

# **Rapport final** **ARE « DYNAMICS »**

Modélisation d'un système dynamique :  
Atomes et Molécules

avec KOITA Dado  
et LANGREE Tommi

L0 MIPI 2018-2019





## **RESUME**

Notre projet s'est ensuite porté sur les formations des molécules dans un espace donné. Nous voulions modéliser le déplacement de différents atomes et la formation des molécules qui résultent de leurs rencontres. L'idée était ainsi d'observer quelles molécules on obtenait à partir de différents facteurs tel que la quantité d'atomes présent, la répartition de ceux-ci ainsi que le temps qu'on laissait aux atomes et molécules pour se déplacer dans l'espace donné. Nous avons bien sur simplifier grandement le phénomène et avons réduit les atomes à 4, les principaux qu'on retrouve à l'état naturel dans l'univers, soient : l'hydrogène, l'hélium, l'oxygène et le carbone. Ainsi nous avons pu observer l'importance de tout les atomes malgré leur faible présence. Notre modèle ne permet pas d'appréhender les grandeurs totalement démesurées de l'univers mais met néanmoins en avant la proportion importante d'hydrogène dans celui-ci.

**-TOMMI**

## **INTRODUCTION**

Notre groupe est constitué de KOITA Dado et LANGREE Tommi. Nous sommes tout deux étudiants en L0 MIPI à Sorbonne Université et réalisons ce projet dans le cadre de l'UE d'ARE DYNAMICS. Nous nous sommes dans un premier temps orienté vers un sujet portant sur la mortalité infantile. L'idée était de modéliser l'évolution de la mortalité infantile dans différents pays en prenant en compte différents facteurs comme le PIB par habitant, le pourcentage de PIB du pays investit dans la santé, l'accès au soin etc.... Pour cela, nous nous sommes appuyés sur le code que nous avons étudié lors des premières séances, modélisant la répartition d'un gène entre 3 îles en prenant en compte la migration, la mortalité et le remplacement de la population sur plusieurs générations. Cependant, après plusieurs séances nous avons rapidement eu du mal à avancer. En effet, les facteurs causant la mortalité infantiles sont très nombreux et divers et souvent difficile à déterminer. Nous avons réalisé un modèle sur différents pays, modélisant leur taux de mortalité mais il n'était pas vraiment dynamique. Il était assez difficiles de déterminer quelles conséquences l'on pouvait obtenir en faisant varier ne serait-ce qu'un facteur. Sur conseil des encadrants, nous avons donc pris la décision de changer de sujet pour ne pas nous enliser dans quelque chose de trop peu fructueux.

C'est alors que nous avons commencé par modéliser de simples particules qui bougent dans un espace donné et nous avons imaginés que celles-ci pouvait se heurter et ainsi former quelque chose de nouveau. C'est de là que nous est venu l'idée de travailler sur la modélisation d'atomes et de molécules interagissant entre elles. Nous pourrions ainsi observer ce qu'on obtient après un certain nombre de temps et un certain nombre d'atomes de départ. Pour ce qui est de la modélisation en tant que telle nous avons travailler sur jupyter notebook en python 3 avec les modules matplotlib et scatter plot pour obtenir des nuages de points.

Dans un premier temps nous allons décrire la thématique de notre projet. Dans un seconde plan nous expliquerons le code réalisé pour obtenir une modélisation dynamique, pour ensuite analyser les résultats. Enfin nous concluerons.

**-TOMMI**

## THEMATIQUE

Notre sujet porte sur les atomes et la création des molécules. Pour des raisons techniques et de manque de temps nous avons grandement simplifié le phénomène naturel. Nous avons tout d'abord retenu seulement 4 atomes pour notre « expérience ». En effet, dans l'univers la répartition des atomes est la suivante : 75 % d'Hydrogène, 23 % d'Hélium, 1 % d'Oxygène, 0,5 % de Carbone, 0,13 % de Néon, 0,11 % de fer etc.... Ainsi, après les principaux on retrouve les autres en quantité infimes, à l'échelle de l'univers ces quantités restent suffisantes pour former de nombreuses molécules différentes mais pour notre modèle nous avons gardé seulement les 4 premiers pour représenter au mieux ce qu'il peut se produire et pour avoir assez de formation intéressantes. Nous aurions pu décider de choisir seulement l'Hydrogène et l'Hélium qui à eux deux représentent 98 % des différents atomes mais nous n'aurions pas obtenu grand-chose étant donné que l'hélium ne forme pas de liaisons avec d'autres atomes, celui-ci étant un gaz noble.

De plus, nous avons négligé le processus de liaison avec les électrons qui se partagent entre atomes, le modèle aurait été trop complexe. Enfin nous avons également esquivé les propriétés propres à chaque molécules, c'est à dire leur attractivité sur les autres, leur vitesse etc.... tout le monde sera donc logés à la même enseigne !

Notre modèle reposera donc sur le fait que les atomes se déplacent initialement dans une direction aléatoire et de façon linéaire. Ceux-ci seront limités dans un espace donné pour favoriser leur interaction, ils devront donc rebondir lorsqu'ils atteignent cette limite. De plus, tout les atomes ne peuvent pas former toute les molécules, même si nous ne prenons pas en compte le système des électrons nous respecterons néanmoins les lois fondamentales de la physique et de la chimie. Ainsi lorsqu'une paire d'atomes ou de molécules seront assez proche, nous verrons l'apparition d'un nouvel objet au profit des deux autres. Nous pourrons, de plus modifier tous ces paramètres à notre guise pour coller au mieux avec le modèle initialement observé.

**-TOMMI**

## MODELISATION

*Note :Le code produit initialement reposait sur de nombreuses listes mais qui posait quelques bugs au moment de la simulation. Il a donc été nécessaire de tout reprogrammé en utilisant des dictionnaires pour éviter les bugs et gagner en visibilité et en facilité sur le code en lui même.*

Tout d'abord, nous avons commencé par l'initialisation des particules. Elles sont réparties aléatoirement dans un repère à deux axes, tout deux allant de 0 à 1. L'idée est donc de créer une liste allant de 0 à n représentant chaque particule qui est liée à un dictionnaire de coordonnées et un dictionnaire de type, c'est à dire de la nature de la particule (si c'est un atome d'Oxygène, d'Hydrogène ...). Les coordonnées sont générées aléatoirement. Ensuite un dictionnaire de direction est créé, répartition des directions aléatoire à chaque coordonné et donc à chaque atome.

Une fonction « update » est définie par la suite. C'est celle-ci qui va gérer l'actualisation des positions des atomes, soit leur déplacement. Pour chaque coordonné on additionne la direction qui lui a été associé précédemment, positive ou négative selon un pas qui est défini par l'utilisateur. Nous utiliserons pour la suite, un « déplacement » de particule de 0,05 entre chaque frame. A cela, nous avons pris en compte les rebonds. Lorsque la particule atteint une des 4 limites, nous modifions sa direction (qui sera alors contraire) et elle se retrouve « initialisée » sur l'axe limite, pour ensuite poursuivre son chemin dans le sens inverse.

De plus, nous créons une fonction qui programme la « fusion » des deux particules qui se rencontrent, en additionnant les valeurs indiqués dans le dictionnaire type, si la somme correspond à un ensemble défini auparavant, une des deux particules est effacé et celle restante prend la valeur correspondant à la somme des deux : c'est son nouveau type. L'utilisation des dictionnaires est ainsi importante car cela permet de modifier et de supprimer les valeurs correspondantes à chaque particules.

Enfin, pour conclure ce code on définit le nombre d'images qu'on souhaite (t), on paramètre la figure et ses axes puis on appelle les fonctions pour initialiser les premières particules. On appelle ensuite les différentes fonctions dans le bon ordre pour que l'on obtienne le résultat voulu. Puis, on paramètre les particules visuellement avec des paramètres visuels correspondant au numéro type comme nous pouvons le voir ci-dessous.

```
for e in N:
    (a,b)=position[e]
    if typ[e]==2:  #H
        plt.scatter(a,b,c="gray")
    if typ[e]==20:  #O
        plt.scatter(a,b,c="red")
    if typ[e]==43:  #C
        plt.scatter(a,b,c="black")

    if typ[e]==4:  #H2
        plt.scatter(a,b,c="gray",s=150)
    if typ[e]==22:  #OH
        plt.scatter(a,b,c="red",s=200,marker='s')
    if typ[e]==65:  #COH
        plt.scatter(a,b,c="red",s=300,marker='s')

    if typ[e]==40:  #O2
        plt.scatter(a,b,c="red",s=150)
    if typ[e]==60:  #O3
        plt.scatter(a,b,c="red",s=300)
    if typ[e]==63:  #CO
        plt.scatter(a,b,c="red",s=100,marker='p')
    if typ[e]==83:  #CO2
        plt.scatter(a,b,c="red",s=200,marker='p')
    if typ[e]==24:  #H2O
        plt.scatter(a,b,c="blue",s=200,marker='x')
```

## **ANALYSE :**

Nous pouvons tout d'abord voir avec une répartition égale entre les 4 atomes l'apparition des différentes molécules que nous pouvons obtenir, c'est un bon signe, cela signifie que notre système fonctionne dans un environnement optimal. En effet, en ayant répartis les particules en parts égales nous favorisons ainsi l'émergence de nouveaux objets qui vont à leur tour en former ou non.

Désormais, nous appliquons les valeurs de répartition existantes dans l'univers. Le résultat est sans appel, il est maintenant beaucoup plus rare d'obtenir une variété importante d'objets. On observe principalement du dihydrogène ainsi que de l'hélium sur le long terme. L'hélium étant un gaz noble, il est attendu qu'il en reste autant, et l'hydrogène permet de créer toute sorte de molécules, cependant ici notre modèle étant limité techniquement et par nos choix d'atomes on obtient dès le début que très rarement un autre élément et donc il se passe très rarement grand-chose. On peut donc dire que l'hydrogène joue un rôle important dans la diversité des éléments que nous connaissons et en quantité infiniment plus grande que notre simulation avec tous les éléments existant il en va sans dire que de nombreux autres éléments sont aussi créés.

**-TOMMI**

Dans le cadre de l'ARE j'ai choisi ce sujet de base même si on en a pris un autre. Je me suis occupée du premier PowerPoint de présentation en terme de correction. Je suis celle qui me suis occupée de la recherche des données et de la gestion du drive et github pour la gestion et partage du travail. Dans un premier temps j'ai recueilli toutes les données concernant le sujet que j'ai aussi sauvegardé sur une clé USB. Lors de notre changement de projet je me suis à travailler sur la notion de collision de particules, j'ai fait de nombreuses recherches sur cette notion scientifique. En effet, j'avais réalisé un brouillon, ensuite Tommi s'est occupé du code avec mon aide, pendant ce temps je faisais des recherches sur matplotlib et les animations codées sur python. Lorsqu'on codait je regardais comment corriger et utiliser les fonctions. Si Tommi voulait de l'aide sur une fonction de matplotlib je regardais comment on l'utilise.

**-DADO**

## CONCLUSION

### Journal de bord :

- Séance du lundi 11 mars

Dado et Tommi :

Recherche de statistiques sur les 3 pays, choix de certains critères à utiliser pour la fonction, retranscription de ces données en paramètres dans un nouveau notebook et début de réflexion sur le fonctionnement de la fonction au brouillon. Création du journal de bord et listage des sources utilisées.

- Séance du lundi 18 mars

Dado et Tommi :

Recherche documentation sur les statistiques du taux de vaccination

Début de création de la fonction « taux de mortalité » pour obtenir un résultat → problèmes de conception : différents facteurs qui agissent sur les agents, et pb général du résultat attendu

Réorientation du projet, nouveau sujet : déplacement de point dans un espace qui peuvent fusionner ... Parallèle avec les molécules etc...

- Séance du lundi 25 mars

Réalisation des déplacements des particules et notamment des rebonds

- Séance du lundi 1 avril

Tommi : Finalisation de la fonction fusion après de nombreux essais, cela semble désormais fonctionner. Tentative d'animations mais sans résultats, le module ne semble pas fonctionner sur le pc. Essai également avec Tkinter mais je ne parviens pas à adapter mon code et à prendre en main cet outil.

- Séance du lundi 8 avril

Tommi : Correction des derniers bugs qui pouvaient subsister. Tentative de correction d'un en particulier qui envoyait un KeyError pour une raison inconnue. Après de nombreuses tentatives et l'aide des encadrants, je recode avec cette fois-ci des dictionnaires à la place des listes pour éviter les problèmes. Le résultat fonctionne totalement !

**-TOMMI**

Pour conclure, notre sujet et problématique étaient sur la mortalité infantile dans le monde. Cependant cette question nous fut beaucoup trop complexe à mettre sous forme de

dynamique simple. Malgré cela nous avons décidé d'accepter ce projet sur la collision des particules. Le travail de groupe fut partagé du travail. J'ai acquis des compétences sur python notamment dans l'utilisation des bibliothèques de codage. J'ai pu apprendre le sens de la recherche d'une notion précise et ne pas se baser uniquement de Wikipedia.

**-DADO**

De mon côté, l'ARE a été une expérience enrichissante même si nous avons eu de nombreux problèmes et avons dû recommencer un nouveau projet en cours de route. Cela m'a permis non seulement de me remémorer les compétences de 1I001 que j'avais acquis mais aussi de les approfondir, surtout sur le plan graphique et simulation. L'exercice de recherche et de simulation sur un sujet donné sera sans aucun doute bénéfique pour la suite de mon parcours .