilde{e}_{k+1,i} & = \frac{(h_i)^2[\tilde{e}_{k,i+1} - \bar{e}_{k,i}]}{(h_i)^2 - (h_{i+k+1})^2} \qquad i \in \{j, j+1...j+\tilde{l}\}, \quad k \in \{1, ..., l-1\}. \\ \bar{e}_{k+1,i} & = \frac{(h_{i+k+1})^2[\tilde{e}_{k,i+1} - \bar{e}_{k,i}]}{(h_i)^2 - (h_{i+k+1})^2} \\ \bar{x}_{i,i+1,...,i+k} & = \bar{x}_{i+1,i+2,...,i+k} + \bar{e}_{k,i} \end{cases} \tag{4.1.20}$$

Par ailleurs, la suite de pas utilisée dans notre algorithme sera $h_i = \frac{\tau}{2^i}$. De plus, le processus de Bulirsch-Stoer sera effectué avec un maximum de 7 points et le pas minimal utilisé sera h_{11}. Le programme va donc effectuer le processus de Bulirsch-Stoer avec, dans l'ordre, les ensembles de pas suivants : $\{h_1\}$, $\{h_1, h_2\}$, ... $\{h_1, h_2, ..., h_7\}$, $\{h_2, h_3, ..., h_8\}$,... $\{h_5, h_6, ..., h_{11}\}$. Il s'arrêtera au premier ensemble de pas donnant une erreur relative sur $x(t_0 + \tau)$ inférieure à 10^{-14}. Si aucun de ses ensembles ne permet d'atteindre cette erreur, le processus sera répété avec une réduction préalable du pas τ.

Notons également que l'estimation de l'erreur relative sera déterminée en divisant $\bar{e}_{k,i}$ par l'ordre de grandeur de $\bar{x}_{i+1,i+2,...,i+k}$. En effet, $\bar{e}_{k,i}$ étant la mise à jour du polynôme d'interpolation, il est un bon candidat pour estimer l'erreur.

L'explication de la technique de Bulirsch-Stoer termine notre analyse des modifications apportées à l'algorithme Helio afin de tenir compte des rencontres proches avec le corps central du système. Nous allons à présent nous intéresser aux modifications apporter à SyMBA pour tenir compte de ces rencontres proches.

4.1.2 Algorithme avec rencontres proches entre les planètes

Nous allons donc à présent expliciter les modifications apportées à l'algorithme décrit à la section 2.2.7 afin de tenir compte des rencontres proches avec le corps central. Dans le cadre de SyMBA, nous avions le découpage de l'hamiltonien :

$$\mathcal{H} = \mathcal{H}^{HC}_{Jump} + \mathcal{H}^{HC}_{IntNR} + (\mathcal{H}^{HC}_{IntR} + \mathcal{H}^{HC}_{Kep}), \tag{4.1.21}$$

avec

$$\mathcal{H}_{IntR}^{HC} = -\sum_{i=2}^{n-1} \sum_{i<j\leqslant n} \left(\frac{Gm_i m_j}{\|r_{ij}\|} H(\|r_{ij}\|) \right)$$

et

$$\mathcal{H}_{IntNR}^{HC} = -\sum_{i=2}^{n-1} \sum_{i<j\leqslant n} \left(\frac{Gm_i m_j}{\|r_{ij}\|} (1 - H(\|r_{ij}\|)) \right),$$

où

$$H(r) = \begin{cases} 0 & \text{si } r > r_{crit} \\ 1 & \text{sinon.} \end{cases}$$

En cas de rencontre proche entre une planète et le corps central, nous allons injecter $\mathcal{H}_{Jump}^{HC}$ dans $\mathcal{H}_{Kep}^{HC}$ de la même manière que pour les modifications apportées à l'algorithme Helio. En pratique, nous allons donc remplacer $\mathcal{H}_{Jump}^{HC}$ et $\mathcal{H}_{Kep}^{HC}$ dans (4.1.21) par $\tilde{\mathcal{H}}_{Jump}^{HC}$ et $\tilde{\mathcal{H}}_{Kep}^{HC}$. Nous obtiendrons alors le découpage

$$\mathcal{H} = \tilde{\mathcal{H}}_{Jump}^{HC} + \mathcal{H}_{IntNR}^{HC} + (\mathcal{H}_{IntR}^{HC} + \tilde{\mathcal{H}}_{Kep}^{HC}).$$

L'évolution sera alors effectuée en appliquant la formule (2.2.13) et nous trouvons

$$f(t_{k+1}) = \exp\left(\frac{\tau}{2}\tilde{T}_{Jump}^{HC}\right) \exp\left(\frac{\tau}{2}T_{IntNR}^{HC}\right) \exp\left(\tau\left[\tilde{T}_{Kep}^{HC} + T_{IntR}^{HC}\right]\right) \exp\left(\frac{\tau}{2}T_{IntNR}^{HC}\right) \exp\left(\frac{\tau}{2}\tilde{T}_{Jump}^{HC}\right) f(t_k), \quad k \in \mathbb{N},$$

où $\tilde{T}_{Jump}^{HC}$, T_{IntNR}^{HC} et $\tilde{T}_{Kep}^{HC} + T_{IntR}^{HC}$ sont les opérateurs linéaires associés à $\tilde{\mathcal{H}}_{Jump}^{HC}$, $\mathcal{H}_{IntNR}^{HC}$ et $\tilde{\mathcal{H}}_{Kep}^{HC} + \mathcal{H}_{IntR}^{HC}$ respectivement. Cette équation revient donc faire évoluer le système d'un demi-pas suivant $\tilde{\mathcal{H}}_{Jump}^{HC}$, ensuite d'un demi-pas suivant $\mathcal{H}_{IntNR}^{HC}$, suivi d'une évolution d'un pas complet suivant $\tilde{\mathcal{H}}_{Kep}^{HC} + \mathcal{H}_{IntR}^{HC}$, puis à nouveau d'un demi-pas suivant $\mathcal{H}_{IntNR}^{HC}$ et enfin d'un demi-pas suivant $\tilde{\mathcal{H}}_{Jump}^{HC}$.

L'évolution sous $\tilde{\mathcal{H}}_{Jump}^{HC}$ a été étudiée dans la section 4.1.1 et celle sous $\mathcal{H}_{IntNR}^{HC}$ l'a été à la section 2.2.7. Enfin l'évolution sous $\tilde{\mathcal{H}}_{Kep}^{HC} + \mathcal{H}_{IntR}^{HC}$ est effectuée suivant le même procédé que celui explicité à la section 2.2.7.

Evolution sous $\tilde{\mathcal{H}}_{Kep}^{HC} + \mathcal{H}_{IntR}^{HC}$

Sous $\tilde{\mathcal{H}}_{Kep}^{HC} + \mathcal{H}_{IntR}^{HC}$, l'évolution est, de manière analogue à l'évolution présentée à la section 2.2.7,

$$f(t_{k+1}) = \exp\left(\tau\left[T_{Kep}^{HC} + \sum_{i=0}^{\infty} T_{IntR,i}^{HC}\right]\right) f(t_k)$$

$$\approx \exp\left(\frac{\tau}{2}T_{IntR,0}^{HC}\right) \exp\left(\tau\left[\tilde{T}_{Kep}^{HC} + \sum_{i=1}^{\infty} T_{IntR,i}^{HC}\right]\right) \exp\left(\frac{\tau}{2}T_{IntR,0}^{HC}\right) f(t_k)$$

$$\approx \exp\left(\frac{\tau}{2}T_{IntR,0}^{HC}\right) \left\{ \exp\left(\frac{\tau}{2n}T_{IntR,1}^{HC}\right) \exp\left(\frac{\tau}{n}\left[\tilde{T}_{Kep}^{HC} + \sum_{i=2}^{\infty} T_{IntR,i}^{HC}\right]\right) \exp\left(\frac{\tau}{2n}T_{IntR,1}^{HC}\right) \right\}^n \exp\left(\frac{\tau}{2}T_{IntR,0}^{HC}\right) f(t_k)$$

$$\approx \exp\left(\frac{\tau}{2}T^{HC}_{IntR,0}\right)\left\{\exp\left(\frac{\tau}{2n}T^{HC}_{IntR,1}\right)\left\{\exp\left(\frac{\tau}{2n^2}T^{HC}_{IntR,2}\right)\exp\left(\frac{\tau}{n^2}\left[\tilde{T}^{HC}_{Kep}+\sum_{i=3}^{\infty}T^{HC}_{IntR,i}\right]\right)\right.\right.$$
$$\left.\left.\exp\left(\frac{\tau}{2n^2}T^{HC}_{IntR,2}\right)\right\}^n\exp\left(\frac{\tau}{2n}T^{HC}_{IntR,1}\right)\right\}^n\exp\left(\frac{\tau}{2}T^{HC}_{IntR,0}\right)f(t_k)$$
$$\vdots$$

où $k \in \mathbb{N}$, $\tilde{T}^{HC}_{Kep}$ et $T^{HC}_{IntR,i}$ sont les opérateurs linéaires associés à $\tilde{\mathcal{H}}^{HC}_{Kep}$ et $\mathcal{H}^{HC}_{IntR,i}$ respectivement, avec $\mathcal{H}^{HC}_{IntR} = \sum_{i=0}^{\infty}\mathcal{H}^{HC}_{IntR,i}$.

Comme pour l'algorithme SyMBA, une réduction successive du pas sera donc appliquée pour les différents ordres de grandeur des termes de $\mathcal{H}^{HC}_{IntR}$. L'intégration sera donc analogue à celle décrite à la section 2.2.7. L'unique différence réside dans l'utilisation de $\tilde{\mathcal{H}}^{HC}_{Kep}$ en lieu et place de $\mathcal{H}^{HC}_{Kep}$.

4.1.3 Code

Nous allons à présent détailler les différentes fonctions qui sont développées dans SyMBA sur base de la théorie explicitée aux sections 4.1.1 et 4.1.2.

- *helioq_bs.f* : effectue la technique d'intégration de Bulirsch-Stoer expliquée à la section 4.1.1 pour les systèmes (4.1.16) et (4.1.11). Si l'entrée *iwhich* est égale à 1, la fonction résout le système (4.1.16) si *iwhich* est égale à 2 c'est le système (4.1.11) qui est résolu.

- *helioq_bs_der1.f* : calcule le membre de droite du système (4.1.16). Cette fonction calcule donc la variation des positions héliocentriques canoniques et des vitesses barycentriques sous l'action de $\tilde{\mathcal{H}}^{HC}_{Jump}$.

- *helioq_bs_der2.f* : calcule le membre de droite du système (4.1.11). Cette fonction calcule donc la variation des positions héliocentriques canoniques et des vitesses barycentriques sous l'action de $\tilde{\mathcal{H}}^{HC}_{Kep}$.

- *helioq_chk_sun.f* : détermine si un corps a une rencontre proche avec le corps central.

- *helioq_extr.f* : calcule l'extrapolation de Richardson au moyen de l'algorithme de Neuville. Cette fonction consiste en le système (4.1.20).

- *helioq_getf.f* : calcule $F\left(\left\|r^{HC}_2\right\|^2, \left\|r^{HC}_3\right\|^2, ..., \left\|r^{HC}_n\right\|^2\right)$, tel qu'exprimé à l'équation (4.1.1), et $\frac{\partial\left[\tilde{f}\left(\left\|r^{HC}_i\right\|^2\right)\right]}{\partial\left\|r^{HC}_i\right\|^2}\prod_{j=2,j\neq i}^{n}\left[1-\tilde{f}\left(\left\|r^{HC}_j\right\|^2\right)\right]$.

- *helioq_mmid.f* : applique l'intégration du point milieu modifiée explicitée aux équations (4.1.18) et (4.1.19).

- *helioq_setbs.f* : effectue l'évolution sous $\tilde{\mathcal{H}}^{HC}_{Jump}$ si l'input *iwhich* vaut 1 et sous $\tilde{\mathcal{H}}^{HC}_{Kep}$ si le même input vaut 2. Cette fonction va donc appeler successivement *helioq_bs.f* si l'algorithme de Bulirsch-Stoer a nécessité une réduction du pas. Dans le contraire, il ne va l'appeler qu'une seule fois.

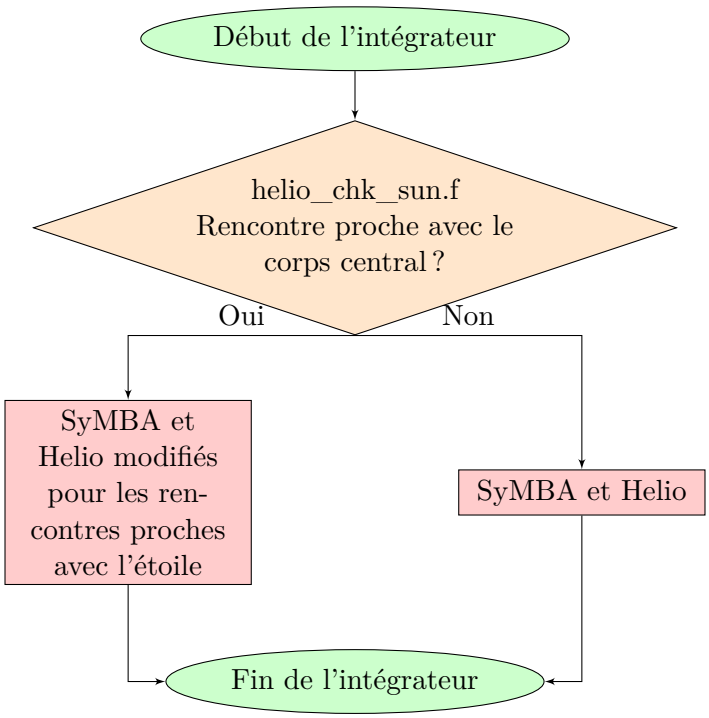
- *symba5q_step_helio.f* : calcule la modification de l'algorithme Helio en cas de rencontre proche entre un corps et le corps central.

- *symba5q_step_interp.f* : effectue l'intégration en utilisant l'algorithme SyMBA modifié pour tenir compte des rencontres proches entre un corps et le corps central.

- *symba5q_step_pl.f* : choisit quel intégrateur utiliser entre Helio, SyMBA et leur version modifiée pour les rencontres proches avec le corps central.
- *symba5q_step_recur.f* : effectue l'évolution sous $\tilde{\mathcal{H}}^{HC}_{Kep} + \mathcal{H}^{HC}_{IntR}$.

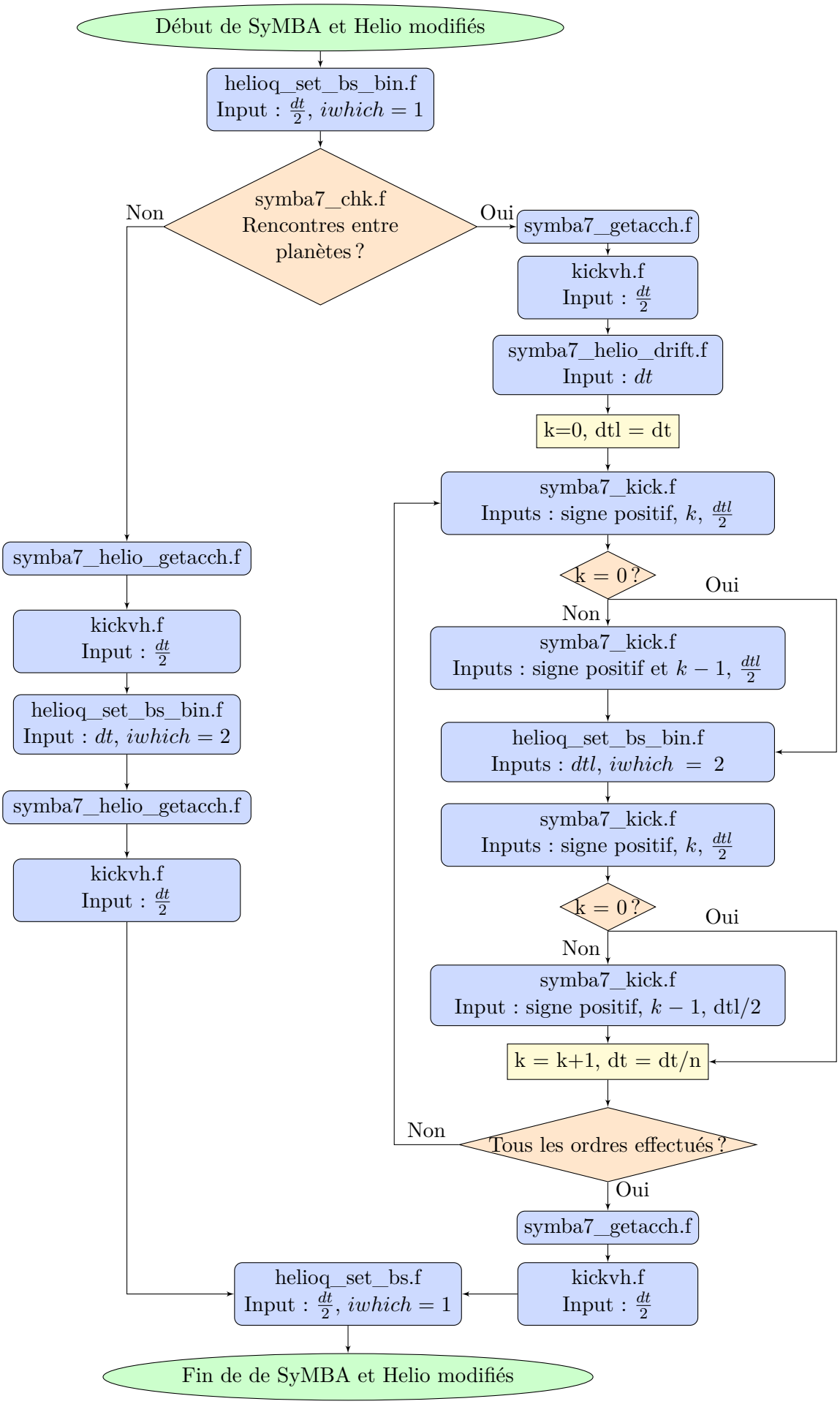
4.1.4 Organigramme

Nous allons à présent réaliser l'organigramme de ce nouveau code. Celui-ci est identique à celui de la section 2.4, sauf pour le diagramme "Intégrateur", dont nous détaillons la nouvelle version ci-dessous.



La partie "SyMBA et Helio" correspond au diagramme "Intégrateur" de la section 2.4. Nous allons à présent détailler le nouveau diagramme "SyMBA et Helio modifiés pour les rencontres proches avec l'étoile".

SyMBA et Helio modifiés pour les rencontres proches avec l'étoile



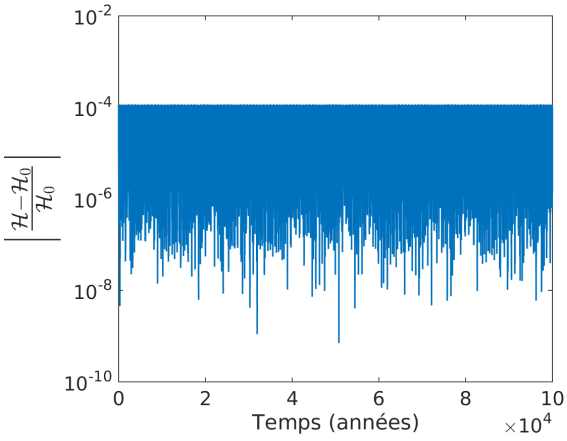


FIGURE 4.4 – Conservation de l'hamiltonien pour le système de la table 4.1 sans le corps 2 avec l'intégrateur SyMBA modifié pour les rencontres proches avec le corps central.

4.1.5 Validation

Le code explicité ci-dessus a été validé dans l'article [14]. Nous constatons effectivement qu'il permet bien de résoudre le problème de conservation de l'hamiltonien comme l'illustre la figure 4.4. Cette figure a été réalisée au moyen de la nouvelle version du code SyMBA, pour les rencontres proches avec le corps central, appliquée au système de la table 4.1 sans le compagnon stellaire avec un pas de 10^{-2} ans. La conservation est clairement meilleure que celle obtenu avec SyMBA à la figure 4.3.

4.2 Systèmes binaires

Comme nous l'avons vu précédemment, le problème qui survient en cas de rencontres proches entre une planète et le corps central pour SyMBA survient également dans sa modification aux systèmes binaires. Nous allons donc adapter les changements explicités pour Helio et SyMBA à la section précédente à nos algorithmes pour les systèmes binaires. Commençons par traiter le cas sans rencontres proches entre les planètes.

4.2.1 Algorithme sans rencontres proches entre les planètes

Pour rappel, la décomposition de l'hamiltonien est la suivante :

$$\mathcal{H} = \mathcal{H}^{BE}_{Kep} + \mathcal{H}^{BE}_{Int} + \mathcal{H}^{BE}_{Jump},$$

avec

$$\mathcal{H}^{BE}_{Kep} = \frac{\left\|p^{BE}_B\right\|^2}{2\mu_{BE}} - \frac{Gm_{sys}\mu_{BE}}{\left\|r^{BE}_B\right\|} + \sum_{i=3}^{n}\left(\frac{\left\|p^{BE}_i\right\|^2}{2m_i} - \frac{Gm_Am_B}{\left\|r^{BE}_i\right\|}\right),$$

$$\begin{aligned}\mathcal{H}^{BE}_{int} = &-\sum_{i=3}^{n-1}\sum_{i<j\leqslant n}\frac{Gm_im_j}{\|r_{ij}\|} + Gm_Am_B\left(\frac{1}{\left\|r^{BE}_B\right\|} - \frac{1}{\left\|r^{BE}_B + s\right\|}\right)\\ &+ Gm_B\sum_{i=3}^{n}m_i\left(\frac{1}{\left\|r^{BE}_B\right\|} - \frac{1}{\left\|r^{BE}_B - r^{BE}_i + s\right\|}\right)\end{aligned}$$

et

$$\mathcal{H}^{BE}_{Jump} = \frac{\left\|\sum_{i=3}^{n}p^{BE}_i\right\|^2}{2m_A},$$

où

$$s = \frac{\sum_{i=3}^{n} m_i r_i^{BE}}{m_{red}}, \qquad r_{ij} = r_i^{BE} - r_j^{BE}, \qquad \mu_{BE} = \frac{m_B m_{red}}{m_{sys}},$$

$$m_{red} = m_A + \sum_{i=3}^{n} m_i \qquad \text{et} \qquad m_{sys} = m_A + m_B + \sum_{i=3}^{n} m_i.$$

De plus, comme illustré à la figure 4.3, le problème provient, comme dans les systèmes sans compagnon binaire, de l'augmentation de $\mathcal{H}_{Jump}^{BE}$. Nous allons donc, à nouveau, introduire progressivement $\mathcal{H}_{Jump}^{BE}$ dans $\mathcal{H}_{Kep}^{BE}$ au moyen d'une fonction de transition $F^{BE}\left(\left\|r_3^{BE}\right\|^2, \left\|r_4^{BE}\right\|^2, ..., \left\|r_n^{BE}\right\|^2\right)$ analogue à celle utilisée dans la section 4.1. Nous aurons donc

$$\mathcal{H} = \tilde{\mathcal{H}}_{Kep}^{BE} + \mathcal{H}_{Int}^{BE} + \tilde{\mathcal{H}}_{Jump}^{BE},$$

où

$$\tilde{\mathcal{H}}_{Kep}^{BE} = \frac{\left\|p_B^{BE}\right\|^2}{2\mu_{BE}} - \frac{G m_{sys}\mu_{BE}}{\left\|r_B^{BE}\right\|} + \sum_{i=3}^{n}\left(\frac{\left\|p_i^{BE}\right\|^2}{2m_i} - \frac{G m_A m_B}{\left\|r_i^{BE}\right\|}\right) + \frac{1}{2m_1}\left\|\sum_{i=3}^{n} p_i^{BE}\right\|^2 F^{BE}\left(\left\|r_3^{BE}\right\|^2, \left\|r_4^{BE}\right\|^2, ..., \left\|r_n^{BE}\right\|^2\right)$$

et

$$\tilde{\mathcal{H}}_{Jump}^{BE} = \frac{1}{2m_1}\left\|\sum_{i=3}^{n} p_i^{BE}\right\|^2 \left[1 - F^{BE}\left(\left\|r_3^{BE}\right\|^2, \left\|r_4^{BE}\right\|^2, ..., \left\|r_n^{BE}\right\|^2\right)\right],$$

où

$$F^{BE}\left(\left\|r_3^{BE}\right\|^2, \left\|r_4^{BE}\right\|^2, ..., \left\|r_n^{BE}\right\|^2\right) = 1 - \prod_{i=3}^{n}\left[1 - \tilde{f}^{BE}\left(\left\|r_i^{BE}\right\|^2\right)\right], \tag{4.2.1}$$

avec

$$\tilde{f}^{BE}\left(\left\|r_i^{BE}\right\|^2\right) = \begin{cases} 1 & \text{si } \left\|r_i^{BE}\right\|^2 < R_1^2 \\ 10\left(\frac{R_2^2 - \left\|r_i^{BE}\right\|^2}{R_2^2 - R_1^2}\right)^3 - 15\left(\frac{R_2^2 - \left\|r_i^{BE}\right\|^2}{R_2^2 - R_1^2}\right)^4 + 6\left(\frac{R_2^2 - \left\|r_i^{BE}\right\|^2}{R_2^2 - R_1^2}\right)^5 & \text{si } R_1^2 < \left\|r_i^{BE}\right\|^2 < R_2^2 \\ 0 & \text{si } \left\|r_i^{BE}\right\|^2 > R_2^2. \end{cases}$$

Notons que nous pouvons utiliser des formules analogues à celles de F et $\tilde{f}$ pour F^{BE} et $\tilde{f}^{BE}$ car les positions des planètes dans le repère binaire éloignée sont repérées par rapport à l'étoile centrale du système. Cette définition est donc analogue à celle utilisée dans le repère héliocentrique canonique.

L'évolution se fera alors au moyen de l'équation (2.2.13) et nous aurons alors :

$$f(t_{k+1}) = \exp\left(\frac{\tau}{2}\tilde{T}_{Jump}^{BE}\right)\exp\left(\frac{\tau}{2}T_{Int}^{HC}\right)\exp\left(\tau\tilde{T}_{Kep}^{BE}\right)\exp\left(\frac{\tau}{2}T_{Int}^{BE}\right)\exp\left(\frac{\tau}{2}\tilde{T}_{Jump}^{BE}\right)f(t_k), \qquad k \in \mathbb{N},$$

où $\tilde{T}_{Kep}^{BE}$, T_{Int}^{BE} et $\tilde{T}_{Jump}^{BE}$ sont les opérateurs linéaires associés à $\tilde{\mathcal{H}}_{Kep}^{BE}$, $\mathcal{H}_{Int}^{BE}$ et $\tilde{\mathcal{H}}_{Jump}^{BE}$ respectivement. Nous allons donc, faire évoluer le système d'un demi-pas suivant $\tilde{\mathcal{H}}_{Jump}^{BE}$, ensuite d'un demi-pas suivant $\mathcal{H}_{Int}^{BE}$, suivi d'une évolution d'un pas complet suivant $\tilde{\mathcal{H}}_{Kep}^{BE}$, puis à nouveau d'un demi-pas suivant $\mathcal{H}_{Int}^{BE}$ et enfin d'un demi-pas suivant $\tilde{\mathcal{H}}_{Jump}^{BE}$. L'évolution sous $\mathcal{H}_{Int}^{BE}$ a déjà été traitée à la section 3.2.1. Nous allons donc nous intéresser aux évolutions sous $\tilde{\mathcal{H}}_{Kep}^{BE}$ et sous $\tilde{\mathcal{H}}_{Jump}^{BE}$.

Evolution sous $\tilde{\mathcal{H}}_{Kep}^{BE}$

L'évolution sous $\tilde{\mathcal{H}}_{Kep}^{BE}$ se détermine au moyen des équations d'Hamilton (2.2.2) :

$$\begin{cases} \dfrac{dr_{i,c}^{BE}}{dt} &= \dfrac{\partial \tilde{\mathcal{H}}_{Kep}^{BE}}{\partial p_{i,c}^{BE}} \\ \dfrac{dp_{i,c}^{BE}}{dt} &= -\dfrac{\partial \tilde{\mathcal{H}}_{Kep}^{BE}}{\partial r_{i,c}^{BE}} \end{cases} \qquad i = A, B, 3, ...n, \quad c = 1, 2, 3, \tag{4.2.2}$$

où c correspond aux coordonnées x, y et z.

Par ailleurs, de manière analogue à $\tilde{\mathcal{H}}_{Kep}^{HC}$, $\tilde{\mathcal{H}}_{Kep}^{BE}$ ne dépend ni de $r_A^B E$, ni de p_A^{BE} et nous avons donc

$$\begin{cases} \frac{dr_{A,c}^{BE}}{dt} &= 0 \\ \frac{dp_{A,c}^{BE}}{dt} &= 0 \end{cases} \qquad c = 1, 2, 3.$$

r_1^{BE} et p_1^{BE} sont donc invariants sous $\tilde{\mathcal{H}}_{Kep}^{BE}$. Le système (4.2.2) se réécrit donc

$$\begin{cases} \dfrac{dr_{i,c}^{BE}}{dt} &= \dfrac{\partial \tilde{\mathcal{H}}_{Kep}^{BE}}{\partial p_{i,c}^{BE}} \\ \dfrac{dp_{i,c}^{BE}}{dt} &= -\dfrac{\partial \tilde{\mathcal{H}}_{Kep}^{BE}}{\partial r_{i,c}^{BE}} \end{cases} \qquad i = B, 3...n, \quad c = 1, 2, 3. \tag{4.2.3}$$

De plus, $\tilde{\mathcal{H}}_{Kep}^{BE}$ peut être décomposé en deux parties qui peuvent être résolues séparément. Nous avons alors la décomposition :

$$\tilde{\mathcal{H}}_{Kep}^{BE} = \tilde{\mathcal{H}}_{Kep,1}^{BE} + \tilde{\mathcal{H}}_{Kep,2}^{BE},$$

avec

$$\tilde{\mathcal{H}}_{Kep,1}^{BE} = \frac{\left\| p_B^{BE} \right\|^2}{2\mu_{BE}} - \frac{G m_{sys} \mu_{BE}}{\left\| r_B^{BE} \right\|}$$

et

$$\tilde{\mathcal{H}}_{Kep,2}^{BE} = \sum_{i=3}^{n} \left(\frac{\left\| p_i^{BE} \right\|^2}{2m_i} - \frac{G m_A m_B}{\left\| r_i^{BE} \right\|} \right) + \frac{1}{2m_1} \left\| \sum_{i=3}^{n} p_i^{BE} \right\|^2 F^{BE} \left(\left\| r_3^{BE} \right\|^2, \left\| r_4^{BE} \right\|^2, ..., \left\| r_n^{BE} \right\|^2 \right).$$

Sur base de cette décomposition, nous allons pouvoir faire évoluer séparément le compagnon stellaire et les planètes sous $\tilde{\mathcal{H}}_{Kep}^{BE}$.

En effet, l'évolution du compagnon stellaire sous $\tilde{\mathcal{H}}_{Kep}^{BE}$ est donnée par (4.2.3) avec i = B. Nous avons alors

$$\begin{cases} \dfrac{dr_{B,c}^{BE}}{dt} &= \dfrac{\partial \tilde{\mathcal{H}}_{Kep}^{BE}}{\partial p_{B,c}^{BE}} \\ \dfrac{dp_{B,c}^{BE}}{dt} &= -\dfrac{\partial \tilde{\mathcal{H}}_{Kep}^{BE}}{\partial r_{B,c}^{BE}} \end{cases} \qquad c = 1, 2, 3.$$

De plus, comme $\tilde{\mathcal{H}}_{Kep,2}^{BE}$ ne dépend ni de r_B^{BE}, ni de p_B^{BE}, nous avons

$$\begin{cases} \dfrac{dr_{B,c}^{BE}}{dt} &= \dfrac{\partial \tilde{\mathcal{H}}_{Kep,1}^{BE}}{\partial p_{B,c}^{BE}} \\ \dfrac{dp_{B,c}^{BE}}{dt} &= -\dfrac{\partial \tilde{\mathcal{H}}_{Kep,1}^{BE}}{\partial r_{B,c}^{BE}} \end{cases} \qquad c = 1, 2, 3.$$

L'évolution du compagnon stellaire sous $\tilde{\mathcal{H}}^{BE}_{Kep}$ est donc équivalente à son évolution sous $\tilde{\mathcal{H}}^{BE}_{Kep,1}$. De plus, comme $\tilde{\mathcal{H}}^{BE}_{Kep,1}$ ne dépend que des positions et vitesses du compagnon stellaire, cette évolution peut être calculée indépendamment de l'évolution des autres corps. Par ailleurs, $\tilde{\mathcal{H}}^{BE}_{Kep,1}$ est égal à $\mathcal{H}^{BE}_{Kep,B}$ détaillé à la section 3.2.1. Les équations d'évolution sous $\tilde{\mathcal{H}}^{BE}_{Kep}$ sont donc données à l'équation (3.2.5) et valent :

$$\begin{cases} r_B^{BE}(t_{k+1}) &= f(\nu) r_B^{BE}(t_k) + g(\nu) v_B^{BE}(t_k) \\ v_B^{BE}(t_{k+1}) &= \dot{f}(\nu) r_B^{BE}(t_k) + \dot{g}(\nu) v_B^{BE}(t_k) \end{cases} \quad k \in \mathbb{N}, \tag{4.2.4}$$

où $\nu = m_{sys}$ et f et g sont les fonctions de Gauss évoquées à la section 2.2.4.

Intéressons-nous à présent à l'évolution des planètes sous $\tilde{\mathcal{H}}^{BE}_{Kep}$. Celle-ci est donnée par le système (4.2.3) avec $i = 3...n$:

$$\begin{cases} \frac{dr^{BE}_{i,c}}{dt} &= \frac{\partial \tilde{\mathcal{H}}^{BE}_{Kep}}{\partial p^{BE}_{i,c}} \\ \frac{dp^{BE}_{i,c}}{dt} &= -\frac{\partial \tilde{\mathcal{H}}^{BE}_{Kep}}{\partial r^{BE}_{i,c}} \end{cases} \quad i = 3...n, \quad c = 1,2,3. \tag{4.2.5}$$

De plus, comme $\tilde{\mathcal{H}}^{BE}_{Kep,1}$ ne dépend d'aucun r_i^{BE} et p_i^{BE}, avec $i = 3,...n$, le système (4.2.5) devient :

$$\begin{cases} \frac{dr^{BE}_{i,c}}{dt} &= \frac{\partial \tilde{\mathcal{H}}^{BE}_{Kep,2}}{\partial p^{BE}_{i,c}} \\ \frac{dp^{BE}_{i,c}}{dt} &= -\frac{\partial \tilde{\mathcal{H}}^{BE}_{Kep,2}}{\partial r^{BE}_{i,c}} \end{cases} \quad i = 3...n, \quad c = 1,2,3.$$

Faire évoluer les planètes sous $\tilde{\mathcal{H}}^{BE}_{Kep}$ est donc équivalent à faire évoluer le système sous $\tilde{\mathcal{H}}^{BE}_{Kep,2}$. De plus, comme $\tilde{\mathcal{H}}^{BE}_{Kep,2}$ ne dépend que des positions et moments associés des planètes, les évolutions peuvent se faire indépendamment de celle du compagnon stellaire. Par ailleurs $\tilde{\mathcal{H}}^{BE}_{Kep,2}$ est analogue à $\tilde{\mathcal{H}}^{HC}_{Kep}$. Nous pouvons donc déterminer les équations d'évolution sous $\tilde{\mathcal{H}}^{BE}_{Kep,2}$ d'une manière similaire à celle utilisée à la section 4.1.1 pour l'évolution sous $\tilde{\mathcal{H}}^{HC}_{Kep}$. Nous trouvons alors :

$$\begin{cases} \frac{dr^{BE}_{i,c}}{dt} &= \frac{p^{BE}_{i,c}}{m_i} + \frac{1}{m_1}\left[\sum_{j=3}^{n} p^{BE}_{j,c}\right] F^{BE}\left(\left\|r_3^{BE}\right\|^2, \left\|r_4^{BE}\right\|^2, ..., \left\|r_n^{BE}\right\|^2\right) \\ \frac{dp^{BE}_{i,c}}{dt} &= \frac{Gm_1 m_i}{\left\|r_i^{BE}\right\|^3} r^{BE}_{i,c} + \frac{r^{BE}_{i,c}}{m_1}\left\|\sum_{j=3}^{n} p_j^{BE}\right\|^2 \frac{\partial\left[\tilde{f}^{BE}\left(\left\|r_i^{BE}\right\|^2\right)\right]}{\partial\left\|r_i^{BE}\right\|^2} \prod_{j=3, j\neq i}^{n}\left[1 - \tilde{f}^{BE}\left(\left\|r_j^{BE}\right\|^2\right)\right] \end{cases} \begin{matrix} i = 3...n, \\ c = 1,2,3, \end{matrix} \tag{4.2.6}$$

avec F^{BE} défini à l'équation (4.2.1) et

$$\frac{\partial\left[\tilde{f}^{BE}\left(\left\|r_i^{BE}\right\|^2\right)\right]}{\partial\left\|r_i^{BE}\right\|^2} =$$

$$= \begin{cases} -30\frac{\left(R_2^2 - \left\|r_i^{HC}\right\|^2\right)^2}{\left(R_2^2 - R_1^2\right)^3} + 60\frac{\left(R_2^2 - \left\|r_i^{HC}\right\|^2\right)^3}{\left(R_2^2 - R_1^2\right)^4} - 30\frac{\left(R_2^2 - \left\|r_i^{HC}\right\|^2\right)^4}{\left(R_2^2 - R_1^2\right)^5} & \text{si } R_1^2 < \left\|r_i^{HC}\right\|^2 < R_2^2 \\ 0 & \text{sinon.} \end{cases}$$

Par ailleurs, en utilisant les équations (3.1.5), (4.2.6) se réécrit, en terme de vitesses,

$$\begin{cases} \frac{dr^{BE}_{i,c}}{dt} &= v^{BE}_{i,c} + \frac{1}{m_1}\left[\sum_{j=3}^{n} m_j v^{BE}_{j,c}\right] F^{BE}\left(\left\|r_3^{BE}\right\|^2, \left\|r_4^{BE}\right\|^2, ..., \left\|r_n^{BE}\right\|^2\right) \\ \frac{dv^{BE}_{i,c}}{dt} &= \frac{Gm_1}{\left\|r_i^{BE}\right\|^3} r^{BE}_{i,c} + \frac{r^{BE}_{i,c}}{m_1 m_i}\left\|\sum_{j=3}^{n} m_j v_j^{BE}\right\|^2 \frac{\partial\left[\tilde{f}^{BE}\left(\left\|r_i^{BE}\right\|^2\right)\right]}{\partial\left\|r_i^{BE}\right\|^2} \prod_{j=3, j\neq i}^{n}\left[1 - \tilde{f}^{BE}\left(\left\|r_j^{BE}\right\|^2\right)\right] \end{cases} \begin{matrix} i = 3...n, \\ c = 1,2,3. \end{matrix} \tag{4.2.7}$$

De plus, ces équations d'évolution ne peuvent être résolues analytiquement et seront donc résolues en utilisant la technique de Bulirsch-Stoer détaillée à la section 4.1.1.

Nous allons à présent nous intéresser à l'évolution sous $\tilde{\mathcal{H}}^{BE}_{Jump}$.

Evolutions sous $\tilde{\mathcal{H}}^{BE}_{Jump}$

$\tilde{\mathcal{H}}^{BE}_{Jump}$ étant analogue à $\tilde{\mathcal{H}}^{HC}_{Jump}$, les équations d'évolution sous cette partie de l'hamiltonien s'obtiennent par un processus analogue à celui sous $\tilde{\mathcal{H}}^{HC}_{Jump}$ dont le calcul est explicité à la section 4.1.1. En appliquant le même calcul, nous trouvons les équations d'évolution :

$$\begin{cases} \frac{dr^{BE}_{i,c}}{dt} &= \frac{1}{m_1}\left[\sum_{j=3}^{n} p^{BE}_{j,c}\right]\left[1 - F^{BE}\left(\left\|r^{BE}_3\right\|^2, \left\|r^{BE}_4\right\|^2, ..., \left\|r^{HC}_n\right\|^2\right)\right] & i = 3...n, \\ \frac{dp^{BE}_{i,c}}{dt} &= \frac{r^{BE}_{i,c}}{m_1}\left\|\sum_{j=3}^{n} p^{BE}_{j}\right\|^2 \frac{\partial\left[\tilde{f}^{BE}\left(\left\|r^{BE}_i\right\|^2\right)\right]}{\partial\left\|r^{BE}_i\right\|^2} \prod_{j=3, j\neq i}^{n}\left[1 - \tilde{f}^{BE}\left(\left\|r^{BE}_j\right\|^2\right)\right] & c = 1,2,3, \end{cases}$$

qui se réécrivent en terme de vitesses en utilisant (3.1.5) :

$$\begin{cases} \frac{dr^{BE}_{i,c}}{dt} &= \frac{1}{m_1}\left[\sum_{j=3}^{n} m_j v^{BE}_{j,c}\right]\left[1 - F^{BE}\left(\left\|r^{BE}_3\right\|^2, \left\|r^{BE}_4\right\|^2, ..., \left\|r^{HC}_n\right\|^2\right)\right] & i = 3...n, \\ \frac{dv^{BE}_{i,c}}{dt} &= \frac{r^{BE}_{i,c}}{m_1 m_i}\left\|\sum_{j=3}^{n} m_j v^{BE}_{j}\right\|^2 \frac{\partial\left[\tilde{f}^{BE}\left(\left\|r^{BE}_i\right\|^2\right)\right]}{\partial\left\|r^{BE}_i\right\|^2} \prod_{j=3, j\neq i}^{n}\left[1 - \tilde{f}^{BE}\left(\left\|r^{BE}_j\right\|^2\right)\right] & c = 1,2,3. \end{cases} \tag{4.2.8}$$

Notons que ces équations ne pouvant être résolues analytiquement, elles seront résolues au moyen d'un algorithme de Bulirsch-Stoer tel qu'explicité à la section 4.1.1.

Nous allons à présent nous intéresser aux modifications apportées à l'algorithme gérant les rencontres proches entre planètes dans les systèmes binaires.

4.2.2 Algorithme avec rencontres proches entre les planètes

Nous allons utiliser un raisonnement analogue à celui de la section 4.1.2 avec le découpage de l'hamiltonien explicité à la section 3.2.2 :

$$\mathcal{H} = \mathcal{H}^{BE}_{Jump} + \mathcal{H}^{BE}_{IntNR} + (\mathcal{H}^{BE}_{IntR} + \mathcal{H}^{BE}_{Kep}),$$

avec

$$\mathcal{H}^{BE}_{IntR} = -\sum_{i=3}^{n-1}\sum_{i<j\leqslant n}\left(\frac{Gm_i m_j}{\|r_{ij}\|}H(\|r_{ij}\|)\right)$$

et

$$\begin{aligned} \mathcal{H}^{BE}_{IntNR} = & -\sum_{i=3}^{n-1}\sum_{i<j\leqslant n}\left(\frac{Gm_i m_j}{\|r_{ij}\|}(1 - H(\|r_{ij}\|))\right) + Gm_A m_B\left(\frac{1}{\|r^{BE}_B\|} - \frac{1}{\|r^{BE}_B + s\|}\right) \\ & + Gm_B\sum_{i=3}^{n} m_i\left(\frac{1}{\|r^{BE}_B\|} - \frac{1}{\|r^{BE}_B - r^{BE}_i + s\|}\right), \end{aligned}$$

où

$$H(r) = \begin{cases} 0 & \text{si } r > r_{crit} \\ 1 & \text{sinon,} \end{cases}$$

tel qu'explicité à la section 2.2.7. Nous remplaçons donc dans cette composition $\mathcal{H}^{BE}_{Jump}$ et $\mathcal{H}^{BE}_{Kep}$ par $\tilde{\mathcal{H}}^{BE}_{Jump}$ et $\tilde{\mathcal{H}}^{BE}_{Kep}$ tels que définis à la section 4.2.1.

Nous avons alors, sur base de l'équation (2.2.13), l'équation d'évolution :

$$f(t_{k+1}) = \exp\left(\frac{\tau}{2}\tilde{T}^{BE}_{Jump}\right)\exp\left(\frac{\tau}{2}T^{BE}_{IntNR}\right)\exp\left(\tau\left[\tilde{T}^{BE}_{Kep}+T^{BE}_{IntR}\right]\right)$$
$$\exp\left(\frac{\tau}{2}T^{BE}_{IntNR}\right)\exp\left(\frac{\tau}{2}\tilde{T}^{BE}_{Jump}\right)f(t_k), \qquad k\in\mathbb{N},$$

où $\tilde{T}^{BE}_{Jump}$, T^{BE}_{IntNR} et $\tilde{T}^{BE}_{Kep}+T^{BE}_{IntR}$ sont les opérateurs linéaires associés à $\tilde{\mathcal{H}}^{BE}_{Jump}$, $\mathcal{H}^{BE}_{IntNR}$ et $\tilde{\mathcal{H}}^{BE}_{Kep}+\mathcal{H}^{BE}_{IntR}$ respectivement. Nous allons ainsi faire évoluer le système d'un demi-pas suivant $\tilde{\mathcal{H}}^{BE}_{Jump}$, ensuite d'un demi-pas suivant $\mathcal{H}^{BE}_{IntNR}$, suivi d'une évolution d'un pas complet suivant $\tilde{\mathcal{H}}^{BE}_{Kep}+\mathcal{H}^{HC}_{IntR}$, puis à nouveau d'un demi-pas suivant $\mathcal{H}^{BE}_{IntNR}$ et enfin d'un demi-pas suivant $\tilde{\mathcal{H}}^{BE}_{Jump}$.

L'évolution sous $\tilde{\mathcal{H}}^{BE}_{Jump}$ a été décrite à la section 4.2.1 et celle sous $\mathcal{H}^{BE}_{IntNR}$ l'a été à la section 3.2.2. Il ne reste donc que l'évolution sous $\tilde{\mathcal{H}}^{BE}_{Kep}+\mathcal{H}^{BE}_{IntR}$ à traiter.

Evolution sous $\tilde{\mathcal{H}}^{BE}_{Kep}+\mathcal{H}^{BE}_{IntR}$

Cette évolution est analogue à celle sous $\mathcal{H}^{BE}_{Kep}+\mathcal{H}^{BE}_{IntR}$ décrite à la section 3.2.2. Nous avons donc l'équation d'évolution :

$$\begin{aligned}
f(t_{k+1}) =& \exp\left(\tau\left[T^{BE}_{Kep}+\sum_{i=0}^{\infty}T^{BE}_{IntR,i}\right]\right)f(t_k)\\
\approx& \exp\left(\frac{\tau}{2}T^{BE}_{IntR,0}\right)\exp\left(\tau\left[\tilde{T}^{BE}_{Kep}+\sum_{i=1}^{\infty}T^{BE}_{IntR,i}\right]\right)\exp\left(\frac{\tau}{2}T^{BE}_{IntR,0}\right)f(t_k)\\
\approx& \exp\left(\frac{\tau}{2}T^{BE}_{IntR,0}\right)\left\{\exp\left(\frac{\tau}{2n}T^{BE}_{IntR,1}\right)\exp\left(\frac{\tau}{n}\left[\tilde{T}^{BE}_{Kep}+\sum_{i=2}^{\infty}T^{BE}_{IntR,i}\right]\right)\exp\left(\frac{\tau}{2n}T^{BE}_{IntR,1}\right)\right\}^n\\
&\exp\left(\frac{\tau}{2}T^{BE}_{IntR,0}\right)f(t_k)\\
\approx& \exp\left(\frac{\tau}{2}T^{BE}_{IntR,0}\right)\left\{\exp\left(\frac{\tau}{2n}T^{BE}_{IntR,1}\right)\left\{\exp\left(\frac{\tau}{2n^2}T^{BE}_{IntR,2}\right)\exp\left(\frac{\tau}{n^2}\left[\tilde{T}^{BE}_{Kep}+\sum_{i=3}^{\infty}T^{BE}_{IntR,i}\right]\right)\right.\right.\\
&\left.\left.\exp\left(\frac{\tau}{2n^2}T^{BE}_{IntR,2}\right)\right\}^n\exp\left(\frac{\tau}{2n}T^{BE}_{IntR,1}\right)\right\}^n\exp\left(\frac{\tau}{2}T^{BE}_{IntR,0}\right)f(t_k)\\
&\vdots
\end{aligned}$$

où $k\in\mathbb{N}$, $\tilde{T}^{BE}_{Kep}$ et $T^{BE}_{IntR,i}$ sont les opérateurs linéaires associés à $\tilde{\mathcal{H}}^{BE}_{Kep}$ et $\mathcal{H}^{BE}_{IntR,i}$ respectivement, avec $\mathcal{H}^{BE}_{IntR}=\sum_{i=0}^{\infty}\mathcal{H}^{BE}_{IntR,i}$. Nous avons donc une réduction successive du pas comme pour l'algorithme SyMBA pour les systèmes binaires. L'unique différence réside dans l'évolution sous $\mathcal{H}^{BE}_{Kep}$ qui est remplacée par l'évolution sous $\tilde{\mathcal{H}}^{BE}_{Kep}$ telle que décrite à la section 4.2.1.

4.2.3 Code

Nous allons à présent détailler les différentes fonctions qui ont été créées pour tenir compte des rencontres proches avec le corps central pour les systèmes binaires.

- *helioq_bs_der1_bin.f* : calcule le membre de droite du système (4.2.8). Cette fonction calcule donc la variation des positions et vitesses dans le repère binaire éloignée sous l'action de $\tilde{\mathcal{H}}^{BE}_{Jump}$.
- *helioq_bs_der2_bin.f* : calcule le membre de droite du système (4.2.7). Cette fonction calcule donc la variation des positions et vitesses dans le repère binaire éloignée sous l'action de $\tilde{\mathcal{H}}^{BE}_{Kep}$.

- *helioq_getf_bin.f* : calcule $F^{BE}\left(\left\|r_3^{BE}\right\|^2, \left\|r_4^{BE}\right\|^2, ..., \left\|r_n^{BE}\right\|^2\right)$, tel qu'exprimé à l'équation (4.2.1), et $\frac{\partial\left[\tilde{f}^{BE}\left(\left\|r_i^{BE}\right\|^2\right)\right]}{\partial\left\|r_i^{BE}\right\|^2} \prod_{j=3, j\neq i}^{n}\left[1-\tilde{f}^{BE}\left(\left\|r_j^{BE}\right\|^2\right)\right]$.

- *helioq_setbs_bin.f* : effectue l'évolution sous $\tilde{\mathcal{H}}_{Jump}^{HC}$ si l'input *iwhich* vaut 1 et sous $\tilde{\mathcal{H}}_{Kep}^{HC}$ si le même input vaut 2. Cette fonction va donc appeler successivement *helioq_bs.f* si l'algorithme de Bulirsch-Stoer a nécessité une réduction du pas. Dans le cas contraire, il ne va l'appeler qu'une fois. De plus, si *iwhich* vaut 2, cette fonction appellera également la fonction *drift_one.f* afin d'effectuer l'évolution décrite à l'équation (4.2.4).

- *symba5q_step_helio_bin.f* : applique l'algorithme décrit à la section 4.2.1.

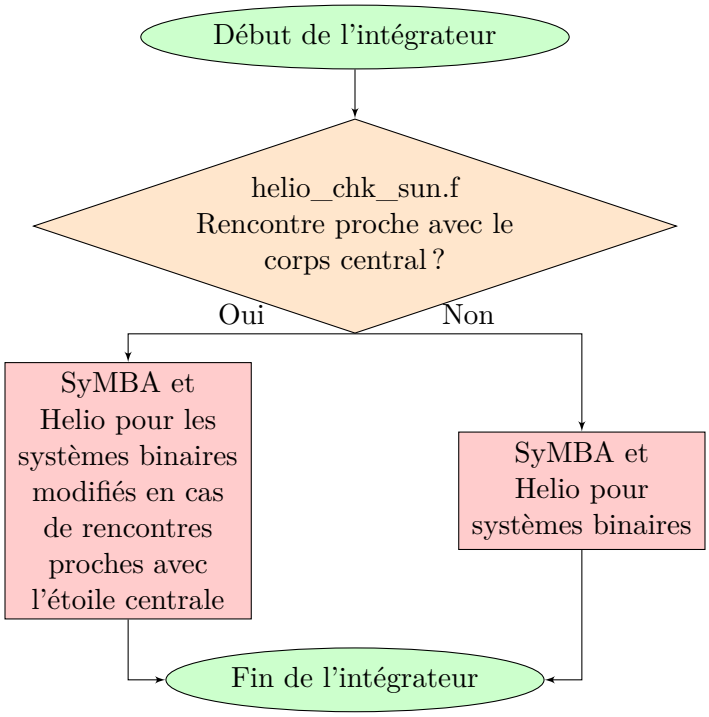
- *symba5q_step_interp_bin.f* : applique l'algorithme décrit à la section 4.2.2.

- *symba5q_step_pl_bin.f* : choisit quel intégrateur utiliser entre ceux des sections 4.2 et 3.2.

- *symba5q_step_recur_bin.f* : effectue l'évolution sous $\tilde{\mathcal{H}}_{Kep}^{BE} + \mathcal{H}_{IntR}^{BE}$.

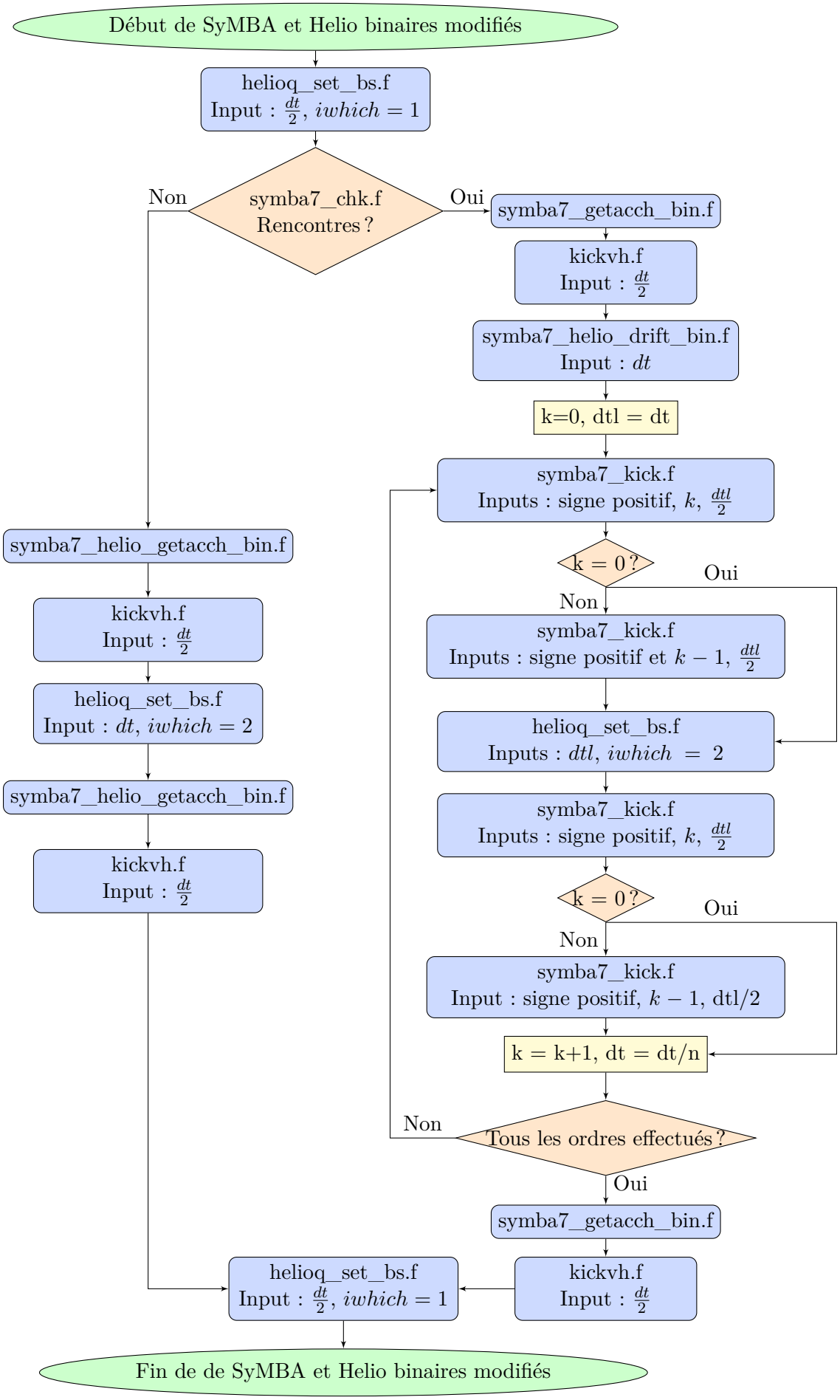
4.2.4 Organigramme

Nous allons à présent réaliser l'organigramme de ce nouveau code. Celui-ci est identique à celui de la section 3.3.2, à l'exception du diagramme "Intégrateur" dont la nouvelle version est reprise ci-dessous



La partie "SyMBA et Helio pour systèmes binaires" correspond au diagramme "Intégrateur" de la section 3.3.2. Le diagramme de "SyMBA et Helio pour les systèmes binaires modifiés en cas de rencontres proches avec l'étoile centrale" est le suivant.

SyMBA et Helio pour les systèmes binaires modifiés en cas de rencontres proches avec l'étoile centrale



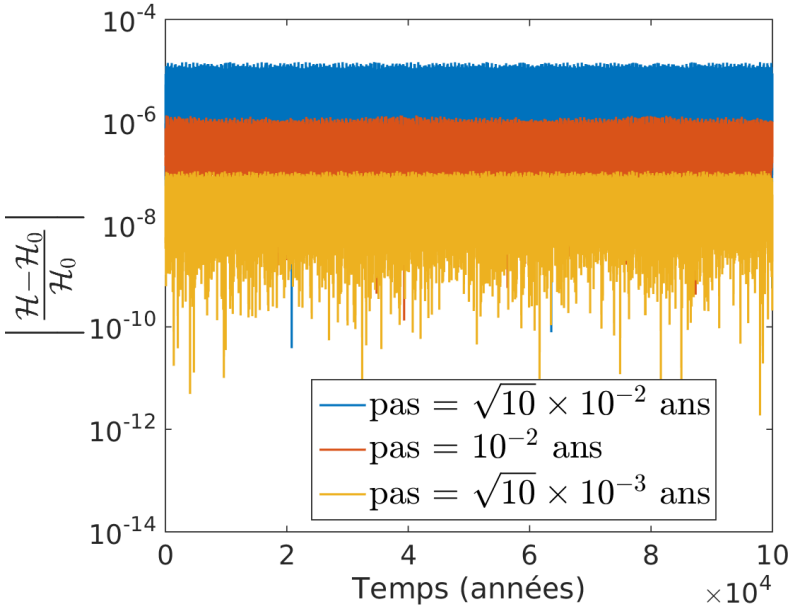


FIGURE 4.5 – Conservation de l'hamiltonien en fonction du pas

4.2.5 Validation

Ordre de l'intégrateur

Tout d'abord, comme à la section 3.4.1, nous multiplions le pas par un facteur α et comparons la conservation de l'hamiltonien pour le système de la table 4.1. En effet, l'intégrateur étant d'ordre τ^2, la multiplication du pas par α, entraîne, en théorie, une variation de la conservation de l'hamiltonien de l'ordre de α^2, comme montré à la section 3.4.1. Nous utilisons les pas $\sqrt{10} \times 10^{-2}$ ans, 10^{-2} ans et $\sqrt{10} \times 10^{-3}$ ans et observons bien à la figure 4.5 un décalage dans l'erreur relative sur l'hamiltonien, de l'ordre de $\sqrt{10}^2 = 10$ entre chaque pas.

Notons que notre schéma d'intégration est bien symplectique. En effet, l'erreur sur l'hamiltonien est bien bornée comme le prédit la théorie des intégrateurs symplectiques. De plus, nous avons bien résolu le problème de conservation de l'hamiltonien illustré à la figure 4.3. Enfin, remarquons que l'erreur relative est plus grande que celle observée à la figure 3.4 malgré des pas utilisés plus faibles. Cela s'explique par la proximité de la planète à l'étoile. En effet, une planète plus proche du corps central a une période de révolution plus faible comme le prédit la troisième loi de Kepler. Or une période plus faible nécessite de considérer un pas plus faible. Afin de bien connaître le mouvement du corps. Les valeurs de pas que nous avons considérées ici ne sont pas suffisamment petites pour apporter la même précision que précédemment.

Validation de trajectoire

Nous avons intégré le système de la table 4.1 avec un pas de 10^{-3} ans. Nous avons dans un premier temps étudié la conservation des intégrales du mouvement à la figure 4.6. Nous pouvons bien constater une conservation des différentes quantités. Notons que nous constatons, comme à la section 3.4.2, une propagation des erreurs numériques entraînant un comportement a priori non symplectique de l'hamiltonien mais qui reste néanmoins conservé à l'ordre de 10^{-9} pendant 10^5 ans.

Par ailleurs, l'erreur sur la coordonnée x du moment angulaire total qui semble bien plus élevée que pour les autres coordonnées vient de son ordre de grandeur. En effet, celle-ci est de l'ordre de 10^{-10} tandis que les coordonnées y et z sont de l'ordre de 10^{-4} et 10^3 respectivement.

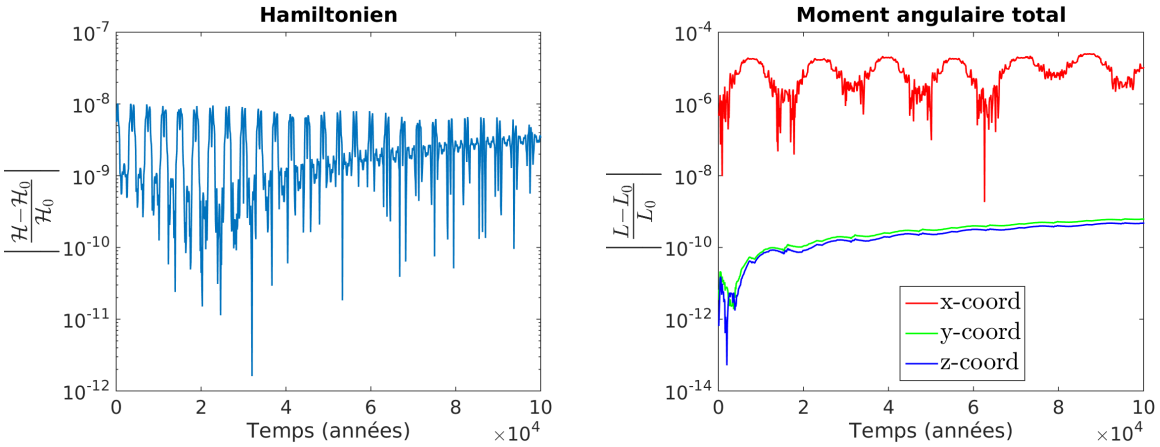
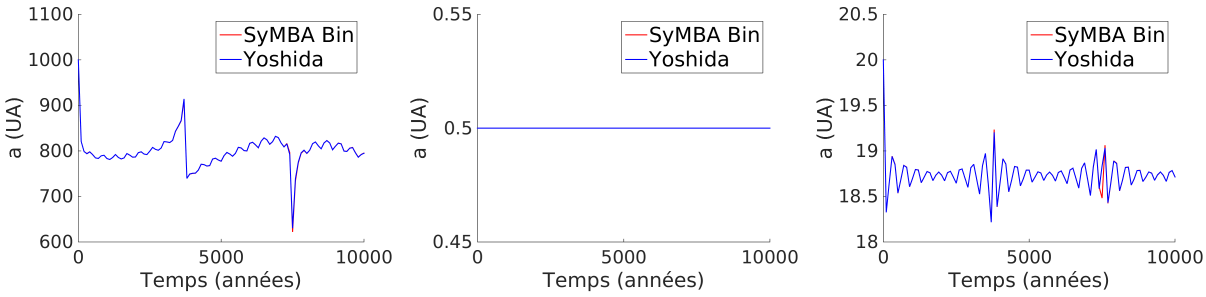


FIGURE 4.6 – Conservation des intégrales premières du mouvement pour le système de la table 4.1 avec un pas de 10^{-3} ans.

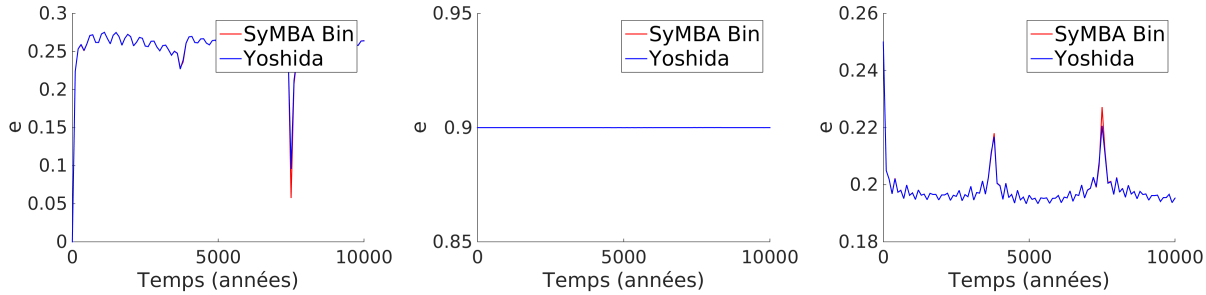
Nous avons ensuite comparé les résultats obtenus avec notre nouvel algorithme à ceux obtenus avec l'algorithme provenant du travail de Chaos et déterminisme [42] présenté à la section 3.4.2. Cette comparaison a été effectuée en gardant un pas de 10^{-3} ans. Nous avons alors obtenu les évolutions des éléments orbitaux reprises aux figures 4.7, 4.8, 4.9, 4.10, 4.11 et 4.12. Notons que le décalage visible entre les deux intégrateurs viendrait de l'instabilité de la méthode de Yoshida utilisée dans le code provenant de [42]. Ce décalage augmente avec le temps d'intégration comme nous pouvons l'observer aux figures 4.13, 4.14, 4.15, 4.16, 4.17 et 4.18. Cependant, l'allure générale est bien similaire.

En conclusion, nous avons modifié le code de façon à pallier aux violations des contraintes (2.2.15) lors de rencontres proches avec l'étoile centrale. Nous possédons donc un code capable de reproduire la dynamique des corps évoluant dans les systèmes binaires.



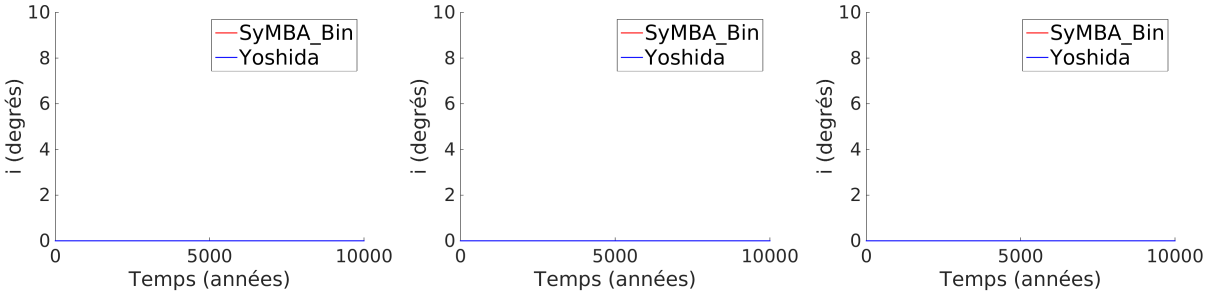
Compagnon stellaire Planète 3 Planète 4

FIGURE 4.7 – Evolutions des demi-grands axes du système de la table 4.1 obtenus avec les intégrateurs SyMBA binaire et Yoshida gérant les rencontres proches avec l'étoile centrale



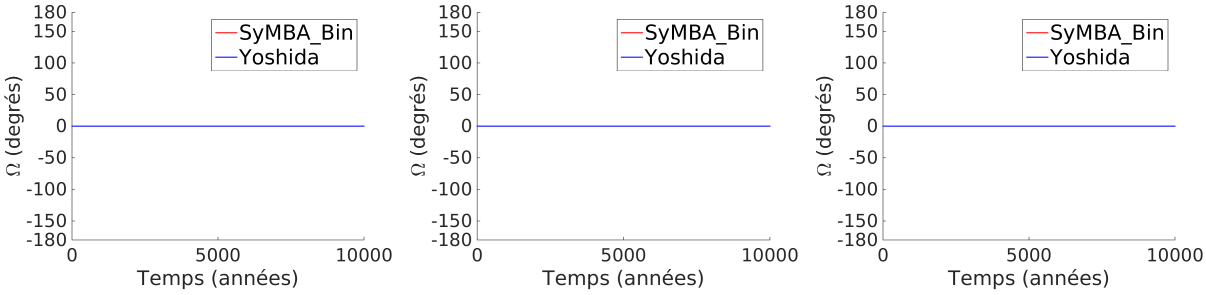
Compagnon stellaire Planète 3 Planète 4

FIGURE 4.8 – Mêmes évolutions pour les excentricités



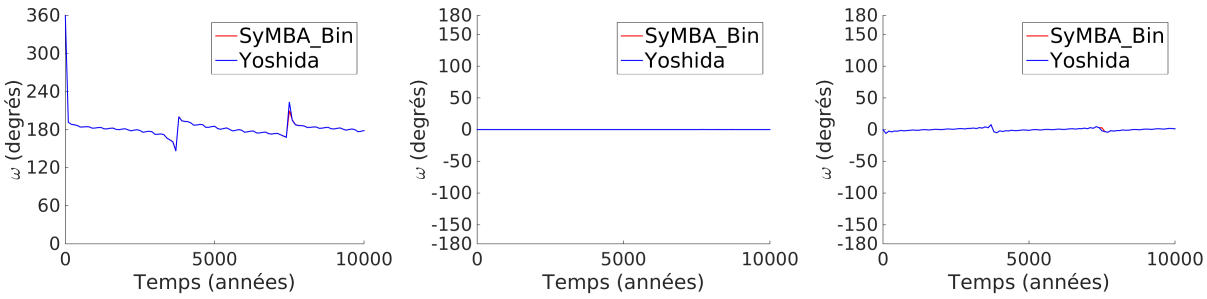
Compagnon stellaire Planète 3 Planète 4

FIGURE 4.9 – Mêmes évolutions pour les inclinaisons



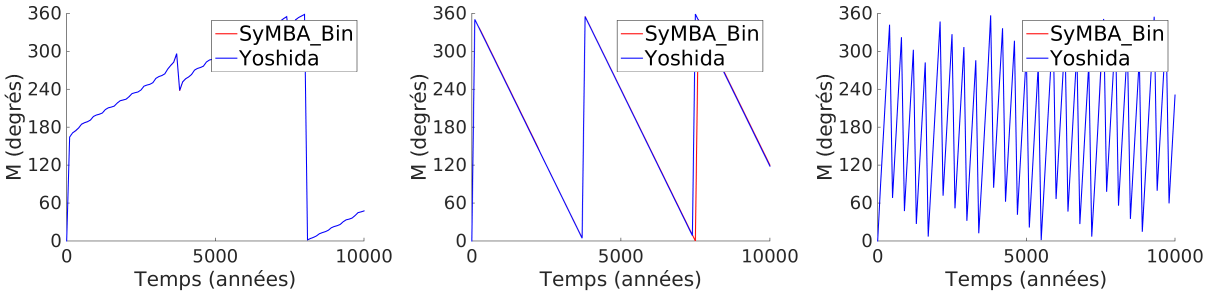
Compagnon stellaire Planète 3 Planète 4

FIGURE 4.10 – Mêmes évolutions pour les longitudes des nœuds ascendants



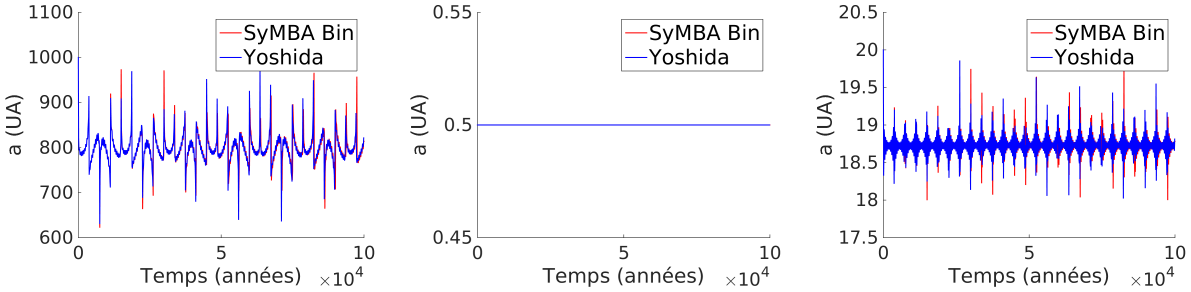
Compagnon stellaire Planète 3 Planète 4

FIGURE 4.11 – Mêmes évolutions pour les arguments des périastres



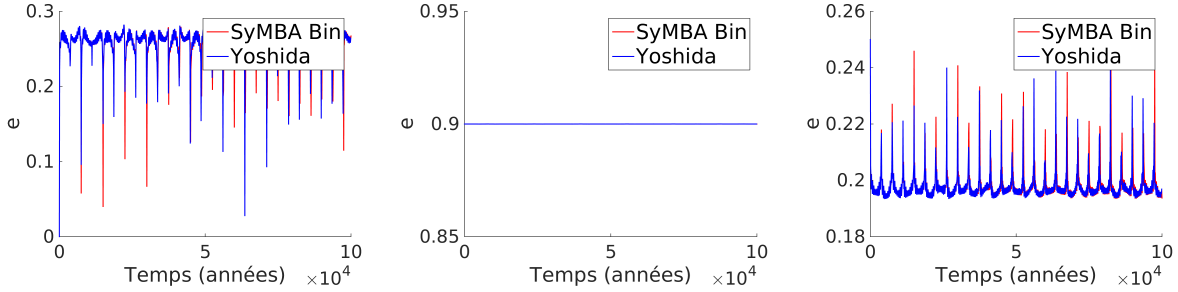
Compagnon stellaire Planète 3 Planète 4

FIGURE 4.12 – Mêmes évolutions pour les anomalies moyennes



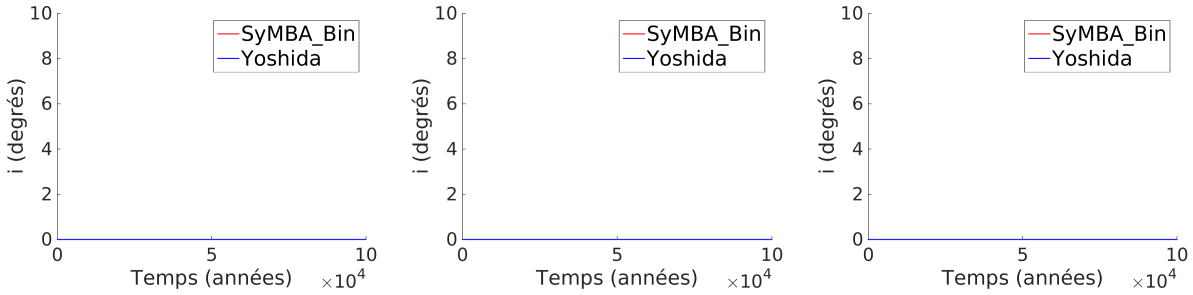
Compagnon stellaire Planète 3 Planète 4

FIGURE 4.13 – Mêmes évolutions des demi-grands axes qu'à la figure 4.7 mais sur une période d'intégration plus longue



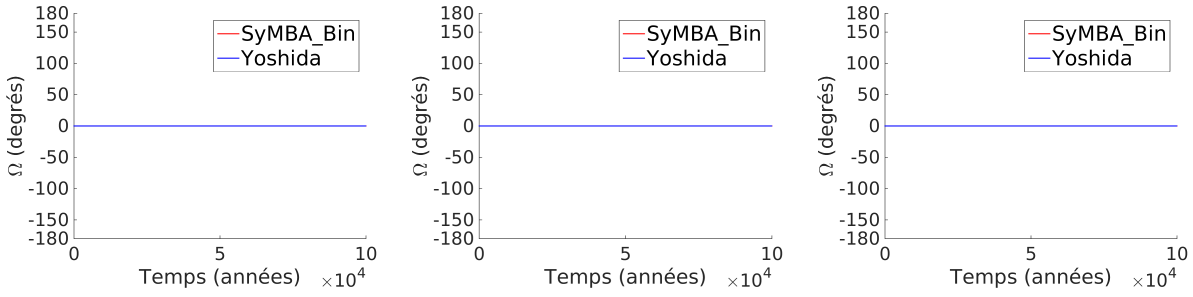
Compagnon stellaire Planète 3 Planète 4

FIGURE 4.14 – Mêmes évolutions des excentricités qu'à la figure 4.8 mais sur une période d'intégration plus longue



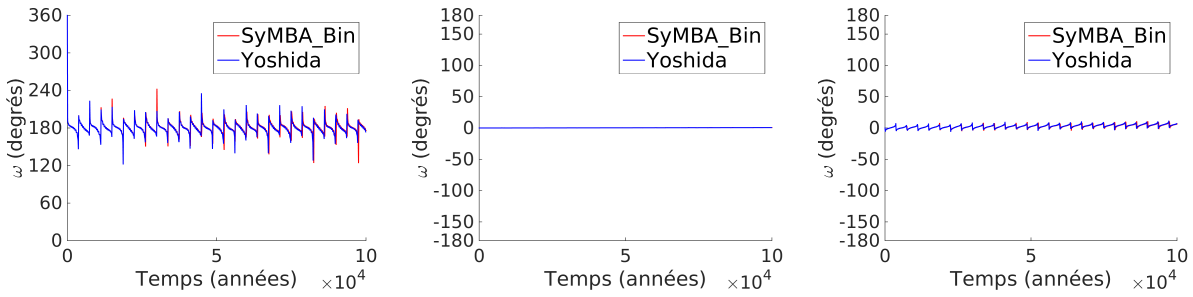
Compagnon stellaire Planète 3 Planète 4

FIGURE 4.15 – Mêmes évolutions des inclinaisons qu'à la figure 4.9 mais sur une période d'intégration plus longue



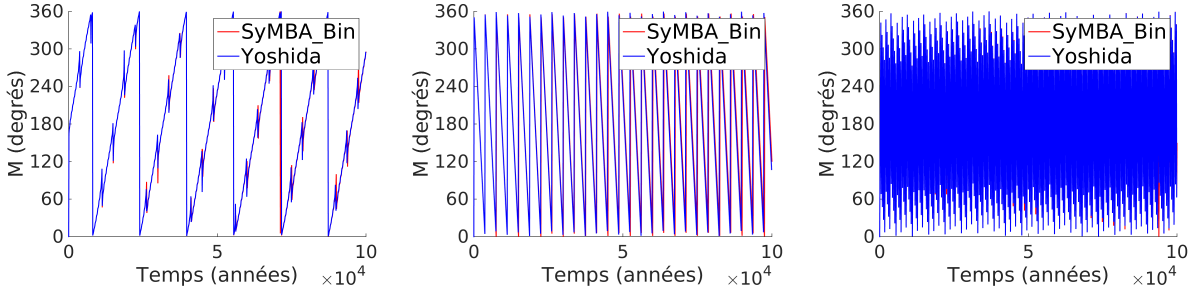
Compagnon stellaire Planète 3 Planète 4

FIGURE 4.16 – Mêmes évolutions des longitudes des nœuds ascendants qu'à la figure 4.10 mais sur une période d'intégration plus longue



Compagnon stellaire Planète 3 Planète 4

FIGURE 4.17 – Mêmes évolutions des arguments des périastres qu'à la figure 4.11 mais sur une période d'intégration plus longue



Compagnon stellaire Planète 3 Planète 4

FIGURE 4.18 – Mêmes évolutions des anomalies moyennes qu'à la figure 4.12 mais sur une période d'intégration plus longue

Chapitre 5

Formation des systèmes de planètes géantes

Nous allons à présent nous intéresser à l'application que nous avons faite du code dans le cadre de ce stage.

De plus en plus de systèmes binaires abritant des exoplanètes sont découverts. Jusqu'à présent, plus de 60 planètes de type S ont été découvertes comme le précise [41]. Dans ce contexte, la question de la formation de tels systèmes se pose. Nous avons donc utilisé notre code afin d'étudier la phase finale de la formation des systèmes planétaires qui est la migration de type II des planètes géantes. Notons que ce processus se produit une fois les planètes formées. Nous n'aborderons donc pas, dans ce manuscrit, la formation des corps planétaires [1]. Notre sujet d'étude sera donc porté sur l'établissement des configurations dynamiques des systèmes planétaires lors de la migration dans le disque de gaz qui les a formés.

Nous avons simulé l'effet du disque de gaz sur les planète via un code développé dans [47]. L'objectif de nos simulations est de reproduire l'étude réalisée dans Libert & Tsiganis [29] en y ajoutant l'influence d'un compagnon stellaire afin d'étudier l'effet que peut avoir ce compagnon stellaire sur la configuration finale des systèmes. Afin de compléter cette étude, nous avons fait varier les paramètres initiaux du compagnon binaire, à savoir le demi-grand axe, l'excentricité, l'inclinaison et la masse du compagnon stellaire.

Dans ce chapitre, nous allons, dans un premier temps, introduire la migration de type II ainsi que les formules utilisées dans nos simulations et émanant de Sotiriadis et al. [47]. Nous allons ensuite détailler notre adaptation de ce code aux systèmes binaires et nous validerons ce nouveau code. Le dernier chapitre de ce manuscrit présentera les résultats des simulations effectuées avec ce code.

5.1 Migration de type II

Comme précisé précédemment, la migration de type II se produit à la fin du processus de formation des systèmes planétaires. Elle résulte de l'interaction entre une planète et le disque de gaz dont elle est issue par accrétion. Notons que cette migration ne s'applique qu'aux planètes de masse élevée (masse supérieure à celle de Saturne). En effet, les planètes moins massives seront soumises à la migration de type I que nous ne détaillerons pas ici mais dont une description peut être trouvées dans [2].

Nous pouvons néanmoins noter que la grande différence entres les migrations de type I et II, au delà de la masse de la planète, provient de l'interaction avec le disque. En effet, une planète

1. Des détails supplémentaires sur cette formation peuvent être trouvés dans [2].

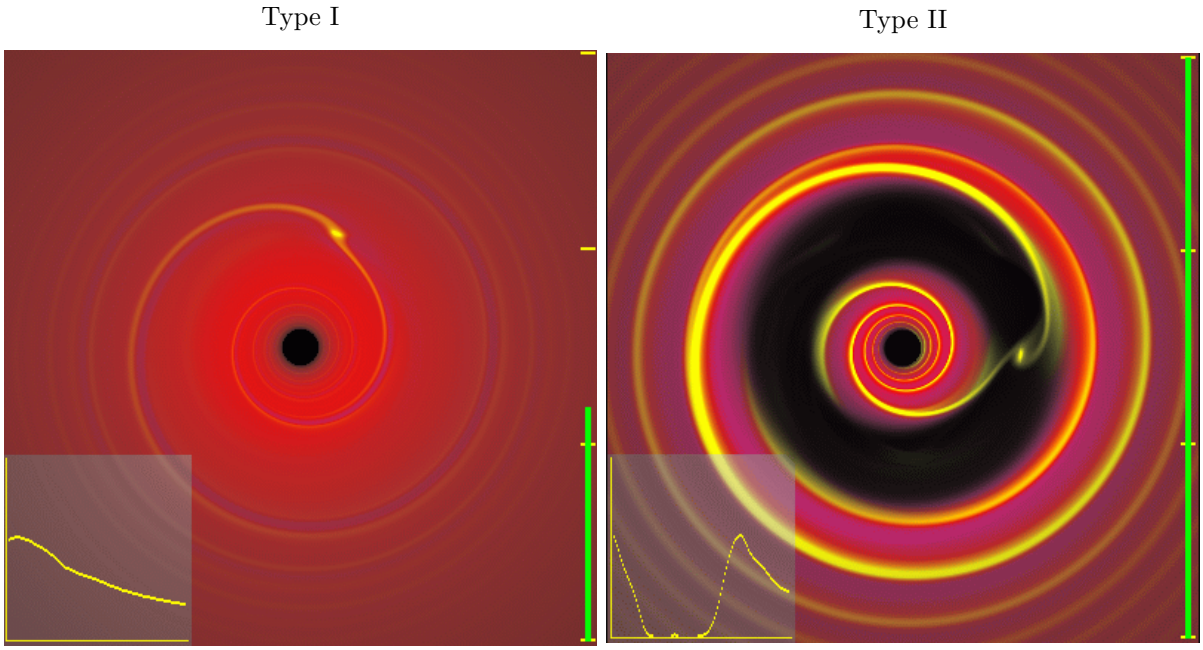


FIGURE 5.1 – Profil du disque de gaz en interaction avec une planète. Figure issue de [3].

plus massive crée une cavité dans le disque gaz, ce que ne parvient pas à faire une planète avec une masse plus faible. Nous pouvons observer cette différence à la figure 5.1. Nous observons bien une cavité au niveau de l'orbite de la planète (point jaune) dans le cas de la migration de type II et qui n'est pas présent pour la migration de type I. Notons que ces figures proviennent de simulations hydrodynamiques.

Nous allons à présent nous intéresser au mécanisme engendrant la formation de cette cavité, en nous basant sur le livre de P. J. Armitage [2]. Celui-ci étudie la variation de vitesse lors de la rencontre proche entre une planète et une partie du disque de gaz ainsi que l'interaction gravitationnelle entre les deux.

Pour ce faire, il prend comme centre de référentiel la planète avec un axe dans la direction de la vitesse de celle-ci avant la rencontre. Il trouve alors que la vitesse perpendiculaire à la trajectoire varie suivant la formule :

$$\delta v_{\perp} = \frac{2Gm_{pl}}{bv_{rel}}, \tag{5.1.1}$$

où $v_{\perp}$ est la vitesse perpendiculaire à la trajectoire, G est la constante gravitationnelle, m_p est la masse de la planète, b est le paramètre d'impact et v_{rel} est la vitesse relative du gaz par rapport à la planète. La détermination de cette formule est détaillée dans [2].

En utilisant la conservation de l'énergie, P. J. Armitage déduit la relation :

$$v_{rel}^2 = \delta v_{\perp}^2 + \left(r_{rel} - \delta v_{/\!/}\right)^2, \tag{5.1.2}$$

où $\delta v_{/\!/}$ est la modification de la vitesse parallèle à la trajectoire. En effet, l'énergie potentielle n'étant pas modifiée par modification des vitesses, il égale les énergies cinétiques avant et après la rencontre. Nous avons alors

$$\frac{m_{pl}\left\|v_{pl,1}\right\|^2}{2} + \frac{m_{gaz}\left\|v_{gaz,1}\right\|^2}{2} = \frac{m_{pl}\left\|v_{pl,2}\right\|^2}{2} + \frac{m_{gaz}\left\|v_{gaz,2}\right\|^2}{2}, \tag{5.1.3}$$

où m_{pl} et m_{gaz} sont respectivement la masse de la planète et celle de la partie du disque de gaz à proximité, $v_{pl,1}$ et $v_{gaz,1}$ sont respectivement la vitesse de la planète et celle du gaz avant

la rencontre proche et $v_{pl,2}$ et $v_{gaz,2}$ sont respectivement la vitesse de la planète et celle du gaz après la rencontre proche. De plus, le référentiel étant centré sur la planète, nous avons $v_{pl,1} = v_{pl,2} = 0$. La formule (5.1.3) devient alors

$$\frac{m_{gaz} \|v_{gaz,1}\|^2}{2} = \frac{m_{gaz} \|v_{gaz,2}\|^2}{2}$$

et celle-ci se simplifie en

$$\|v_{gaz,1}\|^2 = \|v_{gaz,2}\|^2 . \tag{5.1.4}$$

Par ailleurs, avant la rencontre, la vitesse est colinéaire au mouvement. Nous avons donc $v_{gaz,1} = (v_{rel}, 0, 0)$ et $v_{gaz,2} = \Big(v_{rel} - \delta v_{/\!/}, v_\perp, 0\Big)$. En utilisant ces relations dans l'équation (5.1.4), nous retrouvons bien la relation (5.1.2).

Ensuite, en supposant que la variation de vitesse est petite et en utilisant le développement de Taylor d'ordre 1, la formule (5.1.2) se réécrit :

$$v_{rel}^2 \simeq (\delta v_{perp})^2 + v_{rel}^2 - 2 v_{rel} \delta v_{/\!/}.$$

En simplifiant cette équation et en utilisant la relation (5.1.1), nous retrouvons la variation de vitesse parallèle à la trajectoire exprimée dans [2] :

$$\delta v_{/\!/} \simeq \frac{1}{2 v_{rel}} \left(\frac{2 G m_{pl}}{b v_{rel}} \right)^2 . \tag{5.1.5}$$

Cette modification de la vitesse parallèle à la trajectoire du gaz va entraîner une variation de son moment angulaire. En effet, la vitesse parallèle à la trajectoire en cas de rencontre, correspond, à plus grande échelle, à la vitesse tangente à la trajectoire des particules. Nous aurons donc bien une modification du moment angulaire du gaz.

De plus, comme le système est conservatif, la planète subira une variation équivalente et opposée de son moment angulaire afin de garder le moment angulaire total constant. Notons que cette conservation peut également être vue comme l'application d'un moment de force à la planète équivalent et opposé à celui appliqué au disque (car une variation du moment angulaire correspond à un moment de force non nul).

Par ailleurs, par la troisième loi de Kepler, nous avons la proportionnalité :

$$P \sim a^{\frac{3}{2}},$$

où P est la période et a est le demi-grand axe du corps. Un corps plus proche du centre de l'orbite aura donc une période plus faible et "dépassera" les corps plus éloignés.

Le gaz intérieur à la planète va donc la "dépasser". Il aura donc une vitesse relative positive. Nous avons donc $v_{rel} > 0$, d'où $\delta v_{/\!/} > 0$ par la relation (5.1.5). Nous aurons donc une réduction de la vitesse tangente du disque (car $v_{tan} = v_{rel} - \delta v_{/\!/}$) d'où une diminution de son moment angulaire. De plus, comme explicité précédemment, cela entraînera une augmentation du moment angulaire de la planète. Par un raisonnement analogue, la partie du disque extérieure à la trajectoire de la planète subira une augmentation de son moment angulaire et induira une diminution du moment angulaire de la planète. Rappelons que le moment angulaire L est lié au demi-grand axe d'un corps par la formule :

$$L = \sqrt{G m_{pl} a \,(1 - e^2)}.$$

Comme le précise [47], les planètes ont tendance à garder une orbite circulaire dans le disque, la variation du moment angulaire va donc principalement impacter le demi-grand axe. Une augmentation du moment angulaire entraînera alors une augmentation du demi-grand axe et une

diminution du moment angulaire entraînera une diminution du demi-grand axe.

Sur base de la discussion précédente concernant les variations de moment angulaire, nous pouvons donc déduire que le gaz intérieur à l'orbite de la planète subira une diminution de son demi-grand axe et la planète aura, en réaction à celle-ci, une augmentation de son demi-grand axe. La planète et le gaz auront donc tendance à s'éloigner l'un de l'autre. Par un raisonnement analogue, nous pouvons déduire que le demi-grand axe du gaz extérieur à la planète aura tendance à augmenter et celui de la planète à diminuer. Ils auront donc également tendance à s'éloigner l'un de l'autre. Ces mécanismes auront pour conséquence de créer une cavité autour de la planète ainsi qu'une migration de celle-ci. En effet, la planète migrera vers l'étoile lorsque l'effet du disque extérieur est plus important que l'effet du disque intérieur, ce dernier étant généralement moins massif.

Cependant un autre phénomène s'oppose à la formation de la cavité. En effet, le gaz a tendance, par sa viscosité, à homogénéiser le disque ([2]). Cela a pour effet de continuellement reboucher le trou qui se forme. La formation de celui-ci dépend donc du rapport de force entre phénomène d'ouverture et celui de fermeture. L'ouverture étant proportionnelle à la masse de la planète au carré par l'équation (5.1.5), seules les planètes de masse suffisante forment une cavité dans le disque.

Une fois la cavité formée, l'échange de moment angulaire entre la planète et le disque n'aura plus lieu. En effet, il n'y aura plus de rencontre proche entre le disque et la planète car le voisinage de la planète sera vide de gaz. La planète va donc arrêter de migrer et sera alors dans un état stationnaire par rapport au disque.

Par ailleurs, le disque a un mouvement de migration vers l'étoile que nous pouvons observer à la figure 5.1, dû à l'accrétion de celle-ci et à la viscosité du disque. La planète va donc migrer vers l'étoile en suivant le mouvement du disque ([31] et [1]). C'est cette migration qui est appelée migration de type II et que nous avons étudiée durant ce stage.

Notons que les calculs détaillés dans ce chapitre possèdent quelques approximations comme le recours au développement de Taylor. Un autre développement plus précis peut être effectué sur base de la résonance de Linblad et est détaillé dans [2]. Cependant, celui-ci nécessite beaucoup plus de détails et de prérequis. Le but du manuscrit étant d'étudier l'influence du compagnon binaire et non d'étudier les théories de migration, nous avons choisi le développement présenté ci-dessus afin de comprendre les mécanismes entrant en jeu dans la migration de type II. Notons que le recours à la résonance de Linbald est plus solide mathématiquement mais amène aux mêmes résultats avec un effort de raisonnement bien plus important.

Formule de migration

La détermination du taux de migration n'est pas aisée [1] et c'est encore aujourd'hui un domaine très actif de la recherche. Sur base de [12], le taux de migration utilisé dans notre code est celui exprimé dans Sotiriadis et al. [47] :

$$\left|\frac{\dot{a_{pl}}}{a_{pl}}\right| = \frac{1}{\tau_{mig}},$$

où a_{pl} est le demi-grand axe de planète et avec

$$\tau_{mig} = \frac{2}{3}\alpha^{-1}h^{-2}\omega_{pl} \times max\left\{1, \frac{m_{pl}}{m_{dl}}\right\}, \tag{5.1.6}$$

1. Par exemple, la cavité peut ne pas être totalement vide et cela peut avoir une influence sur la migration comme l'a montré [12].

où α est un paramètre du disque nommé paramètre de viscosité de Shakura-Sunyaev introduit dans [46], h est le facteur de forme du disque, ω_{pl} est la vitesse angulaire de la planète pour une orbite circulaire, m_{pl} est la masse de la planète et m_{dl} est la masse locale du disque. Pour nos simulations, nous avons repris les valeurs des paramètres utilisées dans [47], en particulier les valeurs classiques 0.05 et 0.005 pour h et α respectivement.

Pour déterminer m_{dl}, nous avons besoin du profil de densité surfacique du disque. Celui utilisé par [47] est de la forme :

$$\sigma(r) = \sigma_0 r^{-\gamma}, \tag{5.1.7}$$

avec $\gamma = 0.5$. Néanmoins, le code a été construit de manière à avoir trois choix possibles de gamma : 0.5, 1 et 1.5. Sur base de l'équation (5.1.7), nous pouvons calculer la masse d'une partie du disque par la formule :

$$\int\int_S \sigma(r) dS, \tag{5.1.8}$$

où S est la surface considérée du disque. Par ailleurs, [47] nous précise que la partie locale du disque pour une planète est la partie du disque comprise entre les distances $0.2a_{pl}$ et $2.5a_{pl}$. Avec ces bornes et la définition (5.1.7), l'intégrale (5.1.8) se réécrit, pour le disque local, en coordonnées polaires,

$$m_{dl} = \int_{0.2a_{pl}}^{2.5a_{pl}} \int_0^{2\pi} \sigma_0 r^{-\gamma} r d\theta dr.$$

Cette intégrale se résout aisément et nous trouvons

$$m_{dl} = 2\pi\sigma_0 \frac{(2.5a_{pl})^{2-\gamma} - (0.2a_{pl})^{2-\gamma}}{2-\gamma} = \left(2.5^{2-\gamma} - 0.2^{2-\gamma}\right) 2\pi\sigma_0 \frac{a_{pl}^{2-\gamma}}{2-\gamma}.$$

En utilisant la formule (5.1.7), la masse du disque local se réécrit alors

$$m_{dl} = \left(2.5^{2-\gamma} - 0.2^{2-\gamma}\right) \frac{2\pi}{2-\gamma} \sigma(a_{pl}) a_{pl}^2.$$

Si nous utilisons $\gamma = 0.5$ comme précisé précédemment, nous avons

$$m_{dl} = \left(2.5^{1.5} - 0.2^{1.5}\right) \frac{4\pi}{3} \sigma(a_{pl}) a_{pl}^2.$$

Notons qu'en pratique, dans le code, $(2.5^{1.5} - 0.2^{1.5})$ étant une constante, elle sera incorporée dans σ_0.

Par ailleurs, [47] précise également que lorsque nous approchons de la limite intérieure du disque, une zone de transition a été adoptée au moyen de la fonction hyperbolique :

$$f(r) = tanh\left(\frac{r - R_{in}}{\delta r}\right),$$

où R_{in} est la limite intérieure du disque et δr est la largeur de la zone de transition. Dans la zone dont la distance à l'étoile est comprise dans l'intervalle $[R_{in}, R_{in} + \delta r]$, la masse du disque sera donc multipliée par $f(r)$. Cette zone de transition modélise la dissipation qui survient au bord du disque.

Concernant ω_{pl}, celle-ci a pour définition :

$$\omega_{pl} = \frac{2\pi}{T_{pl}},$$

où T_{pl} est la période de la planète. Cette équation se réécrit, en utilisant la troisième loi de Kepler, $T_{pl} = \sqrt{\frac{4\pi^2 a_{pl}^3}{G m_1}}$, avec m_1 la masse de l'étoile :

$$\omega_{pl} = 2\pi \sqrt{\frac{G m_1}{4\pi^2 a_{pl}^3}}. \tag{5.1.9}$$

De plus, dans nos simulations, la masse de l'étoile centrale sera toujours égale à 1 masse solaire. Or dans le système de référence adopté par le code, afin que $G = 1$ (comme précisé à la section 2.3), une masse solaire correspond à $4\pi^2$. Ainsi, l'équation (5.1.9) se réécrit

$$\omega_{pl} = 2\pi\sqrt{\frac{1}{a_{pl}^3}}.$$

Nous pouvons à présent réécrire (5.1.6) comme

$$\tau_{mig} = \frac{2}{3}0.005^{-1}0.05^{-2}2\pi\sqrt{\frac{1}{a_{pl}^3}} \times max\left\{1, \frac{m_{pl}}{(2.5^{1.5} - 0.2^{1.5})\,(4\pi/3)\sigma(a_{pl})a_{pl}^2}\right\}.$$

Remarquons que la migration ne sera effectuée, dans nos simulations, que sur la planète la plus extérieure, comme le proposent [29] et [47].

De plus, pour insérer la migration dans le code SyMBA, [47] utilise l'accélération présentée dans [35] :

$$a_{mig} = -\frac{v_{pl}}{\tau_{mig}},$$

où v_{pl} est la vitesse de la planète qui migre et a_{mig} est l'accélération due à la migration.

Le disque a également une influence sur l'excentricité et l'inclinaison des planètes présentes dans celui-ci. [47] prend en compte ces effets en insérant dans le code les amortissements en excentricité et en inclinaison calculés par [6] à partir de simulations hydrodynamiques. Nous allons détailler les formules utilisées par [47] et reproduites dans notre code pour modéliser cette influence.

Influence sur l'excentricité

Nous commençons par nous intéresser à l'excentricité des planètes dans le disque. Le disque a pour effet de réduire l'excentricité des planètes qu'il contient. Le taux d'amortissement de l'excentricité est classiquement considéré comme proportionnel au taux de migration des planètes :

$$\frac{\dot{e}}{e} = -K\left|\frac{\dot{a}}{a}\right|, \tag{5.1.10}$$

où K est un paramètre arbitraire. Cette formule impose donc un choix pour le paramètre K. C'est sur cette formule qu'est basée l'étude [29].

Une nouvelle formule a été déterminée récemment dans Bitsch et al. [6] au moyen de simulations hydrodynamiques permettant de se détacher de ce choix arbitraire. Pour ce faire, les auteurs ont étudié l'évolution de planètes massives fortement excentriques et inclinées dans le disque de gaz. Sur base de ces simulations hydrodynamiques, ils ont déduit des formules analytiques approchées pour les amortissements en excentricité et en inclinaison de la planète $\dot{e}_{pl}$ et $\dot{i}_{pl}$. Ces dernières sont implémentées dans le code de [47]. Pour l'excentricité, nous avons

$$\dot{e_{pl}} = -\frac{m_{dl}}{0.025m_1}\left(a_{exc}\left[i_{pl} + i_d\right]^{-2b_{exc}} + c_{exc}i_{pl}^{-2d_{exc}}\right)^{-\frac{1}{2}} + 12.65\frac{m_{pl}m_{dl}}{m_1^2}e_{pl}\exp\left\{-\left[\frac{i_{pl}}{m_{pl}}\right]^2\right\}, \tag{5.1.11}$$

avec a_{pl}, e_{pl} i_{pl} et m_{pl} respectivement le demi-grand axe, l'excentricité, l'inclinaison et la masse de la planète, m_{dl} la masse locale du disque, m_1 la masse de l'étoile centrale, $i_d = \frac{m_{pl}}{3}$ degrés,

$$a_{exc} = 80e_{pl}^{-2}\exp\left\{-e_{pl}^2\frac{m_{pl}}{0.26}\right\}15^{m_{pl}}\left(20 + 11m_{pl} - m_{pl}^2\right),$$

$$b_{exc} = 0.3m_{pl},$$

$$c_{exc} = 450 + 2^{m_{pl}}$$

et

$$d_{exc} = -1.4 + \frac{\sqrt{m_{pl}}}{6}.$$

Notons que l'ensemble des angles présents dans ces formules sont exprimés en degrés. Par ailleurs, cette formule est légèrement différente de celle exposée dans [6]. En effet, après discussion avec S. Sotiriadis, il est apparu que le premier coefficient de $\dot{e}$ devait être proportionnel à $\frac{1}{0.025m_1}$ et non à $\frac{1}{0.01m_1}$, comme indiqué dans [6].

L'influence sur l'excentricité sera également exprimée en termes d'accélération suivant les formules de [35] :

$$a_e = -2\frac{(v_{pl}^H \cdot r_{pl}^H) r_{pl}^H}{\left\| r_{pl}^H \right\|^2 \tau_{exc}},$$

où v_{pl}^H et r_{pl}^H sont respectivement la position et la vitesse de la planète dans un repère héliocentrique et

$$\tau_{exc} = \left| \frac{e_{pl}}{\dot{e_{pl}}} \right|.$$

Enfin, notons que, contrairement à la migration, cette évolution sera appliquée à toutes les planètes présentes dans le disque.

Influence sur l'inclinaison

Nous allons à présent nous intéresser à l'influence du disque sur l'inclinaison des planètes. Cette influence n'avait pas été considérée dans Libert & Tsiganis [29]. En effet, cette influence est supposée plus faible que celle sur l'excentricité et avait donc été négligée.

Le taux d'amortissement en inclinaison trouvé par [6] est le suivant :

$$\dot{i_{pl}} = -\frac{m_{dl}}{0.025m_1}\left(a_{inc} i_{pl}^{-2b_{inc}} \exp\left\{ \frac{-(i_{pl}/g_{inc})^2}{2} \right\} + c_{inc}\left[\frac{i_{pl}}{40} \right]^{-2d_{inc}} \right)^{-\frac{1}{2}}, \tag{5.1.12}$$

avec

$$a_{inc} = 1.5 \times 10^4 (2 - 3e_{pl}) m_{pl}^3,$$

$$b_{inc} = 1 + \frac{m_{pl} e_{pl}^2}{10},$$

$$c_{inc} = \frac{1.2 \times 10^6}{(2 - 3e_{pl})\left(5 + e_{pl}^2 \left[m_{pl} + 2\right]^3\right)},$$

$$d_{inc} = -3 + 2e_{pl}$$

et

$$g_{inc} = \sqrt{\frac{3m_{pl}}{e_{pl} + 0.001}}.$$

Notons que, comme le signale [47], un problème numérique survient si $2 - 3e_{pl} < 0$. En effet, dans ce cas là, a_{inc} et c_{inc} deviennent négatifs et donc $a_{inc} i_{pl}^{-2b_{inc}} \exp\left\{ \frac{-(i_{pl}/g_{inc})^2}{2} \right\} + c_{inc}\left[\frac{i_{pl}}{40} \right]^{-2d_{inc}} < 0$. Or ce terme est le radicant d'une racine carrée dans l'équation (5.1.12). Afin de résoudre ce problème, S. Sotiriadis a imposé la valeur 10^{-5} à $2 - 3e_{pl}$ si celui-ci venait à devenir négatif dans le code.

Comme pour l'excentricité, [47] utilise les formules de [35] afin de transformer cette influence en accélération :

$$a_i = -2\frac{(v\cdot k)k}{\tau_{inc}},$$

où k est le vecteur vertical unitaire et

$$\tau_{inc} = \left|\frac{i_{pl}}{\dot{i}_{pl}}\right|.$$

Notons que cette accélération sera appliquée à toutes les planètes présentes dans le disque.

Dissipation du disque

Suite à un phénomène de dissipation, le disque de gaz va progressivement disparaître. Le code de [47] permet de choisir entre deux modèles de dissipation du disque que nous allons présenter dans cette section.

Le premier modèle arrête instantanément l'interaction avec le disque à la fin de son temps de vie. Le disque va donc être considéré comme constant jusqu'à sa dissipation complète à un temps fixé au préalable.

Le deuxième modèle est une réduction exponentielle progressive du disque. σ_0 sera alors modifié progressivement durant l'intégration suivant la formule :

$$\sigma_0(t) = \sigma_0(0) \times e^{-\frac{t}{T_0}},$$

où T_0 est un paramètre fixé. La masse totale du disque m_{disk} sera alors donnée par :

$$m_{disk}(t) = m_{disk}(0) \times e^{-\frac{t}{T_0}},$$

où $m_{disk}(0)$ est la masse initiale du disque. De plus, le code considérera le disque comme entièrement dissipé et arrêtera donc toute interaction avec celui-ci quand

$$|\dot{m}_{disk}(t)| = m_{disk}(0)\frac{e^{-\frac{t}{T_0}}}{T_0}$$

sera inférieur à un seuil critique.

C'est ce dernier modèle que nous avons utilisé dans nos simulations.

Influence globale

Les équations (5.1.11) et (5.1.12) ont été déterminées dans [6] en ne considérant qu'une seule planète dans le disque. Cependant, si des planètes sont proches dans le disque, celles-ci peuvent, potentiellement, partager la même cavité dans le disque de gaz. L'influence du disque s'en trouvera alors modifiée. Afin de tenir compte de ce cas de figure, [47] propose les adaptations suivantes :

- si deux planètes sont dans le même espace vide de gaz, il propose de réduire $\dot{e}_{pl}$ et $\dot{i}_{pl}$ sont réduits de moitié. En effet, dans ce cas-là, une planète subit l'influence du disque intérieur et l'autre subit celle du disque extérieur à la trajectoire des planètes. L'influence du disque sur chaque planète sera donc diminuée de moitié.
- si le trou comporte trois planètes, l'influence du disque sur celle du milieu est divisée par un facteur 100, tandis que celle sur les deux autres planètes l'est par un facteur 2. En effet, la planète au centre ne ressent que très peu l'influence du disque.

Afin de déterminer la proximité des planètes, [47] utilise un critère basé sur les sphères de Hill des planètes [1]. Le code considère donc que deux planètes partagent une même cavité si et seulement si

$$a_{pl,1} - a_{pl,2} < 2(r_{hill,1} + r_{hill,2}),$$

où $a_{pl,i}$ et $r_{hill,i}$ sont respectivement le demi-grand axe et le rayon de la sphère de Hill de la planète i, $i \in \{1, 2\}$.

L'influence totale du disque sera alors donnée, en terme d'accélération, par :

$$a_{disque} = a_{mig} + a_{exc} + a_{inc}$$

pour la planète la plus extérieure dans le disque et par :

$$a_{disque} = a_{exc} + a_{inc}$$

pour les autres planètes dans le disque. Cette accélération aura pour effet de modifier la vitesse des planètes. Nous aurons alors l'équation d'évolution :

$$v_i^H(t_{k+1}) = v_i^H(t_k) + \tau a_{disque,i}^H, \tag{5.1.13}$$

où τ est le pas d'intégration, $a_{disque,i}^H$ est la modification de l'accélération sur la planète i due au disque et $v_i^H(t)$ est la vitesse de la planète i au temps t.

Notons par ailleurs que l'incorporation de cet effet est effectuée de manière à conserver le caractère symplectique de notre intégrateur. Pour ce faire, [47] a suivi la méthode proposée par [25], qui consiste à recourir à la formule d'intégration (5.1.13) :

$$f(t_{k+1}) = E_{disk}\left(\frac{\tau}{2}\right) \exp\left(T\tau\right) E_{disk}\left(\frac{\tau}{2}\right) f(t_k),$$

où T est l'opérateur linéaire associé à l'hamiltonien $\mathcal{H}$ du système à n corps constitué des planètes et des étoiles et $E_{disk}\left(\frac{\tau}{2}\right)$ est l'évolution due au disque exprimée par l'équation (5.1.13).

5.2 Modifications de SyMBA

Pour simuler les interactions avec le disque de gaz décrites ci-dessus, S. Sotiriadis a modifié l'algorithme SyMBA présenté précédemment. Dans cette section, nous allons décrire les différentes fonctions qu'il a implémentées ainsi que l'organigramme de nouveau code.

5.2.1 Code

Les deux nouvelles fonctions réalisant la migration sont les suivantes :

- *dissipative_kick.f* : calcule la modification d'accélération induite par le disque selon l'équation (5.1.13).
- *local_init_migrate.f* : importe les différents paramètres du disque de gaz. Ceux-ci proviennent du nouveau fichier d'input migrate.in que nous allons présenter par la suite.

De plus, les fonctions *symba7_step_pl.f* et *symba7_step_helio.f* ont également été modifiées de manière à incorporer l'influence du disque.

Par ailleurs comme précisé précédemment, un nouveau fichier d'entrées a été créé, il s'agit du fichier migrate.in qui est de la forme :

1. La sphère de Hill est la zone où l'influence de la planète est supérieure à celle de l'étoile centrale.

```
Npl
Mstar,Sigma0
Rin,Rout,DeltaR
T0,gamma_coef,flag_disk
tstop_interaction,ta_coeff
```

Nous allons à présent détailler chacun de ces éléments :

- Première ligne :
  - Npl : nombre de planètes dans le système.
- Deuxième ligne :
  - Mstar : masse de l'étoile centrale du système en masse de Jupiter.
  - Sigma0 : constante de proportionnalité utilisée pour la surface de densité du disque. Elle correspond au produit $\sigma_0 \left(2.5^{2-\gamma} - 0.2^{2-\gamma}\right)$.
- Troisième ligne :
  - Rin : limite intérieure du disque de gaz.
  - Rout : limite extérieure du disque de gaz.
  - deltaR : largeur de la zone de transition pour la limite intérieure du disque.
- Quatrième ligne :
  - T0 : constante utilisée pour la dissipation progressive du disque si ce modèle est choisi.
  - gamma_coef : coefficient γ utilisé dans le profil de densité du disque.
  - flag_disk : prend la valeur 1 si le modèle de dissipation progressive du disque est choisi, la valeur 0 s'il n'y a pas de disque et la valeur -1 pour un arrêt brutal de l'influence d'un disque de masse constante après un temps fixé.
- Cinquième ligne :
  - tstop_interaction : temps de dissipation du disque si le modèle avec le disque de masse constante est choisi.
  - ta_coeff : facteur de proportionnalité utilisé afin de diminuer ou d'augmenter l'interaction avec le disque si nécessaire.

Voici un exemple de fichier migrate.in :

```
3
1047.355d0 200.d0
0.05d0 30.00d0 0.001d0
2.8d5 0.5 1
8.d5 1
```

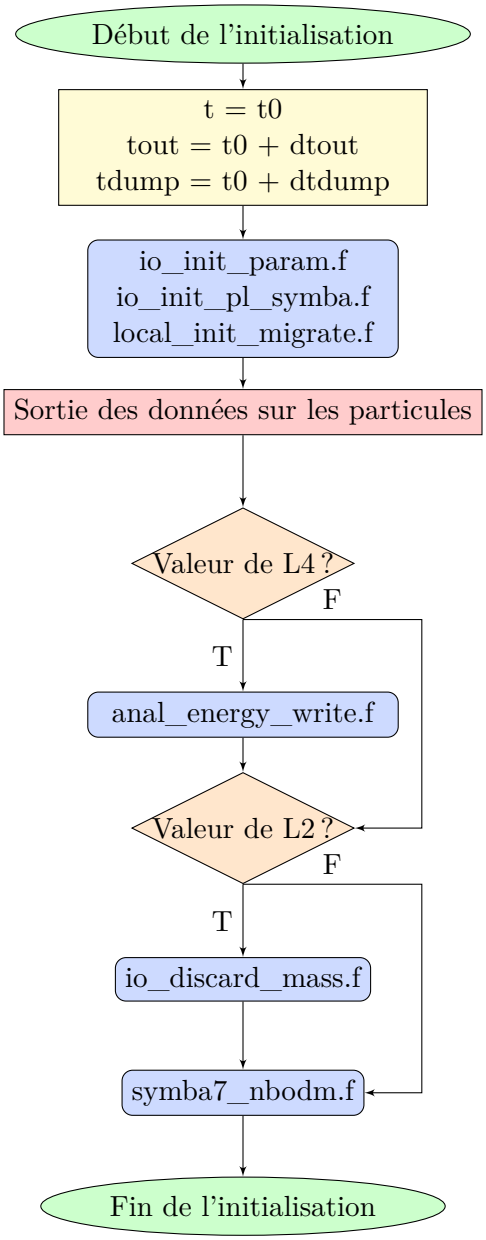
5.2.2 Organigramme

Nous allons à présent nous intéresser à l'organigramme du code incluant l'influence du disque de gaz. Notons que nous effectuons ce diagramme pour un disque se dissipant progressivement, qui est le modèle que nous avons adopté pour nos simulations. De plus, notons que nous présentons le diagramme où le disque est encore présent. Quand $|\dot{m}_{disk}(t)|$ sera inférieur au seuil critique M_{crit}, le code réutilisera l'organigramme de la section 4.1.4.

L'organigramme est similaire à celui expliqué à la section 4.1.4 mis à part pour les diagrammes "Intégration" et "Initialisation".

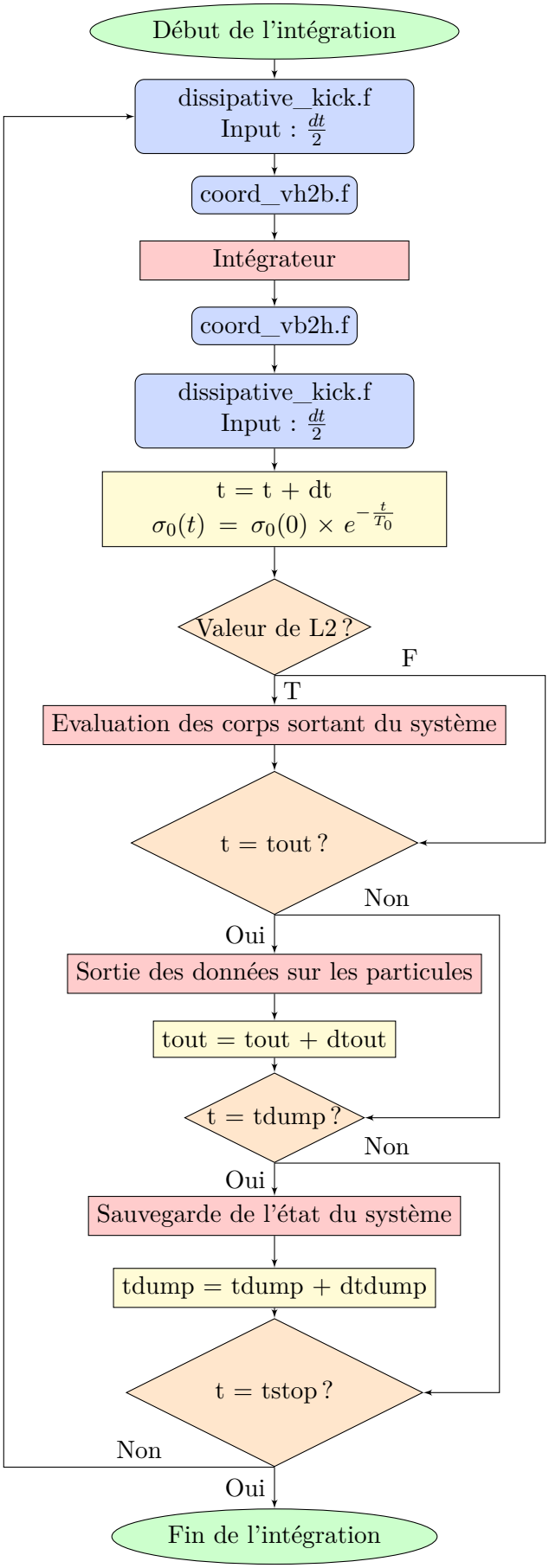
Initialisation

Ce diagramme est une version modifiée de celui de la section 2.4.1.



Intégration

Ce diagramme est une version modifiée de celui de la section 2.4.2.



5.3 Adaptation aux systèmes binaires

5.3.1 Code

Afin d'étudier la migration de type II dans les systèmes binaires, nous avons repris la théorie détaillée dans [47]. En effet, le compagnon stellaire étant éloigné, nous ne tenons pas compte, dans cette étude, de son influence sur le disque. L'influence du disque sur les planètes sera donc similaire à celle présentée dans [47].

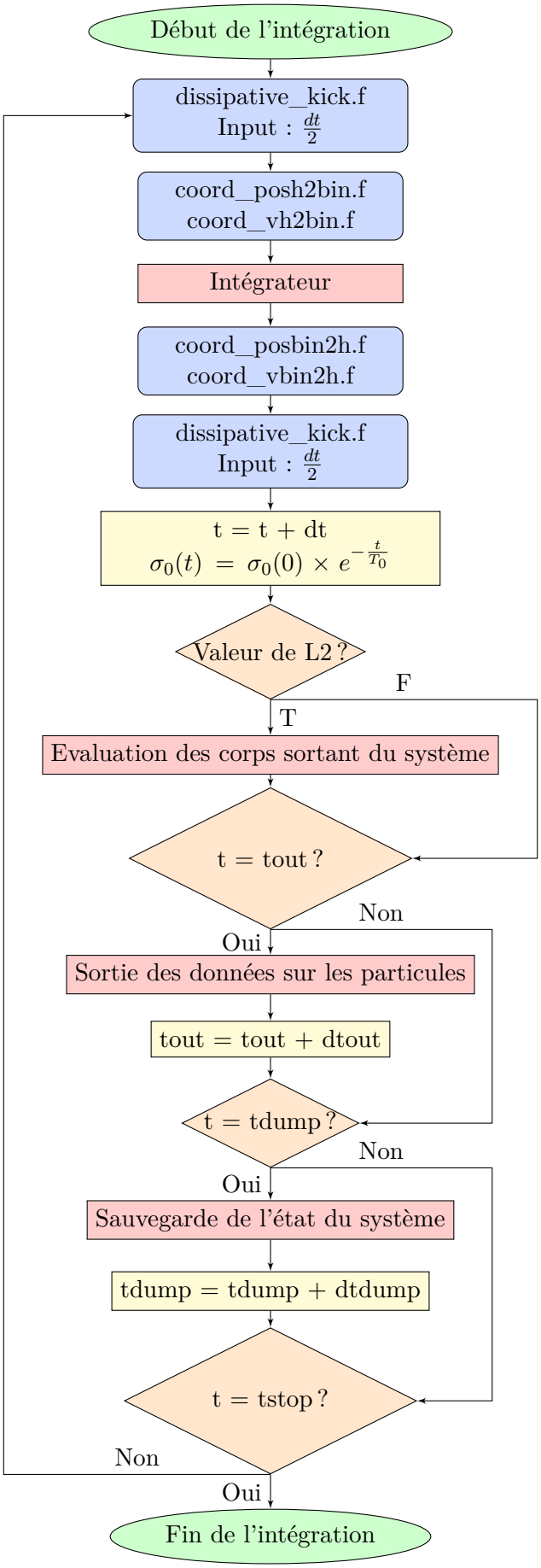
Nous avons donc incorporé le code de S. Sotiriadis dans notre intégrateur pour les systèmes binaires (avec rencontres proches et rencontres avec l'étoile) en tenant compte de la structure de notre programme. Nous avons gardé la fonction *local_init_migrate.f* inchangée et nous avons adapté la fonction *dissipative_kick.f* en la renommant *dissipative_kick_bin.f*. Cette dernière est disponible à l'annexe A.3.

5.3.2 Organigramme

Au niveau de l'organigramme du code, il s'agit de l'organigramme présenté à la section 4.2.4 en remplaçant le diagramme "Initialisation" par celui de la section précédente en modifiant le diagramme "Intégration".

Intégration

Le nouveau diagramme "Intégration" est une modification de celui de la section 3.3.2.



Planète	Masse	a (UA)	e	i (°)	Ω (°)	ω (°)	M (°)
1	0.5 MJ	5	0.1	10	0.01	0.01	0.01
2	2 MJ	10.5	0.02	0.01	0.01	0.01	0.01
3	1 MJ	25	0.05	1	0.01	0.01	0.01

TABLE 5.1 – Conditions initiales du système étudié pour la validation du code.

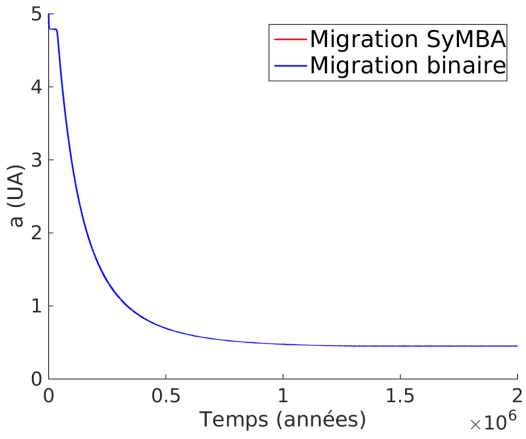
5.3.3 Validation

Nous allons valider, dans cette section, l'influence du disque de gaz sur les planètes. Or, celle-ci est considérée comme indépendante du compagnon binaire. Afin de valider cette interaction, nous avons donc considéré un compagnon stellaire de masse nulle et comparé nos résultats avec ceux obtenus avec le code de S. Sotiriadis [47]. En l'absence de compagnon stellaire, notre code se réduit, en effet, à celui de S. Sotiriadis puisque les coordonnées binaire éloignée (section 3.1) se réduisent aux coordonnées héliocentriques canoniques (section 2.1.3) et les différents morceaux de l'hamiltonien évoqués à la section 2.2.6 se réduisent à ceux de la section 3.2.

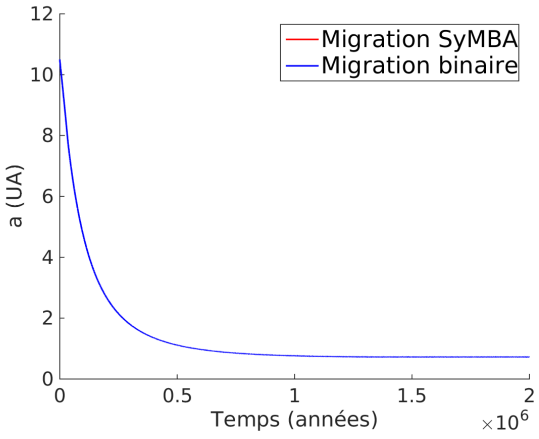
Deux systèmes ont été étudiés, l'un contenant trois planètes et l'autre seulement deux. Les conditions initiales du système à trois planètes sont reprises à la table 5.1. Le systèmes à deux planètes ne contient quant à lui que les planètes 1 et 2 de la table 5.1. De plus, nous avons utilisé une étoile centrale de 1 masse solaire.

Nous pouvons constater de la figure 5.2 à la figure 5.13 que les deux codes donnent bien les mêmes résultats. Nous avons également calculé la différence maximale entre les résultats des deux codes pour chaque élément orbital considéré avec quatre décimales et nous avons trouvé la valeur zéro. Les résultats sont donc identiques. Nous pouvons donc conclure que l'interaction avec le disque a été bien simulée dans notre code.

Notons que nous n'étudions pas la conservation de l'hamiltonien ni du moment angulaire total car le système constitué des corps sans le disque n'est pas conservatif suite aux échanges avec le disque.

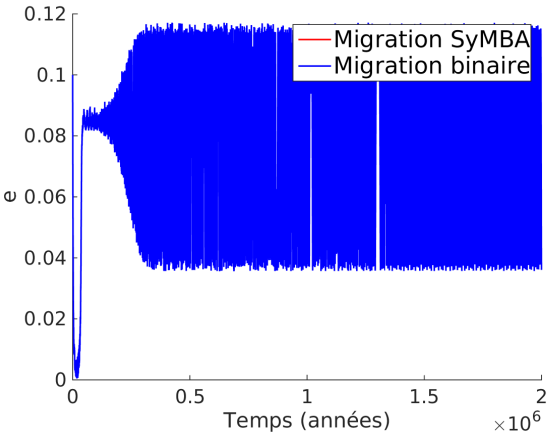


Planète 1

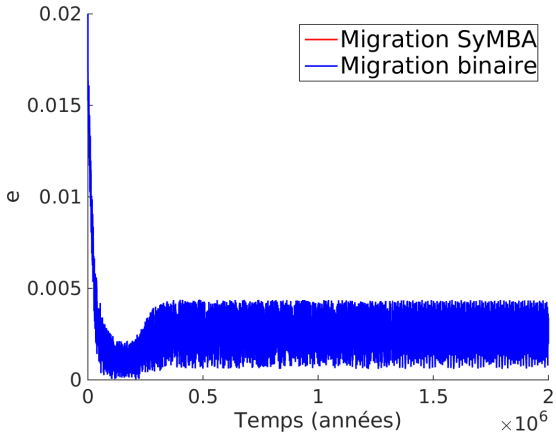


Planète 2

FIGURE 5.2 – Evolutions des demi-grands axes du système formé des planètes 1 et 2 de la table 5.1 lors de la migration de la planète extérieure.

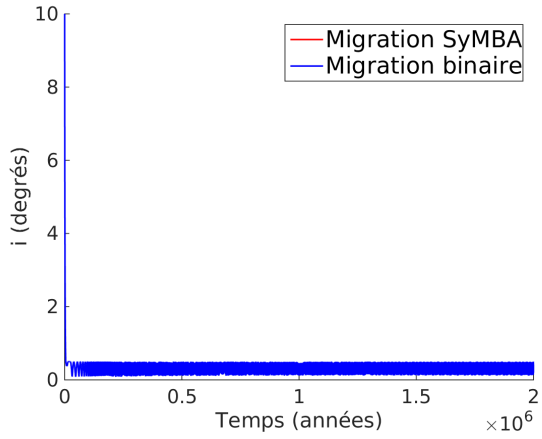


Planète 1

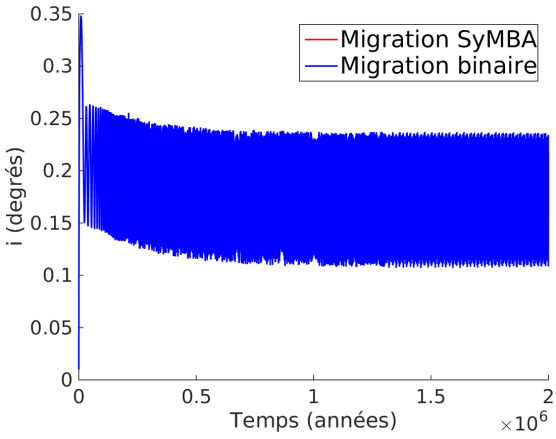


Planète 2

FIGURE 5.3 – Mêmes évolutions pour les excentricités.

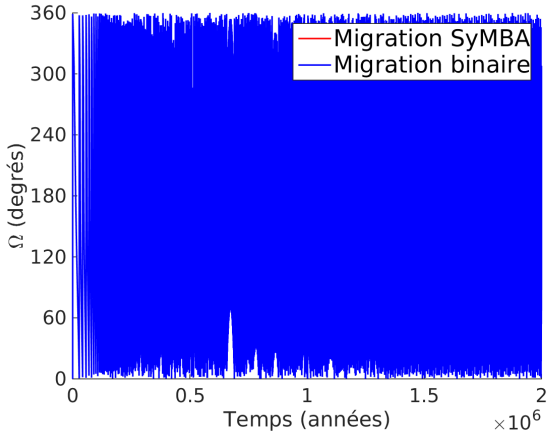


Planète 1

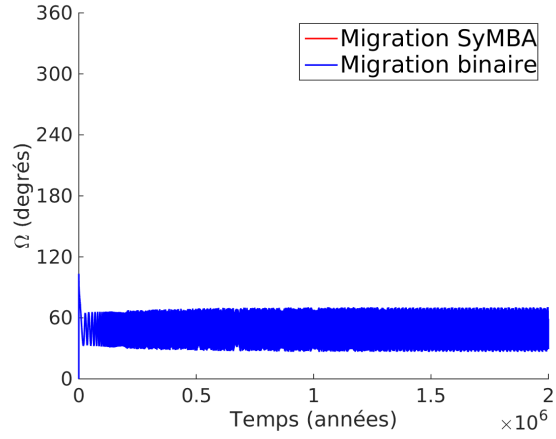


Planète 2

FIGURE 5.4 – Mêmes évolutions pour les inclinaisons.

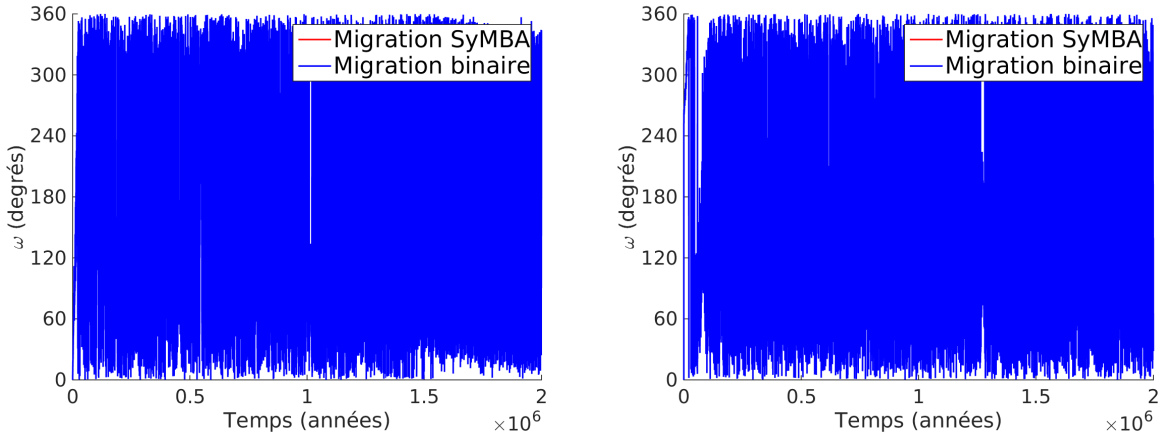


Planète 1



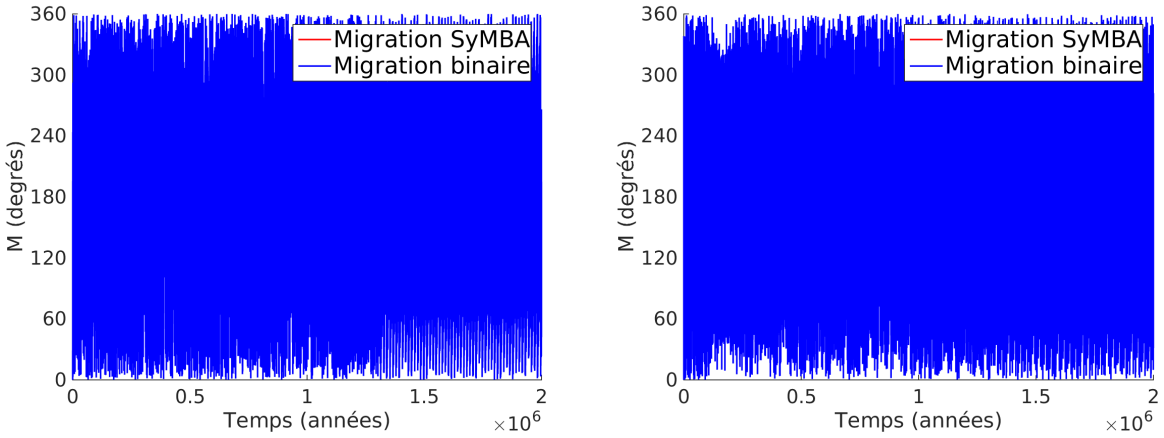
Planète 2

FIGURE 5.5 – Mêmes évolutions pour les longitudes des nœuds ascendants.



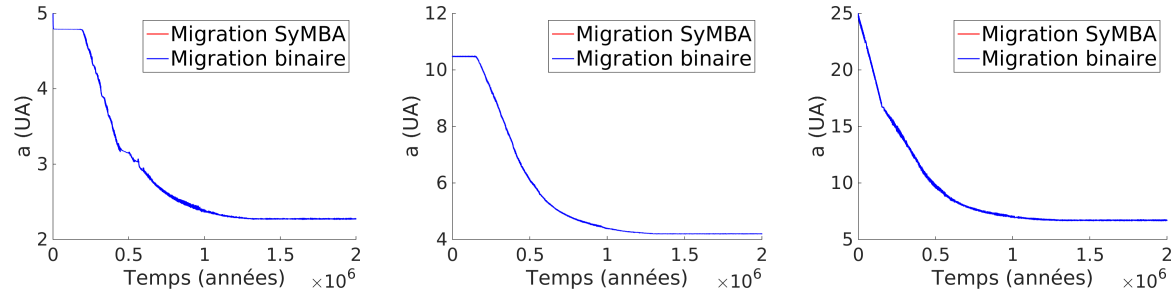
Planète 1 Planète 2

FIGURE 5.6 – Mêmes évolutions pour les arguments des périastres.



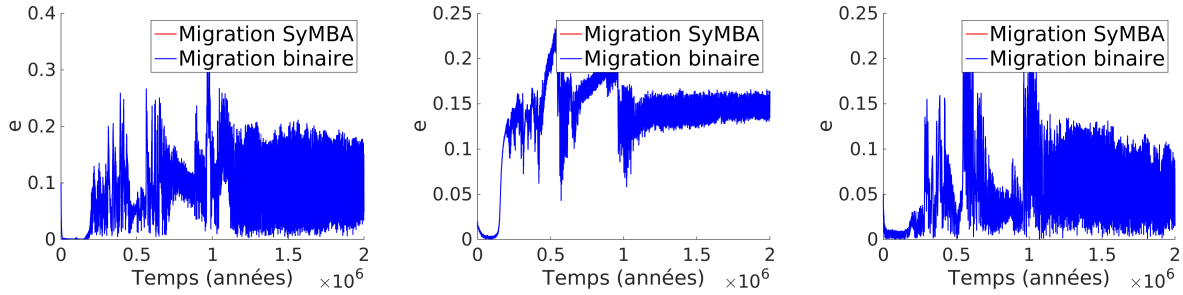
Planète 1 Planète 2

FIGURE 5.7 – Mêmes évolutions pour les anomalies moyennes.



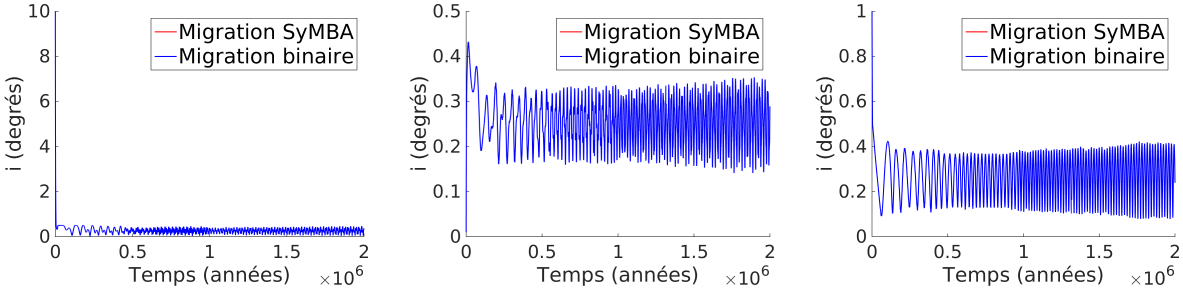
Planète 1 Planète 2 Planète 3

FIGURE 5.8 – Evolutions des demi-grands axes du système de la table 5.1 lors de la migration de la planète extérieure.



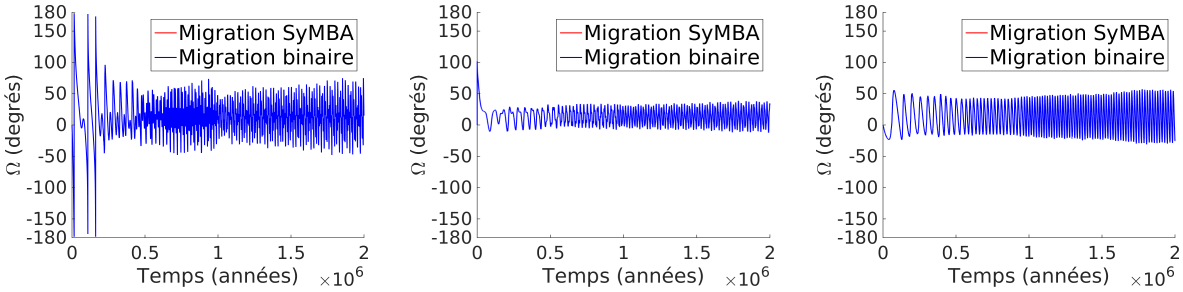
Planète 1 Planète 2 Planète 3

FIGURE 5.9 – Mêmes évolutions pour les excentricités.



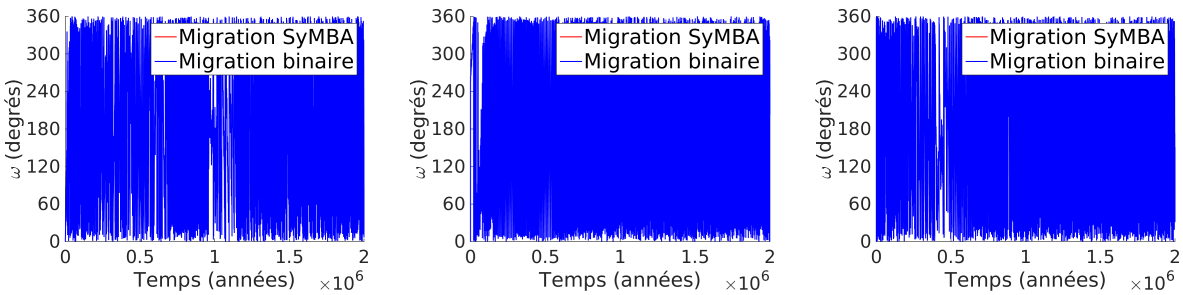
Planète 1 Planète 2 Planète 3

FIGURE 5.10 – Mêmes évolutions pour les inclinaisons.



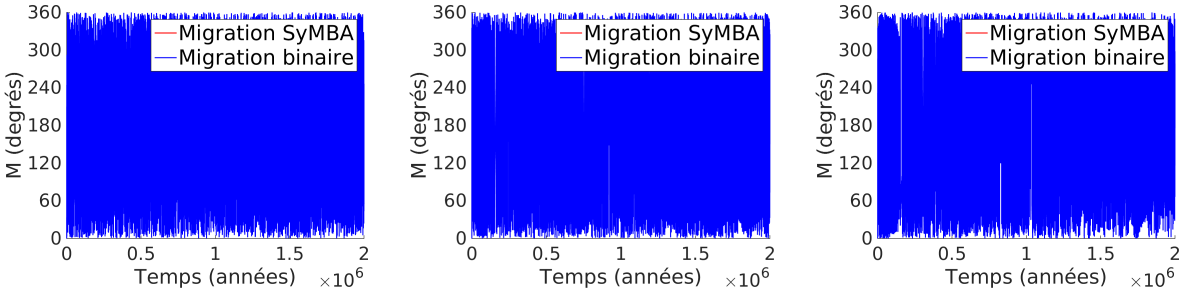
Planète 1 Planète 2 Planète 3

FIGURE 5.11 – Mêmes évolutions pour les longitudes des nœuds ascendants.



Planète 1 Planète 2 Planète 3

FIGURE 5.12 – Mêmes évolutions pour les arguments des périastres.



Planète 1 Planète 2 Planète 3

FIGURE 5.13 – Mêmes évolutions pour les anomalies moyennes.

Chapitre 6

Simulations

Le code décrit au chapitre précédent nous a servi à généraliser l'étude de Libert & Tsiganis [29] aux systèmes binaires. Pour ce faire, nous avons sélectionné quinze systèmes provenant de [29] auxquels nous avons ajouté un compagnon stellaire.

Dans ce chapitre, nous allons présenter quelques évolutions typiques observées dans nos simulations. Nous allons également décrire les phénomènes dynamiques entrant en jeu dans celles-ci. Nous terminerons ce travail en comparant nos résultats avec ceux de Libert & Tsiganis [29] et ceux de Sotiriadis et al. [47] pour les systèmes à une étoile.

6.1 Conditions initiales des systèmes planétaires

Commençons par détailler notre choix de conditions initiales des systèmes formés de deux planètes évoluant dans une étoile binaire. Comme indiqué précédemment, nos configurations initiales sont dérivées de celles de [29]. Notons que, comme précédemment, nous indiçons A l'étoile centrale, B le compagnon stellaire et 1 et 2 les planètes.

La masse de l'étoile centrale est fixée à $m_A = 1$ masse solaire ($M_\odot$). Pour les planètes, nous avons choisi comme conditions initiales, les demi-grands axes $a_1 = 5$ UA et $a_2 = 16.5$ UA, les trois configurations d'excentricités : $e_1 = e_2 = 0.001$, $e_1 = 0.1$ et $e_2 = 0.001$ et $e_1 = 0.001$ et $e_2 = 0.1$ ainsi que les cinq rapports de masses $\frac{m_1}{m_2} = 0.25, 0.5, 1, 2$ et 4. Tous les autres paramètres orbitaux initiaux des planètes ont été fixés à 10^{-5}. Ces conditions initiales sont identiques à celles de [29]. Ayant sélectionné trois configurations initiales d'excentricités et cinq configurations initiales de rapport de masses, nous avons bien sélectionné 15 configurations initiales différentes.

Concernant le compagnon stellaire, nous avons utilisé, comme paramètres initiaux, les demi-grands axes 500 et 1000 UA, les excentricités $10^{-5}, 0.3$ et 0.6, les inclinaisons 10^{-5}°, 30°, 45°, 60° et 90° ainsi que les masses 0.5, 1 et 2 $M_\odot$. Les autres paramètres initiaux du compagnon stellaire ont été fixés à 10^{-5}. Nous avons donc choisi 90 configurations initiales différentes pour le compagnon stellaire.

En combinant les configurations initiales pour les planètes et celles pour le compagnon stellaire, nous avons donc fait évoluer $90 \times 15 = 1350$ systèmes initiaux différents.

Notons également que pour ces évolutions, nous avons utilisé le fichier migrate.in suivant :

```
3
1047.355d0 200.d0
0.05d0 30.00d0 0.001d0
2.8d5 0.5 1
8.d5 1
```

Les paramètres contenus dans ce fichier correspondent à une dissipation progressive d'un disque de gaz ayant une masse initiale de 16 masses de Jupiter. De plus, ce disque a une durée de vie de 1.3×10^6 ans.

Les simulations ont été effectuées avec un pas de 0.01 an pour un temps final de 2×10^6 ans.

6.2 Exemples d'évolutions

Nous allons à présent présenter quelques évolutions rencontrées lors de nos simulations de systèmes binaires. Comme dans [47] et [29], nous allons nous focaliser sur le demi-grand axe, l'excentricité et l'inclinaison de chaque planète, ainsi que sur les angles résonants.

Un premier système est illustré à la figure 6.1. Celui-ci permet d'observer un phénomène de résonance connu sous le nom de résonance en moyen mouvement ([27]). Il se produit, pour deux planètes, lorsqu'une planète parcourt j fois son orbite dans le même temps que l'autre planète la parcourt k fois, où j et k sont des entiers faibles avec $pgcd(j,k) = 1$. Une autre manière d'exprimer cette relation est par le rapport des périodes de planètes. En effet, celui-ci sera tel que $\frac{T_1}{T_2} = \frac{j}{k}$, où T_1 et T_2 sont les périodes de la première et de la deuxième planète respectivement. Notons qu'en utilisant la troisième loi de Kepler, le rapport des périodes se réécrit :

$$\frac{T_1}{T_2} = \frac{\sqrt{\frac{4\pi^2 a_1^3}{G m_A}}}{\sqrt{\frac{4\pi^2 a_2^3}{G m_A}}} = \left(\frac{a_1}{a_2}\right)^{\frac{3}{2}} = \frac{j}{k}.$$

Nous parlons alors de résonance $j : k$. De plus, nous parlons de capture en résonance lors de l'instauration d'une résonance.

Une autre approche pour l'étude des résonances est montrée dans [29] et consiste à étudier les angles résonants :

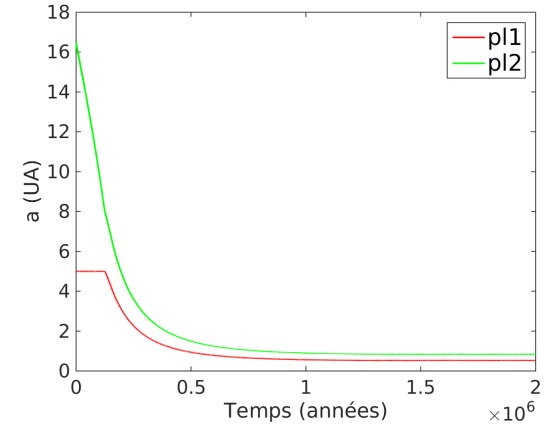
$$\theta_l = j\lambda_1 - k\lambda_2 - (j-k)\overline{\omega}_l \quad l \in \{1,2\},$$

avec $\lambda_i = \omega_i + \Omega_i + M_i$ et $\overline{\omega}_i = \omega_i + \Omega_i$, $i \in \{1,2\}$ où ω_i, Ω_i et M_i sont respectivement l'argument du périastre, la longitude du noeud ascendant et l'anomalie moyenne de la planète i. Deux planètes seront alors considérées comme étant en résonance en moyen mouvement si au moins un de ces angles est en libration.

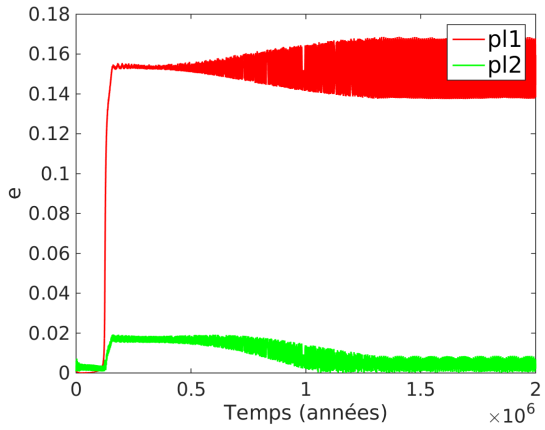
A la figure 6.2, nous avons étudié le rapport de périodes ainsi que les angles résonants pour le système de la figure 6.1. Nous constatons une stabilisation du rapport de périodes à 2, nous sommes donc en présence d'une résonance 2 : 1. Celle-ci est confirmée par les angles résonants qui sont en libration autour de 0°, l'un jusqu'à la fin de la simulation et l'autre jusqu'à 10^6 ans. Notons que cette résonance s'installe après environ 10^5 ans de migration.

Nous constatons la migration de la planète extérieure, induit par le disque de gaz. Lors de celle-ci, le système est capturé en résonance 2 : 1. Une fois la capture effectuée, le système va tendre à rester en résonance. Cette tendance induit la migration de la planète intérieure. En effet, comme précisé précédemment, la migration due au disque n'a été appliquée qu'à la planète 1 dans le code, celle-ci ne vient donc pas directement du disque. De même comme montré dans [47], la conservation de cette résonance par le système tout en poursuivant sa migration va induire une augmentation de l'excentricité des planètes.

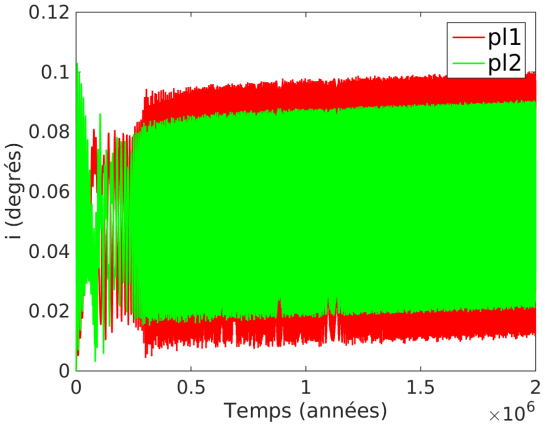
Notons que l'augmentation de l'excentricité est due à la relation existant entre l'excentricité et le demi-grand axe des corps en cas de résonance comme montré dans [26].



Demi-grands axes

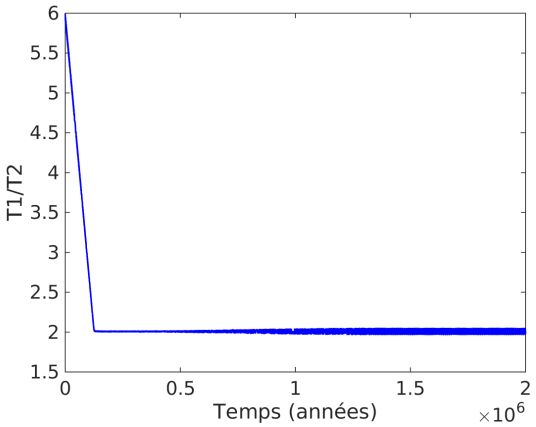


Excentricités

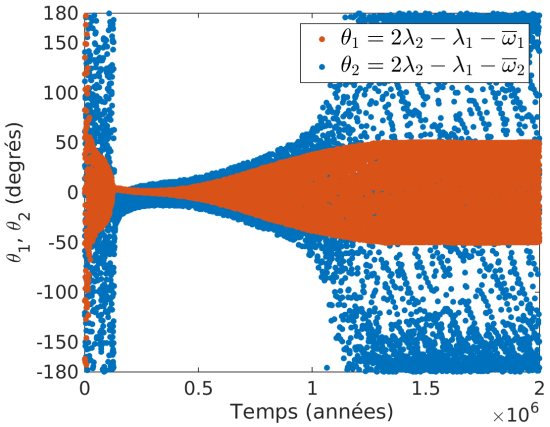


Inclinaisons

FIGURE 6.1 – Evolution du système binaire de conditions initiales $m_B = 2M_\odot$, $a_B = 1000$ UA, $e_B = 0.0$, $i_B = 90°$, $\frac{m_1}{m_2} = 1$ et $e_1 = e_2 = 0.001$

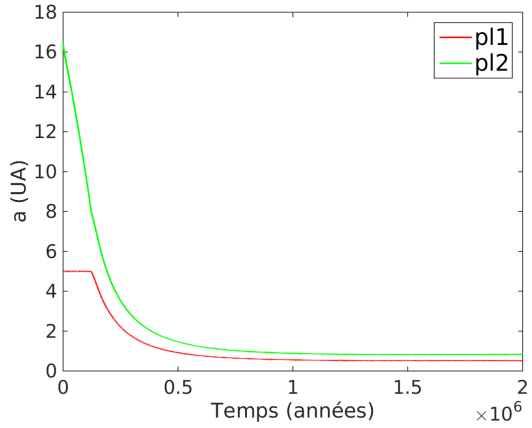


Rapport de périodes

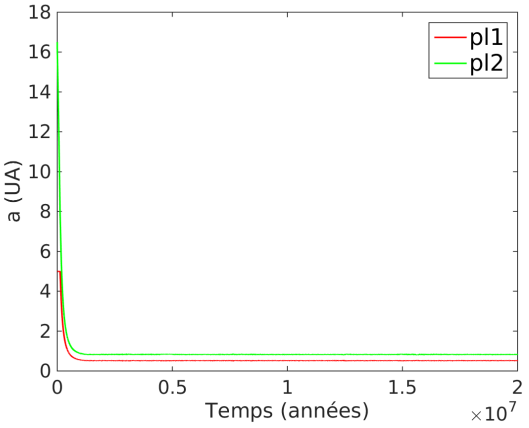


Angles résonants

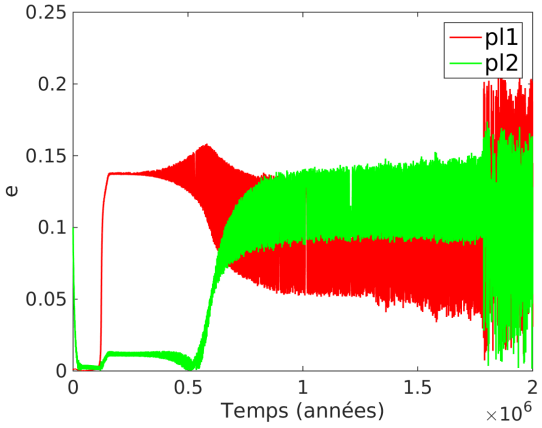
FIGURE 6.2 – Etude de la résonance du système de la figure 6.1



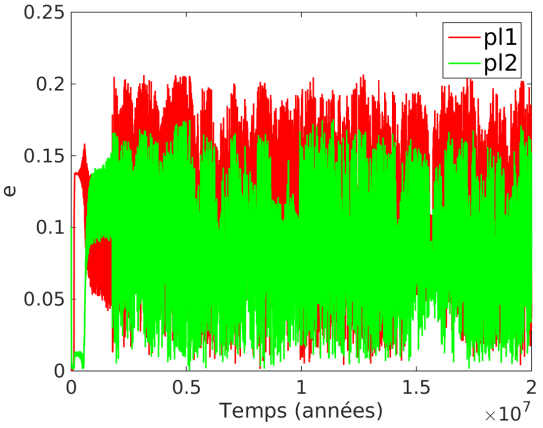
Demi-grands axes



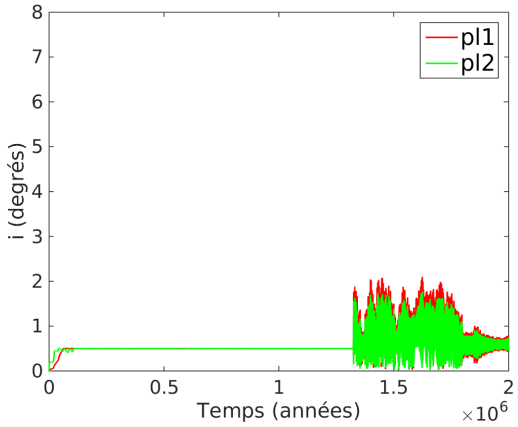
Demi-grands axes



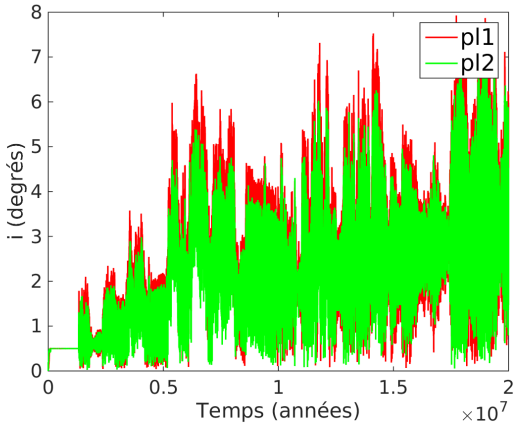
Excentricités



Excentricités



Inclinaisons



Inclinaisons

FIGURE 6.3 – Evolution du système binaire de conditions initiales $m_B = 1M_\odot$, $a_B = 1000$ UA, $e_B = 0.6$, $i_B = 60°$, $\frac{m_1}{m_2} = 1$, $e_1 = 0.001$ et $e_2 = 0.1$

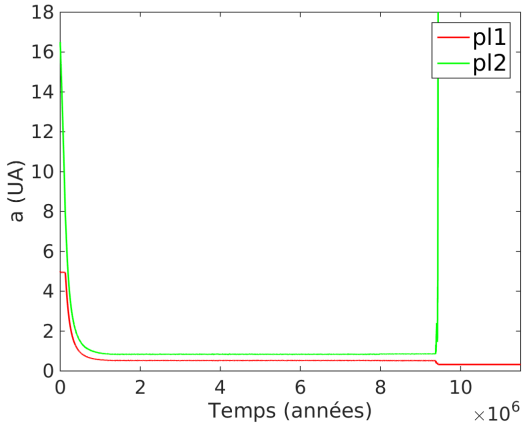
FIGURE 6.4 – Système de la figure 6.3 prolongé sur une plus grande période de temps (2×10^7 ans).

Afin d'avoir également une augmentation des inclinaisons des planètes en résonance, [29] a montré qu'il faut une excentricité relativement élevée (de l'ordre de 0.4), ce qui n'est pas le cas dans ce système. Ceci explique l'absence d'augmentation de l'inclinaison. Notons également que le système semble stable à la fin de l'intégration.

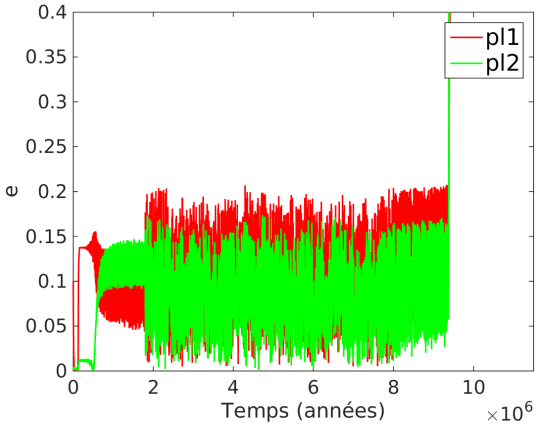
Il n'en va pas de même pour le système de la figure 6.3 qui ne semble pas avoir adopté un comportement régulier à la fin de son intégration. Pour de tels systèmes, nous avons prolongé l'intégration jusqu'à un temps final de 2×10^7 ans afin d'observer leur comportement, comme illustré à la figure 6.4. Soixante-huit systèmes parmi les 1350 systèmes étudiés ont ainsi été prolongés.

Dans certains systèmes, une des planètes a été éjectée du système. Une évolution avec un tel phénomène est illustrée à la figure 6.5. Notons que ces éjections ne se sont jamais produites durant la phase de migration. Elles sont toujours survenues pour des systèmes dont l'évolution a été prolongée. Nous avons observé dix configurations initiales entraînant l'éjection d'une planète.

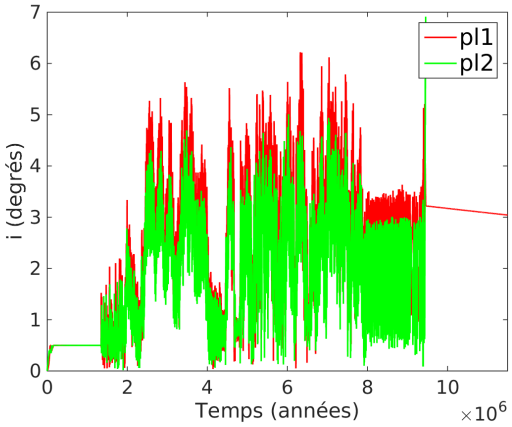
Nous allons à présent discuter un dernier cas. Lorsque le compagnon stellaire a une inclinaison initiale importantes (30°, 45° ou 60°), nous avons constaté une augmentation linéaire de l'inclinaison des planètes à la fin de l'évolution dans plus de 95% des cas. Cette augmentation est illustrée à la figure 6.6. Nous avons donc prolongé ces simulations jusqu'à 2×10^8 ans afin d'observer l'évolution de ces inclinaisons. Nous avons alors observé une oscillation de l'inclinaison des planètes autour d'une valeur moyenne correspondant à l'inclinaison du compagnon stellaire, comme nous pouvons l'observer à la figure 6.7. Plus particulièrement, l'oscillation est comprise entre 0° et $2i_B$° pour les planètes. En voyant les grandes variations de l'inclinaison, nous avons soupçonné une résonance de Kozai telle que décrite dans [22]. En effet, nous sommes en présence de deux corps massifs (les étoiles) et d'un corps de masse négligeable en comparaison à celle des étoiles (la planète). Cependant, le phénomène a été observé pour une inclinaison du compagnon binaire de 30°. Or la résonance de Kozai n'apparaît qu'au-delà de 39.2° d'inclinaison. De plus, en cas de résonance de Kozai, l'augmentation d'inclinaison coïncide avec une diminution d'excentricité comme le précise [2]. Cette relation entre l'excentricité et l'inclinaison est illustrée à la figure 6.8. Néanmoins, comme nous pouvons l'observer à la figure 6.9, les excentricités restent relativement constantes dans notre cas et ne sont pas couplées aux variations des inclinaisons.



Demi-grands axes

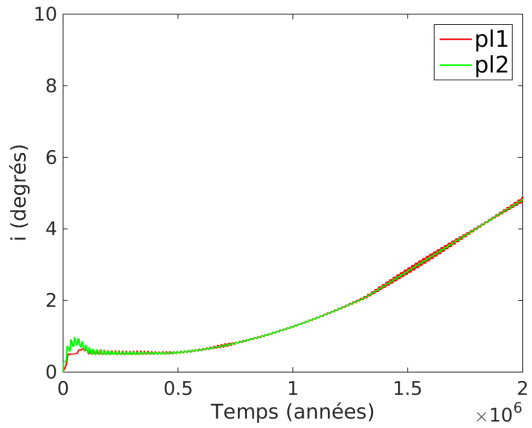


Excentricités

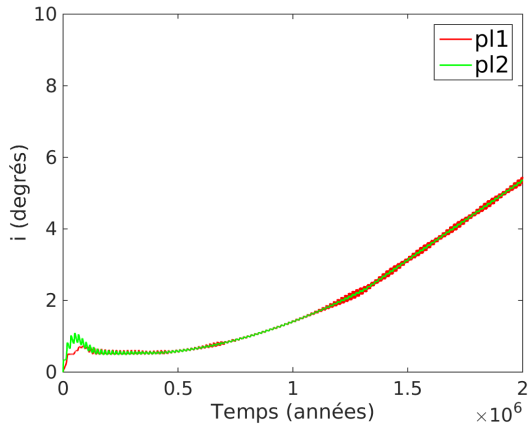


Inclinaisons

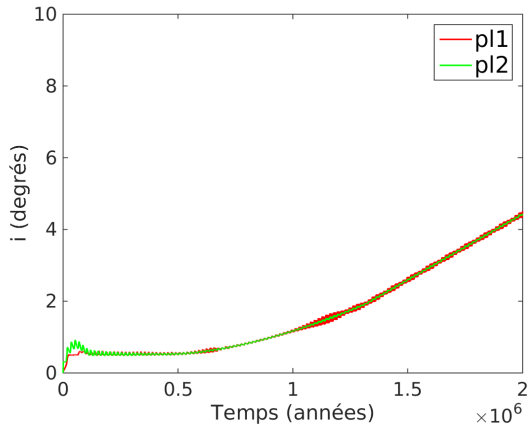
FIGURE 6.5 – Evolution du système binaire de conditions initiales $m_B = 1M_\odot$, $a_B = 1000$ UA, $e_B = 0.3$, $i_B = 30°$, $\frac{m_1}{m_2} = 1$, $e_1 = 0.1$ et $e_2 = 0.001$



$i_B = 30°$

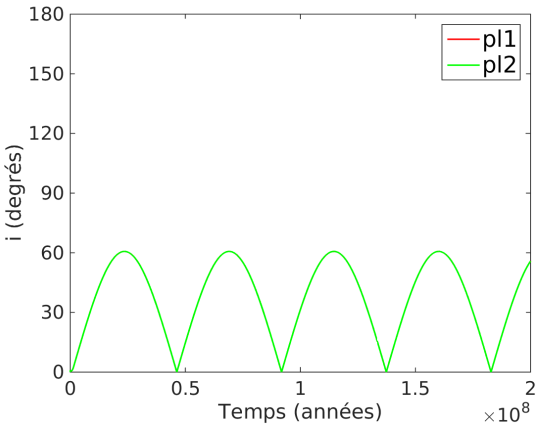


$i_B = 45°$

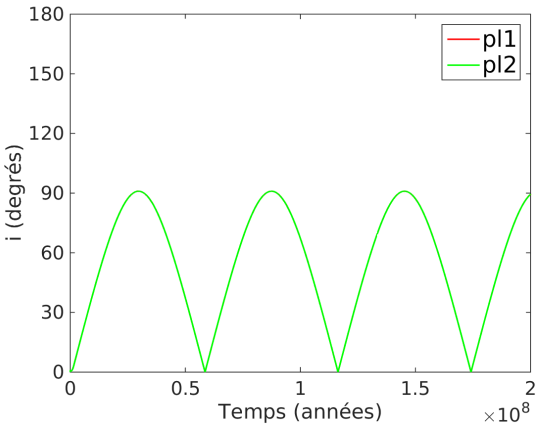


$i_B = 60°$

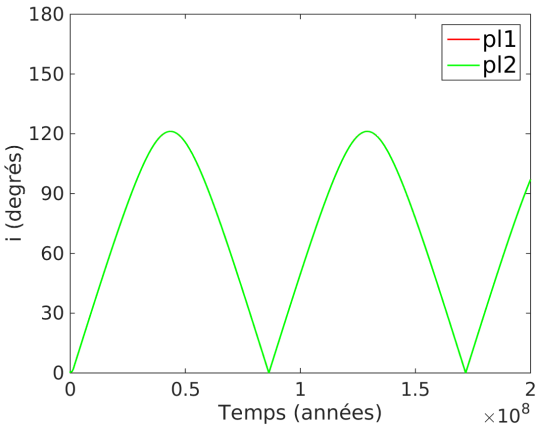
FIGURE 6.6 – Inclinaisons des systèmes binaires de conditions initiales $m_B = 2M_\odot$, $a_B = 1000$ UA, $e_B = 0.6$, $\frac{m_1}{m_2} = 0.25$, $e_1 = e_2 = 0.001$ en prenant en compte différentes inclinaisons pour le compagnon stellaire



$i_B = 30°$



$i_B = 45°$



$i_B = 60°$

FIGURE 6.7 – Systèmes de la figure 6.6 prolongé sur 2×10^8 ans

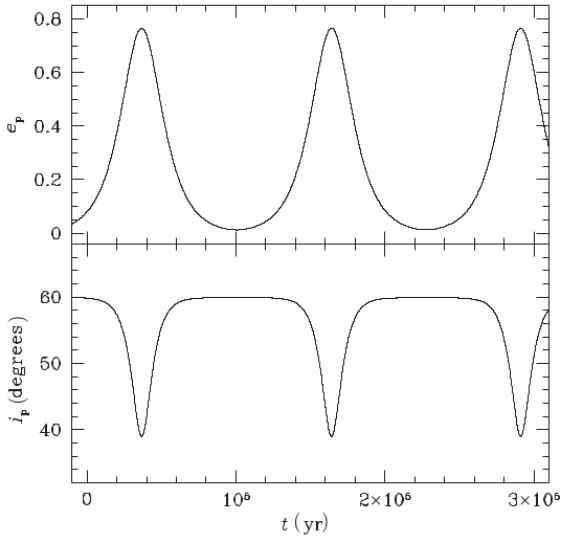
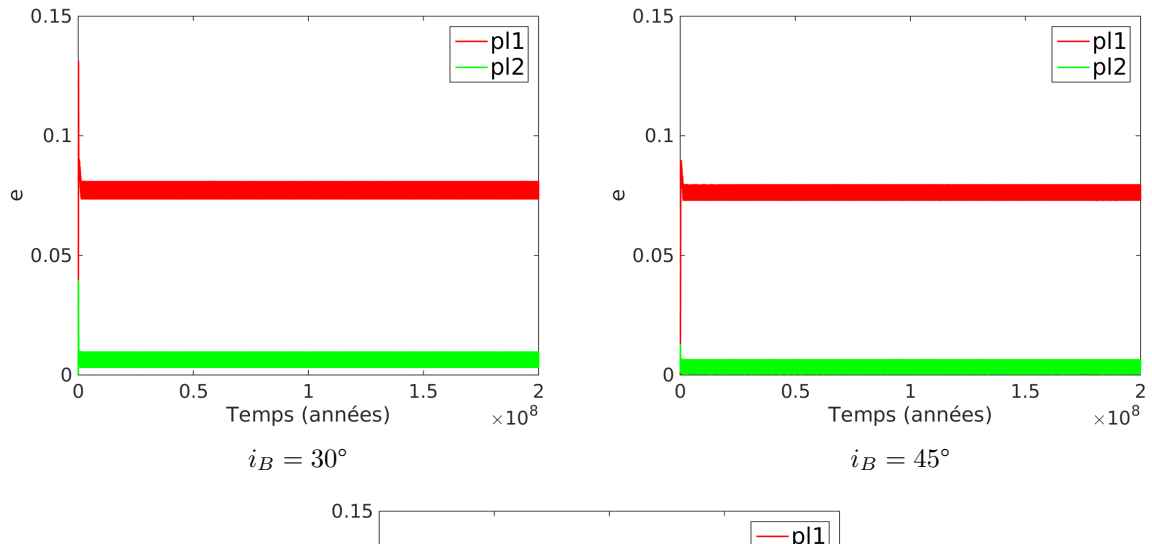
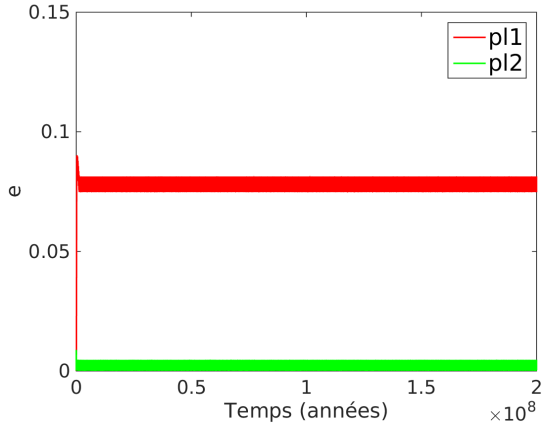


FIGURE 6.8 – Illustration des oscillations couplées de l'inclinaison et de l'excentricité lors d'une résonance de Kozai. Figure issue de [2]



$i_B = 30°$ $i_B = 45°$



$i_B = 60°$

FIGURE 6.9 – Excentricités des systèmes de la figure 6.6

Résonance	Nombre de captures	Pourcentage observé
3 : 2	2	13.3%
2 : 1	11	73.3%
5 : 2	1	6.7%
3 : 1	1	6.7%

TABLE 6.1 – Résonances observées dans les simulations des systèmes sans compagnon stellaire

6.3 Analyse des résultats

Dans un premier temps, nous avons intégré les 15 configurations initiales de planètes sans compagnon stellaire afin de comparer avec les résultats de [29]. En effet, comme explicité à la section 5.1, le modèle d'amortissement des excentricités utilisé dans notre code (formules basées sur des simulations hydrodynamiques) diffère de celui utilisé dans [29] (formule avec paramètre K). De plus, [29] ne prend pas en compte l'amortissement des inclinaisons des planètes. Avant d'ajouter le compagnon stellaire, il nous paraît donc pertinent d'analyser les systèmes sans celui-ci.

Tout d'abord, au niveau des résonances, nous avons observé une majorité de capture en résonance 2 : 1 avec notre code, comme le montre la table 6.1. Les résonances observées dans [29] étaient plus diversifiées. Néanmoins, la comparaison n'est pas aisée. En effet, dans le modèle utilisé par [29], plusieurs valeurs du paramètre K de la formule (5.1.10) sont utilisées et les résonances observées diffèrent selon la valeur du paramètre. Le paramètre K n'étant pas présent dans notre code, la comparaison s'avère difficile. Cependant, nos résultats semblent cohérents avec ceux de [47].

La figure 6.10 montre les valeurs des excentricités à la fin de l'intégration. Celles-ci sont toutes comprises dans l'intervalle $[0, 0.35]$. Ces valeurs sont en accord avec celles de [29] lorsque l'on considère une valeur importante du paramètre K. Ceci semble cohérent avec les propos de [47] qui indiquent que les taux d'amortissement des excentricités et inclinaisons varient au cours du temps dans un rapport avec un taux de migration de l'ordre de K compris entre 1 et 10.

Au niveau de l'inclinaison des corps, nous n'avons constaté aucune augmentation de celle-ci, ce qui s'explique par les valeurs trop faibles des excentricités. En effet, comme le précise [29], l'inclinaison des corps augmente une fois une excentricité suffisante atteinte (de l'ordre de 0.4). Or comme le montre le graphe 6.10, nous n'atteindrons ce seuil dans aucune de nos simulations et n'avons donc pas d'augmentation de l'inclinaison. Cette absence d'augmentation pour l'inclinaison était aussi observée dans [29] pour des choix de K égal à 5 ou 10.

Afin de quantifier l'influence du compagnon stellaire, nous allons à présent comparer nos résultats à ceux de [47]. En effet, nous avons utilisé le même modèle en ajoutant un compagnon stellaire. Il nous paraît donc intéressant de comparer nos résultats. Notons néanmoins que dans notre cas nous étudions des systèmes ayant initialement deux planètes tandis que [47] travaille avec des systèmes contenant initialement trois planètes. Le cas avec trois planètes est un prolongement possible de ce travail.

Tout d'abord au niveau de la répartition finale des demi-grands axes représentée à la figure 6.11, nous constatons une allure similaire des résultats. Cependant nous observons un pic plus important pour l'intervalle $[10^{0.2}, 10^{0.4}]$ avec notre code. Ce pic peut s'expliquer par notre choix restreint des valeurs initiales des demi-grands axes des planètes. Il serait intéressant d'étendre notre étude en les faisant varier.

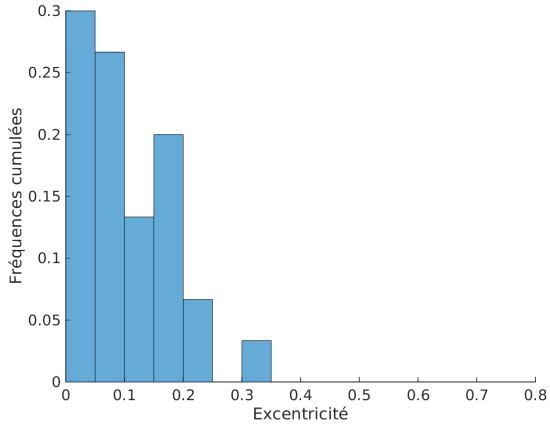


FIGURE 6.10 – Répartition des excentricités observées en l'absence de compagnon stellaire

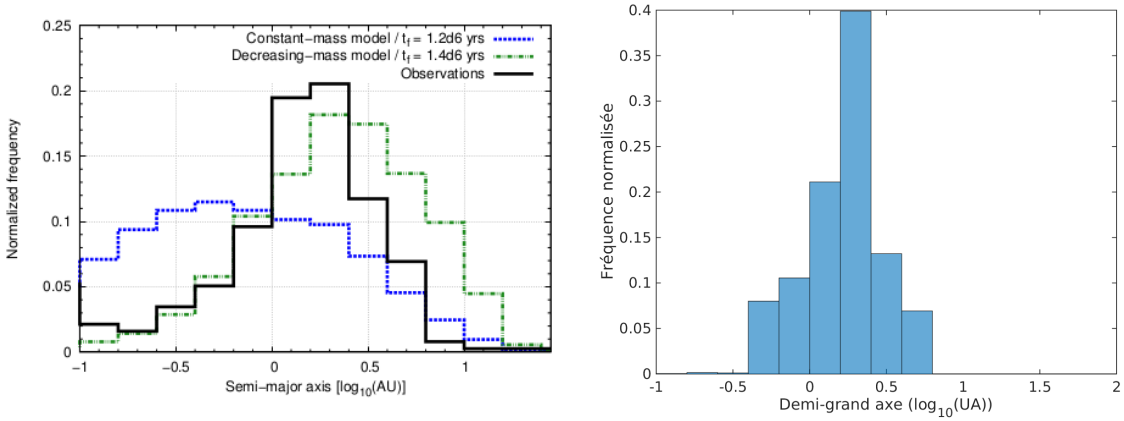


FIGURE 6.11 – Répartition des demi-grands axes. A gauche : figure issue de [47]. A droite : notre étude.

Ensuite, nous avons comparé les fréquences cumulées des excentricités finales à la figure 6.12. Nous constatons, dans notre cas, des excentricités plus faibles ce qui s'explique par le fait que nos simulations ne contiennent que deux planètes et non trois. Cela réduit donc considérablement les interactions possibles entre planètes. Cependant, un premier résultat de notre étude est le fait que le compagnon stellaire semble ne pas exciter les excentricités des planètes.

Enfin, nous avons comparé la distribution des inclinaisons à la figure 6.13. Nous obtenons la même allure que celle de [47]. Cependant ces graphiques ne reprennent que les inclinaisons supérieures à 1° et nous observons, de manière générale, un pic à faible inclinaison important pour nos simulations. Cette différence proviendrait de l'influence qu'exercent les compagnons stellaires possédant une forte inclinaison. En effet, nous avons constaté précédemment, que ceux-ci pouvaient augmenter l'inclinaison sur des grandes échelles de temps, comme illustré à la figure 6.7. Ils pourraient donc avoir la même influence durant la migration dans certaines configurations précises.

Afin de statuer plus précisément sur l'influence du compagnon stellaire, nous pouvons comparer nos résultats obtenus avec ou sans compagnon stellaire. Tout d'abord, au niveau du demi-grand axe, nous constatons à la figure 6.14 des profils similaires. Néanmoins, la présence d'un compagnon stellaire semble augmenter le demi-grand axe des planètes. Notons que ce résultat est à prendre avec des pincettes de par le faible nombre de systèmes traités sans compagnon stellaire.

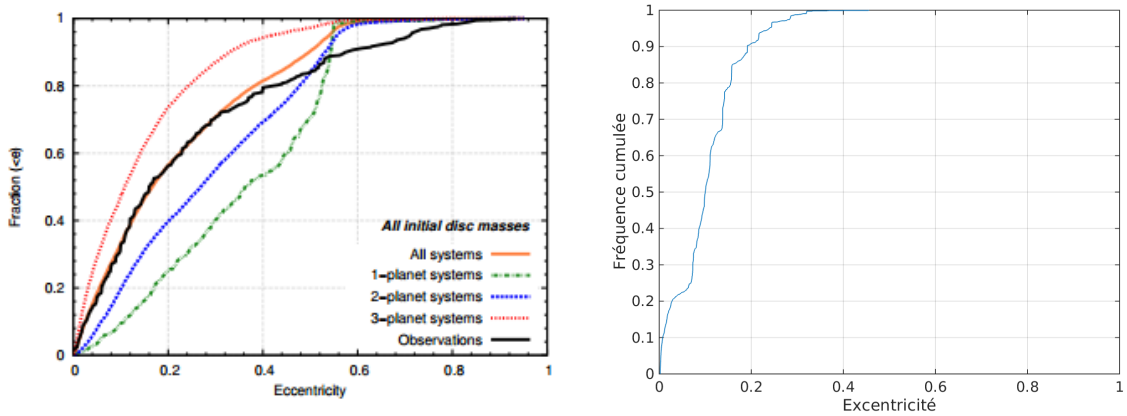


FIGURE 6.12 – Fréquences cumulées d'excentricités. A gauche : figure issue de [47]. A droite : notre étude.

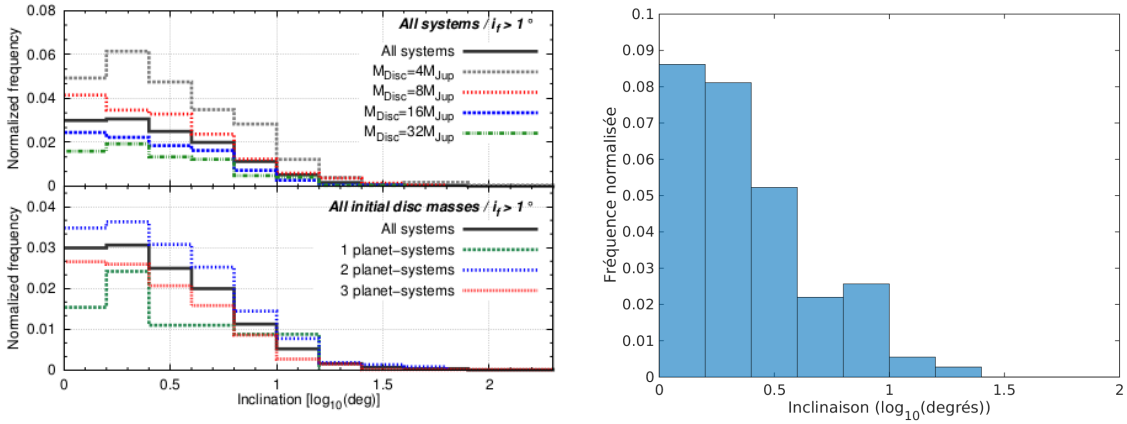


FIGURE 6.13 – Répartition des inclinaisons. A gauche : figure issue de [47]. A droite : notre étude.

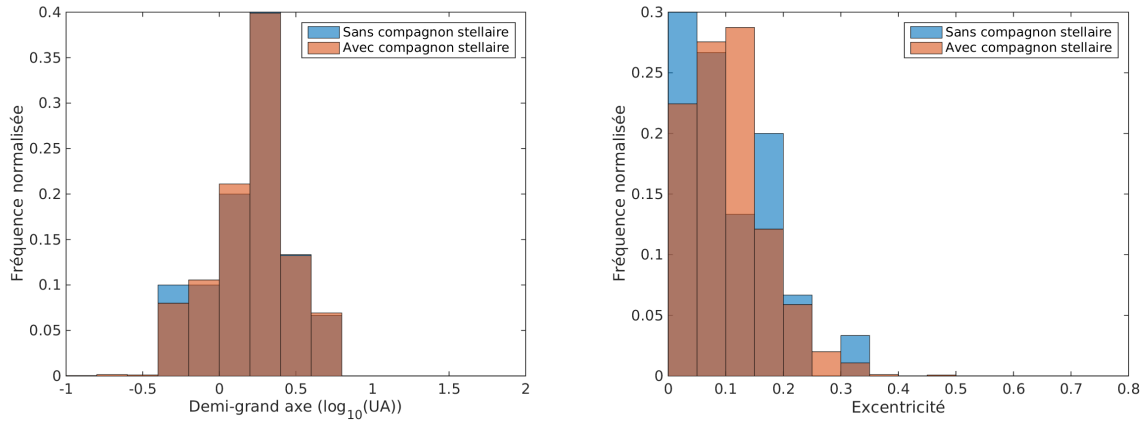


FIGURE 6.14 – Comparaison de nos résultats avec et sans compagnon stellaire

Résonance	Nombre de captures	Pourcentage observé
3 : 2	12	0.89%
2 : 1	1202	89.04%
5 : 2	2	0.14%
3 : 1	102	7.56%
pas de résonance	12	0.89%

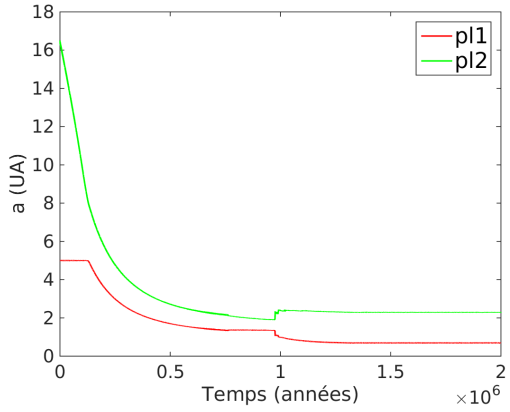
TABLE 6.2 – Résonances observées dans les simulations des systèmes avec compagnon stellaire

Ensuite, en comparant les excentricités à la figure 6.14 (figure de droite), nous constatons que le compagnon stellaire semble influencer l'excentricité. Néanmoins, il ne nous est pas possible de statuer sur une augmentation ou une diminution.

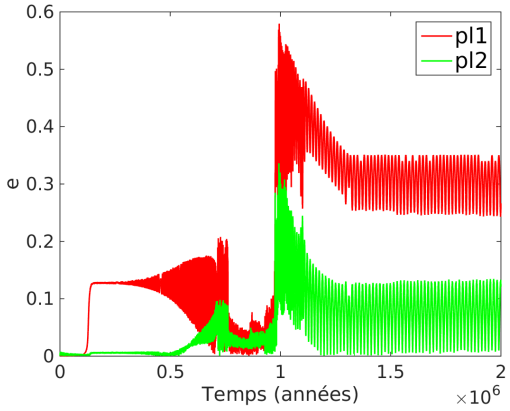
De plus, concernant les inclinaisons, nous avons constaté, qu'en l'absence de compagnon stellaire, les systèmes ne présentaient pas d'augmentation de l'inclinaison, ce qui n'est pas le cas en présence d'un compagnon stellaire. Ce dernier aura donc tendance à augmenter significativement l'inclinaison des planètes.

Enfin, au niveau des captures en résonance, nous pouvons comparer les tables 6.1 et 6.2. Nous constatons une différence dans les pourcentages des différentes résonances. Le compagnon stellaire semble favoriser la résonance 2 : 1. Notons qu'en présence d'un compagnon stellaire, certains corps vont également sortir de la résonance durant la migration. Un tel système est illustré à la figure 6.15. Nous constatons bien une sortie de résonance confirmée par les angles résonants qui commencent à circuler. Ce cas ne s'est pas produit sans compagnon stellaire.

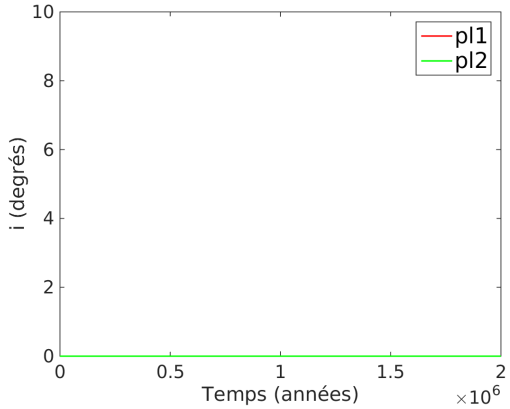
Notons également qu'à la sortie de la phase de migration, les systèmes semblent avoir atteint leur configuration finale dans les simulations avec une seule étoile. Par contre, dans les systèmes avec un compagnon stellaire, nous avons constaté précédemment que certains systèmes n'avaient pas encore atteint leur configuration finale (e.g. figure 6.3) et avaient demandé une intégration plus longue. Nous avons donc comparé les résultats à la fin de la migration à ceux obtenus après une période d'intégration plus longue. Ces résultats sont illustrés à la figure 6.16. Nous pouvons constater que les paramètres finaux sont assez semblables dans les deux cas. Seule l'inclinaison mutuelle semble être légèrement plus élevée. Pour rappel, l'inclinaison mutuelle correspond à



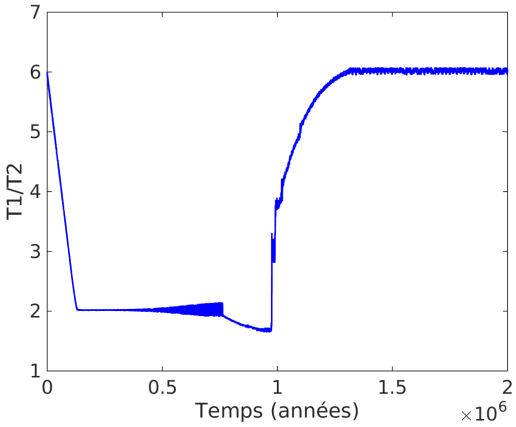
Demi-grands axes



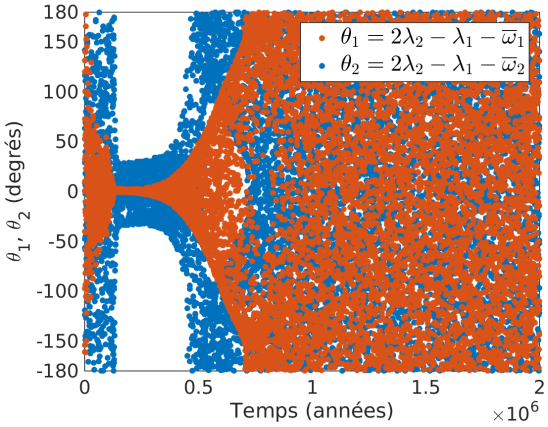
Excentricités



Inclinaisons



Rapport de périodes



Angles résonants

FIGURE 6.15 – Evolution du système binaire de conditions initiales $m_B = 1M_\odot$, $a_B = 1000$ UA, $e_B = 0.6$, $i_B = 0°$, $\frac{m_1}{m_2} = 0.5$ et $e_1 = e_2 = 0.001$

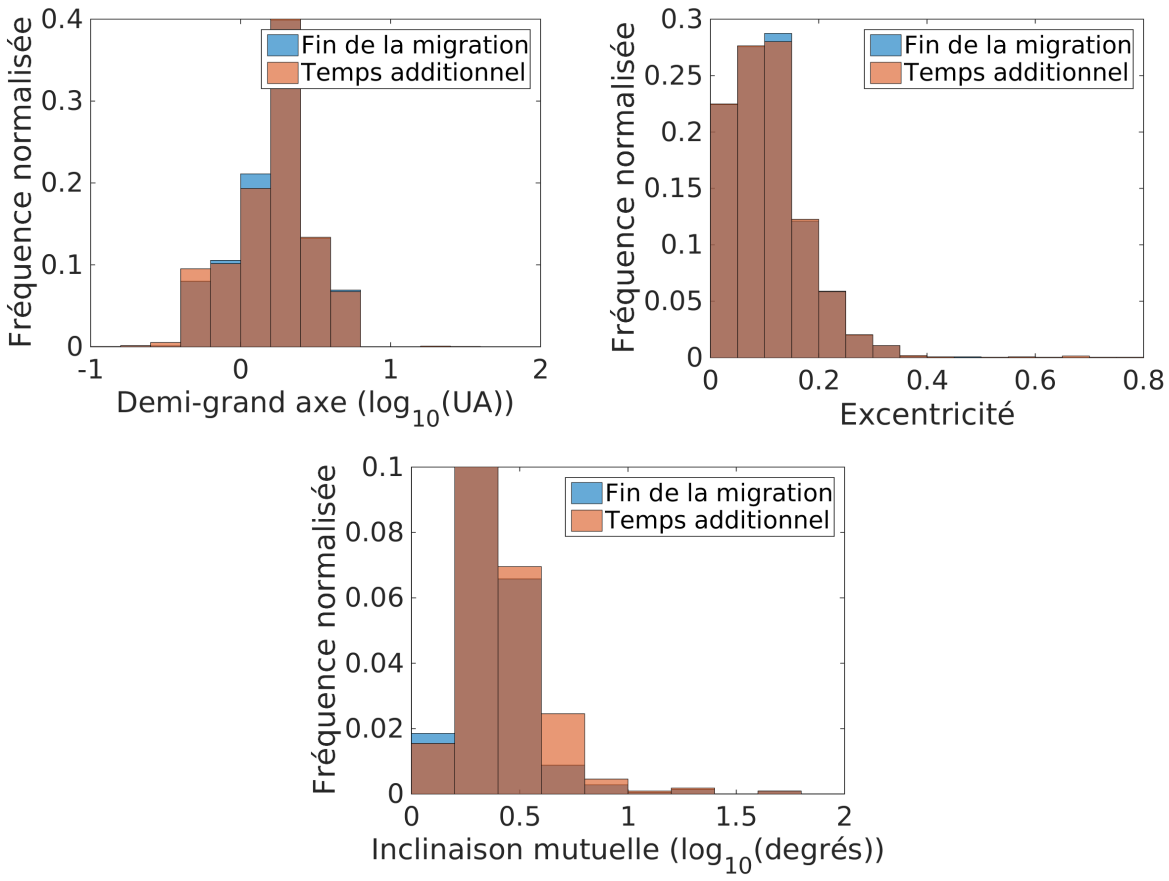


FIGURE 6.16 – Comparaison des architectures finales à la fin de la migration (1.3×10^6) et pour une période d'intégration plus longue (2×10^7)

l'inclinaison de la planète 1 par rapport à la planète 2 et est donnée par la formule :

$$cos(i_{mut}) = cos(i_1)cos(i_2) + sin(i_1)sin(i_2)cos(\Omega_1 - \Omega_2),$$

où i_{mut}, i_l et Ω_l sont respectivement l'inclinaison mutuelle, l'inclinaison de la planète l et la longitude du noeud ascendant de la planète l. Notons que nous utilisons l'inclinaison mutuelle afin de modérer le phénomène illustré à la figure 6.7.

En conclusion de ce chapitre, nous insistons sur l'importance de l'influence que semble avoir un compagnon stellaire sur la migration de type II. L'influence la plus marquante est l'augmentation des inclinaisons que le compagnon stellaire suscite sur les planètes.

Conclusion

Dans le cadre de ce stage, nous avons, dans un premier temps, amélioré notre code calculant la dynamique de systèmes à n corps possédant deux corps massifs. Nous avons ainsi traité les rencontres proches avec l'étoile centrale du système.

Nous avons ensuite implémenté la migration de type II pour les planètes géantes évoluant dans les systèmes binaires. Notre code est une adaptation de celui développé dans [47]. Il nous a servi à mener une étude paramétrique de la formation des planètes géantes dans les systèmes binaires.

Plus de mille simulations avec des configurations initiales différentes ont été réalisées. Nous sommes parvenus à la conclusion que le compagnon stellaire semble influencer significativement la migration de type II. En particulier, il semble tendre à augmenter les inclinaisons planétaires. Celui-ci semble également tendre à rendre les systèmes plus instables.

Nous terminons cette conclusion en évoquant quelques perspectives futures. Tout d'abord, il serait intéressant de raffiner notre modèle de migration. En effet, nous avons appliqué un modèle supposant que le compagnon stellaire n'influençait pas le disque de gaz. La migration a donc été calculée en utilisant les formules de [6] telles que présentées dans [47]. Cependant, le compagnon stellaire a une influence sur le disque et il serait donc intéressant de généraliser l'étude faite dans [6] aux systèmes binaires, afin de raffiner notre modèle. Ensuite, il faudrait poursuivre le travail réalisé dans ce manuscrit en diversifiant les configurations initiales considérées. Nous n'avons malheureusement pas pu le faire durant ce stage par manque de temps et de puissance de calcul. En effet, l'étude présentée dans ce manuscrit, a demandé plus de 40 jours de simulation avec la puissance de calcul mise à notre disposition. Le recours à un cluster nous paraît dès lors tout indiqué. Enfin, il serait également intéressant de considérer le cas des systèmes binaires possédant trois planètes. Cela reviendrait à généraliser le travail effectué dans [47] aux systèmes binaires.

Bibliographie

[1] R. Alexander, *Planet migration*, University of Leicester, 2011.

[2] P. J. Armitage, *Astrophysics of planet formation*, Cambridge University Press, 2010.

[3] P. J. Armitage, *Gap opening and planet migration*, `http://jila.colorado.edu/~pja/planet_migration.html`, consulté le 4 mai 2017.

[4] G. Bacon, *Astronomers use Hubble to 'weigh' dog star's companion*, `http://hubblesite.org/newscenter/archive/releases/2005/36/image/b`, 2005, consulté le 26 mars 2016.

[5] D. Benningfield, *Capricornus*, `https://stardate.org/radio/program/capricornus-0`, 2015, consulté le 26 mars 2016.

[6] B. Bitsch, A. Crida, A.-S. Libert et E. Lega, *Highly inclined and eccentric massive planets I. Planet-disc interactions*, Astronomy & Astrophysics, 555, 2013.

[7] T. Carletti, *Module intégration numérique,* Cours de Chaos et Déterminisme, Université de Namur, 2016.

[8] T. Carletti et A.-S. Libert, *Chaos et déterminisme*, Université de Namur, 2016.

[9] J. E. Chambers, *An hybrid symplectic integrator that permits close encounters between massive bodies*, Monthly Notices of the Royal Astronomical Society, 304, pages 793–799, 1999.

[10] J. E. Chambers, *N-Body integrators for planets in binary star systems*, chapitre 9 du livre *Planets in binary star systems*, éditeur : H. Haghighipour, Astrophysics and Space Science Library, première édition, pages 239-263, 2010.

[11] J. E. Chambers, E. V. Quintana, M. J. Duncan et J. J. Lissauer, *Symplectic integrator algorithms for modeling planetary accretion in binary star systems*, The Astronomical Journal, 123, pages 2884–2894, 2002.

[12] A. Crida et A. Morbidelli, *Cavity opening by a giant planet in a protoplanetary disc and effects on planetary migration*, Monthly Notices of the Royal Astronomical Society, 377, pages 1324–1336, 2007.

[13] J. Danby, *Fundamentals of celestial mechanics*, Willmann-Bell, deuxième édition, 1988.

[14] M. Ducan et H. Levison, *Symplectically integrating close encounters with the sun*, The Astronomical Journal, 120, pages 2117–2123, 2000.

[15] M. Duncan, H. Levison et M. H. Lee, *A multiple time step symplectic algorithm for integrating close encounters*, The Astronomical Journal, 116, pages 2067–2077, 1998.

[16] M. J. Duncan et H. F. Levison, *The long-term dynamical behavior of short-period comets*, Icarus, 108, pages 18–36, 1994.

[17] A. Duquennoy et M. Mayor, *Multiplicity among solar-type stars in the solar neighbourhood. II - Distribution of the orbital elements in an unbiased sample*, Astronomy & Astrophysics, 248, pages 485–524, 1991.

[18] R. Fitzpatrick, *Jacobi integral*, `http://farside.ph.utexas.edu/teaching/336k/Newtonhtml/node121.html`, 2011, consulté le 19 avril 2016.

[19] M. Gerbaldi et G. Theureau, *Les systèmes stellaires multiples*, `https://media4.obspm.fr/public/FSU/pages_doubles/impression.html`, 2016, consulté le 12 mars 2016.

[20] N. Haghighipour, *Dynamical stability and habitability of the γ Cephei binary-planetary system*, The Astrophysical Journal, 644, pages 543–550, 2006.

[21] D. E. Kaufmann, *Swifter — an improved solar system integration software package*, `http://www.boulder.swri.edu/swifter`, consulté le 22 novembre 2016.

[22] Y. Kozai, *Secular perturbations of asteroids with high inclination and eccentricity*, The Astronomical Journal, 67, pages 591–598, 1962.

[23] J. Laskar, *Systèmes de variables et éléments,* du livre *Les méthodes modernes de la mécanique céleste,* auteurs : D. Benest et C. Froeschlé, Editions Frontières, première édition, pages 63-87, 1989.

[24] J. Laskar et P. Robutel, *High order symplectic integrators for perturbed hamiltonian systems*, Celestial Mechanics and Dynamical Astronomy, 80, pages 39–62, 2001.

[25] M. H. Lee et S. J. Peale, *Dynamics and origin of the 2 :1 orbital resonances of the GJ 876 planets*, The Astrophysical Journal, 567, pages 596–609, 2002.

[26] A. Lemaitre, *Mécanique du point et du solide*, Université de Namur, 2014.

[27] A. Lemaître, *Resonances : models and captures,* du livre *Dynamics of small solar system bodies and exoplanets,* éditeurs : J. J. Souchay, et R. Dvorak, Springer Berlin Heidelberg, pages 1-62, 2010.

[28] A.-S. Libert, *Géométrie symplectique*, Université de Namur, 2015.

[29] A.-S. Libert et K. Tsiganis, *Trapping in high-order orgital resonances and inclination excitation in extrasolar systems*, Monthly Notices of the Royal Astronomical Society, 400, pages 1373–1382, 2009.

[30] D. Malmberg, M. B. Davies et J. E. Chambers, *The instability of planetary systems in binaries : how the Kozai mechanism leads to strong planet-planet interactions*, Monthly Notices of the Royal Astronomical Society : Letters, 377, pages L1–L4, 2007.

[31] C. Mordasini et H. Klahr, *Orbital migration*, University of Heidelberg, 2011-2012.

[32] NASA, *The nearest star about the image*, `http://imagine.gsfc.nasa.gov/features/cosmic/nearest_star_info.html`, 2015, consulté le 19 avril 2016.

[33] NASA, *Kepler a search for habitable planets*, `http://kepler.nasa.gov`, 2016, consulté le 18 avril 2016.

[34] NASA, *Kepler discoveries*, `http://kepler.nasa.gov/Mission/discoveries`, 2016, consulté le 18 avril 2016.

[35] J. C. B. Papaloizou et J. D. Larwood, *On the orbital evolution and growth of protoplanets embedded in a gaseous disc*, Monthly Notices of the Royal Astronomical Society, 315, pages 823–833, 2000.

[36] D. Pierce, *FXDR*, `http://meteora.ucsd.edu/~pierce/fxdr_home_page.html`, 1995, consulté le 21 avril 2016.

[37] W. H. Press, S. A. Teukolsky, W. T. Vetterling et B. P. Flannery, *Numerical recipes in fortran 77. The art of scientific computing*, pages 718–726, Press Syndicate of the University of Cambridge, deuxième édition, 1992.

[38] D. Raghavan, H. A. McAlister, T. J. Henry, D. W. Latham, G. W. Marcy, B. D. Mason, D. R. Gies, R. J. White et T. A. ten Brummelaar, *A Survey of Stellar Families : Multiplicity of Solar-type Stars*, The Astrophysical Journal Supplement, 190, pages 1–42, 2010.

[39] K. P. Rauch et M. Holman, *Dynamical chaos in the Widsom-Holman integrator : origins and solutions*, The Astronomical Journal, 117, pages 1087–1102, 1999.

[40] L. F. Richardson et J. A. Gaunt, *The deferred approach to the Limit. Part I. single lattice. part II. Interpenetrating lattices*, Philosophical Transactions of the Royal Society of London Series A, 226, pages 299–361, 1927.

[41] T. Roell, R. Neuhäuser, A. Seifahrt et M. Mugrauer, *Extrasolar planets in stellar multiple systems (Research Note)*, Astronomy & Astrophysics, 542, page A92, 2012.

[42] A. Roisin, *Implémentation d'un détecteur de chaos pour les systèmes célestes*, Université de Namur, 2017.

[43] L. Sacco, *Kepler a découvert une Tatooine avec deux couchers de soleil*, `http://www.futura-sciences.com/magazines/espace/infos/actu/d/astronomie-kepler-decouvert-tatooine-deux-couchers-soleil-33455`, 2011, consulté le 28 avril 2016.

[44] R. Schwarz, *Dynamics and observational prospects of co-orbital planets in double stars*, `http://www.univie.ac.at/adg/schwarz/COORBIN/coorbin.html`, 2016, consulté le 17 avril 2016.

[45] R. Schwarz, *Binary catalogue of exoplanets*, `http://www.univie.ac.at/adg/schwarz/bincat_binary_star.html`, 2016, consulté le 18 avril 2016.

[46] N. I. Shakura et R. A. Sunyaev, *Black holes in binary systems. Observational appearance*, Astronomy & Astrophysics, 24, pages 337–355, 1973.

[47] S. Sotiriadis, A.-S. Libert, B. Bitsch et A. Crida, *Highly inclined and eccentric massive planets II. Planet-planet interactions during the disc phase*, Astronomy & Astrophysics, 598, 2017.

[48] G. Terwagne, *Physique générale - mécanique*, Université de Namur, 2016.

[49] Wikipedia, *the free encyclopedia*, `https://en.wikipedia.org`, pages : *Alpha1 Capricorni, Alpha2 Capricorni, Alpha Centauri, binaire astrométrique, binaire spectroscopique, étoile binaire, étoile double, Jupiter, Neptune, Saturne, Sirius, Uranus*, consulté entre le 26 février 2016 et le 6 avril 2017.

[50] J. Wisdom et M. Holman, *Symplectic maps for the N-body problem*, The Astronomical Journal, 102, pages 1528–1538, 1991.

[51] H. Yoshida, *Construction of higher order symplectic integrators*, Physics Letter A, 150, pages 262–268, 1990.

Annexe A

Codes

A.1 Coordonnées

A.1.1 coord_posbin2h.f

```
c**********************************************************************
c                          COORD_POSBIN2H.F
c**********************************************************************
c     PURPOSE: Converts positions from binary to heliocentric coords.
c     ARGUMENTS:  Inputs are
c                    nbod                    ==> number of bodies
c                                                (must be less than NBMAX)
c                                                (integer)
c                    mass(*)                 ==> masses (real array)
c                    xbin(*),ybin(*),zbin(*) ==> binary particle positions
c                                                (real array)
c                    mred                    ==> mass of the system
c                                                without the companion
c                                                star(real scalar)
c                 Returned are
c                    xh(*),yh(*),zh(*)       ==> heliocentric particle
c                                                positions (real array)
c
c Author :  Arnaud Roisin
c Date :    2016
c
c The coordinates comes from Chambers et al. 2002, AJ, 123, 2884

subroutine coord_posbin2h(nbod,mass,xbin,ybin,
& zbin,xh,yh,zh,mred)

include '../swift.inc'

c...  Inputs:
integer nbod
real*8 mass(nbod), mred
real*8 xbin(nbod),ybin(nbod),zbin(nbod)

c...  Outputs:
real*8 xh(nbod),yh(nbod),zh(nbod)

c...  Internals:
real*8 xtmp,ytmp,ztmp
integer n

c----
c...  Executable code

xh(1) = xbin(1)
yh(1) = ybin(1)
zh(1) = zbin(1)

xtmp = 0.d0
ytmp = 0.d0
ztmp = 0.d0

do n=3,nbod
  xh(n) = xbin(n)
  yh(n) = ybin(n)
  zh(n) = zbin(n)
  xtmp = xtmp + mass(n)*xbin(n)
  ytmp = ytmp + mass(n)*ybin(n)
  ztmp = ztmp + mass(n)*zbin(n)
enddo

xtmp = xtmp/mred
```

```
ytmp = ytmp/mred
ztmp = ztmp/mred

xh(2) = xbin(2) + xtmp
yh(2) = ybin(2) + ytmp
zh(2) = zbin(2) + ztmp

return
end     ! coord_posh2bin.f
```

A.1.2 coord_posh2bin.f

```
c*************************************************************************
c                        COORD_POSH2BIN.F
c*************************************************************************
c     PURPOSE: Converts positions from Heliocentric to binary coords.
c     ARGUMENTS:  Inputs are
c                    nbod                    ==> number of bodies (must
c                                                be less than NBMAX)
c                                                (integer)
c                    mass(*)                 ==> masses (real array)
c                    xh(*),yh(*),zh(*)       ==> heliocentric particle
c                                                positions (real array)
c                 Returned are
c                    xbin(*),ybin(*),zbin(*) ==> binary particle
c                                                positions (real array)
c                    mred                    ==> mass of the system
c                                                without the companion
c                                                star(real scalar)
c
c Author:  Arnaud Roisin
c Date:    2016
c
c The coordinates comes from Chambers et al. 2002, AJ, 123, 2884

subroutine coord_posh2bin(nbod,mass,xh,yh,zh,xbin,
&          ybin,zbin,mred)

include '../swift.inc'

c...  Inputs:
integer nbod
real*8 mass(nbod)
real*8 xh(nbod),yh(nbod),zh(nbod)
real*8 mred

c...  Outputs:
real*8 xbin(nbod),ybin(nbod),zbin(nbod)

c...  Internals:
real*8 xtmp,ytmp,ztmp
integer n

c----
c...  Executable code

xbin(1) = xh(1)
ybin(1) = yh(1)
zbin(1) = zh(1)

xtmp = 0.d0
ytmp = 0.d0
ztmp = 0.d0

do n=3,nbod
  xbin(n) = xh(n)
  ybin(n) = yh(n)
  zbin(n) = zh(n)
  xtmp = xtmp + mass(n)*xh(n)
  ytmp = ytmp + mass(n)*yh(n)
  ztmp = ztmp + mass(n)*zh(n)
enddo

xtmp = -xtmp/mred
ytmp = -ytmp/mred
ztmp = -ztmp/mred

xbin(2) = xh(2) + xtmp
ybin(2) = yh(2) + ytmp
zbin(2) = zh(2) + ztmp

return
end     ! coord_posh2bin.f
```

A.1.3 coord_vbin2h.f

```
c*************************************************************************
c                        COORD_VBIN2H.F
c*************************************************************************
c     PURPOSE: Converts from binary to Helio coords.
c              Velocity only
c     ARGUMENTS:  Input are
c                    nbod                        ==> number of bodies (must
c                                                    be less than NBMAX)
c                                                    (integer)
c                    mass(*)                     ==> masses (real array)
c
c                    vxbin(*),vybin(*),vzbin(*) ==> Binary particle
c                                                    velocities
c                                                    (real array)
c                 Returned are
c                    vxh(*),vyh(*),vzh(*)        ==> Helio particle
c                                                    velocities
c                                                    (real array)
c
c Author :  Arnaud Roisin
c Date :    2016
c
c The coordinates comes from Chambers et al. 2002, AJ, 123, 2884

subroutine coord_vbin2h(nbod,mass,vxbin,vybin,vzbin,vxh,vyh,vzh)

include '../swift.inc'

c...  Inputs:
integer nbod
real*8 mass(nbod)
real*8 vxbin(nbod),vybin(nbod),vzbin(nbod)

c...  Outputs:
real*8 vxh(nbod),vyh(nbod),vzh(nbod)

c...  Internals:
integer i
real*8 vxint,vyint,vzint

c----
c...  Executable code

vxbin(1) = mass(3)*vxbin(3)
vybin(1) = mass(3)*vybin(3)
vzbin(1) = mass(3)*vzbin(3)

do i=4,nbod
  vxbin(1) = vxbin(1) + mass(i)*vxbin(i)
  vybin(1) = vybin(1) + mass(i)*vybin(i)
  vzbin(1) = vzbin(1) + mass(i)*vzbin(i)
enddo

vxint = vxbin(1)/mass(1)
vyint = vybin(1)/mass(1)
vzint = vzbin(1)/mass(1)

do i=3,nbod
  vxh(i) = vxbin(i)  + vxint
  vyh(i) = vybin(i)  + vyint
  vzh(i) = vzbin(i)  + vzint
enddo

vxh(2) = vxbin(2)  + vxint
vyh(2) = vybin(2)  + vyint
vzh(2) = vzbin(2)  + vzint

return
end     ! coord_vbin2h.f
```

A.1.4 coord_vh2bin.f

```
c*************************************************************************
c                        COORD_VH2BIN.F
c*************************************************************************
c     PURPOSE: Converts from Heliocentric to Binary coords.
c              Velocity only
c     ARGUMENTS:  Inputs are
c                    nbod                        ==> number of bodies (must
c                                                    be less than NBMAX)
c                                                    (integer)
c                    mass(*)                     ==> masses (real array)
c                    vxh(*),vyh(*),vzh(*)        ==> heliocentric particle
c                                                    velocities
c                                                    (real array)
c                 Returned are
c                    vxbin(*),vybin(*),vzbin(*) ==> binary particle
```

```
c                                                       velocities
c                                                       (real array)
c                     msys                          ==> Total mass of the
c                                                       system (real scalar)
c Author :  Arnaud Roisin
c Date :    2016
c
c The coordinates comes from Chambers et al. 2002, AJ, 123, 2884

subroutine coord_vh2bin(nbod,mass,vxh,vyh,vzh,vxbin,vybin,vzbin,
&     msys, mred)

include '../swift.inc'

c...  Inputs:
integer nbod
real*8 mass(nbod)
real*8 vxh(nbod),vyh(nbod),vzh(nbod)
real*8 msys, mred

c...  Outputs:
real*8 vxbin(nbod),vybin(nbod),vzbin(nbod)

c...  Internals:
real*8 vxtmp,vytmp,vztmp, vxb, vyb, vzb
integer n

c----
c...  Executable code

msys = mass(1)
vxtmp =0.d0
vytmp =0.d0
vztmp =0.d0

do n=3,nbod
  msys = msys + mass(n)
  vxtmp = vxtmp + mass(n)*vxh(n)
  vytmp = vytmp + mass(n)*vyh(n)
  vztmp = vztmp + mass(n)*vzh(n)
enddo

vxb = -vxtmp/mred
vyb = -vytmp/mred
vzb = -vztmp/mred

do n=2,nbod
  vxbin(n) = vxh(n) + vxb
  vybin(n) = vyh(n) + vyb
  vzbin(n) = vzh(n) + vzb
enddo

msys = msys +mass(2)
vxtmp = vxtmp + mass(2)*vxh(2)
vytmp = vytmp + mass(2)*vyh(2)
vztmp = vztmp + mass(2)*vzh(2)

vxbin(1) = -vxtmp/msys
vybin(1) = -vytmp/msys
vzbin(1) = -vztmp/msys

return
end     ! coord_vh2bin.f
```

A.2 Intégrateur pour systèmes binaires

A.2.1 symba7_helio_getacch_bin.f

Code pour les systèmes binaires

```
c*************************************************************
c                        SYMBA7_HELIO_GETACCH_BIN.F
c*************************************************************
c This subroutine calculates the acceleration on the massive
c particles in the Binary frame.
c             Input:
c                 iflg               ==>  =0 calculate forces
c                                           =1 don't (int scalar)
c                 nbod               ==>  number of massive
c                                           bodies (int scalar)
c                 nbodm              ==>  The last massive
c                                           particle (int scalar)
c                 mass               ==>  mass of bodies
c                                           (real array)
c                 j2rp2,j4rp4        ==>  J2*radii_pl^2 and
c                                           J4*radii_pl^4
c                                           (real scalars)
c                 xbin,ybin,zbin     ==>  position in binary
c                                           coord (real arrays)
c             Output:
c                 axbin,aybin,azbin  ==>  acceleration in binary
c                                           coord (real arrays)
c
c Author :  Arnaud Roisin
c Date :    2016
c
c Modification of symba7_helio_getacch.f, designed by Hal
c Levison, for binary star systems.
c The modifications comes from Chambers et al. 2002, AJ, 123,
c 2884

subroutine symba7_helio_getacch_bin(iflg,nbod,nbodm,mass,
&     j2rp2,j4rp4,xbin,ybin,zbin,axbin,aybin,azbin,mred)

include '../swift.inc'

c...  Inputs:
integer nbod,nbodm,iflg
real*8 mass(nbod),j2rp2,j4rp4
real*8 xbin(nbod),ybin(nbod),zbin(nbod)
real*8 mred

c...  Outputs:
```

Code d'origine

```
c*************************************************************
c                        SYMBA7_HELIO_GETACCH.F
c*************************************************************
c This subroutine calculates the acceleration on the massive
c particles in the HELIOCENTRIC frame.
c             Input:
c                 iflg        ==>  =0 calculate forces
c                                  =1 don't (int scalor)
c                 nbod        ==>  number of massive bodies
c                                  (int scalor)
c                 nbodm       ==>  The last massive particle
c                                  (int scalor)
c                 mass        ==>  mass of bodies (real array)
c                 j2rp2,j4rp4 ==>  J2*radii_pl^2 and
c                                  J4*radii_pl^4 (real scalars)
c                 xh,yh,zh    ==>  position in heliocentric
c                                  coord (real arrays)
c             Output:
c                 axh,ayh,azh ==>  acceleration in helio coord
c                                  (real arrays)
c
c Remarks Based on helio_getacch.f
c Author:  Hal Levison
c Date:    9/12/99
c Last revision:

subroutine symba7_helio_getacch(iflg,nbod,nbodm,mass,
&     j2rp2,j4rp4,xh,yh,zh,axh,ayh,azh)

include '../swift.inc'

c...  Inputs:
integer nbod,nbodm,iflg
real*8 mass(nbod),j2rp2,j4rp4
real*8 xh(nbod),yh(nbod),zh(nbod)

c...  Outputs:
real*8 axh(nbod),ayh(nbod),azh(nbod)

c...  Internals:
integer i,j
real*8 aoblx(NTPMAX),aobly(NTPMAX),aoblz(NTPMAX)
real*8 axhl(NTPMAX),ayhl(NTPMAX),azhl(NTPMAX)
```

```
real*8 axbin(nbod),aybin(nbod),azbin(nbod)

c...  Internals:
integer i,j
real*8 aoblx(NTPMAX),aobly(NTPMAX),aoblz(NTPMAX)
real*8 axbinl(NTPMAX),aybinl(NTPMAX),azbinl(NTPMAX)
real*8 ir3bin(NTPMAX),irbin(NTPMAX)
real*8 irb3, rb2
real*8 dx,dy,dz,rji2,faci,facj,irij3
real*8 sx,sy,sz
real*8 axint, ayint, azint
real*8 bxsx, bysy, bzsz
real*8 rbs2, irbs3
real*8 bxixsx, byiysy, bzizsz
real*8 rbis2, irbis3
real*8 mabred

save axbinl,aybinl,azbinl     ! Note this !!

c----
c...  Executable code

if(iflg.eq.0) then

  sx = 0.d0
  sy = 0.d0
  sz = 0.d0

  do i=1,nbod
    axbinl(i) = 0.d0
    aybinl(i) = 0.d0
    azbinl(i) = 0.d0
  enddo

  do i=3,nbodm
    sx = sx+mass(i)*xbin(i)
    sy = sy+mass(i)*ybin(i)
    sz = sz+mass(i)*zbin(i)
  enddo

  sx = sx*(1.d0/mred)
  sy = sy*(1.d0/mred)
  sz = sz*(1.d0/mred)

  bxsx = xbin(2) + sx
  bysy = ybin(2) + sy
  bzsz = zbin(2) + sz
  rbs2 = bxsx*bxsx + bysy*bysy + bzsz*bzsz
  irbs3 = 1.d0/(rbs2*sqrt(rbs2))
```

```
real*8 ir3h(NTPMAX),irh(NTPMAX)
real*8 dx,dy,dz,rji2,faci,facj,irij3

save axhl,ayhl,azhl     ! Note this !!

c----
c...  Executable code

if(iflg.eq.0) then

  do i=1,nbod
    axhl(i) = 0.0
    ayhl(i) = 0.0
    azhl(i) = 0.0
  enddo

  do i=2,nbodm
    do j=i+1,nbod
      dx = xh(j) - xh(i)
      dy = yh(j) - yh(i)
      dz = zh(j) - zh(i)
      rji2 = dx*dx + dy*dy + dz*dz

      irij3 = 1.0d0/(rji2*sqrt(rji2))
      faci = mass(i)*irij3
      facj = mass(j)*irij3

      axhl(j) = axhl(j) - faci*dx
      ayhl(j) = ayhl(j) - faci*dy
      azhl(j) = azhl(j) - faci*dz

      axhl(i) = axhl(i) + facj*dx
      ayhl(i) = ayhl(i) + facj*dy
      azhl(i) = azhl(i) + facj*dz
    enddo
  enddo

endif

c...  Now do j2 and j4 stuff
if(j2rp2.ne.0.0d0) then
  call getacch_ir3(nbod,2,xh,yh,zh,ir3h,irh)
  call obl_acc(nbod,mass,j2rp2,j4rp4,xh,yh,zh,irh,
  &        aoblx,aobly,aoblz)
  do i = 2,nbod
    axh(i) = axhl(i) + aoblx(i) - aoblx(1)
    ayh(i) = ayhl(i) + aobly(i) - aobly(1)
    azh(i) = azhl(i) + aoblz(i) - aoblz(1)
  enddo
else
```

```
rb2 = xbin(2)*xbin(2) + ybin(2)*ybin(2) + zbin(2)*zbin(2)
irb3 = 1.d0/(rb2*sqrt(rb2))

mabred = mass(1)*mass(2)/mred

axbinl(2) = axbinl(2) + mass(1) *irb3* xbin(2)
aybinl(2) = aybinl(2) + mass(1) *irb3* ybin(2)
azbinl(2) = azbinl(2) + mass(1) *irb3* zbin(2)
axbinl(2) = axbinl(2) - mass(1) *irbs3* bxsx
aybinl(2) = aybinl(2) - mass(1) *irbs3* bysy
azbinl(2) = azbinl(2) - mass(1) *irbs3* bzsz

do i=3,nbod
  axbinl(i) = axbinl(i) - mabred * irbs3 * bxsx
  aybinl(i) = aybinl(i) - mabred * irbs3 * bysy
  azbinl(i) = azbinl(i) - mabred * irbs3 * bzsz
enddo

axint = 0.d0
ayint = 0.d0
azint = 0.d0

do i=3, nbod
  bxixsx = xbin(2) - xbin(i) + sx
  byiysy = ybin(2) - ybin(i) + sy
  bzizsz = zbin(2) - zbin(i) + sz
  rbis2 = bxixsx*bxixsx + byiysy*byiysy + bzizsz*bzizsz
  irbis3 = 1.d0/(rbis2*sqrt(rbis2))

  axbinl(i) = axbinl(i) + mass(2) * irbis3 * bxixsx
  aybinl(i) = aybinl(i) + mass(2) * irbis3 * byiysy
  azbinl(i) = azbinl(i) + mass(2) * irbis3 * bzizsz

  axint = axint + mass(i) * irbis3 * bxixsx
  ayint = ayint + mass(i) * irbis3 * byiysy
  azint = azint + mass(i) * irbis3 * bzizsz
enddo


do i=3,nbod
  axbinl(i) = axbinl(i) - (mass(2)/mred) * axint
  aybinl(i) = aybinl(i) - (mass(2)/mred) * ayint
  azbinl(i) = azbinl(i) - (mass(2)/mred) * azint

  axbinl(2) = axbinl(2) + mass(i) * irb3 * xbin(2)
  aybinl(2) = aybinl(2) + mass(i) * irb3 * ybin(2)
  azbinl(2) = azbinl(2) + mass(i) * irb3 * zbin(2)
enddo

axbinl(2) = axbinl(2) - axint
aybinl(2) = aybinl(2) - ayint
azbinl(2) = azbinl(2) - azint
  do i = 2,nbod
    axh(i) = axhl(i)
    ayh(i) = ayhl(i)
    azh(i) = azhl(i)
  enddo
endif

return
end      ! symba7_helio_getacch

c-----------------------------------------------------------------
```

```
  do i=3,nbodm
    do j=i+1,nbod
      dx = xbin(j) - xbin(i)
      dy = ybin(j) - ybin(i)
      dz = zbin(j) - zbin(i)
      rji2 = dx*dx + dy*dy + dz*dz

      irij3 = 1.0d0/(rji2*sqrt(rji2))
      faci = mass(i)*irij3
      facj = mass(j)*irij3

      axbinl(j) = axbinl(j) - faci*dx
      aybinl(j) = aybinl(j) - faci*dy
      azbinl(j) = azbinl(j) - faci*dz

      axbinl(i) = axbinl(i) + facj*dx
      aybinl(i) = aybinl(i) + facj*dy
      azbinl(i) = azbinl(i) + facj*dz
    enddo
  enddo
endif

axbinl(2) = axbinl(2) * ((mass(2)+mred)/mred)
aybinl(2) = aybinl(2) * ((mass(2)+mred)/mred)
azbinl(2) = azbinl(2) * ((mass(2)+mred)/mred)

c...  Now do j2 and j4 stuff
if(j2rp2.ne.0.0d0) then
  call getacch_ir3(nbod,2,xbin,ybin,zbin,ir3bin,irbin)
  call obl_acc(nbod,mass,j2rp2,j4rp4,xbin,ybin,zbin,irbin,
  &        aoblx,aobly,aoblz)
  do i = 2,nbod
    axbin(i) = axbinl(i) + aoblx(i) - aoblx(1)
    aybin(i) = aybinl(i) + aobly(i) - aobly(1)
    azbin(i) = azbinl(i) + aoblz(i) - aoblz(1)
  enddo
else
  do i = 2,nbod
    axbin(i) = axbinl(i)
    aybin(i) = aybinl(i)
    azbin(i) = azbinl(i)
  enddo
endif

return
end      ! symba7_helio_getacch_bin.f

c-----------------------------------------------------------------
```

A.3 Influence du disque de gaz

A.3.1 dissipative_kick_bin.f

```
subroutine dissipative_kick_bin(nbod,nbodm,Mstar,Mdisc,Rin,Rout,
&  DeltaR,mass,xh,yh,zh,vxj,vyj,vzj,dth,rhill,Npl,first_write)

c...  Positions and velocities in the heliocentric frame

include '../swift.inc'

c...  Inputs Only:
integer nbod,nbodm,Npl,first_write
real*8 dth,Rin,Rout,DeltaR,ta_coeff
real*8 mass(NTPMAX),rhill(nbod)
real*8 xh(NTPMAX),yh(NTPMAX),zh(NTPMAX)

c...  Inputs and Output:
real*8 vxj(nbod),vyj(nbod),vzj(nbod)

c...  Internals:
integer i,j,ialpha,flag_edge
integer middle,outer(Npl),inner(Npl)
real*8 axh(nbod),ayh(nbod),azh(nbod)
real*8 taua(NTPMAX),taue(NTPMAX),taui(NTPMAX),didt
real*8 c1(NTPMAX),c2(NTPMAX),c3(NTPMAX)
real*8 F1(NTPMAX),F2(NTPMAX),F3(NTPMAX)
real*8 el(6),v,r,gmsum,vdotr,r2,normec,normeF
real*8 a(Npl),e(Npl+2),inc(Npl+2),incdeg(Npl+2)
real*8 capom(nbod),omega(nbod),capm(nbod),xi(Npl+2)
real*8 aaa,bbb,ccc,ddd,ggg,fff
real*8 M,Mstar,Mdisc,LocalDisc,Sigma00,temp_Mdisc,Mdot
real*8 temp_taua,tecc,tegap(nbod),tigap(nbod)

real*8 timet,init_sigma,tinc

c...  Common blocks
common/migrate/ taua,taue,taui,ta_coeff
common/sigma/ Sigma00,init_sigma

common timet
c----
c...  Executable code

cvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvv
c...   Obtain the orbital elements
do i=3,nbod

  if (mass(i) .gt. 1.d-4) then

c... Initialize the 2 coefficients for gaps
    tegap(i) = 1. ; tigap(i) = 1.

    ialpha = -1
    gmsum = mass(1)+mass(i)

c...     The last output variable is argument of latitude xi (not M)
    call orbel_xv2el_u(xh(i),yh(i),zh(i),vxj(i),vyj(i),vzj(i),
    &    gmsum,ialpha,el(1),el(2),el(3),el(4),el(5),el(6))

    a(i-2)=el(1) ! semi-major axis
    e(i)=el(2) ; if (e(i) .eq. 0) e(i) = 1.d-14 ! eccentricity
    inc(i)=el(3) ; incdeg(i)=inc(i)*180.D0/pi !Inclination
    xi(i) = el(6); ! Argument of latitude
    xi(i) = dmod(xi(i),2*PI)
    if (xi(i) .lt. 0.0d0) xi(i) = xi(i) + 2*PI ! In interval [0,2Pi)

    endif

  enddo
c^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^

cvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvv
c...   Check which planet is the inner, the middle and the outer one
c...   In our model: only the outer is migrating
outer = maxloc(a)+2
inner = minloc(a)+2
do i=1,Npl
  if ((i+2.ne.inner(1)).and.(i+2.ne.outer(1))) middle = i+2
enddo
c^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^
cvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvv
c...   If the outer planet is outside of 30AU then the "middle" one
c...   start migration. The outer one will remain in the system if
c...   it is inside of the "ejection from the system" criterion
```

```
c... 3 planets in the system
if (Npl .eq. 3) then

  if ( a(outer(1)-2) .lt. Rout ) then

    taua(outer(1))=ta_coeff*9.4*0.005*0.05*0.05*
    &            (a(outer(1)-2)**(-1.5))
    taua(inner(1)) = 1.d-15 ; taua(middle) = 1.d-15

  elseif ( a(outer(1)-2) .ge. Rout ) then

    if ( a(middle-2) .lt. Rout) then
      taua(middle)=ta_coeff*9.4*0.005*0.05*0.05*
      &                  (a(middle-2)**(-1.5))
      taua(inner(1)) = 1.d-15 ; taua(outer(1)) = 1.d-15
    else
      taua(middle) = 1.d-15 ; taua(inner(1)) = 1.d-15
      taua(outer(1)) = 1.d-15 ;
    endif

  endif

c... 2 planets in the system
elseif (Npl .eq. 2) then

  if ( a(outer(1)-2) .lt. Rout ) then

    taua(outer(1)) = ta_coeff*9.4*0.005*0.05*0.05*
    &            (a(outer(1)-2)**(-1.5)) ; taua(inner(1)) = 1.d-15

  else

    taua(outer(1)) = 1.d-15 ; taua(inner(1)) = 1.d-15

  endif

c... 1 planet in the system. No migration (?)
else

  taua(inner(1)) = ta_coeff*9.4*0.005*0.05*0.05*
  &            (a(inner(1)-2)**(-1.5))

endif
c^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^
cvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvv
c...   Check if the planets share a common gap. "4 R_Hill" criterion
c...   The condition is (a_i+2*R_Hill_i) > (a_j-2*R_Hill_i)

c...  3 planets in the system
if ( Npl .eq. 3 ) then

c...    All of the them share a common gap
  if ( ((a(inner(1)-2)+rhill(inner(1))*2) .ge.
  &        (a(middle-2)-rhill(middle)*2))    .and.
  &       ((a(middle-2)+rhill(middle)*2)     .ge.
  &       (a(outer(1)-2)-rhill(outer(1))*2))  ) then

    tegap(inner(1)) = 2. ; tigap(inner(1)) = 2.
    tegap(middle) = 100. ; tigap(middle) = 100.
    tegap(outer(1)) = 2. ; tigap(outer(1)) = 2.

c...    Only the inner planets share a gap
  elseif ( (a(inner(1)-2)+rhill(inner(1))*2) .ge.
  & (a(middle-2)-rhill(middle)*2) ) then
    tegap(inner(1)) = 2. ; tigap(inner(1)) = 2.
    tegap(middle) = 2. ; tigap(middle) = 2.

c...    Only the outer planets share a gap
  elseif ( (a(middle-2)+rhill(middle)*2) .ge.
  & (a(outer(1)-2)-rhill(outer(1))*2) ) then

    tegap(middle) = 2. ; tigap(middle) = 2.
    tegap(outer(1)) = 2. ; tigap(outer(1)) = 2.

  endif

c...  2 planets in the system
elseif ( Npl .eq. 2) then

c...    The two planets share a common gap
  if ( (a(inner(1)-2)+rhill(inner(1))*2) .ge.
  &    (a(outer(1)-2)-rhill(outer(1))*2) )   then

    tegap(inner(1)) = 2. ; tigap(inner(1)) = 2.
    tegap(outer(1)) = 2. ; tigap(outer(1)) = 2.

  endif

endif
c^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^
```

```
cvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvv
c...    Keep the initial value for re-use if one of the planets
c       is inside the cavity
temp_Mdisc = Mdisc
c^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^

c#########################################################################
c...    ************* THE LOOP FOR THE DAMPING **********************
c#########################################################################
do i=3,nbod ! Start the loop for every planet

  if (mass(i) .gt. 1.d-4) then !ONLY FOR GIANTS

cvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvv
c...     Initialize accelarations and the mass of the disk
     axh(i) = 0 ; ayh(i) = 0 ; azh(i) = 0
     Mdisc = temp_Mdisc
c^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^

cvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvv
c...    If e > e_crit the code crashes / Let SyMBA do the
c       ejection/merging
     if (e(i) .gt. 0.95) cycle
c^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^

cvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvv
c...    Inner disk boundary / No damping inside the inner cavity
     temp_taua = taua(i)  ; flag_edge = 1
     if (a(i-2) .lt. Rin+DeltaR) then
       Mdisc = Mdisc*tanh( (a(i-2)-Rin) / DeltaR  )
       if (Mdisc .lt. 0.0) then
         Mdisc = 0.0d0
         temp_taua = 0.d0
         flag_edge = 0
       endif
     endif
c^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^

cvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvv
c...     Check which case: "Planet dominated" or "Disc dominated" case
     if (taua(i).gt. 1.d-14) then

c...       Mass of the planet in Jupiter masses
       M = (Mstar/mass(1))*mass(i)

c...       ! Compute "Local disc", gamma exponent = 0.5
       LocalDisc = 4.*PI*Sigma00*a(i-2)**(1.5)/3.

c...       Change in "Planet dominated" case
       if (M .gt. LocalDisc) temp_taua=temp_taua*(LocalDisc/M)

     endif
c^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^

cvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvv
c...     Update the accelarations due to migration
     if (temp_taua .gt. 1.d-14) then

c       write(*,*) i,temp_taua

       axh(i) = -vxj(i)*temp_taua
       ayh(i) = -vyj(i)*temp_taua
       azh(i) = -vzj(i)*temp_taua
     endif
c^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^

cvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvv
c...   The formula from eccentricity and inclination damping after
c...   hydro simulations of B.Bitsch et al. 2013.

c...     Check if the planet is inside the disk's boundaries
     if ( (flag_edge .ne. 0) .and. ( a(i-2) .lt. Rout ) ) then

c...       The mass of the planet in Jupiter masses
       M=(Mstar/mass(1))*mass(i)

c...       The "local" mass of the disk for each planet
       LocalDisc = 4.*PI*Sigma00*a(i-2)**(1.5)/3. ! gamma=-0.5
       Mdisc = LocalDisc

c...       calculate v.r and r^2
       vdotr = xh(i)*vxj(i) + yh(i)*vyj(i) + zh(i)*vzj(i) !v.r
       r2 = xh(i)**2 + yh(i)**2 + zh(i)**2 !r^2

c...      %%%%%%%% %%%%%%%%%%%%%%%%%%%%%%%% %%%%%%
c...      %%%%%%%% Damping for eccentricity %%%%%%%
c...      %%%%%%%% %%%%%%%%%%%%%%%%%%%%%%%% %%%%%%
c...       The code works properly for M < 11Mj (problem with merging)
c...       The execution will stop if we have a merging and the final
c...       mass is higher than ~11Mj
```

```
      taue(i)=12.65d0*(M*Mdisc/Mstar**2)*e(i)*exp(-(incdeg(i)/M)**2)

      aaa=(80.d0/e(i)**2)*exp(-e(i)**2*M/0.26D0)*15.D0**M *
      &     (20.D0+11.D0*M-M**2)
      bbb=0.3D0*M
      ccc=450.D0+2.D0**M
      ddd=-1.4D0+sqrt(M)/6.D0
      fff=(aaa*(incdeg(i)+M/3.D0)**(-2*bbb)+
      &      ccc*incdeg(i)**(-2*ddd))**(-0.5)
      fff = (-Mdisc/(2.93)*fff)
c...     F_e + G_e , the damping is scaled for "local mass" at 5.2 AU
      taue(i)=taue(i)+fff

c...      Damping scaled to have ./year and not ./orbit when
c...      G=4*pi**2 (mass(1)=4*pi**2)
      taue(i)=taue(i)*dsqrt(1./a(i-2)**3)

c...      Reduce the damping if the the planets share a common gap
      taue(i) = taue(i) / tegap(i)
c...      Eccentricity damping timescale
      if ( taue(i) .lt. 0.d0) then
        tecc = dabs(e(i)/taue(i))
      else
        tecc = - dabs(e(i)/taue(i)) + 1.d-12
      endif

c...      Update the accelarations after eccentricity damping
      axh(i) = axh(i) - (2.0*vdotr*xh(i)) / (r2*tecc)
      ayh(i) = ayh(i) - (2.0*vdotr*yh(i)) / (r2*tecc)
      azh(i) = azh(i) - (2.0*vdotr*zh(i)) / (r2*tecc)

c...     %%%%%  end for eccentricity damping %%%%%
c...     %%%%%%%% %%%%%%%%%%%%%%%%%%%%%%%%%%% %%%%%%

c...     %%%%%%%% %%%%%%%%%%%%%%%%%%%%%%%%%% %%%%%%
c...     %%%%%%%% Damping for inclination %%%%%%%%
c...     %%%%%%%% %%%%%%%%%%%%%%%%%%%%%%%%%% %%%%%%

c...     Need this check because if i<10^-8 then problem with
c...     all the angles
      if (incdeg(i) .gt. 0.5) then ! We choose as I_crit=0.5 degrees

c...      The formula doesn't work for e > 2/3.
c...      We just leave aaa and ccc coefficients only propotional
c...      to the mass if e > 2/3
        if (e(i) .lt. 0.6666665d0) then
          aaa=1.5D0*10.D0**4*(2.D0-3.D0*e(i))*M**3
          ccc=1.2D0*10.D0**6/((2.D0-3.D0*e(i))
          &              *(5.D0+e(i)**2*(M+2.D0)**3))
        else
          aaa=1.5D0*10.D0**4*10.d0**(-5)*M**3
          ccc=1.2D0*10.D0**6/((10.d0**(-5))*
          &               (5.D0+e(i)**2*(M+2.D0)**3))
        endif

        bbb=1.D0+M*(e(i)**2)/10.D0
        ddd=-3.D0+2.D0*e(i)
        ggg=dsqrt(3.D0*M/(e(i)+0.001D0))
        fff=(aaa*incdeg(i)**(-2*bbb)*exp(-((incdeg(i)/ggg)**2)/2.)
        &     +ccc*(incdeg(i)/40.D0)**(-2*ddd))**(-0.5)

c...    the damping is scaled for "local mass" at 5.2 AU
        didt=(-(Mdisc/(2.93))*fff)

c...    Damping scaled to have rad/year and not rad/orbit when
c...    G=4*pi**2 (mass(1)=4*pi**2)
        didt=didt*pi/180.D0 !In radians
        didt=didt*dsqrt(1./(a(i-2)**3))

c...    Reduce the damping if the the planets share a common gap
        didt = didt / tigap(i)

c...    Update the accelarations after inclination damping
       tinc = dabs(incdeg(i)/(didt*180.D0/pi))
       azh(i) = azh(i) - (2.0*vzj(i)) / (tinc)

      endif endif for inclination damping
c...     %%%%%%  end for inclination damping %%%%%
c...     %%%%%%%% %%%%%%%%%%%%%%%%%%%%%%%%%% %%%%%%

c... endif for the damping formula between the boundaries of the disk
    endif
c^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^
cvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvv
c...    Update the velocities after the interaction with the disk
```

```
    vxj(i) = vxj(i) + axh(i)*dth
    vyj(i) = vyj(i) + ayh(i)*dth
    vzj(i) = vzj(i) + azh(i)*dth
c^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^
  endif ! ONLY FOR GIANTS
enddo ! endding of the loop for the damping
c#####################################################################
c#####################################################################

cvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvv
c...   Take the initial value of the mass of the disk again
Mdisc = temp_Mdisc
c^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^
return
end
c---------------------------------------------------------------------
```