Dynamique de la chute libre

THÉMATIQUE

- Mécanique, trajectoire d'un objet en mouvement.

CONCEPTS OU NOTIONS ABORDÉS

- Liens entre position, vitesse et accélération
- 2e loi de Newton
- Méthode d'Euler

OBJECTIFS DE FORMATION

- Déterminer les coordonnées du vecteur vitesse à partir de celles du vecteur accélération.
- Déterminer les coordonnées de position à partir de celles du vecteur vitesse.
- Utiliser la 2e loi de Newton pour déterminer la trajectoire d'un projectile en chute libre.
- Utiliser la méthode d'Euler pour calculer les coordonnées du vecteur vitesse à partir de celles du vecteur accélération, les coordonnées de position à partir de celles du vecteur vitesse.

Introduction

Consulter la page éduscol associée au thème « Programmer en physique-chimie ».

Présentation de l'activité

Dans la situation proposée, les élèves fixent les paramètres initiaux de la trajectoire d'un projectile en chute libre : position et vitesse initiales, angle de lancer. Ils réalisent ensuite un programme de calcul numérique pour déterminer les coordonnées du projectile et de son vecteur vitesse en s'appuyant sur la méthode d'Euler.

Place possible dans la progression

Cette activité peut être proposée après la de résolution analytique des équations différentielles issues de la 2e loi de Newton pour projectile en chute libre et conduisant à la détermination des coordonnées de la vitesse, de la position et de la trajectoire du projectile. Le recours à informatique peut contribuer à une meilleure compréhension des grandeurs position, vitesse et accélération et des mécanismes mathématiques d'intégration.

Le choix de l'outil informatique dépend des compétences initiales des élèves et peut être laissé au choix de l'élève ; quel que soit l'outil utilisé, une certaine maîtrise en est nécessaire.

Une activité plus clé en main peut être proposée dès la classe de seconde dans le cadre de l'analyse d'un mouvement, ce qui permet de cibler un outil informatique en particulier en début de formation. À ce niveau, une activité de simulation peut venir en complément d'une étude vidéo et peut par exemple consister en :

- l'affichage de la trajectoire, les valeurs ou la boucle de calcul étant données ;
- la modification des paramètres initiaux et l'analyse des effets sur la trajectoire.

Pistes de validation, prolongements

Un prolongement peut consister à ajouter des forces de frottement de l'air. C'est un des intérêts de la simulation numérique mise en place : il est possible d'accéder à des effets de frottement sans outil mathématique supplémentaire ou solution analytique connue de l'équation associée et de confronter les résultats du programme aux données expérimentales[1].

LOGICIELS UTILISÉS

- Tableur Calc (LibreOffice).
- Logiciel de géométrie dynamique GeoGebra.
- Langage Python.

NOTIONS ET COMPÉTENCES INFORMATIQUES TRAVAILLÉES

- Définir des paramètres numériques.
- Tracer des graphiques.
- Effectuer des calculs numériques.
- Mettre en œuvre des boucles.

Exemples de contextualisation

- Déterminer l'angle optimal pour un lancer de poids ou de Vortex®[2].
- Choisir le club de golf le mieux adapté à la trajectoire souhaitée.
- Déterminer les conditions d'un lancer-franc réussi.

De la situation physique au traitement numérique

On considère un objet dans le champ de pesanteur terrestre, supposé vertical uniforme d'intensité *g*. L'objectif de l'activité est l'étude du mouvement de ce projectile dans le référentiel terrestre considéré comme galiléen. Les forces exercées par l'air sont négligées dans un premier temps, le projectile n'est donc soumis qu'à son poids. Il est admis (ou cela a été montré précédemment) que le mouvement est plan et un repère orthonormal (O,x,z) est associé à ce plan. Le projectile a une vitesse initiale $\vec{v}_0=(v_0\cos(\alpha), v_0\sin(\alpha))$ et une position initiale, le point M_0 (x_0,z_0).

La 2^e loi de Newton permet d'obtenir les équations suivantes :

$$\frac{dv_x}{dt} = 0 \qquad (1.a)$$

$$\frac{dv_z}{dt} = -g \qquad (1.b)$$

[1] On peut, pour ce faire, effectuer des expérimentations numériques sur la simulation mise en place jusqu'à ce qu'elle corrobore avec le mouvement réel du projectile.

[2] Voir la ressource d'accompagnement du programme de cycle 4 « Comment optimiser une performance en lancer de Vortex® ? ».

Par ailleurs, par définition de la vitesse :

$$v_x(t) = \frac{dx}{dt} \qquad (2.a)$$

$$v_z(t) = \frac{dz}{dt} \qquad (2.b)$$

Les équations (1.a) et (1.b) permettent d'estimer, pas à pas, les composantes de la vitesse, sans procéder à une intégration, sans passer par une primitive.

On utilise pour ce faire le développement limité (à l'ordre 1) des fonctions vitesses horizontale et verticale, qui donne une estimation de la valeur de ces fonctions à un instant $t+\tau$ à partir de leurs valeurs à l'instant *t* :

$$v_x(t+\tau) = v_x(t) + \frac{dv_x}{dt}.\tau$$

$$v_z(t+\tau) = v_z(t) + \frac{dv_z}{dt}.\tau$$

τ joue le rôle de *pas* de calcul.

On déduit des équations (1.a) et (1.b) :

$$v_x(t+\tau) = v_x(t)$$

(la composante horizontale de la vitesse est constante)

$$v_z(t+\tau) = v_z(t) - g.\tau$$

Un raisonnement similaire donne :

$$x(t+\tau) = x(t) + v_x(t).\tau$$

$$z(t+\tau) = z(t) + v_z(t).\tau$$

Il reste alors à choisir la valeur du pas tau et à calculer pas à pas, par itération, les valeurs successives de v_x, v_z, x et z.

La méthode repose sur des approximations faites grâce à des développements limités. Ceci entraîne un écart entre les valeurs obtenues par calcul numérique (ou intégration « numérique ») et celles obtenues par intégration analytique (par recherche d'une primitive) des équations de Newton. Plus la valeur du pas *tau* est faible, meilleure est la précision. En revanche, le choix d'une valeur faible de *tau* entraîne davantage de calculs et un temps de traitement plus long.

Ce que les élèves doivent retenir

- Il suffit que les coordonnées du vecteur accélération d'un objet et les conditions initiales soient connues pour accéder aux coordonnées du vecteur vitesse de cet objet et à ses coordonnées de position à l'aide de la mise en œuvre d'une succession de calculs numériques effectué par itération (méthode d'Euler).
- D'après la 2^e loi de Newton, dans le référentiel terrestre, l'accélération d'un objet en chute libre est égale à celle du champ de pesanteur terrestre. La composante horizontale de la vitesse est donc constante tout au long du mouvement.

Activité : calcul numérique de la trajectoire d'un projectile en chute libre par la méthode d'Euler

Les propositions suivantes ne sont pas prescriptives. Il s'agit de bases pour illustrer la situation d'apprentissage qu'il convient d'ajuster en fonction de sa place dans la progression et dans la séquence pédagogique, du niveau de maîtrise par les élèves de l'outil informatique choisi, etc.

Exemples de consignes pour les élèves

Programmation (compétence Réaliser)

1. Programmer les paramètres numériques initiaux de la chute libre : champ de pesanteur g=9.81 m.s^{-2}, coordonnées de position initiales (x_0,z_0), angle de lancer (α), vitesse initiale (v_0), pas de calcul (τ=0.2 s).
2. Programmer les coordonnées de position et du vecteur vitesse du projectile et calculer par itération les valeurs de ces variables.

$$x(t+\tau) = x(t) + v_x(t).\tau$$

$$z(t+\tau) = z(t) + v_z(t).\tau$$

$$v_x(t+\tau) = v_x(t)$$

$$v_z(t+\tau) = v_z(t) - g.\tau$$

3. Afficher le graphique représentant l'évolution de l'ordonnée z en fonction de l'abscisse *x* du projectile.

Expérimentation numérique et validation de la programmation (compétence Valider)

1. Faire varier l'angle *α*. Observer les évolutions de la trajectoire et déterminer la valeur de *α* pour laquelle la distance parcourue par le projectile est maximale (aide : la portée correspond à la valeur de x correspondant à une valeur de l'ordonnée z=0, c'est à dire au niveau du « sol »).
2. Afficher le graphique de la trajectoire de référence, d'équation :

$$z_{ref}(x) = -\frac{g}{2v_0^2\cos^2(\alpha)}.x^2 + \tan(\alpha).x + z_0$$

La courbe issue de la programmation et la courbe de référence sont-elles en adéquation ?

3. Faire varier τ. Observer les variations de la trajectoire. Préciser pour quelles valeurs de *tau* la trajectoire calculée est la plus proche de la trajectoire de référence.

Conclusion (compétence Communiquer)

Proposition 1 : Lister les étapes (en utilisant des verbes à l'infinitif) pour parvenir à l'affichage des différentes courbes z(x) obtenue par calcul numérique d'une part et obtenue par la résolution analytique (courbe de référence) et à la réduction de l'écart entre ces deux courbes.

Proposition 2 : Produire un tutoriel ou une fiche méthode (s'il n'en est pas de disponible dans l'établissement) sur (au choix) :

- l'affichage de courbe avec l'outil utilisé ;

- la détermination des valeurs d'une fonction à partir de sa dérivée à l'aide de la méthode d'Euler.

Exemples d'aides et de réalisations pour la programmation avec Libreoffice Calc

1. Ouvrir le fichier « *chute libre avec tableur – à compléter.ods* » présent dans le dossier « [Dynamique de la chute libre.zip]() ». Compléter la colonne « durée de la chute *t* » de manière à ce que les valeurs prises par *t* augmentent de *tau* à chaque itération (calcul).
2. Compléter la ligne correspondant à *t=0* à l'aide des paramètres numériques initiaux.

Remarque : il est nécessaire de convertir les angles en radians avant d'en calculer le cosinus ou le sinus, soit par le calcul, soit en utilisant la fonction « RADIANS() ». Le recours au signe « $ » est également indispensable pour les adresses des cellules qui ne doivent pas être modifiées lors de la copie des formules.

3. Compléter la 2e ligne (correspondant à *t=tau*) en utilisant les expressions données par la méthode d'Euler :

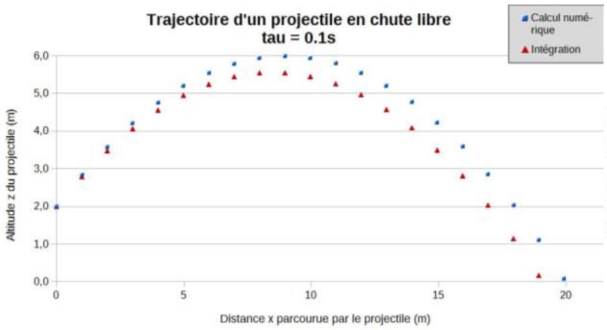
$$x(t+\text{tau}) = x(t) + v_x(t).\text{tau}$$
$$z(t+\text{tau}) = z(t) + v_z(t).\text{tau}$$
$$v_x(t+\text{tau}) = v_x(t)$$
$$v_z(t+\text{tau}) = v_z(t) - \text{g.tau}$$

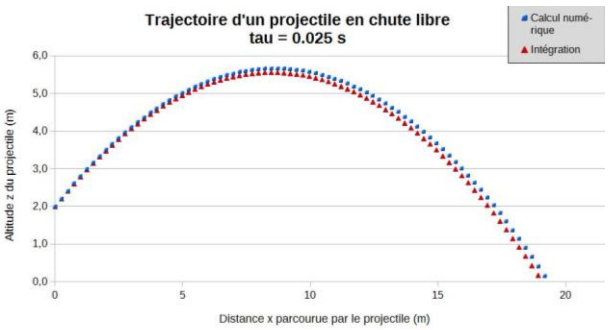
Remarque : la transcription des équations dans un tableur n'est pas transparente. Les cellules ne portent en effet pas le nom des grandeurs manipulées. L'itération des calculs implique par ailleurs des références absolues et relatives (utilisation ou pas du symbole « $ ») auxquelles il faut être très attentif. Une certaine dextérité dans l'utilisation du tableur est donc nécessaire. En revanche, la modification d'un paramètre a l'avantage d'être directement répercutée dans le graphique.

Voir le fichier *Chute libre avec tableur.ods* fourni dans le dossier Dynamique de la chute libre.

Exemples de résultats obtenus avec Libreoffice Calc

pour tau=0,1 s et tau = 0,025 s (α=40° v_0=13 m.s^{-1}) – en bleu la courbe obtenue par calcul numérique, en rouge la courbe de référence.





Exemples d'aides et de réalisations pour la programmation avec Geogebra

Remarque : il n'y a pas de flèche sur les vecteurs dans Geogebra.

1. Ouvrir Geogebra et créer les objets définissant les conditions initiales du mouvement du projectile :
   - positionner un point G de manière à définir le vecteur g(0,-9,81) ;
   - positionner un point V_0 de manière à définir le vecteur vitesse initiale v_0 ;

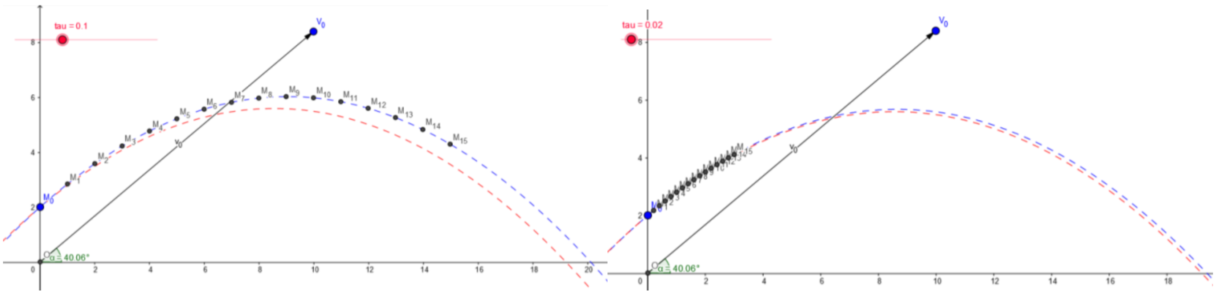
- positionner un point M_0 sur l'axe des ordonnées de manière à définir la position initiale du projectile ;
- définir le pas tau de calcul. Lui associer un curseur pouvant varier entre 0 et 0,5 s par exemple.

2. Pour n variant de 1 à 12 (par exemple), construire par itérations les vecteurs vitesse et les positions occupées par le projectile.
   - Quelle valeur de k choisir pour définir les vecteurs vitesse successifs de la manière suivante, dans le formalisme Geogebra : $v_{n+1} = v_n + k.\, \tau$? Justifier. Construire ensuite ces vecteurs (il pourra ensuite être utile, pour alléger l'affichage, de ne pas les faire apparaître).
   - Construire les points Mn définis par récurrence par l'expression $M_{n+1} = M_n + v_n.\, \tau$.
3. Construire la courbe correspondant à la trajectoire de référence obtenue par intégration de la 2[e] loi de Newton en utilisant l'outil *conique*.

Voir le fichier *chute libre avec geogebra.ggb dans le dossier* Dynamique de la chute libre.

Exemples de résultats obtenus avec Geogebra

tau=0,1 s et tau = 0,02 s (α=40° v_0=13 m/s) – en bleu la courbe obtenue par calcul numérique, en rouge la courbe de référence.



Exemples d'aides et de réalisations pour la programmation en Python

1. Ouvrir le fichier « *chute libre avec python – a completer.py* » présent dans le dossier « [Dynamique de la chute libre.zip]() ». Compléter les valeurs des paramètres.

```
#déclaration des variables et de leur valeur initiale

g=9.81 #m.s-2
v=    #m/s
alpha=   #°
tau=0.1 #s c'est le PAS du calcul
x=    #m
z0=   #m
z=z0 #m ainsi z sera traité comme une variable (au sens mathématique) et z0 reste constant
t=0 #s
```

Le pas a été ici fixé à 0.1 s mais pourra être modifié.

2. Quelles formules peuvent-être tapées dans la boucle ci-dessous ?

```
while z>=0:
    x =
    z =
    t =
        vz =
```

3. Afficher les deux graphiques (l'un relatif au calcul numérique, l'autre issu de l'intégration des équations différentielles du mouvement) de l'altitude en fonction de la distance parcourue. Soigner l'affichage : échelle, titres, légende, etc.

Remarque : il est proposé ici de compléter un fichier pré-rempli, afin que l'élève reste concentré sur le contenu physique. Le tracé de courbe avec Python est cependant régulièrement pratiqué en mathématiques et les élèves ont recours aux listes dans le cadre de cet enseignement. La conception de l'ensemble du programme est donc envisageable. Un travail sur le tracé de graphiques peut par ailleurs avoir déjà été mené en amont, dès la classe de seconde, et peut ainsi avoir préparé à cette activité.

Point de vigilance : pour que les coordonnées $x(t+\tau)$ et $z(t+\tau)$ soient calculées à partir des valeurs des composantes à l'instant *t*, il est nécessaire de positionner les calculs des composantes de la vitesse après les calculs des coordonnées de position dans la boucle « while ».

Dans le dossier « Dynamique de la chute libre.zip », vous trouverez le fichier chute libre avec python.py présentant un exemple de réalisation.

Exemples de résultats obtenus avec Python pour tau=0,1 s et tau = 0,02 s (α=40° v_0=13 m.s^{-1})