



UNIVERSITÉ
LIBRE
DE BRUXELLES

INFO-F308 - PROJET D'ANNÉE

SIMULATION DE LA RÉALITÉ

Auteurs :

Romain LIEFFERINCKX - 000591790

Manuel ROCCA - 000596086

Matteo MORBÉE - 000549684

Damian KAZBERUK - 000589811

Martin GOUVERNEUR - 000586541

Professeurs :

Gilles GERAERTS

Promoteur :

Pascal TRIBEL

Année académique 2025-2026

Table des matières

1	Introduction	2
2	Equations utilisées	2
3	Conclusion	5
3.1	test	5

1 Introduction

Le Printemps des Sciences est un événement scientifique annuel coordonné par le Département Infosciences de l'ULB en région bruxelloise. Cet événement a "pour objectif de rassembler l'ensemble des outils de diffusion et de promotion des sciences de la Faculté des Sciences et de permettre ainsi à tout un chacun de goûter aux sciences, d'en découvrir les multiples facettes, de l'approche expérimentale à la compréhension des enjeux sociaux et démocratiques qui en relèvent, en passant par le simple plaisir de la découverte".¹

Dans le cadre du cours INFO-F308 intitulé Projets d'informatique 3 transdisciplinaire, les étudiants de troisième année de bachelier en sciences-informatiques ont été amenés à former des groupes de 5 étudiants. Une fois cette étape terminée chaque groupe fut présenté avec un choix de sujet à présenter au Printemps des Sciences. Nous avons, pour notre part, choisi le sujet "Simulation de la réalité".

Nous devons créer un modèle expérimental représentant un système physique et recréer ce système au mieux pour pouvoir ensuite, à l'aide de simulations, comparer notre modèle à la théorie. Le nombre d'expériences à représenter est très grand. Après avoir pesé les pour et les contres de certaines suggestions comme la simulation d'un trou noir, le refroidissement d'une tasse ou encore des éclairs, nous avons finalement décidé de simuler l'interaction entre les corps dans l'espace.

La première partie de ce projet est donc la création d'un modèle expérimental. (détails à ajouter)

Concernant la simulation, plusieurs choix s'offrent à nous :

- équations physiques classiques (Newton) en 2D
- équations 3D
- simulation stochastique (Bayes)
- réseaux de neurones

Certaines options sont plus complexes que d'autres. Comme notre but est de faire découvrir une science accessible à tout le monde nous avons décidé d'adopter une approche de complexité croissante ; commencer par la simulation la plus simple (à savoir les équations de gravitations classiques) et potentiellement, si le temps et nos compétences le permettent, s'essayer à d'autres options (comme le machine learning et les réseaux neuronaux).

2 Equations utilisées

Dans cette section, nous détaillons les équations qui nous ont permises de modéliser les différents phénomènes physiques dans notre simulateur.

Nous allons commencer par les équations de la physique newtonienne que nous avons utilisées dans notre simulateur.

En deux dimensions, on utilise la deuxième loi de Newton pour modéliser le mouvement des objets comme ceci :

$$F = ma$$

où :

F = la force

m = la masse de l'objet

a = l'accélération de l'objet

$$F_{grav} = -G \frac{Mm}{r^2}$$

1. Source : <https://sciences.brussels/printemps/equipe/>

où :

$$\begin{aligned}F_{grav} &= \text{la force gravitationnelle} \\G &= \text{la constante gravitationnelle} \\M &= \text{la masse de la source de gravité} \\m &= \text{la masse de l'objet attiré par la source} \\r &= \text{la distance entre les deux masses}\end{aligned}$$

Ces deux équations sont à la base de la physique newtonienne en deux dimensions, or comme nous le savons, notre univers est en trois dimensions. Nous devons donc adapter ces équations pour qu'elles fonctionnent dans un espace tridimensionnel pour coller avec notre thème : la simulation de la réalité.

A noter que r part de la source (0, 0) donc le soleil vers l'objet (x, y), la bille.

Ceci nous permet d'obtenir les composantes de la force gravitationnelle dans chaque direction et ainsi d'obtenir la version vectorielle de la loi de la gravitation universelle :

$$F_{grav} = -G \frac{M_{soleil} m_{bille}}{r^2} (x, y)$$

Attention, le signe négatif indique que la force est attractive, donc dirigée vers le soleil. Puisque comme dit plus haut, r part du soleil vers la bille, nous devons inverser le sens de la force pour qu'elle pointe bien vers le soleil.

Maintenant que nous avons notre formule de la force gravitationnelle vectorielle, nous pouvons l'intégrer dans la deuxième loi de Newton pour obtenir l'accélération de notre bille dans chaque direction :

$$F_{grav} = ma = -G \frac{M_{soleil} m_{bille}}{r^2} (x, y)$$

donc :

$$a = -G \frac{M_{soleil}}{r^2} (x, y)$$

A partir de ce moment la, nous pourrions nous dire que nous avons tout ce qu'il nous faut pour simuler le mouvement de notre bille dans le champ gravitationnel du soleil, cependant, il ne faut pas oublier les frottements comme dit plus haut, car nous voulons simuler la réalité.

Formule générale de la friction :

$$F_{friction} = -\mu mv$$

où :

$$\begin{aligned}F_{friction} &= \text{la force de friction} \\ \mu &= \text{le coefficient de friction} \\ m &= \text{la masse de l'objet} \\ v &= \text{la vitesse de l'objet}\end{aligned}$$

Par conséquent, l'accélération due à la friction :

$$a_{friction} = -\mu v$$

Nous pouvons maintenant combiner les deux accélérations pour obtenir l'accélération totale de

notre bille pour chaque composantes :

$$a_{total} = a_{grav} + a_{friction} = -G \frac{M_{soleil}}{r^2}(x, y) - \mu v$$

Pour terminer, regardons comment nous utilisons cette accélération pour mettre à jour la position et la vitesse de notre bille à chaque itération. Nous savons grâce à Newton encore une fois, que :

$$\frac{dx}{dt} = v$$

La dérivée de la position par rapport au temps est égale à la vitesse. De plus, nous savons que :

$$\frac{dv}{dt} = a$$

La dérivée de la vitesse par rapport au temps est égale à l'accélération.

Malgré cela, dans notre simulation, nous utilisons des pas de temps discrets, nous allons donc approximer ces dérivées par des différences finies grâce à la méthode d'Euler explicite :

$$\begin{aligned} v_x(t + \Delta t) &= v_x(t) + a_x(t) \Delta t \\ x(t + \Delta t) &= x(t) + v_x(t + \Delta t) \Delta t \end{aligned}$$

où :

$$\begin{aligned} \Delta t &= \text{le pas de temps} \\ t &= \text{le temps à l'instant } t \\ x(t) &= \text{la position à l'instant } t \\ v_x(t) &= \text{la vitesse à l'instant } t \\ a_x(t) &= \text{l'accélération à l'instant } t \end{aligned}$$

Ceci nous permet de mettre à jour la position et la vitesse de notre bille à chaque itération de la simulation.

Maintenant que nous avons vu les équations de la physique newtonienne, nous allons nous pencher sur comment nous allons modéliser le drap avec un puit de gravité au centre. Pour modéliser le drap, nous avons choisi d'utiliser une fonction gaussienne bidimensionnelle centrée en (0, 0) (le centre du drap) qui va représenter la déformation du drap due à la présence du soleil, ou en tout cas une masse.

La fonction représentant notre drap :

$$f(x, y) = A \exp \left(-\frac{(x - x_0)^2 + (y - y_0)^2}{2\sigma^2} \right)$$

- A : amplitude = $-depth$ (négative puisqu'on veut un puit)
- (x_0, y_0) : centre de la gaussienne = (0, 0)
- σ : écart-type = "largeur" de la cloche

après simplification, nous arrivons à la fonction qui modélise notre puit :

$$f(x, y) = -depth \exp \left(-\frac{x^2 + y^2}{2\sigma^2} \right)$$

La dernière étape, consiste à regarder pour chaque itération avec notre x et y mis à jour, ou se

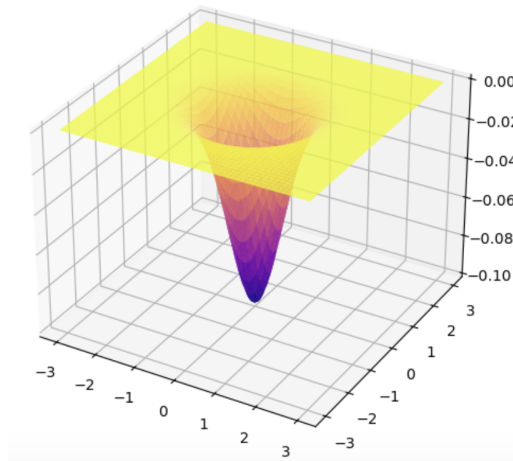


FIGURE 1 – Représentation graphique de notre fonction de drap avec puit de gravité au centre.

trouve notre z grâce à la fonction de notre drap.

$$z = f(x, y) = -depth \exp\left(-\frac{x^2 + y^2}{2\sigma^2}\right)$$

Nous avons ainsi notre position en 3D (x, y, z) pour chaque itération de notre simulation.

3 Conclusion

Au terme de ce projet, nous avons appris de nombreuses choses intéressantes. Bien que ce projet n'ait été pas été aussi complexe que ses prédécesseurs (PDA1 et PDA2) il fut bien plus intéressant car nous avons touché à un aspect très intéressant de l'informatique : la diversité.

Pour réaliser ce projet, nous nous sommes penchés sur des concepts théoriques physiques, sur de la pédagogie, sur de la vulgarisation et des aspects techniques (création du modèle expérimental). Tout ceci en restant dans un cadre scientifique correct en témoigne ce rapport. Bien entendu, nous avons pu construire sur nos compétences de base de travail de groupe déjà initiées l'année passée pour le merveilleux *Tetris Royale*².

En somme, ce projet sensationnel conclut avec panache notre troisième et dernière année de bachelier en sciences-informatiques au sein de l'ULB. Nous souhaitons également remercier Tribel Pascal, notre assistant assigné pour ce projet qui, tout au long de ce projet, répondait présent (sauf sur teams si on ne le tague pas :)) et nous a aidé et conseillé de manière particulièrement pertinente.

3.1 test

Modèle physique — Formules clés

2. <https://github.com/romainlief/Tetris-Royal>

Gravité (vectorielle)

$$\vec{F}_{grav} = G \frac{Mm}{r^2}(x, y)$$

Accélération due à la gravité

$$F = ma \quad \Rightarrow \quad \vec{a}_{grav} = G \frac{M}{r^2}(x, y)$$

Frottements

$$\vec{F}_{fric} = -\mu m \vec{v} \quad \Rightarrow \quad \vec{a}_{fric} = -\mu \vec{v}$$

Accélération totale

$$\vec{a} = \vec{a}_{grav} + \vec{a}_{fric} = G \frac{M}{r^2}(x, y) - \mu \vec{v}$$

Drap

$$z = f(x, y) = -\text{depth} \exp\left(-\frac{x^2 + y^2}{2\sigma^2}\right)$$

Donne la coordonnée z pour chaque (x, y) .