POLYTECHNIQUE MONTRÉAL
LE GÉNIE EN PREMIÈRE CLASSE

PHS4700
Physique pour les applications multimédia

PAGE COUVERTURE **OBLIGATOIRE** POUR TOUS LES DEVOIRS

Numéro du groupe : 01

Numéro de l'équipe : 11

Numéro de devoir : 2

Nom: Yuan	Prénom : Ming Xiao	matricule: 1949477
Signature :		
Nom: Podgorica	Prénom : Andi	matricule: 1955913
Signature :		
Nom: Charles	Prénom : Cassy	matricule: 1947025
Signature : Cassy Charles		
Nom: Houidhek	Prénom : Noursen	matricule: 1949635
Signature : Noursen Houidhek		

Table des matières

Introduction

Dans le devoir 2 du cours de PHS4700, Physique pour application multimédia, la discipline que nous étudions est la cinématique. La cinématique est définie comme l'étude du mouvement des solides et des particules. La résolution numérique d'EDO a été vue en cours et nous allons s'en servir pour faire cette étude. En se servant de cette méthode, nous étudierons dans ce travail le mouvement d'une fusée possédant un propulseur.

D'abord, nous avons programmé une application permettant de simuler le lancement de cette fusée de la surface de la Terre vers l'espace. Le but de ce devoir est de programmer une fonction Matlab (Octave) afin de tracer la trajectoire de la fusée de son point de départ avec un angle donné jusqu'à ce qu'elle dépasse la distance de 10 000 km du centre de la Terre ou qu'elle s'écrase sur celle-ci. La simulation possède alors deux contraintes pour qu'elle prenne fin. Ces deux contraintes sont présentées dans la section 1.5.

Dans la section 1, la théorie ainsi que les équations nous permettant de faire la simulation de cette fusée sont présentées. Ensuite, dans la section 2, l'analyse de nos résultats est mise en évidence en profondeur. Pour terminer, dans la conclusion, les difficultés rencontrées dans ce devoir sont également abordées.

1. Théorie et équations

1.1. Méthodologie utilisée

En sachant que le but de ce devoir est de tracer la trajectoire de la fusée, nous avons utilisé la solution numérique. L'algorithme de Runge-Kutta d'ordre quatre a été privilégié car elle offre une meilleure précision que la méthode Euler. À l'aide de cette méthode, nous avons trouvé la position $r_c(t)$ du centre de masse, la vitesse $v_c(t)$ du centre de masse et la vitesse angulaire *ω(t)* du centre de masse de la fusée à chaque intervalle de temps Δt pendant la simulation. Afin d'obtenir ces données, l'équation suivante a été utilisée :

$$\frac{d\vec{q}(t)}{dt} = \vec{g}(\vec{q}(t), t)$$

Avec des conditions initiales:

À t=to=0s, $\vec{q}(t_0) = \vec{q}_0$

Alors, pour chaque instant *t,* l'algorithme de Runge-Kutta a été utilisée afin de faire le calcul du vecteur *q(t)*. D'abord, ce vecteur est construit de trois composantes du vecteur vitesse à l'instant *t*. Ensuite, le vecteur *q(t)* possède trois composantes du vecteur position de la fusée à cet instant.

De plus, ce vecteur possède trois composantes de la vitesse angulaire à l'instant *t* ainsi que trois composantes de la position angulaire à l'instant *t.*

$$\vec{q}(t) = \begin{pmatrix} q_1(t) \\ q_2(t) \\ q_3(t) \\ q_4(t) \\ q_5(t) \\ q_6(t) \\ q_7(t) \\ q_8(t) \\ q_9(t) \\ q_{10}(t) \\ q_{11}(t) \\ q_{12}(t) \end{pmatrix} = \begin{pmatrix} v_{c,x}(t) \\ v_{c,y}(t) \\ v_{c,z}(t) \\ x_c(t) \\ y_c(t) \\ z_c(t) \\ \omega_x(t) \\ \omega_y(t) \\ \omega_z(t) \\ \Omega_x(t) \\ \Omega_y(t) \\ \Omega_z(t) \end{pmatrix} \begin{matrix} \left.\vphantom{\begin{matrix} a \\ a \\ a \end{matrix}}\right\} \vec{v_c}(t) \\ \left.\vphantom{\begin{matrix} a \\ a \\ a \end{matrix}}\right\} \vec{r_c}(t) \\ \left.\vphantom{\begin{matrix} a \\ a \\ a \end{matrix}}\right\} \vec{\omega}(t) \\ \left.\vphantom{\begin{matrix} a \\ a \\ a \end{matrix}}\right\} \vec{\Omega}(t) \end{matrix}$$

$$\vec{g}(\vec{q}, t) = \begin{pmatrix} a_{c,x}(\vec{q},t) \\ a_{c,y}(\vec{q},t) \\ a_{c,z}(\vec{q},t) \\ q_1(t) \\ q_2(t) \\ q_3(t) \\ \alpha_x(\vec{q},t) \\ \alpha_y(\vec{q},t) \\ \alpha_z(\vec{q},t) \\ \omega_x(t) \\ \omega_y(t) \\ \omega_z(t) \end{pmatrix} \begin{matrix} \left.\vphantom{\begin{matrix} a \\ a \\ a \end{matrix}}\right\} \vec{a_c}(t) \\ \left.\vphantom{\begin{matrix} a \\ a \\ a \end{matrix}}\right\} \vec{v_c}(t) \\ \left.\vphantom{\begin{matrix} a \\ a \\ a \end{matrix}}\right\} \vec{\alpha}(t) \\ \left.\vphantom{\begin{matrix} a \\ a \\ a \end{matrix}}\right\} \vec{\omega}(t) \end{matrix}$$

Dans le fichier *Devoir2.m*, l'algorithme de Runge-Kutta implémenté prend en paramètres:

- Le vecteur : $q(t_{n-1}) = [\, v_{c,x}\ (t_{n-1})\ \ v_{c,y}\ (t_{n-1})\ \ v_{c,z}\ (t_{n-1})\ \ r_{c,x}\ (t_{n-1})\ \ r_{c,y}\ (t_{n-1})\ \ r_{c,z}\ (t_{n-1})\ \ \omega_x(tn-1)\ \ \Omega_x(tn-1)\ ]$
  Ici, v_c représente le vecteur vitesse du centre de masse, r_c représente le vecteur position du centre de masse, ω représente la vitesse angulaire et Ω représente la position angulaire. *t* représente l'instant présent lors de la simulation et t_{n-1} représente l'instant *t* - Δt. Dans ce TP, la vitesse angulaire est seulement en x ainsi que la position angulaire, car pour les vecteurs y et z, leurs valeurs correspondent à 0. À l'instant t=0s, ce vecteur est trouvé grâce aux données entrées par l'usager.
- La vitesse angulaire de la fusée
- La différence de temps entre deux instants *t* successifs: Δt
- La fonction *g* dépendant de l'accélération de la fusée et calculant le vecteur donné par : $g(t_n) = [\, a_{c,x}(t_n)\ \ a_{c,y}(t_n)\ \ a_{c,z}(t_n)\ \ v_{c,x}(t_n)\ \ v_{c,y}(t_n)\ \ v_{c,z}(t_n)\ \ \alpha_x(t_n)\ \ \omega_x(t_n)\ ]$. Comme pour le vecteur $q(t_{n-1})$, l'accélération angulaire α et la vitesse angulaire ω sont seulement en x, car pour les vecteurs y et z, leurs valeurs correspondent à 0.

Dans la section 1.2, nous présentons les fonctions utilisées pour le calcul de l'accélération $a_c(t)$. Cette accélération dépend des forces agissant sur la fusée.

- Pour chaque instant *t*, l'algorithme de Ruge Kutta donne :
- $q(t_n) = [\, v_{c,x}(t_n)\ \ v_{c,y}(t_n)\ \ v_{c,z}(t_n)\ \ r_{c,x}(t_n)\ \ r_{c,y}(t_n)\ \ r_{c,z}(t_n)\ )\ \omega_x(t_n)\ \Omega_x(t_n)]$
- $v_c(t_n)$ est le vecteur vitesse du centre de masse à l'instant *t*
- $r_c(t_n)$ est le vecteur position du centre de masse
- $\omega(tn)$ est le vecteur vitesse angulaire de la fusée
- $\Omega(tn)$ est le vecteur position angulaire de la fusée

- $t_n = t$ est l'instant présent de la simulation

1.2. Calcul de l'accélération

Afin de calculer l'accélération, nous avons utilisé la deuxième loi de Newton pour les différentes forces subies par la fusée. Pour ce faire, nous avons utilisé $\sum \vec{F} = m\vec{a}$ avec m la masse du solide et $\vec{F}$ les différentes forces subies par la fusée.

1.2.1. Force gravitationnelle

La force gravitationnelle $\overrightarrow{F_g}$ agit sur la fusée. Soit $\sum \vec{F} = m\vec{g}$ avec $\vec{g} = -GM_T \frac{\vec{r}}{||\vec{r}||3}$ avec m, la masse de la fusée. Nous avons d'abord calculé $\vec{g}$ avec la constante gravitationnelle et la masse de la Terre données dans l'énoncé ainsi que $\vec{r}$, la position de la fusée dans le référentiel global. Cette force est calculée dans le référentiel global.

1.2.2. Force de frottement visqueux

La fusée subit une force de frottement visqueux $\overrightarrow{F_{vis}} = -\frac{1}{2}A\rho C_{vis}|\vec{v}|\vec{v}$ avec $\rho = \rho_O e^{(R_T-|\vec{r}|)/h_o}$, $\rho_o = 1.275\frac{kg}{m^3}, ho = 7200m$ et $A = (\pi R^2 \cos\alpha + 2RHsin\alpha)$.

L'angle α représente l'angle entre le vecteur vitesse et l'axe de la fusée. La figure suivante montre l'angle α

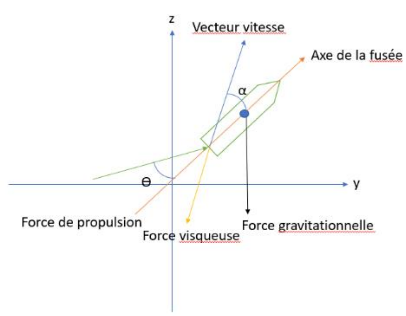

FIGURE 1: REPRÉSENTATION DE L'ANGLE ALPHA SUR LA FUSÉE

Le vecteur vitesse correspond au vecteur de vitesse du centre de masse de la fusée qui est déjà appliqué dans le référentiel global. Pour trouver l'axe de la fusée dans le référentiel global, une matrice de rotation en x : $\begin{pmatrix} 1 & 0 & 0 \\ 0 & \cos(\Omega_x) & -sin(\Omega_x) \\ 0 & sin(\Omega_x) & \cos(\Omega_x) \end{pmatrix}$ avec pour angle la position angulaire en x, Ω_x a été appliquée sur l'orientation initiale de la fusée a été utilisée avec : $\begin{pmatrix} 0 \\ 0 \\ 1 \end{pmatrix}$. Ensuite, le calcul de l'angle α est fait à partir du produit scalaire de l'axe de la fusée dans le référentiel global et le vecteur vitesse. Cette force est donc calculée dans le référentiel global.

1.2.3. Force de Propulsion

En plus de la force gravitationnelle et la force de frottement visqueux, la troisième force appliquée à la fusée est la force de propulsion. Pour calculer cette force, nous avons utilisé la formule : $\vec{F} = -\mu\vec{v}_{gaz}$, ou $\mu = 1200kg/s$ et $\vec{v}_{gaz} = -|\overrightarrow{v}_{gaz}|\begin{pmatrix} 0 \\ sin\theta \\ cos\theta \end{pmatrix}$ avec θ qui représente l'angle de la force de propulsion par rapport à la fusée. Il est à noter que θ reste constant durant le vol en raison d'une défaillance mécanique. Avant d'appliquer cette force, nous avons vérifié si la masse de la fusée est égale à 20 tonnes, ce qui équivaut à la soustraction de la masse totale initiale 320 tonnes et du carburant 300 tonnes. Si cette masse est égale à 20 tonnes, cela signifie qu'il n'y a plus de carburant et que la force de propulsion ne doit plus être appliquée, d'où le vecteur $[0 \quad 0 \quad 0]$. Dans le cas contraire, la fusée continue à être propulsée. À cette force de propulsion, vu qu'elle est dans un référentiel local, nous avons appliqué la matrice de rotation en x : $\begin{pmatrix} 1 & 0 & 0 \\ 0 & \cos(\Omega_x) & -sin(\Omega_x) \\ 0 & sin(\Omega_x) & \cos(\Omega_x) \end{pmatrix}$ avec pour angle la position angulaire en x, Ω_x pour la calculer dans le référentiel global.

1.3. Calcul de l'accélération angulaire

Pour le calcul de l'accélération angulaire, nous avons calculé le moment d'inertie tel qu'énoncé dans le devoir.

$$I_{c,zz} = \frac{m}{2}r^2$$
$$I_{c,xx} = I_{c,yy} = \frac{m}{4}r^2 + \frac{m}{12}l^2$$

FIGURE 2: FORMULE DU MOMENT D'INERTIE

Le moment d'inertie est celui d'un cylindre plein de rayon R et de hauteur H (hauteur totale de la fusée) et de masse m(t) (masse totale de la fusée à l'instant t). Après le calcul du moment d'inertie dans le référentiel local, nous avons appliqué une matrice de rotation en x de nouveau : $\begin{pmatrix} 1 & 0 & 0 \\ 0 & \cos(\Omega_x) & -sin(\Omega_x) \\ 0 & sin(\Omega_x) & \cos(\Omega_x) \end{pmatrix}$ avec pour angle la position angulaire en x, Ω_x, pour le calculer dans le référentiel global.

Le tau τ est le produit vectoriel du vecteur distance fourni dans l'énoncé : $\begin{pmatrix} 0 \\ 0 \\ -25 \end{pmatrix}$ et de la force de propulsion dans le référentiel local.

Le calcul de l'accélération linéaire est tiré du produit scalaire de l'inverse de la matrice du moment d'inertie et du vecteur tau.

1.4. Choix de Δt

Dans ce devoir, il fallait choisir le bon Δt pour que les simulations soient bonnes. Donc, il fallait choisir un Δt qui n'est pas trop grand pour ne pas manquer de précision dans les simulations et un Δt qui n'est pas trop petit pour ne pas faire des simulations prenant beaucoup de temps.

De plus, dans l'énoncé du devoir, la précision requise pour les simulations correspond à des erreurs maximales sur la position finale de la fusée en x, y, z de $\pm$ 10 m. Alors, nous avons essayé plusieurs Δt pour respecter cette précision requise.

Par exemple, avec notre algorithme deltaT.m , on calcule notre thêta, avec une précision de +-10m, ce qui résulte à un delta de 0.0005s. Mais cette simulation prend beaucoup de temps et les instances ne sont pas comprises entre 100 et 1000. C'est pourquoi nous avons choisi $\Delta t = 2.54$. Avec $\Delta t = 2.54$, nous respectons la contrainte donnée d'avoir des instances entre 100 et 1000 points.

1.5. Arrêt de la simulation

Dans le présent devoir, nous devions tracer la trajectoire de la fusée de son point de départ jusqu'à ce qu'elle dépasse la distance de 10 000 km du centre de la Terre ou qu'elle s'écrase sur la surface de la Terre. D'abord, la simulation s'arrête lorsque la fusée s'éloigne de la Terre si ses coordonnées $x^2 + y^2 + z^2$ sont égales ou dépassent $10^{14}\ m^2$. La seconde contrainte pour que la simulation s'arrête est lorsque la fusée s'écrase sur le sol, c'est-à-dire lorsque son centre de masse franchit la surface de la Terre. Pour que cette situation se produise, il faut que les coordonnées $x^2 + y^2 + z^2$ soient plus petites ou égales au rayon de la Terre au carré.

2. Présentation et analyse des résultats

La fusée subit les trois forces données : force de propulsion, force gravitationnelle et la force de frottement visqueux. On a remarqué que l'angle $\boldsymbol{\theta}$ doit être très petit pour que la fusée puisse décoller, dans les 0.0001+ radians. Pour chaque type de cas, nous avons testé trois simulations. Ainsi, par essai-erreur, nous avons pu trouver un angle thêta limite de 0.000016 rad ou la fusée quitte l'atmosphère. Si l'angle est plus grand que cette valeur, la fusée retombe sur la terre.

2.1. Simulation 1

Voyons maintenant les résultats que nous avons obtenus pour la simulation 1 avec les 3 forces et un thêta égal à zéro.

Parmi les données pouvant être modifiées, nous avons utilisé les conditions initiales suivantes:

TABLEAU 1: TABLEAU DES DONNÉES UTILISÉES LORS DE LA SIMULATION 1

θ (rad)	Δt (s)

0.000000	2.54

Voici le graphique représentant la terre et la trajectoire de la fusée. La ligne rouge représente la trajectoire de la fusée et la boule bleue est la terre.

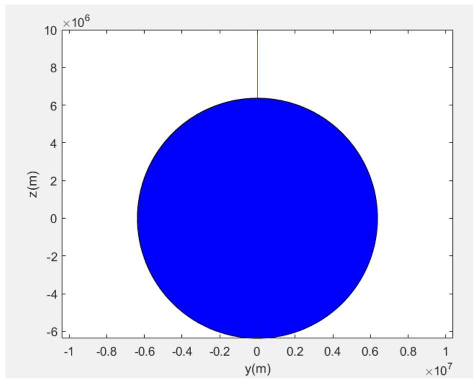

FIGURE 3: SIMULATION DE LA FUSÉE PARTANT EN LIGNE DROITE

Voici maintenant un tableau résumé des résultats obtenus :

TABLEAU 2: RÉSUMÉ DES RÉSULTATS DE LA FUSÉE POUR LA SIMULATION 1

Résultat de la simulation	θ (rad)	Temps final(s)	Vitesse finale (m/s)	Position finale (m)
La fusée s'éloigne de la terre en ligne droite	0.000000	843.28	[0;0;3958.79]	[0;0;10000306]

Si on analyse cette simulation, d'après le graphe aussi, on peut voir que la fusée s'éloigne de la terre en ligne droite jusqu'à ce qu'elle n'ait plus de carburant. Ce cas a aussi été utilisé pour vérifier la validité de nos résultats quand aucun angle n'est appliqué à la fusée. Le temps mis est 843.28 s avec une vitesse finale de 3958.79 m/s en z. Il est normal que les composantes de la vitesse en x et y soit de 0. Cela s'explique car aucune force n'a généré de mouvement de rotation. La position finale est aussi seulement en la composante z avec 10000306 m.

2.2. Simulation 2

Voyons maintenant les résultats que nous avons obtenus pour la simulation 2 avec les 3 forces et avec des thêtas entre 0.000014 à 0.000016.

Parmi les données pouvant être modifiées, nous avons utilisé les conditions initiales suivantes:

TABLEAU 3: TABLEAU DES DONNÉES UTILISÉES LORS DE LA SIMULATION 2

θ (rad)	Δt (s)
0.000014	2.54
0.000015	2.54

0.000016	2.54

Voici le graphique représentant la terre et la trajectoire de la fusée. La courbe rouge représente la trajectoire de la fusée et la boule bleue est la terre.

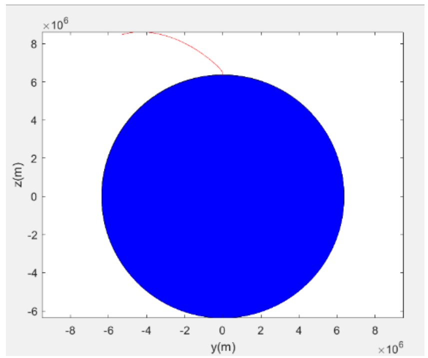

FIGURE 4: GRAPHIQUE REPRÉSENTANT LA FUSÉE QUITTANT L'ASTMOSPHÈRE

Voici maintenant un tableau résumé des résultats obtenus :

TABLEAU 4: RÉSUMÉ DES RÉSULTATS DE LA FUSÉE POUR LA PREMIÈRE SIMULATION 2

Résultat de la simulation	θ (rad)	Temps final(s)	Vitesse finale (m/s)	Position finale (m)
La fusée quitte l'atmosphère	0.000014	1163.32	[0; -3582.46;549.29]	[0; -4012410;9161473]
La fusée quitte l'atmosphère	0.000015	1257.30	[0; -3570.5; -112.28]	[0; -4563375;8899690]
La fusée quitte l'atmosphère	0.000016	1412.24	[0; -3370.2;-998.40]	[0;-5314537;8473110]

Pour ces trois simulations, la fusée quitte l'atmosphère. On voit que pour les positions obtenues, seules les composantes en y et en z ont changé. Cela s'explique par la modification de l'angle thêta, en lui donnant une valeur différente de 0. Vu qu'une vitesse et une position angulaire sont générées, ainsi qu'une rotation, la fusée se déplace avec un léger angle sur le plan yz. La condition d'arrêt de la fusée prend en compte les axes vers lesquels la fusée se dirige. Avec un petit thêta, la fusée s'éloigne de la terre. En effet, en observant le graphe, on peut voir la trajectoire de la fusée qui va vers la gauche. A

2.3. Simulation 3

Voyons maintenant les résultats que nous avons obtenus pour la simulation 3 avec les 3 forces et des thêta supérieurs à 0.000016 rad. Ce qui cause l'écrasement de la fusée sur la terre

Parmi les données pouvant être modifiées, nous avons utilisé les conditions initiales suivantes:

TABLEAU 5: TABLEAU DES DONNÉES UTILISÉES POUR LA SIMULATION 3

θ (rad)	Δt (s)
0.000017	2.54
0.000018	2.54
0.000019	2.54

Voici le graphique représentant la terre et la trajectoire de la fusée. La courbe rouge représente la trajectoire de la fusée et la boule bleue est la terre.

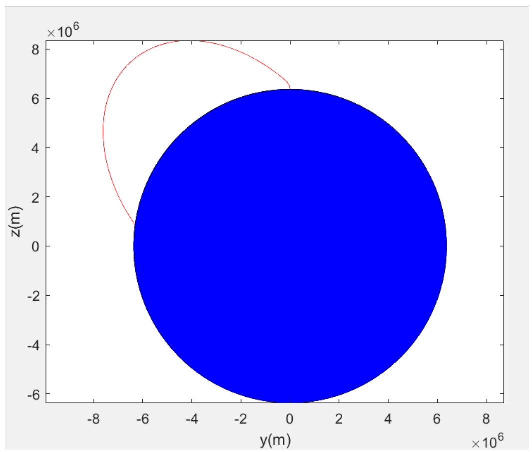

FIGURE 5: SIMULATION DE LA FUSÉE QUI RETOMBE SUR LA TERRE

Voici maintenant un tableau résumé des résultats obtenus :

TABLEAU 6:RÉSUMÉ DES RÉSULTATS DE LA FUSÉE POUR LA SIMULATION 3

Résultat de la simulation	θ (rad)	Temps final(s)	Vitesse finale (m/s)	Position finale (m)
La fusée retombe sur la terre	0.000017	3352.80	[0;65.83; -9.18]	[0; -6309816;880030]
La fusée retombe sur la terre	0.000018	3246.12	[0;66.05; -5.47]	[0; -6349182;526386]
La fusée retombe sur la terre	0.000019	3139.44	[0;66.48; -2.27]	[0;-6367141;218248]

Pour ces simulations, la fusée retombe sur la terre. On voit que pour les positions obtenues, seules les composantes en y et en z ont changé comme pour la simulation 1 Cela s'explique par la modification de l'angle thêta, en lui donnant une valeur différente de 0. Vu qu'une vitesse et une position angulaire sont générées, ainsi qu'une rotation, la fusée se déplace sur le plan yz. La condition d'arrêt de la fusée prend en compte les axes vers lesquels la fusée se dirige. Cette condition stipule que si la somme des coordonnées de la fusée au carré est plus petite que le rayon de la terre, la fusée s'écrase, on voit que cette condition est respectée lorsque le thêta est supérieur ou égal à 0.000017. Ce qui donne un angle plus grand à la force de propulsion. Plus cet angle est grand, plus de chances la fusée aura de s'écraser. Ce qui est le cas ici. On voit aussi que le temps total de la simulation est de 3352.80 secondes avec une vitesse finale de 65.83 m/s en y et -9.18 m/s en y avec une position finale de -6309816 m en y et 880030 m en z. Aussi, vu que la fusée redescend en direction de la terre du côté gauche, il est normal que la vitesse en z soit négative et positive en y. Le temps nécessaire pris pour cette simulation est cohérent car le temps de la retombée de la fusée ajoute un délai.

3. Conclusion

En somme, nous avons réussi avec succès à faire la simulation de la fusée. Cependant, nous avons rencontré multiples difficultés durant ce devoir. La première difficulté fût le choix du Δt. Effectivement, il n'était pas évident de trouver la bonne valeur car il fallait lancer la simulation pour de multiples valeurs afin d'arriver aux résultats attendus qui respectent les précisions demandées dans ce travail. Aussi, une autre difficulté rencontrée fût le débogage de notre code, car malgré notre bonne logique des concepts physiques, notre simulation ne marchait pas bien car nous avions commis des erreurs dans le code. Enfin, l'utilisation de la méthode de Runge-Kutta d'ordre 4 pour tracer la trajectoire de la fusée fût un peu compliquée, car nous ne savions pas comment ajouter des paramètres en plus à la fonction.