À taille humaine, à l'échelle du monde
INSA ROUEN

Projet de Physique P6-3
STPI/P6-3/2009 – 35

Etoiles doubles (et autres systèmes de deux points matériels)

Etudiants :

Eve ARQUIN
Anastacia BILICI
Mylène CHAMPAIN
Arnaud DELANDE
Zineb LAMRANI
Coralie PONSINET

Enseignant-responsable du projet :

Yves MONTIER

Date de remise du rapport : ***21/O6/09***

Référence du projet : ***STPI/P6-3/2009 – Projet n°35***

Intitulé du projet : ***Etoiles doubles (et autres systèmes de deux points matériels)***

Type de projet : ***Simulation***

Objectifs du projet :

Le principal objectif de notre projet est de maîtriser l'étude des systèmes isolés de deux points matériels, c'est-à-dire de comprendre leur fonctionnement.

A partir de l'exploitation des lois de la mécanique, le second objectif est de réaliser par simulations numériques, des animations illustrant les mouvements de ces points matériels.
Le but de telles animations est d'affirmer la compréhension des lois, mais aussi de permettre une visualisation des mouvements (trajectoires, …).

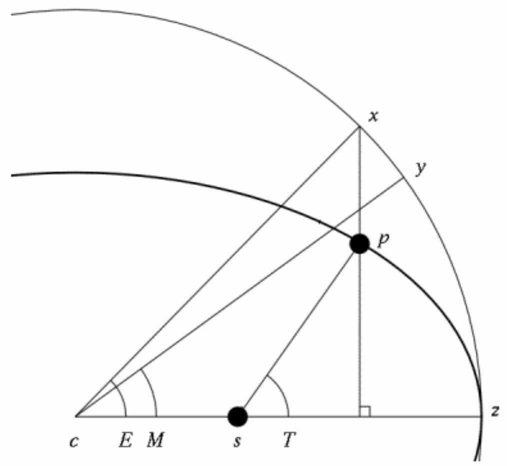
TABLE DES MATIERES

NOTATIONS, ACRONYMES

Référentiel barycentrique $\Re^*$: Référentiel dans lequel le barycentre est fixe.

Anomalie excentrique E :

C'est l'angle entre la direction du périapse et la position courante d'un objet sur son orbite, projetée sur le cercle exinscrit perpendiculairement au grand axe de l'ellipse, mesuré au centre de celle-ci. Dans le diagramme ci-contre, E est l'angle zcx.



Une classe en programmation : C'est l'unité de base de la programmation orienté objet.

Une classe défini un type et l'ensemble des opérations qu'il peut effectuer.

1. INTRODUCTION

Les étoiles doubles sont un ensemble de deux étoiles suffisamment rapprochées pour que l'on puisse observer des effets gravitationnels entre elles. Ce phénomène est très intéressant à étudier, car il est inclus dans une famille de phénomènes beaucoup plus large de la physique, notamment dans celui de l'étude des systèmes à plusieurs corps. Le cas de deux corps en est l'élément le plus simple.

Tout au long de notre projet, nous nous sommes focalisés sur le mouvement elliptique qui caractérise un grand nombre d'étoiles doubles. Il est possible en théorie de rencontrer d'autres types de mouvements comme le mouvement parabolique ou le mouvement hyperbolique mais ils ne rentrent pas dans le cadre de ce projet car ils sont trop particuliers. De ce fait, nous avons dû débuter le projet par la détermination des équations théoriques du mouvement étudié. Nous avons ensuite poursuivi par la recherche de moyens de modéliser le mouvement informatiquement et graphiquement. Ceci dans le but de pouvoir clairement visualiser les caractéristiques inhérentes à ce genre de mouvement.

La modélisation informatique a posé des problèmes mathématiques qu'il nous a fallu résoudre. Ce problème s'était déjà posé l'année dernière pour le projet concernant les champs de force newtoniens. Le groupe de l'année dernière a contourné la difficulté en utilisant le logiciel Maple. Notre objectif était de parvenir à résoudre ce problème par nos propres moyens.

Enfin, notre dernier objectif était de poursuivre le travail effectué l'année dernière afin de le compléter.

2. METHODOLOGIE / ORGANISATION DU TRAVAIL

La méthodologie adoptée par le groupe pour ce projet s'est installée au-fur-et-à-mesure de l'avancement de celui-ci. En effet, nous avons tous commencé par nous documenter sur ce qu'étaient des étoiles doubles, afin d'analyser et de comprendre les objectifs à atteindre.

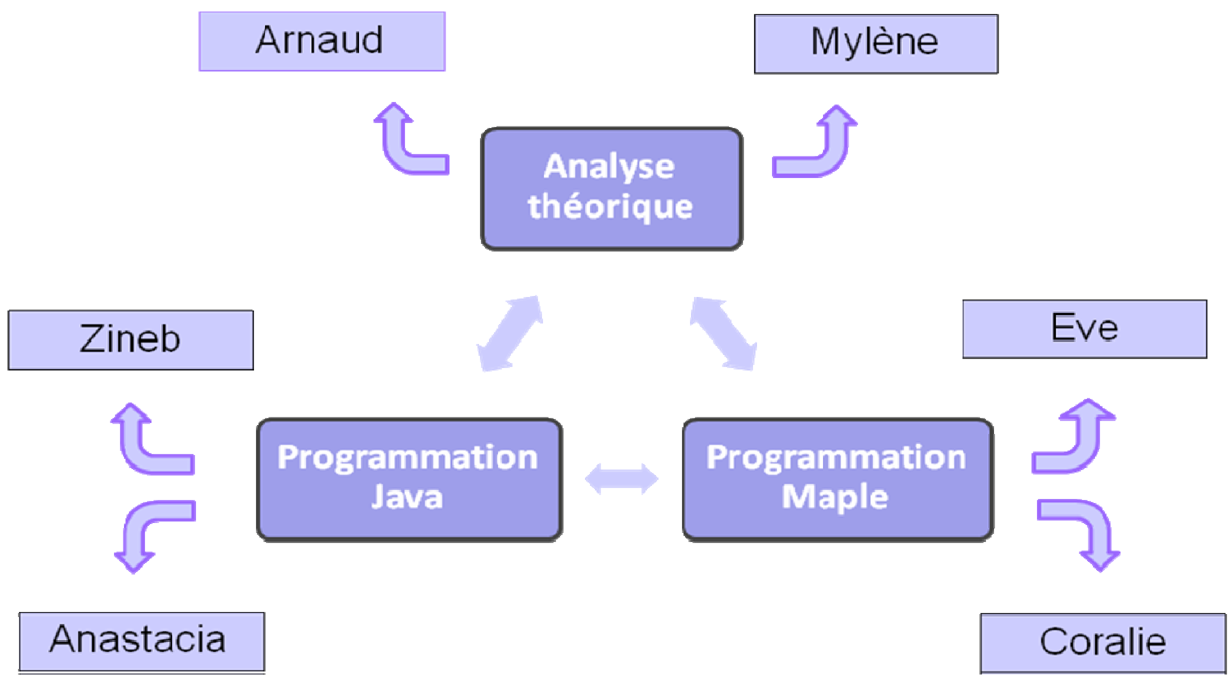
Puis, nous avons tous, durant les premières séances, tenté de réunir les équations déjà existantes et les relations entre elles afin de pouvoir les appliquer à notre cas d'étoile double.
C'est là que le vrai travail du projet s'est mis en œuvre : nous avons eu beaucoup de mal à nous « lancer » dans le projet, c'est-à-dire à véritablement saisir les enjeux de celui-ci. Mais une fois sur la bonne voie, l'organisation s'est installée par elle-même.

Arnaud et Mylène se sont concentrés sur l'analyse théorique, c'est-à-dire à mettre les équations trouvées les une en rapport avec les autres, cela afin de parvenir aux équations souhaitées : la position de chaque étoile du système d'étoile double, en fonction du temps.

Puisque nous voulions illustrer le projet grâce à des logiciels de programmation, nous avons choisi de suivre la programmation du groupe de l'année précédente afin de réaliser des expériences similaires mais pour les étoiles doubles. Pour cela, Eve et Coralie se sont concentrées sur les logiciels Povray et Scilab afin d'obtenir des simulations d'étoiles.

Mais nous voulions aussi tenter une nouvelle approche de la simulation en utilisant un logiciel interactif qui permettrait à l'utilisateur de choisir lui-même le système qu'il souhaite étudier. Zineb et Anastacia ont donc essayé de simuler des étoiles doubles grâce au logiciel Java.

Organigramme des tâches réalisées



3. TRAVAIL REALISE ET RESULTATS

3.1. Etude théorique

Soit dans un référentiel galiléen, un système de deux étoiles qu'on associe à deux points matériels M_1 et M_2 de masses m_1 et m_2. Les deux étoiles sont distantes de la longueur r.

D'après le principe fondamental de la dynamique :

$$m_1 \frac{d^2\overrightarrow{OM_1}}{dt^2} = \overrightarrow{f_1}, \qquad m_2 \frac{d^2\overrightarrow{OM_2}}{dt^2} = \overrightarrow{f_2}$$

Où $\overrightarrow{f_1}$ et $\overrightarrow{f_2}$ sont les forces appliquées aux points M_1 et M_2. Il s'agit ici des forces gravitationnelles donc elles sont réciproques : $\overrightarrow{f_1} = -\overrightarrow{f_2}$

3.1.1. Mouvement du barycentre G dans $\Re$

Le barycentre des deux étoiles est défini tel que : $m_1\overrightarrow{OM_1} + m_2\overrightarrow{OM_2} = (m_1 + m_2)\overrightarrow{OG}$

$$m_1 \frac{d^2\overrightarrow{OM_1}}{dt^2} + m_2 \frac{d^2\overrightarrow{OM_2}}{dt^2} = \overrightarrow{f_1} + \overrightarrow{f_2} = (m_1 + m_2)\frac{d^2\overrightarrow{OG}}{dt^2} = \vec{0}$$

Donc $\frac{d^2\overrightarrow{OG}}{dt^2} = \vec{0} \Rightarrow \overrightarrow{OG}(t) = \overrightarrow{v_G}t + \overrightarrow{OG}(0)$

Le point G décrit donc un mouvement rectiligne uniforme dans $\Re$.

3.1.2. Mouvement de la particule fictive dans le référentiel barycentrique

Intéressons nous maintenant au mouvement de la particule fictive.

$$\frac{d^2\overrightarrow{OM_2}}{dt^2} - \frac{d^2\overrightarrow{OM_1}}{dt^2} = \frac{d^2\overrightarrow{M_1M_2}}{dt^2} = \left(\frac{1}{m_1} + \frac{1}{m_2}\right)\overrightarrow{f_2}$$

Le vecteur $\overrightarrow{M_1M_2}$ est la distance entre les deux étoiles : $\overrightarrow{M_1M_2} = \vec{r}$

Notons μ la masse réduite du système telle que $\boxed{\frac{1}{\mu} = \left(\frac{1}{m_1} + \frac{1}{m_2}\right)}$

Nous avons donc l'équation : $\boxed{\mu \frac{d^2\vec{r}}{dt^2} = \overrightarrow{f_2}}$

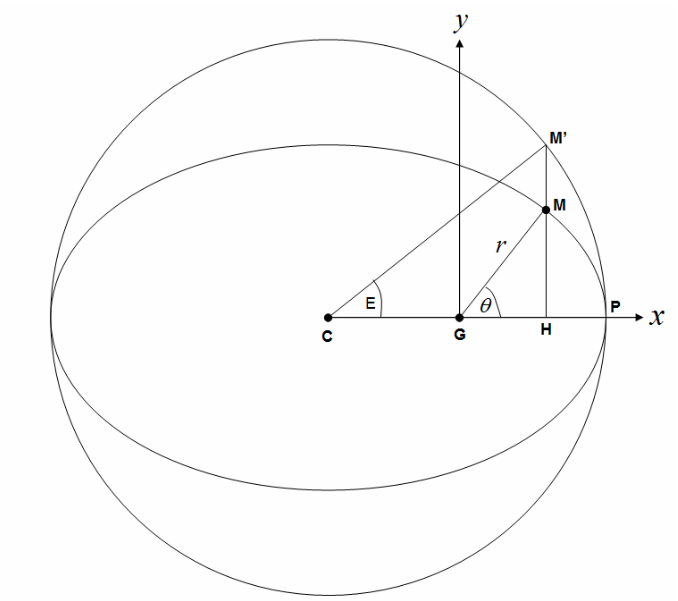
Soit un point M tel que $\overrightarrow{M_1M_2} = \overrightarrow{GM}$

Ainsi l'étude du système d'étoiles doubles revient à étudier la trajectoire du point M de masse μ , sur lequel s'exerce la force $\overrightarrow{f_2}$, autour d'un point G affecté d'une masse $m_1 + m_2$.

3.1.3. Trajectoire elliptique

On sait que toute orbite prend la forme d'une ellipse dont l'excentricité E est fonction des conditions initiales.

On considère le schéma suivant :



Dans un souci de clarté, on définit chaque paramètre :

M : Corps sur son orbite elliptique.

M' : Projeté du point M sur le cercle exinscrit perpendiculairement au grand axe de l'ellipse.

H : Projeté du point M sur le grand axe de l'ellipse.

G : Foyer de l'ellipse (le point autour duquel le corps orbite).

C : Centre de l'ellipse (aussi centre du demi grand axe).

E : Anomalie excentrique.

θ : Anomalie vraie.

Les demi grand axes et demi petit axes sont respectivement a et b.

L'orbite du point M autour du centre de masse G est une trajectoire elliptique.

Rappelons que l'équation en coordonnées polaires d'une ellipse est de la forme $r = \dfrac{p}{1+e\cos\theta}$ où p est le paramètre de la conique, e l'excentricité, $r = GM$ et $\theta = \left(\vec{x}, \overrightarrow{GM}\right)$

L'angle θ est aussi appelé l'anomalie vraie.

Cette équation décrit la trajectoire de M mais le temps n'intervient pas dans cette équation. Nous cherchons donc à exprimer r et θ en fonction du temps.

Pour cela, introduisons l'anomalie excentrique E.

D'après l'équation de Kepler, on a :

$$\frac{2\pi}{T}(t - t_P) = E - e\sin E$$

Avec T la période de révolution du point M et t_P le temps lorsque le point M passe au point P (périhélie). Par la suite, on prendra t_P=0 pour simplifier les calculs.

On a donc E l'anomalie excentrique dépendante du temps.

3.1.4. Détermination du rayon r et θ en fonction du temps

On cherche maintenant à exprimer r et θ en fonction du temps.

Pour cela, on se place dans le triangle GHM, et on calcule les longueurs GH et HM pour en déduire r.

- $GH = CH - CG = a\cos E - ae = a(\cos E - e)$.

  Car par définition, l'excentricité est le rapport de la distance du centre au foyer sur le demi grand axe, soit: $e = \dfrac{CG}{a} \Leftrightarrow CG = ae$.

- On a $\dfrac{HM}{HM'} = \dfrac{b}{a}$

De plus, la longueur du demi grand axe a et celle du demi petit axe b sont liées par l'excentricité e tel que $b = a\sqrt{1-e^2}$.

D'où $HM = \dfrac{b}{a} HM' = b\sin E = a\sqrt{1-e^2}\sin E$

On a donc les coordonnées cartésiennes du point P :

$$x = GH = a(\cos E - e)$$

$$y = HM = a\sqrt{1-e^2}\sin E$$

On peut ainsi calculer le rayon r :

$$r = \sqrt{x^2 + y^2} = \sqrt{a^2(\cos E - e)^2 + a^2(1-e^2)\sin^2 E}$$
$$\Leftrightarrow r = a\sqrt{\cos^2 E - 2e\cos E + e^2 + \sin^2 E - e^2\sin^2 E}$$
$$\Leftrightarrow r = a\sqrt{1 + e^2 - 2e\cos E - e^2(1-\cos^2 E)}$$
$$\Leftrightarrow r = a\sqrt{1 - 2e\cos E + e^2\cos^2 E}$$
$$\Leftrightarrow r = a\sqrt{(1 - e\cos E)^2}$$
$$\Leftrightarrow r = a\left|1 - e\cos E\right|$$

Pour une orbite elliptique $0 < e < 1$ et $-1 < \cos E < 1$

Donc $-1 < e\cos E < 1 \Rightarrow 0 < 1 - e\cos E < 1$

Ainsi $\left|1 - e\cos E\right| = 1 - e\cos E$

On obtient donc $r = a(1 - e\cos E)$

On a l'expression de r(t) et on a vu précédemment que E dépendait du temps.

On a bien **r en fonction de t**.

Ensuite, calculons θ en fonction du temps.

On a $\cos\theta = \dfrac{GH}{GM} = \dfrac{a(\cos E - e)}{r} = \dfrac{a(\cos E - e)}{a(1 - e\cos E)} = \dfrac{\cos E - e}{1 - e\cos E}$

Etant donné que E dépend du temps, et que l'on vient d'exprimer θ en fonction de E, alors on a donc θ **en fonction de t**.

Au final, on a donc r(t) et θ(t) que l'on peut rassembler dans le tableau suivant :

$$r = a(1 - e\cos E)$$
$$\cos\theta = \frac{\cos E - e}{1 - e\cos E}$$
$$\frac{2\pi}{T}(t - t_P) = E - e\sin E$$

On a aussi, en coordonnées cartésiennes, x(t) et y(t) :

$$x = a(\cos E - e)$$
$$y = a\sqrt{1-e^2}\sin E$$
$$\frac{2\pi}{T}(t - t_P) = E - e\sin E$$

3.1.5. *Expression de l'excentricité en fonction des conditions initiales r_0, v_0 et α*

Nous avons initialement choisi d'exprimer l'excentricité en fonction de r_0, v_0 et α, mais nous avons par la suite réalisé que v_0 correspond à la vitesse initiale du point fictive qui est difficile à utiliser pour nos calculs puisque nous ne trouvons sa valeur dans aucun livre. C'est pourquoi, nous avons décidé de fixer T comme condition initiale, valeur qui est donné dans beaucoup de livres astronomiques.

Nous avons décidé de conserver cette démonstration dans le rapport, puisqu'il s'agit d'une autre expression de l'excentricité toute aussi juste.

L'accélération du point M peut s'exprimer grâce à la formule de Binet comme suit :

$$\vec{a} = -C^2u^2\left(u + \frac{du}{d\theta}\right)\vec{u_r}$$

Avec $u = \frac{1}{r}$ et C la constante des aires.

D'après le principe fondamental de la dynamique, on a également : $\mu\vec{a} = \vec{f_2} = \frac{K}{r^2}\vec{u_r}$

Avec $K = -Gm_1m_2 = -G\mu(m_1 + m_2)$

On obtient donc $-\mu C^2u^2(u + u''(\theta))\vec{u_r} = Ku^2\vec{u_r}$

D'où on arrive à l'équation différentielle linéaire du second ordre : $u + u''(\theta) = \frac{-K}{\mu C^2}$

Après résolution, on obtient : $u = \frac{1}{r} = \frac{-K}{\mu C^2} + A\cos\theta$

A est une constante déterminée par les conditions initiales.

En posant $p = \left|\frac{-\mu C^2}{K}\right|$ le paramètre de la conique et $e = \left|\frac{A\mu C^2}{K}\right|$ l'excentricité, on retrouve bien l'égalité $r = \frac{p}{1 + e\cos\theta}$, équation d'une ellipse.

Initialement, on fait le choix de prendre $\theta_0 = 0$

Ainsi $A = \frac{1}{r_0} + \frac{K}{\mu C^2}$

On a ainsi $e = \left| \frac{A\mu C^2}{K} \right|$

La constante des aires C s'exprime : $C = r^2\dot{\theta} = r_0 v_0 \sin\alpha$

Avec $\alpha = (\overrightarrow{GM_0}, \overrightarrow{v_0})$

D'où $A = \frac{1}{r_0} + \frac{-G(m_1 + m_2)}{(r_0 v_0 \sin\alpha)^2}$

Ainsi $e = \left| 1 - \frac{r_0 (v_0 \sin\alpha)^2}{G(m_1 + m_2)} \right|$

3.1.6. *Expression de l'excentricité e en fonction des conditions initiales r_0, T et α*

D'après la troisième loi de Kepler, on a l'égalité :

$$T^2 = \frac{4\pi}{G(m_1 + m_2)} a^3$$

Or $e = \frac{a - r_0}{a} = 1 - \frac{r_0}{a} \Leftrightarrow r_0 = a(1 - e) \Leftrightarrow a = \frac{r_0}{1 - e}$

On remplace dans l'équation de Kepler :

$$T^2 = \frac{4\pi r_0^3}{G(m_1 + m_2)(1 - e)^3}$$

D'où $\boxed{e = 1 - \sqrt[3]{\frac{4\pi r_0^3}{GT^2(m_1 + m_2)}}}$

Au final, on obtient e en fonction des conditions initiales et de l'excentricité e qui dépend des conditions initiales. Donc les seuls paramètres m_1, m_2, T, v_0 et α suffisent à déterminer complètement le système.

3.1.7. Adaptation au problème à deux corps

Nous venons d'étudier le mouvement du point M par rapport au point G, centre de masse du système d'étoiles doubles dans le référentiel barycentrique. Rappelons que le point M est un point fictif.

On a :

$$\overrightarrow{GM_1} = -\frac{m_2}{m_1 + m_2}\overrightarrow{GM}$$

$$\overrightarrow{GM_2} = \frac{m_1}{m_1 + m_2}\overrightarrow{GM}$$

Or $\overrightarrow{GM} = \left(a(\cos E - e)\right)\overrightarrow{e_x} + \left(a\sqrt{1-e^2}\sin E\right)\overrightarrow{e_y}$

Donc on en déduit le mouvement des points M_1 et M_2 dans le référentiel barycentrique.

$$\overrightarrow{GM_1} = -\frac{m_2 a}{m_1 + m_2}\left[(\cos E - e)\overrightarrow{e_x} + \left(\sqrt{1-e^2}\sin E\right)\overrightarrow{e_y}\right]$$

$$\overrightarrow{GM_2} = \frac{m_1 a}{m_1 + m_2}\left[(\cos E - e)\overrightarrow{e_x} + \left(\sqrt{1-e^2}\sin E\right)\overrightarrow{e_y}\right]$$

Avec $\frac{2\pi}{T}(t - t_P) = E - e\sin E$

3.1.8. Conclusion et récapitulatif des formules

Finalement, seulement 3 paramètres doivent être renseignés pour définir les trajectoires des étoiles doubles en fonction du temps : il s'agit de r_0, T et α. Les étapes de résolution sont résumées dans le graphe ci-dessous.

Masse de l'étoile 1 m_1

Distance initiale r_0

Période T

Angle de vitesse α

Masse de l'étoile 2 m_2

Excentricité e

$$e = 1 - \sqrt[3]{\frac{4\pi r_0^3}{GT^2(m_1 + m_2)}}$$

Anomalie excentrique E

$$\frac{2\pi}{T}(t - t_A) = E - e \sin E$$

<u>Position de l'étoile 1</u>

$$x_1 = \frac{-m_2 r_0}{(m_1 + m_2)(1-e)}(\cos E - e)$$

$$y_1 = \frac{-m_2 r_0}{(m_1 + m_2)(1-e)}\sqrt{1-e^2}\sin E$$

<u>Position de l'étoile 2</u>

$$x_2 = \frac{m_1 r_0}{(m_1 + m_2)(1-e)}(\cos E - e)$$

$$y_2 = \frac{m_1 r_0}{(m_1 + m_2)(1-e)}\sqrt{1-e^2}\sin E$$

3.2. Programmation Maple-Povray

3.2.1. *Programmation en Maple*

Le but de la programmation sous Maple est d'obtenir les équations de trajectoire des deux étoiles doubles, pour ensuite les transférer sous Povray où elles seront mises sous la forme d'une simulation. Les données de Maple sont utilisables par Povray grâce à l'intermédiaire du logiciel Scilab.

Les équations de trajectoire à modéliser sont :

x1(t) = m2*r0*(cos(E_t(t))-e) / ((m1+m2)*(1-e))

y1(t) = m2*r0*√(1-e^2)*sin(E_t(t)) / ((m1+m2)*(1-e))

avec : m1 : masse de la première étoile double

m2 : masse de la deuxième étoile double

r0 : distance initiale

e : excentricité

Nous utilisons, donc, les équations trouvées dans la partie théorique.

T est la principale variable, avec r_o, m1 et m2.

L'excentricité se calcule donc comme suit : $e = 1 \quad \sqrt[3]{\frac{4 \times \pi \times ro^3}{G \times T \times (m1+m2)}}$

Pour obtenir des valeurs réalistes, nous avons pris le modèle des étoiles doubles Sirius. Ainsi nous obtenons une excentricité de l'ordre de 0,83. Cependant, afin de pouvoir observer les deux étoiles dans la fenêtre du logiciel Povray, nous avons été obligées de multiplier par $1O^{-11}$ 10^{-11} les valeurs x1, y1 et x2, y2 trouvées.

Afin d'être au plus près de la réalité, nous avons pris de nombreux points.

Le calcul de ces points est réalisé grâce à une boucle : pour t variant de 0 à T (la période), pour chaque T/200, Maple affiche les valeurs de x1, y1, x2, y2 associées à t.

*for i from 0 to evalf(T) by evalf(T/200) do printf(`%g %g %g %g %g\n`,i,x1(i)*1e-11,y1(i)*1e-11,x2(i)*1e-11,y2(i)*1e-11);od:*

Nous avons donc ensuite copié et collé les valeurs affichées dans un fichier texte, afin qu'elles soient récupérées par Scilab pour être ensuite utilisables dans Povray.

3.2.2. Programmation en Scilab

Il a donc ensuite fallu créer un programme pour récupérer les données sous Scilab et les transférer sous Povray. Nous avons repris le programme de l'an dernier en Scilab pour comprendre comment ce logiciel fonctionne. Nous avons aussi fait appel à Monsieur Thierry Bacon, qui a confirmé ce que l'on avait modifié dans le programme et nous a expliqué ce qui restait flou dans la programmation Scilab.

En d'autres termes, nous n'avions pas compris comment le programme Scilab pouvait transférer toutes les données.

Voici donc comment il a fallu programmer sous Scilab.

On crée une fonction appelée CreerPov. Les arguments de cette fonction, mis entre parenthèses lors de sa déclaration, sont : n, t, x, y, u, v, et z. n correspond au nombre de programmes Povray qui vont être créés pour transférer les données.

Un mini programme en Povray va être créé pour chaque ligne du tableau obtenu sous Maple. Dans notre cas, ayant choisi un nombre de points de 200, nous aurons donc 200 fichiers créés. Chacun de ses programmes verra son nom créé en fonction du numéro à 5 chiffres qui lui est attribué quand il est créé (numéro 00001 pour le premier, 00002 pour le deuxième etc.)

Ensuite, t correspond au temps, x, y, u, v sont les coordonnées des deux étoiles doubles au cours du temps (x, y pour la première et u, v pour la seconde). En effet, Povray n'accepte pas les variables avec des numéros (par exemple : x1).

De même, on déclare la variable z de façon à ce que la valeur 0 lui soit automatiquement associée. En effet, même si les équations de mouvement sont planes, le logiciel Povray demande systématiquement 3 variables.

NewCar=strsubst(NewCar,'$v',sprintf('%e',v));

NewCar=strsubst(NewCar,'$z',sprintf('%e',0));

Le signe $ renvoie à la valeur des variables.

On commence par créer le nom du fichier Povray, nom qui contient un numéro comme expliqué juste au dessus. Ensuite, on transfère les données du fichier source (ellipse_base1.pov) dans le nouveau fichier que l'on crée. Enfin, on récupère la valeur de t, de x, de y, de u et de v pour cette ligne du programme.

3.2.3. Programmation Povray

Pour cette partie de la programmation, nous avons repris le même fond étoilé que l'année dernière. En effet, nous l'avons également trouvé sur un site de programmation.

Cela dit, nous avons ensuite dû programmer différemment de l'an dernier, puisque les deux étoiles sont mobiles dans notre cas.

De plus, grâce à Scilab, le logiciel récupérait les valeurs x, y, u, v dans l'ordre. De ce fait, le temps n'intervient plus.

Nous avons tout d'abord déclaré les deux variables pour chaque étoile. Nous avons choisi de représenter les deux étoiles par des sphères de différentes couleurs. Ensuite nous avons associé à chacune la translation qu'elle doit effectuer.

```
// première étoile

sphere { <0.0, 0.0, 0.0>, 2
  finish {
    ambient 0.2
    diffuse 0.8
    phong 1
  }
  pigment { color red 1 green 0 blue 0 }
  translate<$x,$y,$z>
}

// deuxième étoile

  sphere { <0.0, 0.0, 0.0>, 2
  finish {
    ambient 0.2
    diffuse 0.8
    phong 1
  }
  pigment { color red 1 green 1 green 1  }
  translate<$u,$v,$z>
}
```

Ici, $u, $v, $z renvoient aux valeurs que l'on trouve dans le fichier texte.

Après avoir fini le programme ellipse_base1.pov, il nous fallait exécuter le programme Scilab, celui-ci étant programmé de manière à créer un fichier Povray pour chaque valeur (associée à un t donné) de x, y, u, v.

Une fois tous les fichiers créés, un autre problème s'est présenté.

Nous utilisions la fonction « queue » de Povray, qui sert à mettre bout à bout plusieurs fichiers. Cependant, cette fonction ne marchait pour nous que pour 33 fichiers, alors que nous en avions 200 à aligner. Nous avons donc repris rendez vous avec Mr Bacon afin de voir comment résoudre ce problème. Il nous a conseillé de tenter de créer un fichier que nous exécuterions. Le principe de fonctionnement de ce fichier est simple : il doit ouvrir chacun des 200 fichiers en Povray, pour ensuite les exécuter et créer autant d'images, dans le format « .bmp ».

Nous avons suivi ce conseil, et créé, avec son aide, un fichier que nous pourrions exécuter sous Windows.

Une fois lancé, le programme a créé une image en .bmp associée à chaque ligne de valeurs calculées à la base par Maple.

Il ne restait plus qu'à regrouper toutes ces images, les associer en créant un fichier « .avi ».

Nous avons réalisé l'assemblage grâce à Windows Movie maker, qui nous a permis d'obtenir au final un film présentant les trajectoires des deux étoiles doubles sur une période.

3.2.4. *Problèmes rencontrés*

Le premier problème que nous avons rencontré pendant ce projet a concerné la programmation en Maple. La première version de l'étude théorique, sur laquelle nous avions basé nos calculs, s'est révélée être inexacte, du fait de l'utilisation de v0, qui n'a pas de réalité physique. Arnaud et Mylène ont remédié à ce problème, et nous avons dû modifier tout le programme Maple, c'est-à-dire les équations ainsi que l'expression de la période.

En outre, nous avions commencé à travailler sur Maple en salle informatique à l'INSA, mais les ordinateurs ont été changés et il n'y avait plus la même version de Maple sur les nouveaux ordinateurs.

Il a donc fallu télécharger une version de Maple plus ancienne sur nos ordinateurs, recommencer plusieurs fois cette manipulation puisque les versions téléchargées rencontraient des problèmes avec les calculs numériques.

Cependant, nous avions l'avantage de connaître déjà Maple, par le biais des cours de M12 au début du semestre. Ceci n'était pas le cas de Scilab et Povray, qui étaient deux logiciels, dont nous ignorions totalement le fonctionnement. Nous nous sommes basées sur le rapport de l'an dernier pour comprendre en gros la programmation, puis nous avons créé nos propres programmes.

Monsieur Bacon nous a beaucoup aidées pour cela, et nous tenons à le remercier pour sa disponibilité et le temps qu'il nous a accordé.

En effet, même si nous pouvions nous baser sur les travaux de l'an dernier, notre projet se distingue clairement du leur, et des problèmes se sont posés à nous concernant les fonctions à utiliser pour créer des images, mais aussi et surtout pour les assembler sous forme d'un film.

Le dernier problème a concerné le passage des images « .bmp » à un fichier « .avi ». Il existe un logiciel spécial appelé bmp2avi, mais Monsieur Bacon a tenté de l'utiliser pour notre cas et ceci ne fonctionnait pas. Il a alors trouvé un autre logiciel sur son ordinateur qui permettait de faire la conversion, mais une fois encore nous avons voulu télécharger ce logiciel sur nos ordinateurs mais n'y sommes pas parvenues à cause de virus contenus dans le fichier d'installation. Nous avons donc finalement utilisé Windows Movie Maker.

Le film publié, on peut voir les deux étoiles, l'une rouge, l'une jaune se déplaçant l'une part rapport à l'autre. On se rend d'ailleurs compte que l'une tourne autour de l'ellipse plus rapidement, c'est évidemment celle qui a le poids le plus faible. Le mouvement des étoiles est représenté dans le référentiel du point fictif, étant donné que l'application Java montre le mouvement dans le référentiel terrestre.

De même, nous avons reproduit la simulation pour une étoile six fois plus grosse que l'autre. On remarque là aussi que l'étoile la plus petite tourne autour de la plus grosse. Ceci s'explique par le fait que le centre de masse est beaucoup plus proche de l'étoile la plus grosse.

3.3. Programmation en Java

3.3.1. Points de départ et difficultés

Dans cette partie, la source de départ est le fruit de l'étude théorique faite précédemment. Elle utilise les différentes équations de la trajectoire et les différentes variables, c'est-à-dire:

$$x = a(\cos E - e)$$
$$y = a\sqrt{1-e^2}\,\sin E$$
$$\frac{2\pi}{T}(t - t_P) = E - e\sin E \qquad e = \left|1 - \frac{r_0(v_0 \sin\alpha)^2}{G(m_1+m_2)}\right| \qquad T = 2\pi\sqrt{\frac{r_0}{G(m_1+m_2)(1-e)^3}}$$

$$\overrightarrow{GM_1} = -\frac{m_2 a}{m_1+m_2}\left[(\cos E - e)\overrightarrow{e_x} + \left(\sqrt{1-e^2}\sin E\right)\overrightarrow{e_y}\right]$$
$$\overrightarrow{GM_2} = \frac{m_1 a}{m_1+m_2}\left[(\cos E - e)\overrightarrow{e_x} + \left(\sqrt{1-e^2}\sin E\right)\overrightarrow{e_y}\right]$$

Dans notre cas, le problème majeur pour la résolution de ces équations, est l'équation qui relie le temps avec l'anomalie excentrique(E), puisqu'elle n'admet pas de résolution analytique, mais uniquement une solution numérique (un tableau de E et les valeurs du temps).

Pour résoudre cette difficulté, nous avons choisi le java, qui est langage interprété/portable (facile a utiliser par rapport aux autres langages comme le C et le pascal).

3.3.2. Méthode utilisée

3.3.2.1. Partie mathématique

Tout d'abord, considérons l'équation suivante : $\frac{2\pi}{T}(t - t_P) = E - e\sin E$

La méthode la plus logique et la plus adaptée pour ce genre d'équations est la dichotomie.
En effet, cette méthode est un algorithme mathématique qui permet de chercher le zéro d'une fonction continue sur un intervalle [a,b].

Pour pouvoir appliquer cette méthode il faut que la fonction vérifie 2 conditions:

- f(a) et f(b) doivent être de signes opposés.

-f(x)=0, a<x<b doit admettre une unique solution.

Mais comment appliquer cela sur notre équation?

On prend tp=0, passage de l'étoile au périgée au temps t=0.

Trouver E à un instant t fixé revient à trouver : $f(E) = 0$

avec : $$f(E) = E - e\,sin(E) - \frac{2\pi}{T}t$$

et sa dérivée : $$f'(E) = 1 - e\cos(E)$$

Pour t fixé (0<t<T), la fonction est évidemment continue sur l'intervalle [0,2π], et f(0) et f(2π) sont bien de signes opposés. De plus, par un simple calcul, on trouve que la dérivée est positive. La fonction est donc strictement monotone ce qui assure l'unicité de la solution.

Ici, toutes les conditions sont réunies pour pouvoir employer cette méthode.

Pour le reste, c'est-à-dire les autres méthodes utilisées pour obtenir la simulation, sont des méthodes de programmation java et sont détaillées dans le cd.

3.3.2.2. Partie informatique

Afin de pouvoir simuler le mouvement des étoiles doubles informatiquement, 16 classes ont été réalisées, chacune ayant un rôle prédéfini. On peut séparer ces classes en deux grands groupes. Le premier concerne l'aspect calculatoire de la simulation. Le deuxième concerne l'aspect graphique et permet de retranscrire des résultats à l'écran. Un troisième groupe de classe a été créé pour simplifier la programmation.

- **Premier groupe** : *Rassemble les classes Calcul, Calculateur, Constantes et Donnees.*

Tout d'abord, nous avons commencé par calculer les constantes à partir des conditions initiales, puis calculer E en fonction de t et enfin calculer les nouvelles coordonnées de l'étoile (classe Calcul).

Afin d'assurer la cohérence du programme, il nous a fallu utiliser une classe qui se charge d'organiser les calculs (classe Calculateur). C'est cette classe qui se charge de calculer les constantes et les nouvelles coordonnées des étoiles, à l'aide des fonctions de Calcul. Il se charge aussi de transférer ces résultats aux objets concernés, entre autre aux étoiles.

Nous avons réuni les constantes qu'il est nécessaire de calculer pour pouvoir démarrer la simulation (c'est-à-dire e, a et T) dans la classe Constantes. Pour initialiser ces constantes, on se sert de la classe Donnees qui contient les conditions initiales de la simulation (c'est-à-dire r0, alpha, T aussi, m1, m2, la vitesse du centre de gravité et la vitesse de simulation).

- **Deuxième groupe** : *Les classes Repere, RepereReel, Etoile, PointRepere, ZoneAffichage, Trajectoire, InterfaceAcquisitionDonnees, PanneauDessin et FenetreSimulation qui sont essentiellement dédiées au rendu graphique.*

Dans ce second groupe de classes, nous avons d'abord construit un repère muni d'une origine et d'une échelle. La classe Repère est le repère de la fenêtre en pixels et son objectif est d'assurer que les éléments sont affichés aux bonnes coordonnées sur l'écran. Il est l'un des composants d'une ZoneAffichage. Ce repère contient des coordonnées entières.

Afin de pouvoir placer les objets dans un repère qui correspond à la réalité physique, nous avons construit une classe RepereReel qui contient des coordonnées réelles. Il était nécessaire de le différencier de Repere car les entiers et les réels ne sont pas représentés de la même manière par la machine.

Pour ce qui est de la visualisation des étoiles à l'écran, la classe Etoile permet de représenter à chaque instant les données correspondant à l'étoile, c'est-à-dire ses coordonnées, son rayon (arbitraire car s'il ne l'était pas la plupart des simulations nous feraient observer le mouvement de deux pixels sur l'écran), et la couleur choisie pour la représenter ; le tout dans le but de les afficher simplement à l'écran.

Lors de la visualisation des étoiles, il nous a fallu également afficher le centre de gravité G, repéré par ses coordonnées mobiles (classe PointRepere).

Afin de coordonner l'ensemble des informations contenues dans les classes précédentes, la classe ZoneAffichage permet d'afficher les éléments dans une zone de la fenêtre. Munie d'un Repere, tout objet doit passer par cette classe pour pouvoir être afficher à l'écran, ceci afin que la représentation du mouvement des étoiles soit cohérente à l'écran.

Dans notre cas nous avons trois trajectoires à représenter. Celle de G, et celles des deux étoiles. C'est le rôle de la classe Trajectoire qui enregistre les coordonnées successives d'un objet. La trajectoire est ensuite tracée à l'écran en reliant les points enregistrés. C'est pour cette raison que si l'on zoome trop ou que la vitesse de simulation est trop élevée, on voit apparaitre des segments. Il est à noter que les coordonnées enregistrées sont celle qui correspondent à la réalité et non pas celles sur l'écran en pixels.

Nous avons choisi la programmation Java pour son interactivité, et la possibilité de construire une interface d'acquisition de données. La classe InterfaceAcquisitionDonnees correspond à cette fenêtre d'accueil. En effet, elle contient les objets nécessaires à l'acquisition des données auprès de l'utilisateur. Elle permet, de ce fait, de transmettre les données insérées par l'utilisateur à tous les objets qui en ont besoin pour fonctionner. De plus, la classe PanneauDessin correspond à l'unité graphique sur laquelle s'affichent les résultats de la simulation.

Pour une plus grande simplicité au niveau de la programmation, nous avons créé une classe FenetreSimulation. Cette classe permet d'afficher une fenêtre à l'écran qui contient le panneau sur lequel la simulation est affichée. Il est en effet plus délicat de récupérer les données et d'afficher la simulation sur la même fenêtre.

- **Troisième groupe** : *Les classes Contrôle et Animateur*

On peut voir qu'il reste deux classes qui n'appartiennent à aucune des deux catégories. En réalité c'est parce que leur seule utilité est de simplifier la programmation :

La classe Contrôle avait pour rôle d'afficher l'état de certaines données à certains moments pour pouvoir contrôler que le programme fonctionnait correctement, et dans le cas contraire de déterminer ce qui ne fonctionnait pas bien et pourquoi.

La classe Animateur quant à elle permet de gérer la simulation et l'interface d'acquisition des données indépendamment l'une de l'autre. Sans cette classe, lorsque la simulation tournait il était impossible d'effectuer quelque manœuvre que ce soit sur l'autre fenêtre.

4. CONCLUSIONS ET PERSPECTIVES

4.1. Conclusion générale

Ce projet nous a permis d'utiliser et d'approfondir les connaissances acquises de mécanique du point en première année. De plus nous avons pu aborder un phénomène physique concret et étudier les interactions et forces s'appliquant sur le système.

Nous avons commencé par rechercher des informations concernant les systèmes de deux points matériels, tout en analysant le projet réalisé l'année dernière. Puis nous avons défini les objectifs à atteindre. Ceci nous a donc obligés à séparer le travail pour être plus efficaces. Le projet nous a donc permis de nous familiariser avec le travail de groupe, d'autant plus que nous ne nous connaissions pas avant de réaliser le projet. Ainsi une cohésion de groupe a été créée.

L'organisation a nécessité le respect de délais imposés. Nous nous sommes rendu compte que les créneaux horaires hebdomadaires de P6-3 nous permettaient de faire le point et de mesurer l'avancée du travail. De ce fait nous pouvions adapter les objectifs en fonction de problèmes éventuels rencontrés. Cependant nous avons du à de nombreuses reprises nous rencontrer en dehors des créneaux prévus.

La réalisation de projet nécessite également l'intervention d'aide extérieure, comme par exemple par la prise de contact avec des observatoires ou des informaticiens. En effet l'aide des membres du groupe n'était parfois pas suffisante ; elle permettait seulement de régler les problèmes minimes.

4.2. Conclusions personnelles

Eve

Comme, très certainement, d'autres membres du groupe, j'ai choisi le projet des étoiles doubles car celui-ci me semblait intéressant à aborder. En effet, le fait de comprendre des phénomènes physiques m'intéresse particulièrement.

J'ai pu également me rendre compte que dans un groupe de travail, chacun cherche à mettre à disposition ce pour quoi, il est le plus doué. Nous avons d'ailleurs dû faire appel à des spécialistes. D'ailleurs, aller demander de l'aide peut apprendre beaucoup sur un sujet ou un outil qui n'est pas forcement familier. De plus, l'avancement du projet est très variable. Par exemple, le temps de se renseigner sur le sujet, de définir les objectifs et de régler les problèmes peut être plus ou moins long. Il est donc très important de gérer son temps en prévoyant ces possibles impromptus.

Coralie

Concernant l'étude des étoiles doubles, j'ignorais totalement de quoi il s'agissait, et j'avoue que la première réunion m'a fait très peur, quand j'ai constaté que le projet allait faire appel à mes connaissances en mécanique qui sont très minces ainsi qu'à des notions de programmation que je ne possédais pas vraiment.

Quand nous nous sommes partagé le travail, je me suis retrouvée à travailler avec Eve. Voyant qu'elle s'investissait dans le projet, je l'ai suivie et dépassé mes appréhensions.

Une autre point positif concernant le projet de P6-3 est que c'est le premier projet en science depuis 2 ans pour lequel le groupe ne s'est pas fait par affinité, mais par hasard. Je pense que ceci nous a poussés à nous montrer plus rigoureux ; contrairement aux projets de mathématiques ou d'informatique qui étaient plus flexibles.

Au final, je pense que notre projet a bien été mené à bout, et que l'organisation du travail que nous avons adoptée a porté ses fruits.

Zineb

Avant d'entamer ce projet, mes informations sur ce sujet étaient limitées à des informations tirées de la P2. Mais en cherchant, en écoutant les commentaires des autres membres du groupe, j'ai pu enrichir mes connaissances sur l'étude des deux étoiles doubles.

Un autre point positif et important de ce projet, c'est le fait que les membres du groupe étaient responsables et associatifs, je dirais même compétents. Il est vrai que nous avons eu des petits problèmes pour avancer, mais nous avons finalement réussi à nous en sortir. De plus, notre groupe était composé d'élèves de thématiques différentes, ce qui a été très enrichissant pour le projet: chacun pouvait apporter ses connaissances personnelles et ainsi combler les lacunes d'autres personnes.

Anastacia

Ce projet a été très enrichissant pour moi. En effectuant des recherches sur internet, j'ai appris beaucoup de choses sur les étoiles doubles, (notamment du point de vue de la physique, facette moins connue pour moi). Quand j'ai choisis ce projet, je pensais qu'il allait y avoir également une partie pratique (des observations par exemple); le fait de pouvoir observer des étoiles aurait pu être une bonne stimulation.

A part cela, je pense que le fait de travailler en équipe réduite, a permis de conserver une bonne ambiance et donc une bonne qualité de travail.

Arnaud

Ce projet m’a au final beaucoup apporté tant sur un point de vue scolaire, que sur un niveau personnel. Il m’a appris à travailler en groupe, ce dont je ne suis habitué pas à faire, à se partager le travail et le mettre en commun plus tard. Dans ce projet, j’ai essentiellement travaillé sur les équations et sur l’aspect physique, domaine par lequel je suis très intéressé.

Cela m’a permis d’approfondir ce que l’on a abordé en P2 notamment. Il m’a paru très intéressant de s’attaquer à un phénomène concret et de pouvoir le simuler sur ordinateur. La partie informatique est d’ailleurs très intéressante puisqu’elle permet de bien visualiser le phénomène d’interactions entre les astres. Finalement, c’est un projet où je ne me suis pas du tout ennuyé et il fut très agréable de travailler avec ce groupe.

Mylène

J’ai choisi ce projet concernant les étoiles doubles car je ne connaissais pas du tout ce phénomène physique, même si j’en avais déjà entendu parler.

Ce projet a été très enrichissant tant sur le plan du contenu du travail que sur celui du travail d’équipe. En effet, les connaissances de chacun ont pu être mises en avant grâce au travail de groupe, ce qui a permis d’installer dès le début du projet, une ambiance agréable et détendu. De plus, l’idée de travail avec de nouvelles personnes sur un projet, c’est-à-dire des personnes que l’on ne connaissait pas, a été réellement enrichissante.

Enfin, il est vrai que le projet ne se prêter pas réellement à des expériences pratiques, mais il serait intéressant de pouvoir adapter le projet à cela, pour les années à venir par exemple.

5. BIBLIOGRAPHIE

Etude théorique

- **Livres**

- Les Nouveaux précis Bréal – Mécanique PCSI (emprunté à M. Montier).
- Préférence Prépas-Pierre GRECIAS, Jean-Pierre MIGEON-TEC&DOC, Lavoisier.
- Hprépa-Physique 1ère année MPSI, PCSI, PTSI-Jean-Marie BREBEC, HACHETE 2005.

- **Sites Internet**

- Les étoiles doubles : http://fr.wikipedia.org/wiki/%C3%89toile_binaire.
- L'ellipse : http://fr.wikipedia.org/wiki/Ellipse_(math%C3%A9matiques).
- Les anomalies vraie et moyenne : http://kaekoda.free.fr/bup/bup2_html/bup2_htmlse6.html.
- Les lois de Kepler : http://www.phy6.org/stargaze/Fkepl2nd.htm.
- Les systèmes de deux points matériels : http://www.zonegeeks.com/Documents/Cours/prepa/physique/systeme-de-deux-points-materiel.pdf

Programmation Povray et Java

- **Sites internet**

    - Documentation Java : http://java.sun.com/javase/6/docs/api/
    - Téléchargement de Maple : http://www.dynamaths.com/logiciels/calculsym/maple-37.html
    - Téléchargement de SciLab : http://www.scilab.org/
    - Téléchargement de Povray : http://www.infos-du-net.com/telecharger/PovRay,0301-1308.html
    - Tutoriel Povray : http://povray.free.fr/
      http://www.f-lohmueller.de/pov_tut/pov__fra.htm .

- **Autres sources**

- Maple : cours de M12.
- Programmer en java-Claude DELANNOY-EYROLLES 2007

6. ANNEXES

6.1. Listings des programmes réalisés

- ✓ Programmation en Java
- ✓ Programmation sous Maple-Povray