**PHS4700**

**Physique pour les applications multimédia**

Automne 2015

PAGE COUVERTURE **OBLIGATOIRE** POUR TOUS LES DEVOIRS

**Numéro de devoir : 03**

**Numéro de l’équipe : 14**

|  |
| --- |
| Nom: Rose Prénom : Alexandre matricule: 1580973  Signature : |
| Nom: Mainville Prénom : David matricule: 1636075  Signature : |
| Nom: Gosselin Prénom : Antoine matricule: 1588443    Signature : |
| Nom: Farvacque Prénom : Dylan matricule: 1684271  Signature : |

Table des matières

[I – Description du problème 2](#_Toc435460407)

[II – Équations importantes 3](#_Toc435460408)

[III – Méthode de résolution des équations du mouvement 4](#_Toc435460409)

[IV – Description du logiciel 5](#_Toc435460410)

[V – Résultats obtenus 6](#_Toc435460411)

[VI – Analyse des résultats obtenus 7](#_Toc435460412)

[VII - Discussions sur le devoir 8](#_Toc435460413)

# I – Description du problème

Pour ce présent devoir, notre tâche consiste en la simulation d’un tir d’une boite de conserve remplie d’air en chute libre par une balle. Notre simulation devra être capable de fournir les composantes en X, Y et Z des vitesses et centres de masse selon le temps des différents objets. Enfin, en cas de collision entre la balle et la boite, notre simulation devra indique la vitesse du centre de masse ainsi que la vitesse angulaire après la collision entre la boite et la balle.

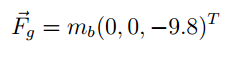
Nous nous intéresserons à quatre scenarios distincts :

* La boite de conserve n’as pas de vitesse angulaire et le lancer de la balle ne possède que des vitesses selon l’axe des z et des z,
* Toujours avec une boite conserve sans vitesse angulaire, le lancer de la balle aura aussi une vitesse en y
* Dans le troisième scenario, la balle n’aura qu’une vitesse en x et en y et la boite de conserve aura une vitesse angulaire selon l’axe des y
* Enfin, dans le dernier scenario, la boite aura toujours une vitesse angulaire selon l’axe des y et le lancer de la balle aura une vitesse non nulle selon tous les axes.

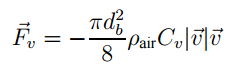
Dans le présent rapport, un bref rappel des équations nécessaires à la simulation sera fait. Ensuite, notre simulation développée sur MATLAB sera présentée. Aussi, nous présenterons nos résultats et en ferons une analyse détaillée. Enfin, nous conclurons par une discussion sur les problèmes que nous avons dû surmonter au cours du devoir en ce qui a trait à la programmation et aux simulations.

# II – Équations importantes

1. Équations du mouvement a résoudre
2. Équations qui contrôlent la simulation
3. Détection de collision et vitesses finales des objets

La force gravitationnelle sur la balle est donnée par une seule multiplication de constante sur l’axe de la hauteur (dans la situation présente l’axe des Z).

Équation 1

 La force de frottement visqueux est donnée par la multiplication du coefficient de frottement visqueux (air) avec la densité de l’air à 30 °C (Cv), la vitesse () multiplié avec sa norme et la moitié de l’aire de la balle ( d2b/8).

Équation 2

# III – Méthode de résolution des équations du mouvement

1. Justification du pas
2. Description des vérifications effectuées pour assurer la précision

# IV – Description du logiciel

Les simulations sont entièrement réalisées à l’aide du logiciel MATLAB. Tout d’abord on initialise un vecteur représentant la zone des buts à l’aide des coordonnées données dans l’énoncé. Suite à quoi on exécute les différentes simulations avec des données en entrée permettant de définir les options désiré (voir les différents scénarios possibles décrits en introduction).

# V – Résultats obtenus

# VI – Analyse des résultats obtenus

.

# VII - Discussions sur le devoir

Dans le cadre de ce second laboratoire, nous avons eu quelques défis à surmonter. Tout d’abord, nous avions remarqué que les calculs pour l’ensemble des simulations étaient très semblables, nous avons donc décidé de découper tout cela dans des classes spécifiques et utiliser l’héritage pour dupliquer le moins possible. Or, MATLAB est avant tout un langage de calcul avant d’être un langage de programmation oriente objet. Nous avons donc dû nous adapter à celui-ci afin d’en tirer le plus d’avantages possibles et de rendre notre code plus lisible et facile d’accès