

Rapport Final

M322 - Modélisation mathématique

- I. **Résumé**
- II. **Introduction**
- III. **Développement**
- IV. **Conclusion**
- V. **Références**

19 DÉCEMBRE 2019

Equipe : Lacrizeomic

Anthony MALVESIN – Eudes CHATIN



I. Résumé

L'objectif premier de ce module est de réaliser un système modélisé, représentant des particules, d'en extraire des informations pour en exploiter les résultats. Le projet doit concilier mathématiques, simulations et analyses c'est ce qui le rend tout aussi compliqué qu'intéressant.

Ici la problématique est la suivante :

Comment considérer un principe physique, le modélisé pour en tirer des résultats ?

Nous nous sommes donc instinctivement attachés aux principes physiques et en particulier aux collisions des particules pour ensuite pouvoir modéliser celles-ci dans les circonstances désirées. L'approche utilisée est donc de réaliser les méthodes nécessaires à la bonne implémentation mathématique des principes physiques.

II. Introduction

Une simulation a pour but de représenter un phénomène physique réel et la collision de particules pose de nombreux problèmes :

- Tout d'abord l'organisation, à savoir les méthodes à employées. Certaines optimise le rendu mais son plus complexe à implémenter.
- La manière dont les particules réagissent entres-elles. Faut-il s'adapter à son environnement ou inversement ?
- La qualité du rendu qui doit être facilement exploitable et compréhensible.
- L'implémentation d'équations mathématiques adaptées au programme et au système choisi.
- Développer plusieurs options de modélisation sur une même simulation

C'est pourquoi le résultat attendu doit permettre d'établir une virtualisation plus réelle que possible. Plusieurs systèmes de simulation peuvent être choisis, l'étude approfondie des mouvements, la dispersion, l'entropie, l'équilibre conditionné d'un gaz mais aussi le changement d'environnement.

Ce rapport sera établi de manière à mettre en évidence les différents points importants auxquels nous avons dû répondre lors de l'élaboration du simulateur de Gaz.

Nous aborderons en premier lieu les systèmes de modélisation choisis pour ensuite aborder la discrétisation numérique. « Procédé qui constitue en général une étape préliminaire à la résolution numérique d'un problème ou sa programmation sur machine » Nous développerons ensuite les choix d'implémentations et enfin les analyses/interprétations du rendu numérique du projet.

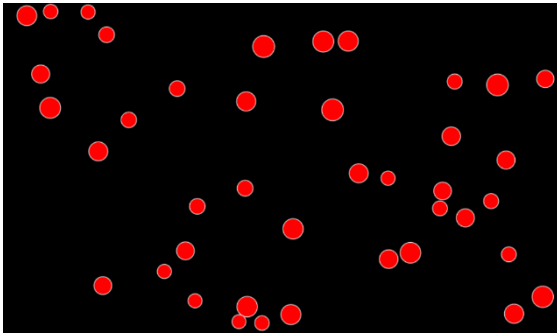
III. Développement

1. Modélisation

Nous avons pour objectif de rendre un travail de qualité et facilement consultable. Le JavaScript est un langage de programmation conciliant ces deux volontés notamment grâce à un rendu fluide, une visualisation simplifiée possiblement consultable sur le web et enfin le JavaScript était encore inconnu pour nous deux, c'était alors l'occasion d'y s'initier avec ce projet.

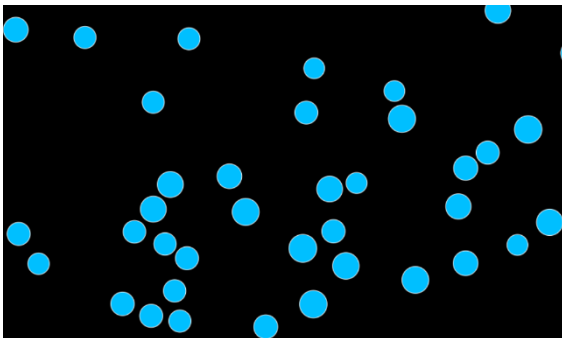
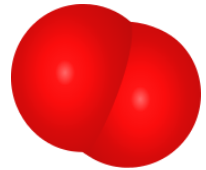
L'intérêt d'une simulation est de représenter un phénomène réel à travers un problème. Ici nous avons décidé de mélanger deux gaz de caractéristiques différentes.

L'hydrogène et le dioxygène.



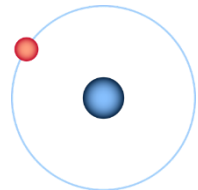
Molécules Bleues - Dioxygène

Plus petites
Moins lourdes
Plus rapides
Vitesse: 333m.s



Molécules Rouges - Hydrogène

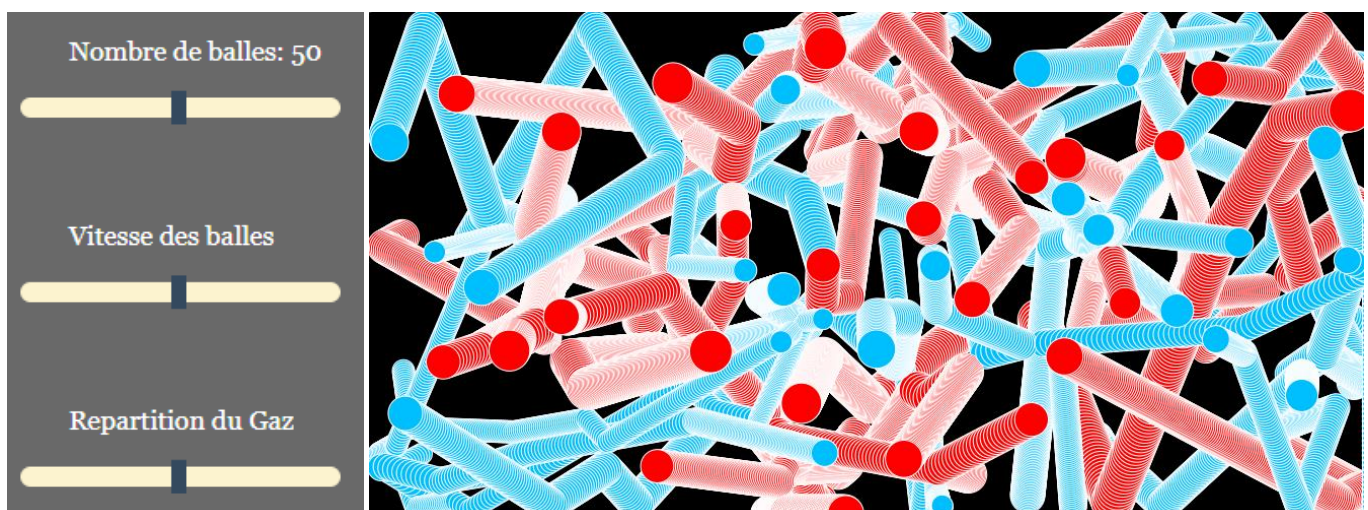
Plus grandes
Plus lourdes
Donc moins rapide
Vitesse: 500m.s



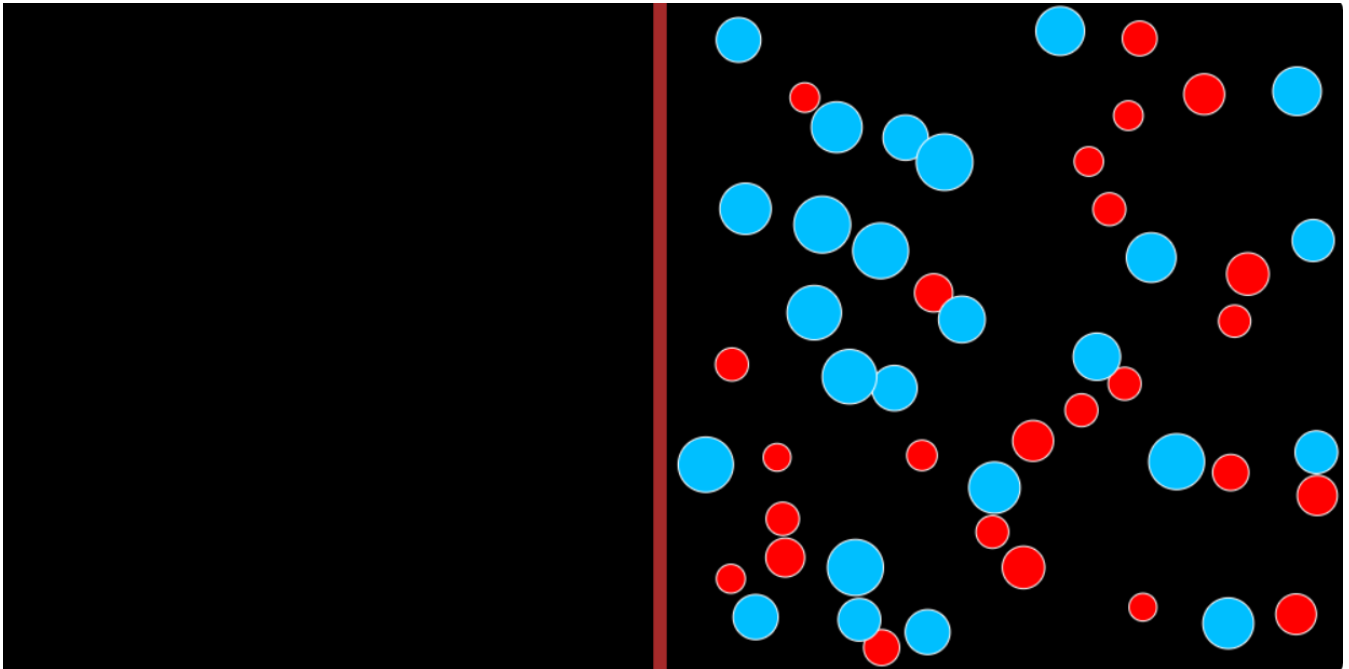
Les systèmes de modélisation choisis permettent l'étude des mouvements et des états dans un environnement modifiable avec un rapport d'état du système grâce aux variables d'entrée comme la vitesse, le nombre de molécules et la répartition des deux Gaz représentés. Et à une/des variables de sortie tels que la pression.

C'est pourquoi la personnalisation du contenu désiré est modifiable en fonction d'un phénomène probable comme le changement brusque de vitesse, de nombre de particules mais encore le mélange de deux gaz différents.

L'ajout d'un « Mode dispersion » ajoute une autre dimension à la simulation. Tracer la trajectoire des molécules permet d'analyser leurs dispersions tout en s'adaptant aux variables d'entrées (Sliders).



Enfin donner à l'utilisateur le pouvoir de modifier l'espace dans lequel évoluaient les molécules était un objectif dès le début du projet. Pour simplifier cette fonction il suffit de presser la touche « Right » ou « Left » du clavier pour en effectuer l'action.



2. Discrétisation numérique

La discrétisation est le procédé d'un passage entre un état continu (qui existe en tous points du temps) et un état discret (qui existe seulement à certains moments comme par exemple quand x est un entier).

Dans le cas de notre projet, c'est le fait d'arriver à donner des valeurs de positions à une balle qui devrait se déplacer sur toute la longueur d'un axe (valeurs continues).

L'énergie cinétique est l'énergie de déplacement d'une particule. Dans notre environnement, les collisions sont élastiques donc l'énergie cinétique totale reste constante.

3. Implémentation

Notre programme ayant pour objectif de modéliser la collision de particules, nous avons tout d'abord cherché à les modélisées.

La collision de particules suit le principe des collisions élastiques, c'est-à-dire que la somme de l'énergie cinétiques de 2 molécules reste le même avant et après la collision, il y a seulement un transfert d'énergie d'une molécule à l'autre.

Le calcul de ce type de collision se fait ainsi :

$$m_1 v_1^2 + m_2 v_2^2 = m_1 v_1'^2 + m_2 v_2'^2$$

Avec m= masse et v= vitesse.

Il nous a donc fallu prendre en compte, afin d'obtenir une modélisation la plus proche de la réalité possible, les différents phénomènes physiques qui ont un impact sur lesdites collisions (vitesse et masse).

Ainsi nous avons décidé de gérer la vitesse d'une molécule par des grandeurs de vitesse en x et y (qui peuvent être vu comme les vecteurs de direction en x et y).

La vitesse que nous obtenons alors en additionnant Vx et Vy et la masse que nous avons attribué selon une échelle de grandeur que nous avons choisie arbitrairement nous permettent donc de réaliser notre calcul.

Nous calculons cet angle grâce à la méthode atan2 fournie par JavaScript, qui permet de calculer l'angle θ entre une position en x et y et l'axe des abscisses.

Par ailleurs nous avons implémenté un autre système physique qu'est la notion de pression.

« La pression est une grandeur physique qui traduit les échanges de quantité de mouvement »

Afin de représenter ces échanges et également pour se faciliter la tâche, nous avons décidé de calculer la pression exercée par notre système sur un des 4 murs englobants les molécules.

L'objet de référence étant un mur, il n'est pas affecté par la pression qui est exercé sur lui (collisions élastique). Ainsi la vitesse de la molécule avant et après rebond est la même, néanmoins, son vecteur de déplacement en x se trouve inversé (parce que notre mur est un mur vertical). La pression exercée par chaque molécule sur le mur est donc de 2 fois son vecteur x.

Le calcul de la pression est le suivant :

$$p = \frac{F}{S}$$

Soit la pression équivaut à la force divisée par la surface de l'objet subissant la force.

Le problème est maintenant d'obtenir une valeur qui correspond à la réalité, pour cela il nous faut récupérer toutes les échelles de tailles que nous avons choisi pour représenter les valeurs réelles.

La surface et la force dépendent de 2 échelles que nous avons mis en place :

- L'échelle de taille
- L'échelle de vitesse

Nous avons une molécule allant de 10 à 20 unités dans notre programme est supposé mesurer 2×10^{-9} mètres dans la réalité, soit la surface de notre mur égale à 500, cela représente 6.6667×10^{-8} dans la réalité.

$$S = 6.6667 * 10^{-8}$$

Le calcul de la force est également à mettre à l'échelle car notre programme déplace nos molécules de plus ou moins 120 pixels par secondes (2pixels par 60fps (+/- car variable)) pour une vitesse réelle de 500 mètre/secondes.

Sachant que 180 pixels représentent 2.4×10^{-8} mètres, nous avons donc une échelle de vitesse de $\frac{1}{5} * 10^{-11}$; $(2.4 * 10^{-8} = 500 * x \Rightarrow x = \frac{2.4 * 10^{-8}}{500})$

$$F = F_{calculé} * 5 * 10^{-11}$$

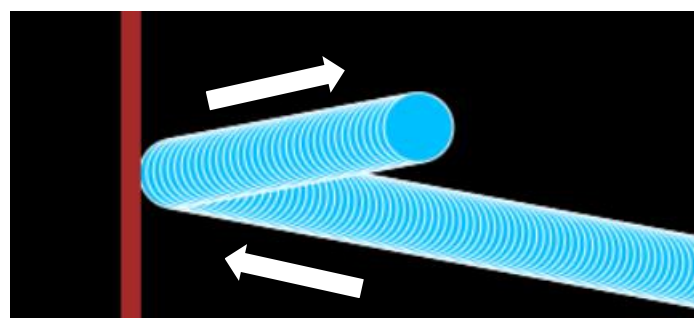
On a donc

$$P(pascal) = F_{calculé} * \frac{1}{5 * 10^{-11} * 6.6667 * 10^{-8}}$$

$$P = F_{calculé} * 2.9999 * 10^{-17}$$

Il nous reste encore à le rendre lisible, car la simulation de notre pression est donc différente de 0 seulement au moment de l'impact de la molécule contre le mur, ce qui le rend visible pendant seulement un cycle d'horloge de notre ordinateur délai qui n'est pas observable à l'œil nu. Nous avons alors décidé de modéliser la pression exercée sur le mur, comme le pression moyenne supportée pendant 5 secondes.

Bien que la mise en place de ce calcul fût simple il nous restait encore à le rendre lisible, car la simulation de notre pression est donc différente de 0 seulement au moment de l'impact de la molécule contre le mur, ce qui le rend visible pendant seulement un cycle d'horloge de notre ordinateur délai qui n'est pas observable à l'œil nu. Nous avons alors décidé de modéliser la pression exercée sur le mur, comme le pression moyenne supportée pendant 5 secondes.



4. Analyses et interprétations des résultats

Nous avons comme objectif au départ de rendre la simulation la plus flexible possible. C'est sur cette idée que différents sliders/variables modifiables seront à disposition de l'utilisateur pour rendre chaque simulation unique. Ainsi les variables de sorties telles que la température réagiront en fonction de l'état et mouvements des particules.

L'interprétation des résultats doit donc être conjointement lié avec la simulation en cours. Celle-ci doit pouvoir montrer les variables de sorties et leur signification en fonction de ce que voit l'utilisateur.

IV. Conclusion

Numériser un phénomène physique demande de l'analyse, de l'organisation, des connaissances mais aussi une logique assez affûtée pour pouvoir concilier la programmation à son sujet. Bien que les recherches internet favorise l'élaboration du projet, l'interprétation et l'études des résultats obtenus permettent d'améliorer l'optimisation du code mais avant tout la modélisation.

Depuis le début du module, de nombreuses versions du gaz ont été simulées avec des résultats uniques permettant l'amélioration du programme comme la gestion d'apparition d'une particule sur une autre ou programme mal optimisé.

Lorsqu'une simulation était un succès, l'amélioration d'un système (comme l'apparition d'un piston pour contracter le gaz) était un objectif nécessaire pour rendre un travail de qualité. A ce stade du projet plusieurs pistes d'amélioration sont envisageables.

- Composer son propre espace pour les molécules.
- La possibilité de choisir plusieurs gaz uniques pour en étudier la réaction d'un mélange.
- Permettre d'établir des graphiques et des statistiques
- Pouvoir revenir en arrière ou stopper la simulation sur un moment précis
- L'utilisateur pourrait tout aussi bien pouvoir composer son propre gaz.

V. Références

Discrétisation : <https://fr.wikipedia.org/wiki/Discr%C3%A9tisation>

Simulation à événements discrets :

https://fr.wikipedia.org/wiki/Simulation_%C3%A0_%C3%A9v%C3%A9nements_discrets

Collisions : https://fr.wikipedia.org/wiki/Choc_%C3%A9lastique

Modèle de collisions :

https://codepen.io/Full_of_Symmetries/pen/qqazdW?editors=0010

Constante de Boltzmann : https://fr.wikipedia.org/wiki/Constante_de_Boltzmann