

**PHS4700**

**Physique pour les applications multimédia**

Automne 2017

PAGE COUVERTURE **OBLIGATOIRE** POUR TOUS LES DEVOIRS

Numéro de devoir : 02

Numéro de l’équipe : 07

Numéro du groupe : 01

|  |  |  |
| --- | --- | --- |
| Nom: Bourgault | Prénom : Gabriel | matricule: 1794069 |
| Signature : |  |  |
| Nom: Chan | Prénom : Kevin Ka Hin | matricule: 1802812 |
| Signature : |  |  |
| Nom: Nguyen | Prénom : Kenny | matricule: 1794914 |
| Signature : |  |  |
| Nom: Silva-Pinto | Prénom : Nuno | matricule: 1799144 |
| Signature : |  |  |

Table des matières

[Introduction 2](#_Toc496468338)

[Théorie et équations 3](#_Toc496468339)

[Équations théoriques 3](#_Toc496468340)

[Équations du mouvement à résoudre 3](#_Toc496468341)

[Équations pour déterminer l’arrêt de la simulation (collision) 4](#_Toc496468342)

[Méthode de résolution des équations du mouvement et justification 4](#_Toc496468343)

[Intervalles de temps t choisis pour la résolution et justification 4](#_Toc496468344)

[Présentation et analyse des résultats 5](#_Toc496468345)

[Option 1 : Force gravitationnelle seulement 6](#_Toc496468346)

[Option 2 : Force gravitationnelle et force visqueuse 7](#_Toc496468347)

[Option 3 : Force gravitationnelle, force visqueuse et force de Magnus 8](#_Toc496468348)

[Vérifications effectuées pour assurer la précision de nos simulations 9](#_Toc496468349)

[Conclusion 10](#_Toc496468350)

[LOREM IPSUM FILLER 10](#_Toc496468351)

# Introduction

Ce deuxième devoir a pour but d’étudier la trajectoire d’une balle au tennis sur table. Pour ce faire, nous avons programmé une application permettant la simulation de cette trajectoire. De plus, les principales caractéristiques de jeu seront simulées. Cela inclut un système de coordonnées de références où l’origine se trouve au sol alignée avec le coin droit de la table, une surface de jeu, un filet ainsi qu’une balle sphérique.

Nous nous intéressons alors à trois situations précises :

1. Seule la force gravitationnelle agit sur la balle.

2. La force gravitationnelle ainsi qu’une force de frottement visqueux agissent sur la balle.

3. La force gravitationnelle, la force de frottement visqueux ainsi qu’une force de Magnus agissent sur la balle.

Chacune de ces situations est testée à l’aide de quatre coups différents. Ainsi, au total, nous avons 12 simulations. Finalement, une simulation se termine lorsque la balle touche le filet, lorsque la balle touche la surface de la table ou lorsque la balle touche le sol.

Pour pouvoir simuler les trois situations précédentes, nous avons programmé une fonction Matlab nous permettant de tout d’abord déterminer si le coup a réussi ou non. Dans le cas échéant, on différencie un coup échoué en trois catégories : la balle frappe la table du côté du joueur, la balle frappe le filet ou la balle est frappée à l’extérieur de la table (après avoir touché le sol). Celle-ci détermine aussi les variables suivantes lorsque la simulation se termine : le temps (en secondes), les positions finales en x, en y et en z (en m) du centre de masse de la balle, le vecteur vitesse final du centre de masse de la balle (en m/s).

Ainsi, le rapport suivant contient la théorie ainsi que les équations auxquelles nous nous sommes tournés pour nous aider à programmer la fonction Devoir2.m. Cette section sera par la suite appuyée par la présentation ainsi que l’analyse des résultats que nous avons obtenus. Finalement, une brève discussion des problèmes rencontrés servira de conclusion à ce devoir.

# Théorie et équations

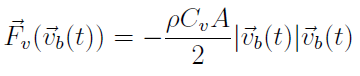
## Équations théoriques

La force gravitationnelle sur la balle est donnée par l’équation 1. Celle-ci indique que la force ne s’applique sur l’axe des z. Cette équation est utilisée dans les trois situations.



Équation 1

La force de frottement visqueux est donnée par l’équation 2. Certaines valeurs ont été fournies dans l’énoncé du devoir : la masse volumique de l’air (air = 1.2 kg/m3), l’aire efficace de la balle de la balle ( A = ( R2b) ) ainsi que Cv = 0.5. b correspond à la vitesse de la balle. Cette équation est utilisée dans la situation 2 et la situation 3.



Équation 2

La force de Magnus est donnée par l’équation 3. La valeur de CM est égale à 0.29 : elle nous a été fournie par l’énoncé du devoir. correspond à la masse volumique de l’air : c’est la même qu’à l’équation 2. Le rayon Rb, la vitesse angulaire b ainsi que b correspondent à ceux de la balle. Cette équation est utilisée dans la situation 3 uniquement.



Équation 3

## Équations du mouvement à résoudre

## Équations pour déterminer l’arrêt de la simulation (collision)

Afin de déterminer l’arrêt de la simulation, nous avons opté pour une formule simple de détection de collision :

*Il y a collision lorsque au moins une des composantes de la balle est en intersection avec un des plans.*

Dans la situation, nous avons défini 3 plans : le filet, la table et le sol. Ainsi, lorsque nous détections qu’une des extrémités de la balle était contenue dans les bornes du plan étudié, on savait que nous avions atteint une collision, et donc une condition d’arrêt de la simulation.

Le pseudocode est le suivant :

Si bornesXmin > cmX + rayon || bornesXmax < cmX – rayon

Alors collision

Si bornesYmin > cmY + rayon || bornesYmax < cmY – rayon

Alors collision

Si bornesZmin > cmZ + rayon || bornesZmax < cmZ – rayon

Alors collision

Sinon, pas de collision

## Méthode de résolution des équations du mouvement et justification

## Intervalles de temps t choisis pour la résolution et justification

Après quelques essais et quelques ajustements suite à ces derniers, nous avons finalement opté pour un t = 0.00001s. Cette valeur nous permettait d’avoir un moins haut risque de manquer un enregistrement de collisions tout en gardant une précision adéquate pour la position finale.

# Présentation et analyse des résultats

**Tableau 1 : Conditions initiales pour les quatre coups à simuler**

|  |  |  |  |
| --- | --- | --- | --- |
| **Essai** | **rbi (m)**  **Position initiale du cm** | **vbi (m/s)**  **Vitesse initiale du cm** | **wbi (rad/s)**  **Vitesse angulaire** |
| 1 | (0,00 0,50 1,10) | (4,00 0,00 0,80) | (0,00 − 70,00 0,00) |
| 2 | (0,00 0,40 1,14) | (10,00 1,00 0,20) | (0,00 100,00 − 50,00) |
| 3 | (2,74 0,50 1,14) | (−5,00 0,00 0,20) | (0,00 100,00 0,00) |
| 4 | (0,00 0,30 1,00) | (10,00 − 2,00 0,20) | (0,00 10,00 − 100,00) |

**Tableau 2 : Résultats des simulations**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Option** | **Essai** | **Coup** | **tf (s)**  **Temps de la simulation** | **rbf (m)**  **Position finale du cm** | **vbf (m/s)**  **Vitesse finale du cm** |
| **1 : Force gravitationnelle seulement** | 1 | 2 | 0,00260 | (1,3572 0,5000 0,8073) | (4,00 0,00 -2,53) |
| 2 | 3 | 0,00316 | (5,0086 0,90009 0,0110) | (10,0 1,00 -4,71) |
| 3 | 2 | 0,00232 | (1,3886 0,5000 0,8361) | (-5,00 0,00 -2,45) |
| 4 | 3 | 0,00306 | (4,6971 -0,6394 0,0129) | (10,0 -2,00 -4,40) |
| **2 : Force gravitationnelle et force visqueuse** | 1 | 1 | 0,00266 | (1,2956 0,5000 0,7777) | (3,32 0,00 -2,51) |
| 2 | 0 | 0,00248 | (2,5725 0,6573 0,7751) | (6,99 0,70 -2,43) |
| 3 | 2 | 0,00244 | (1,3824 0,5000 0,7834) | (-4,12 0,00 -2,50) |
| 4 | 2 | 0,00172 | (1,3523 0,0295 0,9251) | (8,28 -1,66 -1,17) |
| **3 : Force gravitationnelle, force visqueuse et force de Magnus** | 1 | 1 | 0.00270 | (1,2743 0,5000 0,7749) | (3,02 0,00 -2,40) |
| 2 | 0 | 0,00240 | (2,4906 0,6212 0,7789) | (7,47 0,49 -2,70) |
| 3 | 1 | 0,00228 | (1,5832 0,5000 0,7776) | (-3,92 0,00 -2,83) |
| 4 | 2 | 0,00172 | (1,3560 0,0233 0,9245) | (8,32 -1,78 -1,18) |

### Option 1 : Force gravitationnelle seulement

## Option 2 : Force gravitationnelle et force visqueuse

## Option 3 : Force gravitationnelle, force visqueuse et force de Magnus

## Vérifications effectuées pour assurer la précision de nos simulations

Pour l’option 1, la plus simple, nous avons effectué les calculs à la main pour comparer avec le résultat obtenu par la simulation. Puisque les résultats étaient concluants, nous avons gardé les mêmes conditions (i.e. intervalle de temps) pour les autres options. Nous avons aussi vérifié que le point de départ de chacune des courbes correspondait aux conditions initiales de chacun des essais (**Tableau 1**). Nous avons aussi vérifié visuellement les collisions : cela nous a permis de valider la fiabilité de notre algorithme de détection de collision.

De plus, nous avons fait des hypothèses à l’aide des données brutes pour l’ajout de la force de Magnus : nous avons estimé qu’un *backspin* allait permettre à l’option 3 (Magnus) d’avoir une position finale plus loin que les autres. C’est ce qui arrive dans l’essai #3. ***BULLSHIT ALERT.***

# Conclusion

Après avoir eu un notre baptême du feu, suite au premier devoir, nous étions encore une fois en territoire inconnu : il fallait apprendre de nouvelles fonctions Matlab comme l’affichage de courbes. De plus, il était nécessaire de comprendre comment utiliser la méthode de Runge-Kutta. Cette tâche se révélait plus difficile que prévu puisque nous n’avions pas suivi le cours MTH1115 – Équations différentielles. En effet, ce cours n’est plus obligatoire à l’obtention du baccalauréat en génie logiciel. Finalement, nous avons réalisé qu’un exemple applicatif de la méthode se trouvait dans la boîte à outils du cours, alors cela nous a beaucoup aidés à comprendre la méthode.

Vérifier la validité des calculs était une tâche difficile. En effet, il y avait des calculs trop compliqués pour être faits à la main comme ceux de la méthode de Runge-Kutta. Cependant, nous avons fait des calculs pour l’option 1, la plus facile à évaluer, pour s’assurer que nous comprenions bien la base du problème. Puisque la force gravitationnelle revient dans les trois options à évaluer, nous trouvions cette validation pertinente à faire.

Finalement, se réunir en équipe pour ce devoir était difficile puisque nous devions attraper les balles que les différents autres cours nous lançaient en mi-session. Cela ne nous a tout de même pas empêché de terminer ce devoir.