

Enzo Bonnal
(L3)

Maîtres de stage :
Romain Maggiolo
et Gaël Cessateur

*Modélisation numérique de la trajectoire de
particules dans une cavité de la surface glacée
d'un grain de poussière d'un Nuage Moléculaire.*

Stage effectué du 21 Juin au 29 Juillet 2016 dans l'Institut royal d'Aéronomie Spatiale de Belgique,
3 Avenue Circulaire, Bruxelles.

Merci à Romain et Gaël pour leur attention,
leur bonne humeur et leur patience !

Résumé :

La récente observation par la sonde Rosetta d'une grande quantité d'O₂ dans le coma de la comète 67P était inattendue.

Les Scientifiques pensent que le Dioxygène contenu dans les comètes leur a été incorporé durant leur formation, consistant en l'agglomération de petits grains de poussière glacés du nuage moléculaire au début de notre système solaire.

Ce qu'il advient de remarquer est que la couche de glace recouvrant les grains semble être très poreuse et irrégulière mais que les modèles physico-chimiques utilisés pour prédire la quantité d'O₂ présent dans la comète ne prennent pas en compte cette caractéristique et travaillent avec une surface glacée plane.

Pour avancer un peu plus dans cette direction, j'ai été chargé de réaliser un programme qui puisse simuler le mouvement d'atomes d'Hydrogène et d'Oxygène au sein d'irrégularités de la surface glacée et poreuse d'un grain et de déterminer en quelles proportions les atomes se recollent sur la surface ou s'échappent du grain vers la phase gazeuse du nuage moléculaire. J'ai mis au point une modélisation approximative d'une cavité glacée et de la physique d'une particule se déplaçant à l'intérieur de celle-ci. J'ai ensuite réalisé des statistiques à partir de millions de simulations et conclus que l'Hydrogène est beaucoup plus enclin à s'échapper du grain que les particules plus massives telles que l'Oxygène.

Ainsi, après un dégazage ou la collision d'un rayonnement cosmique avec le grain, il y aura plus d'atomes d'Oxygène qui se recolleront sur la surface que d'atomes d'Hydrogène et des réactions chimiques vont avoir lieu avec une part initiale d'Oxygène plus grande que celle prise en compte par les modèles avec surface plane. Et comme dans tout les modèles chimiques le fait d'augmenter la quantité d'Oxygène disponible pour une quantité d'Hydrogène donnée entraîne une plus grande production d'O₂, ce résultat est encourageant pour développer un peu plus cette idée.

Abstract :

The recent observation by the Rosetta probe of large amount of O₂ in the coma of comet 67P was unexpected.

Scientists think that the O₂ in comets has been incorporated to them during their formation, that consists in the agglomeration of icy grains from molecular cloud and composed of a silicate nucleus covered by an ice layer.

The point is that this icy layer seems to be highly porous and uneven and the chemical models used to predict the amount of O₂ in comets composition don't include that feature and consider the grain's surface as a planar icy surface.

To investigate in this way I've been asked to make a program that can simulate the outgassing of H and O atoms from icy non-planar surface and see if they are caught again by the icy surface or if they leave the grain and join the gas phase of the molecular cloud. I had to imagine a way to modelize an icy cavity and the physic of a particule moving inside. Then, I made statistics on millions of simulations and conclude that hydrogen is more likely to out-diffuse from the grain compared to heavier species like O.

After an outgasing or a collision between a cosmic ray and the grain, there will be a higher number of O that will be stucked again on the surface than the number of H : chemical reactions will occurred on the surface of the icy grain with a higher initial percent of O available compared to the one predicted by the planar surface models. Moreover, in all chemical models, if we increase the number of O atoms available for a number of H atoms available given, it will results in a higher production of O₂ : This result can invite us to keep on working on this idea.

Table des matières :

I/ Contexte, Motivations et Problématique du stage

(page 5)

II/ Réalisation du Programme

(page 7)

III/ Utilisation du Programme et Résultats

(page 12)

IV/ Conclusion

(page 17)

V/ Bibliographie

(page 18)

VI/ Annexe

(page 19)

J'ai pris mes fonctions de stagiaire le 21 Juin 2016 dans l'équipe de mes deux maîtres de stage **Romain Maggiolo et Gaël Cessateur**, dirigée par **Johan de Keyser** à l'**IASB de Bruxelles**.

Une des objectifs de cette équipe est d'approfondir nos connaissances sur les comètes en interprétant des données obtenues par la mission Rosetta (l'IASB a participé à la création de ROSINA, l'un des instruments que la sonde a emmenés à son bord pour aller à la rencontre de la comète 67P).

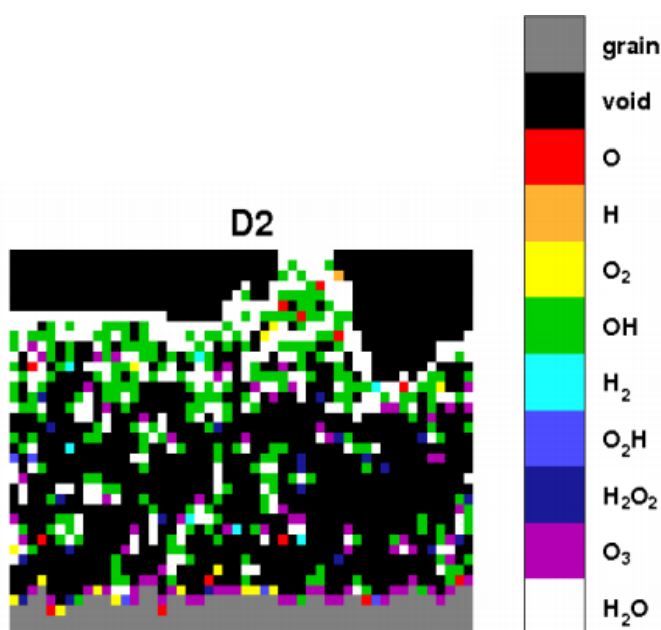
C'est dans ce contexte que j'allais être occupé durant ces 6 semaines, puisque la problématique qui a motivé mon stage est directement issu des mesures réalisées grâce à la sonde Rosetta : il s'agit d'explorer une voie qui pourrait à terme permettre d'expliquer la présence sur la comète 67P d'un taux d'O₂ plus important que celui prévu par les modèles. Je fut occupé dans un premier temps par la lecture d'articles pour me permettre de mieux cerner le contexte et la problématique qui m'avaient été présentés, puis dans un deuxième temps par la **confection d'un programme** de modélisation et enfin par l'**interprétation des résultats** obtenus à l'aide de ce dernier.

I/Contexte, Motivations et Problématique du stage

Parmi les différentes mesures et observations effectuées par les instruments de la sonde Rosetta, on s'intéresse au taux de Dioxygène que contient la comète 67P. Des modèles chimiques ont été réalisés pour prévoir ce taux mais il s'avère qu'il est en réalité bien plus élevé que les estimations théoriques. À noter qu'il est pour l'instant admis que l'O₂ que contient une comète lui est incorporé dès sa formation dans les premiers âges de notre système solaire, lorsque des **grains de silice recouverts de glace** de la nébuleuse protosolaire (on parlera de nuage moléculaire dans la suite) s'agglomèrent pour lui donner naissance.

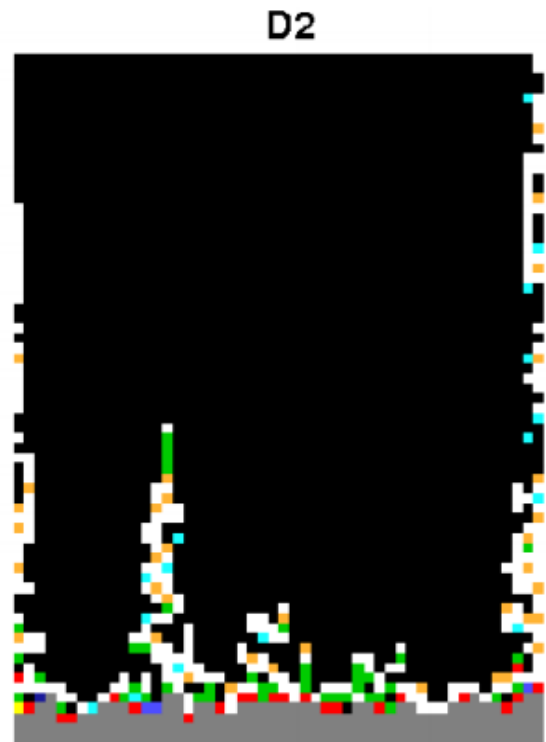
On m'a alors demandé de débiter l'exploration d'une possible piste pour expliquer cet écart : **Le fait que la géométrie de la surface des grains soit irrégulière et poreuse**. En Effet, les modèles chimiques utilisés résolvent un équilibre chimique entre la phase gazeuse du Nuage Moléculaire et la surface glacée et **plane** des grains et aboutissent à un taux d'O₂ inférieur à l'observation.

Le fait de considérer une surface de grain irrégulière plutôt que plane a été encouragé par différentes lectures comme par exemple cette étude numérique de la formation de la couche de glace sur le noyau de silice des grains réalisée par Cuppen et Herbst (2007) :



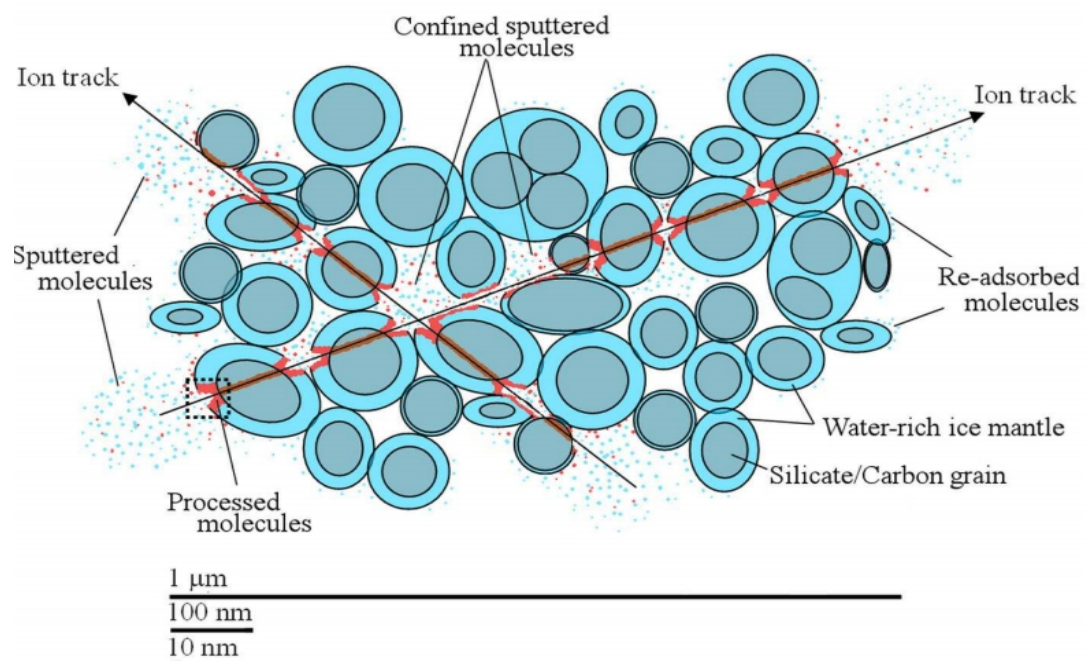
Générations numérique d'une coupe de la couche de glace recouvrant les grains :

On y découvre une surface poreuse (1ère image) ou encore des sortes de renforcements (2ème image).



On retrouve aussi cette caractéristique à plus grande échelle, avec cette représentation d'une agglomération de grains transpercée par des rayons cosmiques lourds retrouvée dans un article de Pillig et al. (2010) : on y aperçoit des cavités ainsi qu'une sorte de tunnel créé par le passage de l'ion.

Typical dense cloud grain



J'ai alors du essayer de réaliser la tâche suivante : **Modéliser le comportement de particules s'échappant de la glace des grains** (après un quelconque apport d'énergie), et plus particulièrement les atomes d'Hydrogène et d'Oxygène, **dans une cavité ou une irrégularité de la surface des grains**. La partie programmation de mon stage pouvait alors commencer.

II/ Réalisation du Programme

1) Simplification du problème à traiter :

Il a tout d'abord fallu prendre conscience que le problème est très complexe et qu'il allait falloir faire énormément d'hypothèses. Finalement, ma modélisation devait grossièrement ne prendre en compte que la facette « **trajectoire d'un corps dans une cavité** » du problème.

2) Principe général et principales caractéristiques du travail réalisé:

On va considérer un volume qui possède une ou deux ouvertures vers l'extérieur : on modélise ainsi une cavité dans la surface glacée d'un grain et plus ou moins ouverte vers l'extérieur. Le but sera de simuler la vie d'une particule au sein de cette cavité : de son arrivée à l'intérieur de la cavité (depuis l'extérieur ou alors depuis la surface gelée, à cause d'un apport d'énergie), en passant par ses différents rebonds sur les parois pour finir par terminer sa course :

- soit parce qu'elle est sortie de la cavité par l'ouverture vers la phase gazeuse extérieure.
- soit parce qu'elle s'est recollée sur la surface.

L'Objectif du programme est de permettre de **sauvegarder les informations** suivantes sur la vie de la particule simulée : Le temps qu'elle a passé dans la cavité, le nombre de contacts effectués contre les parois de la cavité, le résultat de la simulation (c'est-à-dire le fait que la particule soit sortie de la cavité ou bien qu'elle s'y soit recollée).

Ensuite, il doit être en mesure de réaliser des **moyennes** sur un très grand nombre de résultats de simulations, avoir la possibilité de faire **varier un paramètre** de l'expérience ainsi que de traiter les données pour les **tracer dans des graphes** facilement interprétables.

3) Le choix du langage :

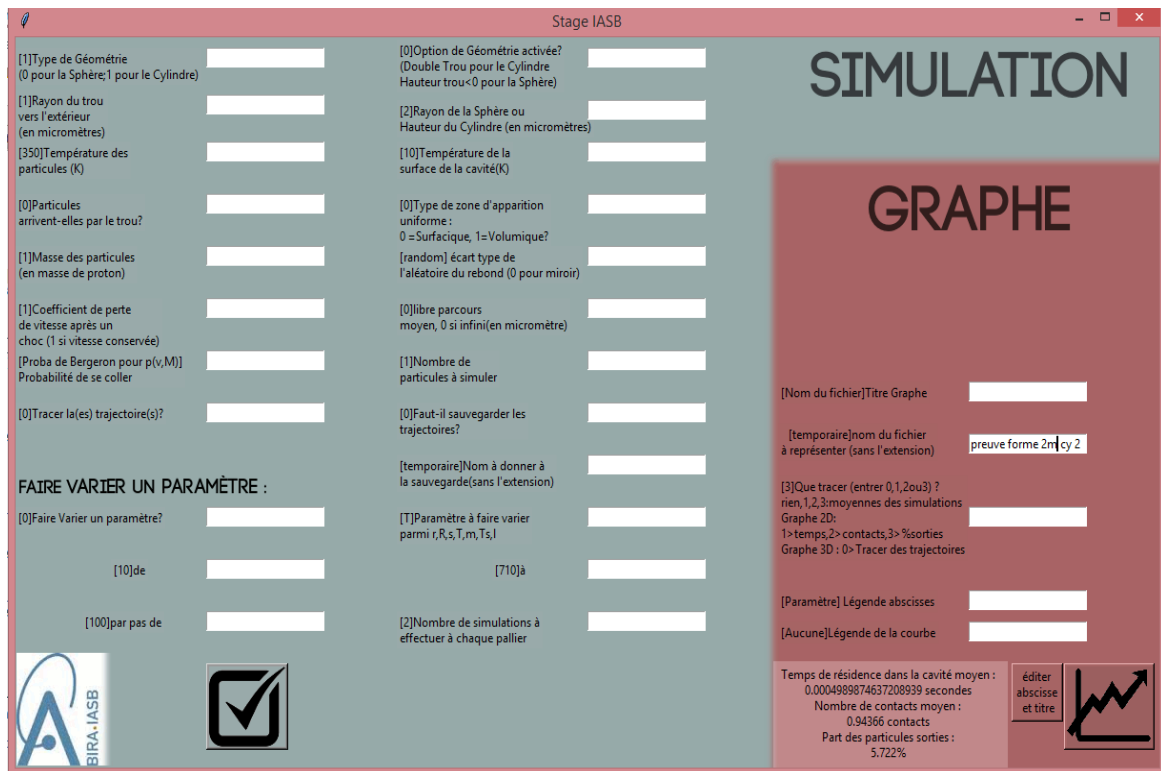
Le choix a été assez restreint par mes connaissances actuelles et s'est donc fait entre le **C(++)** et **Python**. La modélisation étant basé sur le fait de réaliser des moyennes sur de très nombreuses simulations (de l'ordre de la centaine de milliers) , l'utilisation d'un langage compilé comme le C permettrait d'accroître considérablement la vitesse d'obtention des résultats (ou, pour une durée équivalente, d'être plus précis) ... Cependant je me suis vite rendu compte que le temps gagné sur le développement du fait de l'utilisation de Python allait être plus avantageux.

4) Portabilité et facilité d'utilisation :

Comme mon programme allait peut être être réutilisé par d'autres personnes, il a fallu le rendre **intuitif** autant que possible à l'aide d'une interface graphique complète dispensant l'utilisateur d'avoir à se plonger dans le code et me permettant également d'éviter de me perdre dans les paramètres :

De plus, il fallait réfléchir à sa **portabilité** puisque je l'ai développé sous Windows mais que les machines du laboratoire destinées au calcul sont sous une distribution Linux. Une partie du travail a donc été d'adapter le code pour qu'il tourne sans encombre sur le système d'exploitation et la version de python utilisée.

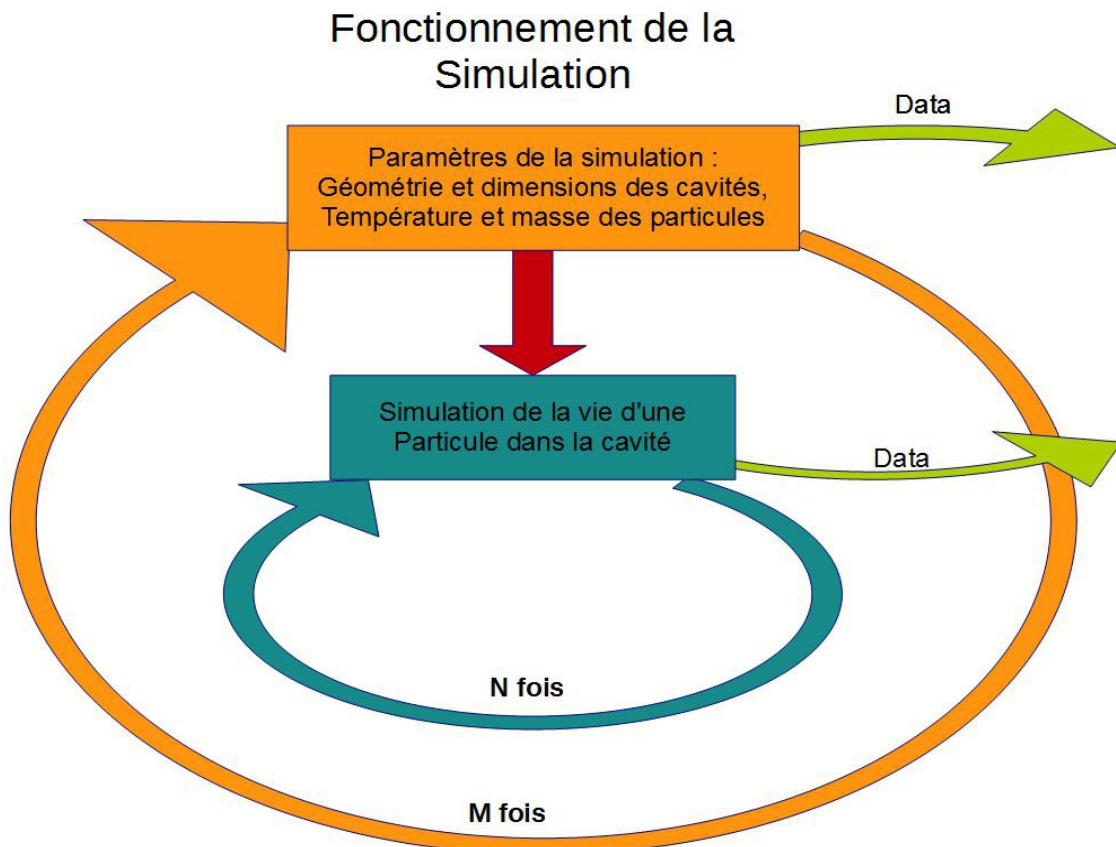
*J'ai également réalisé un semblant de **petit guide d'utilisation** qui se trouve en annexe et qui donne une bonne idée du fonctionnement du programme.*



Allure de l'interface graphique du programme

(réalisée à l'aide de la bibliothèque Tkinter native de Python)

5) Fonctionnement du programme :



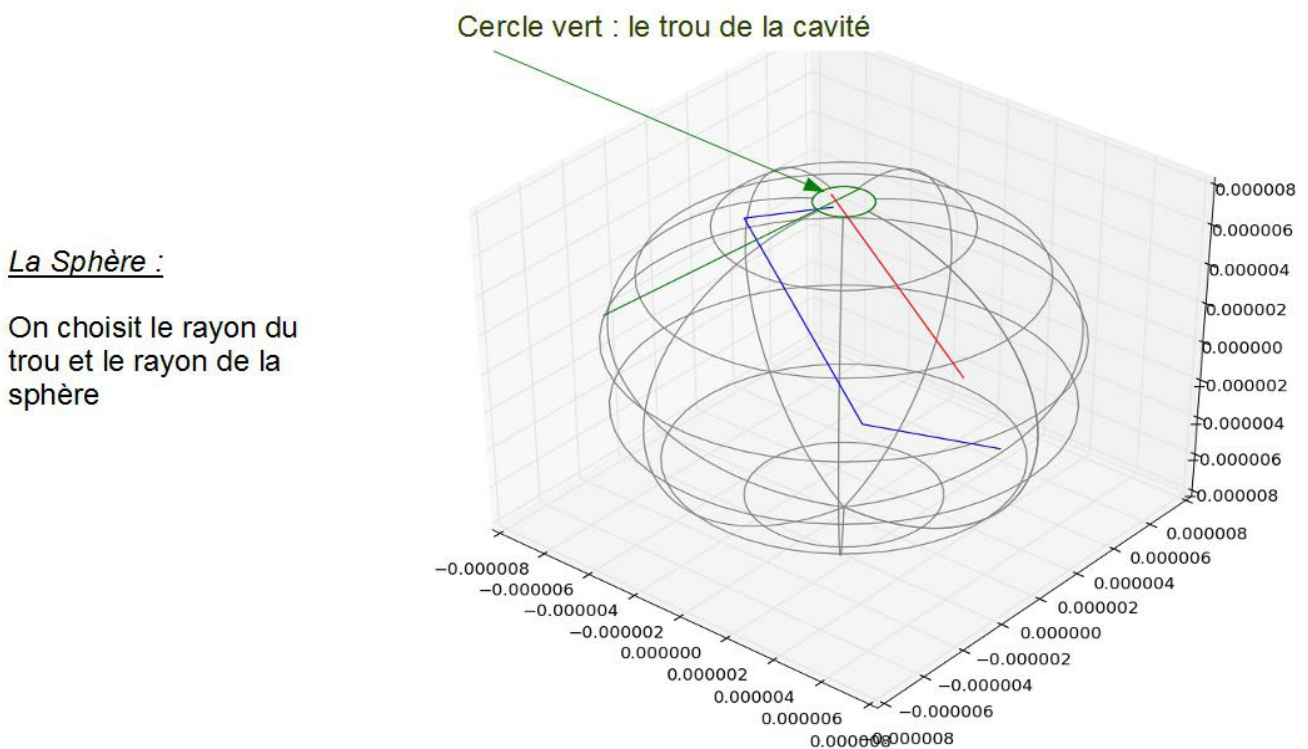
Le programme fonctionne avec comme mécanique principale l'imbrication de **deux boucles** autour d'une action centrale qui est la simulation de la vie d'une particule :

-la **1ère boucle** est réalisée directement sur l'action principale et elle permet de réaliser autant de fois que désiré des simulations à partir des mêmes paramètres.

-La **2ème** est une boucle qui réalise la première boucle autant de fois que désiré et qui va en plus faire varier un paramètre de l'action principale (la simulation de la vie d'une particule), par exemple la masse de la particule.

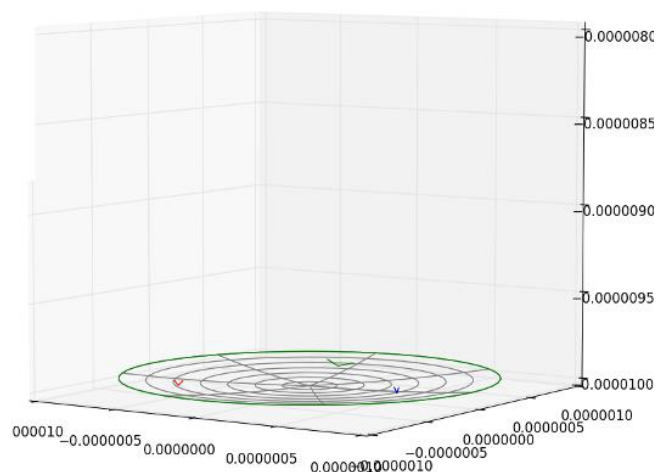
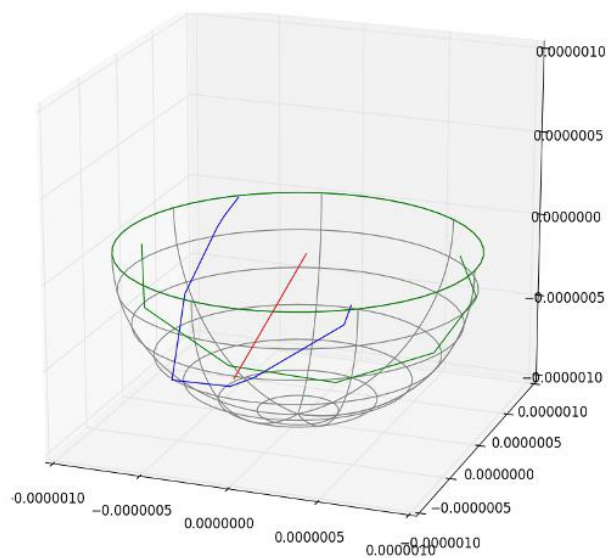
Comme on peut le voir dans l'encadré de la boucle Orange, il faut sélectionner une « géométrie » de la cavité. Le programme n'en prend en compte que deux , La **SPHERE** et le **CYLINDRE** qui, lorsque l'on joue sur leurs dimensions et le positionnement de l'ouverture circulaire vers l'extérieur, offrent de bonnes possibilités de modélisation d'une imperfection de la surface glacée d'un grain:

Ainsi, voici ce que l'on obtient si l'on désire **visualiser les trajectoires** d'une ou plusieurs particules à la suite d'une simulation : (les trois courbes rouge/vert/Bleu sont les trajectoires de trois particules)



(Cette visualisation est réalisée à l'aide de la bibliothèque matplotlib)

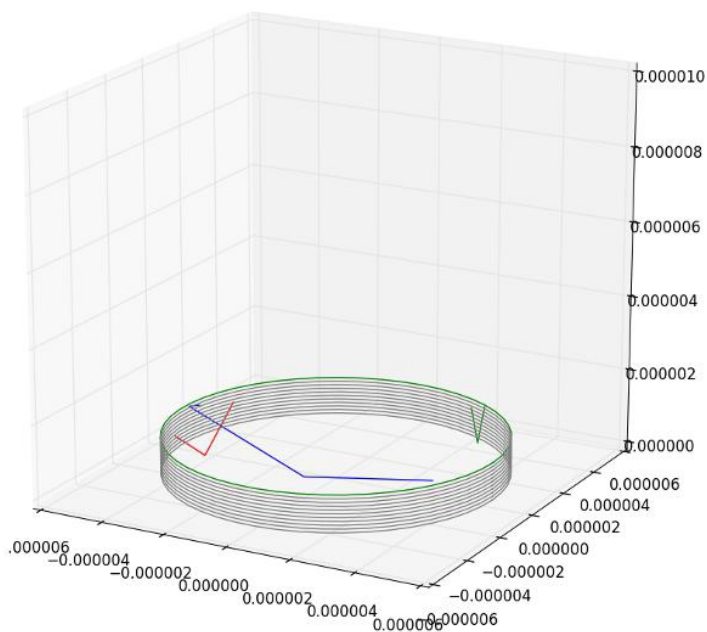
Toujours avec le mode « géométrie sphérique », on va maintenant jouer sur le positionnement de l'ouverture pour obtenir par exemple une cavité en forme de **demi sphère**, puis à l'extrême, une cavité équivalente à **une surface plane** (cf les deux premières figures page suivante)



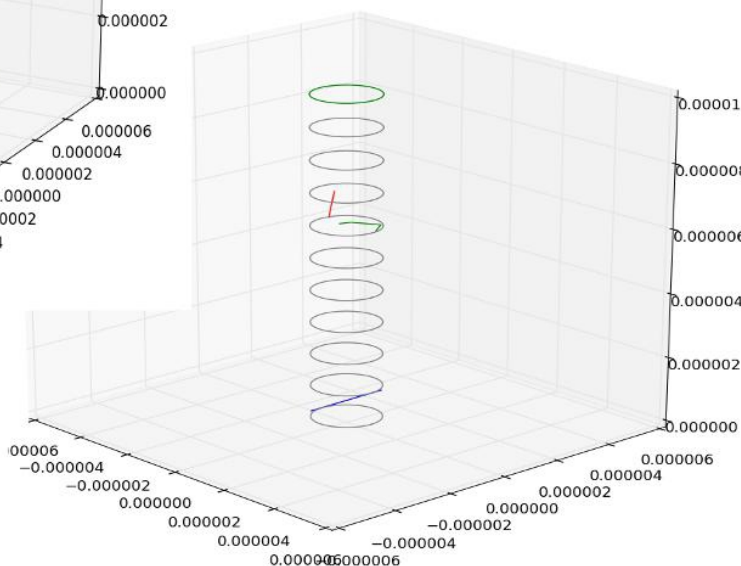
Voici maintenant la « **géométrie cylindrique** » qui nous permet une spécification supplémentaire : en plus du trou en haut du cylindre, on peut également le percer à ses deux extrémités pour reproduire une sorte de **tunnel**.

Le Cylindre :

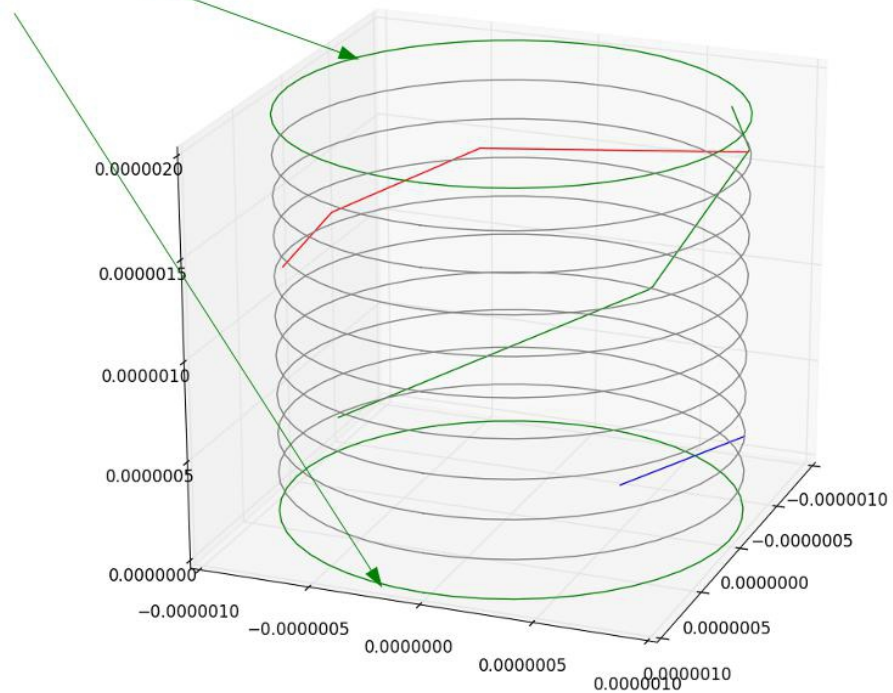
On choisit le rayon du trou et la hauteur du cylindre.



Ici : le fond est plein, il n'y a un trou qu'à *une seule* extrémité.



Ici un Cylindre avec une sortie
à chacune de ses extrémités :



Les sorties et traitement des Datas :

Comme on le voit sur le schéma du fonctionnement des boucles du programme, les datas peuvent être enregistrés à deux endroits (les deux flèches vertes).

Je vous invite à vous référer au guide en Annexe pour plus de détail sur l'enregistrement des données mais voilà à quoi ressemble un fichier de sauvegarde d'une expérience réalisée :

```
legende temporaire natureQ 1 mode2 1 sigma 0.3 r 4.9999
TempsVie(s) ; Nb contacts ; SortieParTrou?
1e-06 5.166132793477091e-10 0 1
1e-06 1.252998185367141e-09 1 1
1e-06 5.554135373070392e-10 0 1
1e-06 1.5880034424668405e-09 1 1
1e-06 1.2105429189673022e-09 1 1
1e-06 4.404740939337954e-10 0 1
1e-06 1.567818975850679e-09 0 1
1e-06 2.4764880883465263e-09 1 1
1e-06 5.56960965558355e-10 1 0
1e-06 4.843752337304791e-10 0 1
1e-06 2.78110876612243e-10 0 1
1e-06 1.021020078328304e-09 0 1
```

La première ligne (qui est 5 fois plus longue en réalité mais que j'ai du couper) contient la valeur de tous les **paramètres** de la simulation.

À partir de ces données, il est possible de tracer des **graphes** tels que *le temps que passent les particules dans la cavité en fonction de la taille de la cavité* ou bien encore *la part des particules qui réussissent à sortir de la cavité en fonction de leur Température* ou encore de leur masse. Il est en fait possible de faire varier 8 paramètres différents. (cf guide)

6) Principes physiques utilisés :

Différentes options de paramétrage sont mises à la disposition de l'utilisateur et parmi elles, certaines s'appuient sur des principes physiques n'étant pas toujours pertinents et il incombe à l'utilisateur de ne pas les utiliser si cela semble hors propos dans la simulation réalisée :

-Température et vitesse des particules :

Un des paramètres de la simulation est la vitesse initiale de la particule, le programme rend possible deux types d'ordre : prendre une **vitesse définie directement** (exemple 1000 m/s) ou bien en choisir une aléatoirement en utilisant la **loi de répartition** des vitesses de Maxwell-Boltzmann.

-Le libre parcours moyen :

Il est possible, si l'on désire simuler un grand nombre de particules dans la cavité par exemple, de simuler des collisions à l'aide d'un ordre en terme de libre parcours moyen : On va ensuite utiliser une loi de densité de probabilité exponentielle décroissante pour générer la distance que la particule va parcourir avant la prochaine collision (distance qui est donc différente pour chaque collision).

-Le type de rebond :

Lorsqu'il y a contact entre la particule et une paroi, on a accès à la normale à la paroi au point de contact et on peut ainsi réaliser aisément un rebond parfait et trouver la nouvelle direction de la vitesse de la particule.

Il est également possible d'appliquer une perte d'énergie après le contact (voire une hausse) à l'aide d'un paramètre représentant la part de sa vitesse que la particule va conserver à la suite du contact (remarque : il suffit de prendre le carré de ce coefficient si l'on désire qu'il représente plutôt la part de son *l'énergie* que la particule conserve) : exemple, si l'on entre 0,8 et que la particule arrive sur la paroi à 1000m/s, elle repartira avec une vitesse de 800m/s.

De plus, il est possible de modéliser les **irrégularités** de la paroi de la cavité par une modification apportée à sa normale suivant une loi gaussienne d'écart-type paramétrable par l'utilisateur . (indiquer 0 avec ce paramètre revient à choisir un rebond parfait). On peut également choisir un mode où le rebond est totalement aléatoire (isotrope dans le demi espace dirigé vers l'intérieur de la cavité).

-La loi de Probabilité de Collage de la particule :

Il est possible de donner une valeur à la probabilité directement, ainsi si l'on entre par exemple 0,1 les particules auront 1 chance sur 10 de se coller à chaque contact avec la paroi. Sinon le programme va utiliser par défaut une loi de probabilité qui diminue avec la vitesse et augmente avec la masse de la particule et qui est valable pour une paroi de glace à 10K : c'est une loi trouvée dans un article de E. Matar dont la référence se trouve dans la bibliographie.

C'est cette loi que j'ai utilisé lors des simulations de la partie suivante :

III/ Utilisation du programme et Résultats

Je me suis servi du programme pour sortir quelques résultats permettant d'éclairer un petit peu la situation suivante, toujours en gardant à l'esprit la problématique initiale:

Est-ce que considérer une surface de grain non plane peut permettre d'expliquer ce fort taux d'O₂ inattendu ?

On considère une surface de grain (de silice recouvert de glace) **irrégulière et/ou poreuse** de température 10K (l'hypothèse de la température de 10 K est nécessaire du fait de la loi de Probabilité de collage utilisé).

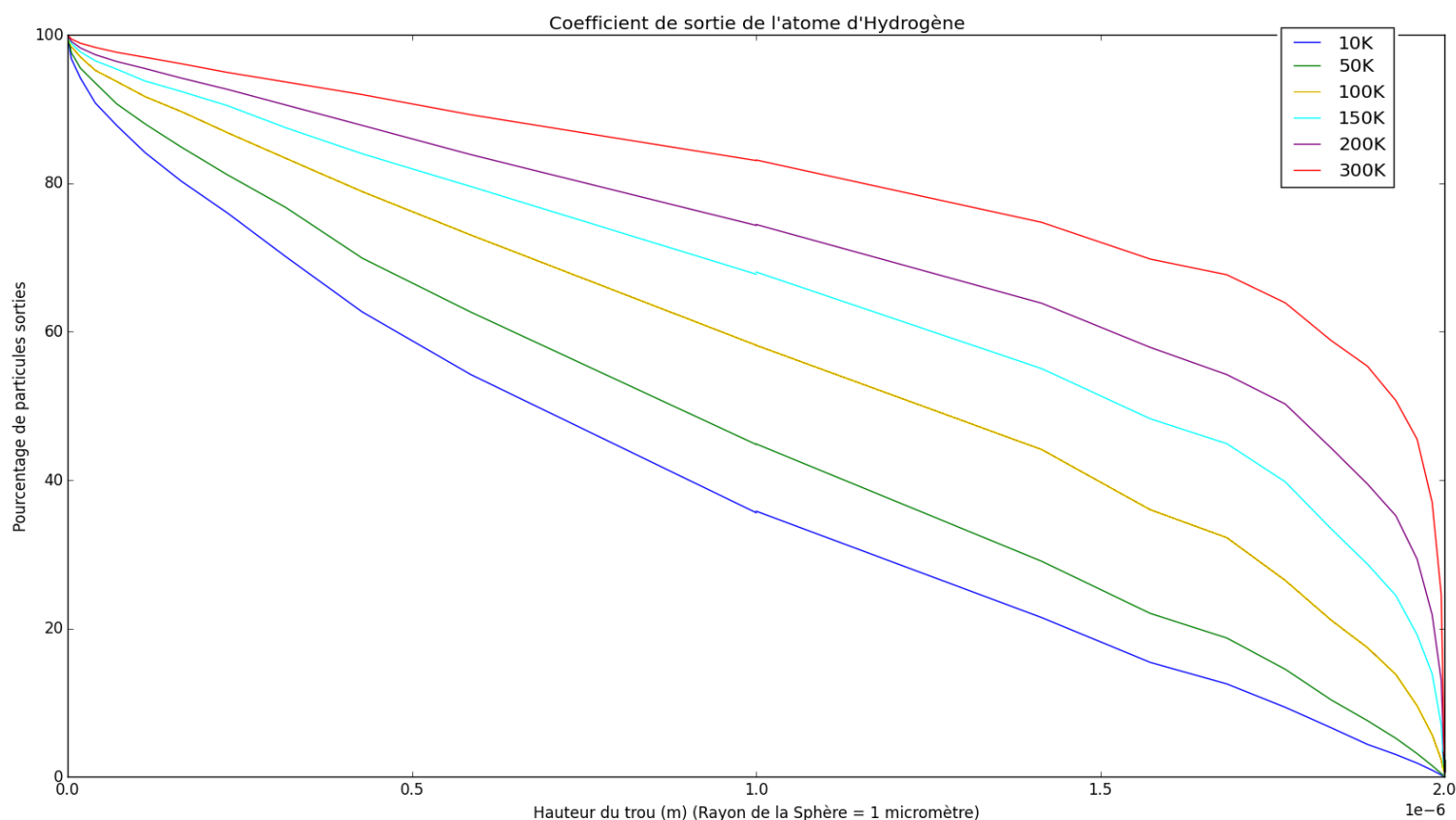
Des **émissions de particules depuis la surface** du grain peuvent avoir lieu, par exemple lorsqu'un rayonnement qui vient casser du H₂O ou lors d'un dégazage.

Ici je me suis intéressé aux atomes d'**Hydrogène** et d'**Oxygène** libérés depuis la surface et plus particulièrement à la part de chacun d'eux qui parvient à quitter la cavité, en fonction de la **forme de la cavité** (ou surface cabossée, vallonnée) ainsi que de la **température** (ou vitesse) des atomes considérés.

Graphes obtenus et interprétations:

Ces graphes sont réalisés avec une « géométrie sphérique » :

L'Hydrogène :



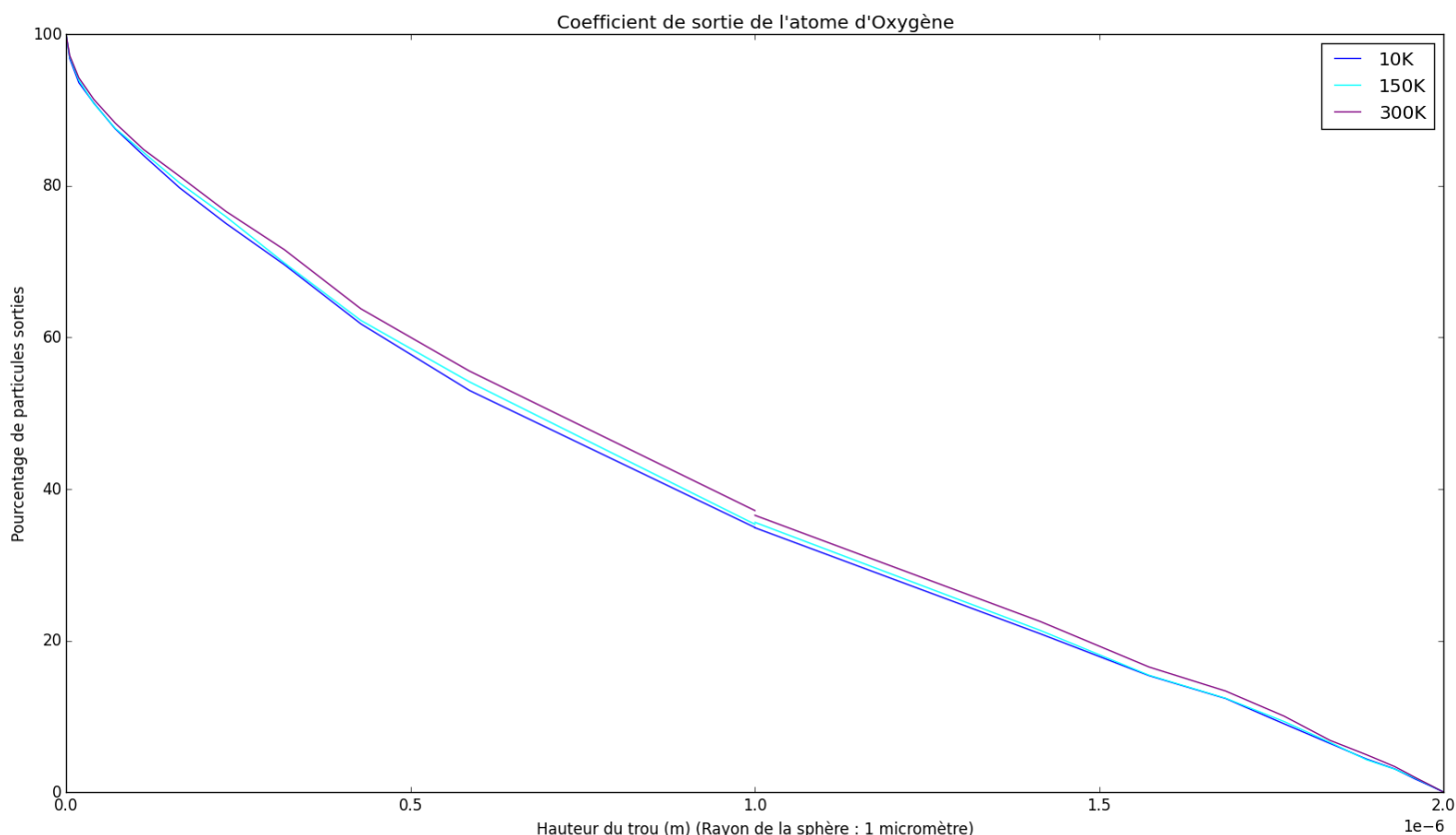
On a ici en abscisse la **Hauteur du trou** pour un rayon de la sphère fixe de 1 micromètre : ainsi lorsque cette grandeur vaut 0 on a une **surface plane**, lorsqu'elle vaut 1 micromètre on a une **demi-sphère** et lorsqu'elle vaut 2 micromètres on a une **sphère totalement bouchée**.

On a donc sur ce graphe la part de l'Hydrogène qui réussit à sortir de la cavité en fonction de l'ouverture de cette dernière sur le milieu extérieur et ce pour différentes températures de l'atome.

Remarque : Je parle de température pour un seul atome et non de vitesse car j'ai ici attribué aux particules des vitesses initiales en suivant la loi de distribution de vitesses le Maxwell-Boltzmann précédemment évoquée.

Interprétation : On voit que dès que l'Hydrogène est un peu excité lorsqu'il commence sa trajectoire dans la cavité il obtient très vite un fort taux de sortie. Le taux de sortie augmente naturellement avec la vitesse de l'atome ainsi que l'ouverture de la cavité.

L'Oxygène :



On a exactement le même type de graphe mais cette fois-ci on considère l'atome d'Oxygène. On voit que contrairement à l'Hydrogène, **même à fortes températures, les atomes d'Oxygène sont recaptés facilement.**

Conclusion qualitative :

À partir de ces résultats je peux tirer ma seule conclusion : La différence entre le comportement d'un atome lourd comme l'Oxygène et celui d'un atome léger tel que l'Hydrogène **croît très vite** avec la vitesse de ces derniers et est très conséquente : Ainsi, pour une surface un peu vallonnée, disons qu'une irrégularité peut ressembler à une **demi sphère** (soit une hauteur de trou égale au rayon de la sphère), et pour des atomes quittant la surface à une **centaine de Kelvins, la part des atomes d'Hydrogène qui se recollent sur la surface est de 40 % alors que c'est le cas de plus de 60% des atomes d'Oxygène dans la même situation !**

Ainsi, si les atomes expulsés dans l'intérieur de la cavité ont une relativement grande Énergie Cinétique lors d'un dégazage ou suite à l'arrivée d'un rayon cosmique, alors l'intérieur de la cavité sera un lieu très intéressant : En effet sa surface contiendra de nombreux atomes d'Oxygène prêts à réagir pour former la nouvelle couche de glace tandis que les H très volatiles auront probablement déjà quitté les lieux.

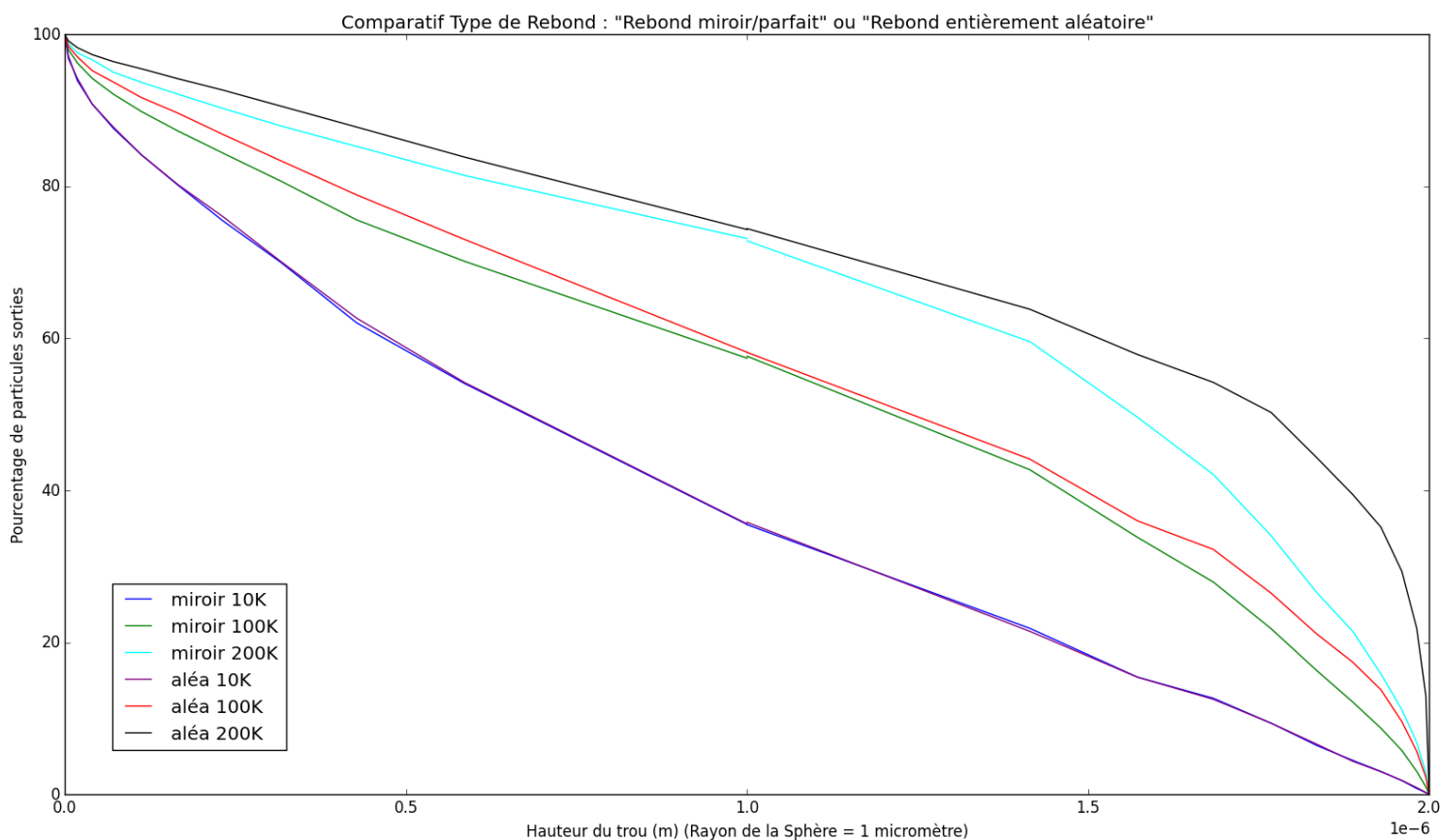
Et ainsi, on peut espérer que ce phénomène encouragera la formation de plus de Dioxygène que prévu !

→ Ce résultat semble donc aller dans le sens désiré, à savoir réussir à expliquer le taux d' O_2 observé sur 67P à l'aide de la géométrie de la surface des grains dont elle constitue l'agglomération.

Choix de modélisation effectués :

Le type de rebond :

-Lors des deux simulations précédentes, après un choc avec la paroi, les particules prennent une direction parfaitement **aléatoire**. Ce choix **arbitraire** de modélisation est donc à justifier :

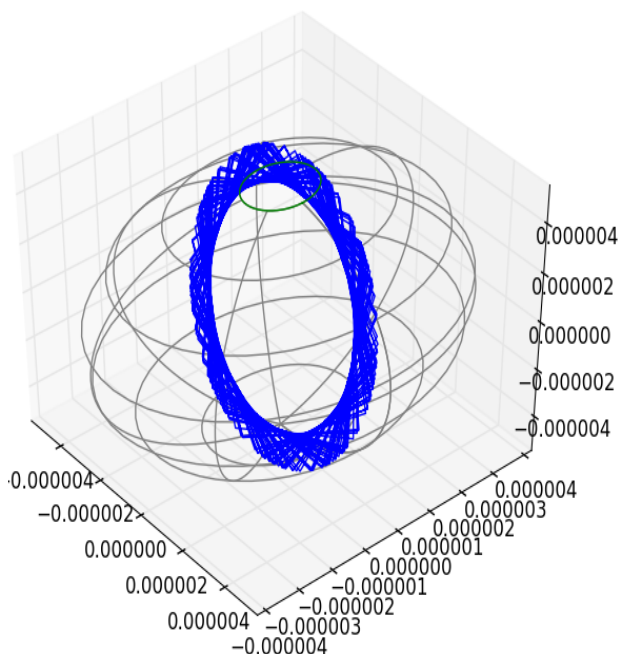


Voici donc un comparatif, pour l'atome d'Hydrogène, entre la modélisation du rebond par un rebond aléatoire ou par un rebond parfait (« miroir »).

On voit que le choix du rebond n'est pas primordial et ne change pas la conclusion tirée !

On remarque également que l'écart est toujours en faveur d'une sortie plus grande en mode aléatoire et que l'écart est plus important à forte température et lorsque la cavité est très fermée : Ceci s'explique par le fait qu'avec un rebond parfait/miroir et une géométrie sphérique, il existe des trajectoires dites « piégées », c'est à dire que même si la particule avait une probabilité nulle de se coller après contact, elle ne pourrait pas sortir :

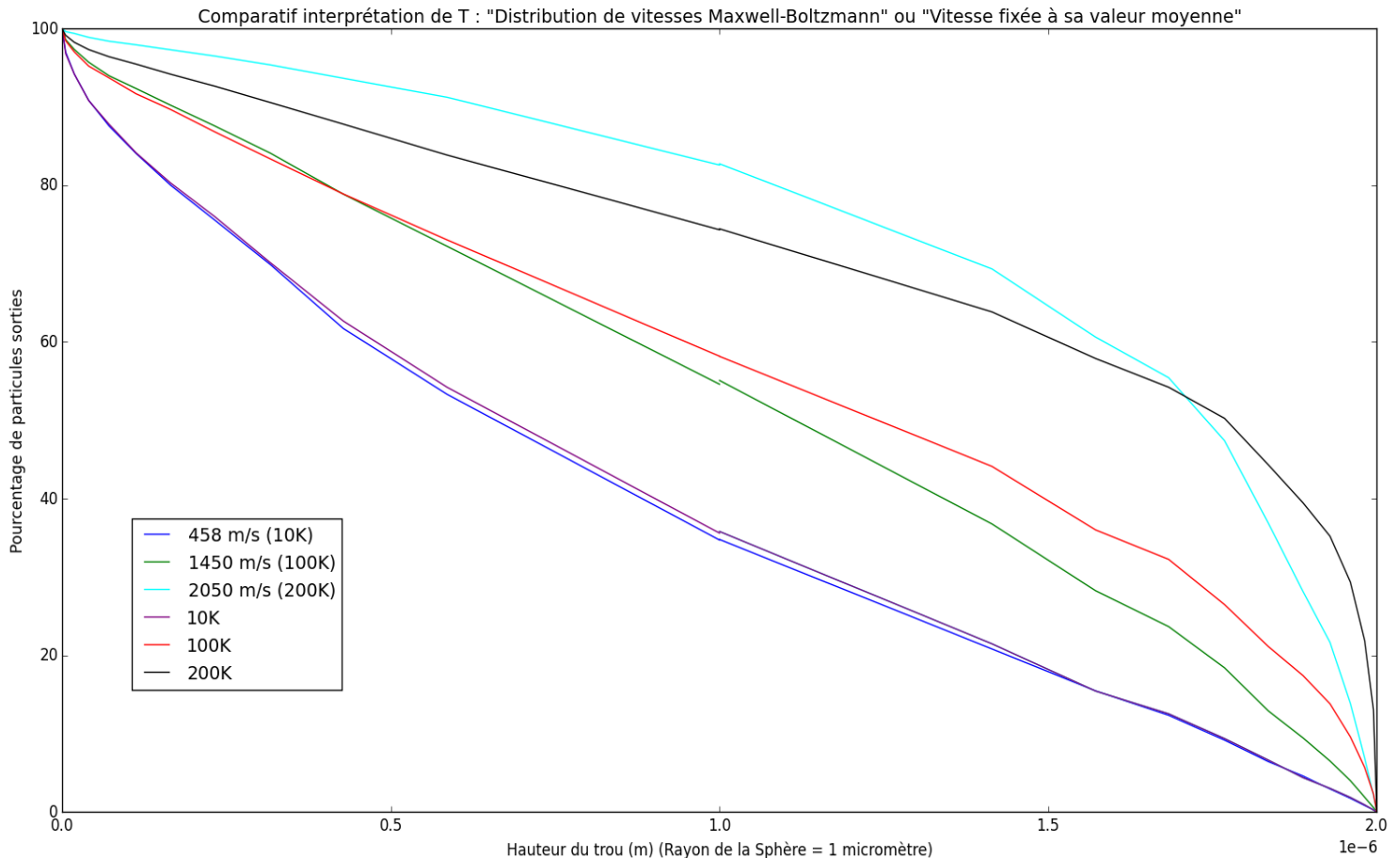
Ci-contre : la trajectoire d'une particule très rapide dans une cavité presque fermée avec l'utilisation du mode « rebond parfait »



Vitesse ou Température ?

Le programme a besoin de la vitesse de la particule pour simuler sa vie dans la cavité et on a vu qu'il peut l'obtenir de deux façon : avec un ordre directement donné en m/s ou un ordre en Kelvins, pour ce dernier il va ainsi attribuer différentes vitesses aux particules de la simulation en respectant la loi de distribution des vitesses de Maxwell-Boltzmann.

Nous allons voir quel est l'influence de ce choix sur le résultat sachant que pour nos simulation, nous avons utilisé un ordre en Kelvins.



Voici un graphe comparant les deux types de caractérisations de la vitesse des particules : (X m/s fixe ou bien une maxwellienne de vitesse moyenne X m/s)

On voit une nouvelle fois que ce choix n'est pas important pour tirer une conclusion qualitative comme celle que nous avons tirée des deux premiers graphes et que les deux modélisations ont des comportements semblables.

Remarque : Par manque d'information, il ne m'a arbitrairement pas paru pertinent d'utiliser les modes du programme introduisant un libre parcours moyen ou une perte d'énergie après contact (rebond inélastique)

Remarque sur les incertitudes : Après avoir réalisé exactement les mêmes simulations, il s'avère que la barre d'erreur numérique est d'environ **2 fois l'épaisseur de la courbe** ce qui rendait superflus le fait de les faire apparaître sur les graphes... (sachant que l'on parle bien d'une barre d'erreur sur la simulation numérique en elle-même et non sur le choix de tous les paramètres et modèles et de l'erreur probablement énorme entraînée par les hypothèses faites).

IV/ Conclusion :

Mon stage s'est terminé sur ces résultats, dont l'interprétation principale est que **lorsque des atomes essaient de s'échapper d'une surface irrégulière de glace, il y a très probablement une très grande différence de réussite selon qu'ils soient légers comme l'Hydrogène ou un peu plus lourds comme l'Oxygène**. Ceci encourage à modifier les modélisations chimiques pour ajouter plus d'Oxygène et pour aller vers l'explication de ce taux de Dioxygène observé par Rosetta s'avérant plus important que les prévisions réalisées.

Ce stage m'a permis de me familiariser un peu plus avec la programmation scientifique, notamment au niveau des erreurs d'arrondis et d'optimisation et aussi à gagner en vitesse et en attention. J'ai également encore plus pris conscience que l'informatique est extrêmement importante dans la recherche scientifique en voyant les chercheurs de l'équipe enchaîner des lignes de codes avec courage et abnégation.

J'ai aussi touché un peu au travail d'équipe puisque deux semaines avant la fin de mon stage un nouveau stagiaire est arrivé (lui aussi de niveau L3 et pour 6 semaines) et la mission qui lui a été donnée est de continuer, à partir des mes résultats et de mon programme, à travailler un peu plus dans cette voie notamment sur l'aspect chimique. Je lui ai donc transmis mon maigre savoir sur ce que j'ai l'impression d'avoir compris de notre problème mais je lui ai surtout appris à utiliser mon programme (ce qui m'a évité d'avoir à commenter 1600 lignes de code) !

J'ai beaucoup apprécié le contexte du stage. Mes maîtres de stage, mon collègue stagiaire, les membres de l'IASB ainsi que tous les Belges sont très sympathiques et détendus ! Bien qu'on puisse s'y attendre vu la proximité, j'ai été surpris par l'important nombre d'expatriés français en Belgique notamment au sein du laboratoire.

C'est donc ravi et le ventre gonflé (de menus frites-gaufres-bières) que j'ai terminé cette petite excursion Belge, avec en tête une très bonne image de la recherche dans ce pays.

V/ Bibliographie :

Articles de recherche :

-GAS TEMPERATURE DEPENDENT STICKING OF HYDROGEN ON COLD AMORPHOUS WATER ICE SURFACES OF INTERSTELLAR INTEREST

E. Matar H. Bergeron F. Dulieu H. Chaabouni M. Accolla J.L. Lemaire :

-RADIOLYSIS OF AMMONIA-CONTAINING ICES BY ENERGETIC, HEAVY AND HIGHLY CHARGED IONS INSIDE DENSE ASTROPHYSICAL ENVIRONMENTS

S. Pilling, E. Seperuelo Duarte, E. F. da Silveira, E. Balanzat, H. Rothard, A. Domaracka, and P. Boduch

-THE STICKING PROBABILITY OF ATOMIC HYDROGEN ON AMORPHOUS WATER ICE

Vijay K. Veeraghattam, Katie Manrodt, Steven P. Lewis, and P. C. Stancil

-IRRADIATION OF ASTROPHYSICAL ICE GRAINS BY COSMIC-RAY IONS: A REAXSIMULATION STUDY

Martin Mainitz, Christian Anders, and Herbert M. Urbassek

-STICKING OF MOLECULES ON NON-POROUS AMORPHOUS WATER ICE

Jiao He, Kinsuk Acharyya, & Gianfranco Vidali

-STICKING OF CO TO CRYSTALLINE AND AMORPHOUS ICE SURFACES

Al-Halabi, A.; van Dishoeck, E. F.; Kroes, G. J.

-SIMULATION OF THE FORMATION AND MORPHOLOGY OF ICE MANTLES ON INTERSTELLAR GRAINS H. M.

Cuppen and Eric Herbst

VI/ Annexe :

Le Guide d'utilisation

Stage IASB

SIMULATION

[1]Type de Géométrie
(0 pour la Sphère;1 pour le Cylindre)

[1]Rayon du trou
vers l'extérieur
(en micromètres)

[350]Température des
particules (K)

[0]Particules
arrivent-elles par le trou?

[1]Masse des particules
(en masse de proton)

[1]Coefficient de perte
de vitesse après un
choc (1 si vitesse conservée)

[Proba de Bergeron pour p(v,M)]
Probabilité de se coller

[0]Tracer la(es) trajectoire(s)?

[0]Option de Géométrie activée?
(Double Trou pour le Cylindre
Hauteur trou<0 pour la Sphère)

[2]Rayon de la Sphère ou
Hauteur du Cylindre (en micromètres)

[10]Température de la
surface de la cavité(K)

[0]Type de zone d'apparition
uniforme :
0 =Surfacique, 1=Volumique?

[random] écart type de
l'aléatoire du rebond (0 pour miroir)

[0]libre parcours
moyen, 0 si infini(en micromètre)

[1]Nombre de
particules à simuler

[0]Faut-il sauvegarder les
trajectoires?

[temporaire]Nom à donner à
la sauvegarde(sans l'extension)

[T]Paramètre à faire varier
parmi r,R,s,T,m,Ts,l

[710]à

[2]Nombre de simulations à
effectuer à chaque pallier

FAIRE VARIER UN PARAMÈTRE :

[0]Faire Varier un paramètre?

[10]de

[100]par pas de

GRAPHE

[Nom du fichier]Titre Graphe

[temporaire]nom du fichier
à représenter (sans l'extension)

[3]Que tracer (entrer 0,1,2ou3) ?
rien, 1,2,3:moyennes des simulations
Graphe 2D:
1>temps,2> contacts,3> %sorties
Graphe 3D : 0> Tracer des trajectoires

[Paramètre] Légende abscisses

[Aucune]Légende de la courbe

Temps de résidence dans la cavité moyen :
0.0004989874637208939 secondes
Nombre de contacts moyen :
0.94366 contacts
Part des particules sorties :
5.722%

éditer
abscisse
et titre

(Python3 (interface Tk/Tcl) + Matplotlib et Numpy)

Sommaire du Guide :

I/ Organisation de l'interface

II/ Types d'actions possibles

III/ Détails des entrées

I/ Organisation de l'interface

L'interface comporte **deux parties** :

→ PARTIE SIMULATION : Les deux premières colonnes d'entrées sont consacrées aux paramètres de simulation et le bouton en bas à gauche *lancera l'expérience désirée et une sauvegarde de cette dernière* sous forme de datas brutes+une visualisation facultative de l'expérience.

→ PARTIE GRAPHE : Les 5 entrées en bas à droite servent à *tracer des courbes et à visualiser des résultats* à partir des fichiers de datas générés à l'aide de la partie gauche. Le tracer est lançable à l'aide du bouton en bas à droite.

Note : Ces deux parties de l'interface sont *indépendantes*.

II/ Type d'actions possibles

Le programme permet de simuler les vies de particules à l'intérieur d'une cavité.

Nous allons clarifier la notion de « *vie d'une particule dans la cavité* » :

Vie d'une particule :

Apparition :

- Apparition aléatoire (uniforme) sur la surface intérieure de la cavité
- ou-Apparition aléatoire (uniforme) dans le volume de la cavité
- ou-Entrée par le trou ou un des trous (position initiale aléatoire et uniforme sur la surface du trou)

Phase de rebonds :

La vitesse initiale de la particule a une direction initiale aléatoire(uniforme) et une norme tirées dans une distribution de Maxwell-Boltzmann correspondant à la température entrée.

Le programme va calculer la trajectoire de la particule jusqu'à ce qu'elle :

-**Touche la paroi de la cavité** : Sa nouvelle vitesse sera soit de même norme, soit de norme inférieure si l'on applique l'option de perte d'énergie, et sa nouvelle direction sera calculé en prenant pour normale de contact la normale théorique liée à la géométrie de la cavité (exemple : orientée vers le centre si l'on a une cavité sphérique) mais affecté d'une modification aléatoire à partir d'une distribution de probabilité gaussienne d'écart-type paramétrable.

-**Entre en collision avec une autre particule** : Ceci a lieu seulement si le paramètre dédié a été activé : la particule subira des changements de vitesse (nouvelle direction uniformément aléatoire et norme remplacée par une nouvelle norme piochée dans la distribution de Maxwell-Boltzmann) , changements générés à partir d'une densité de probabilité associé au libre parcours moyen paramétrable lpm : $\text{Proba}(\text{choc entre } x \text{ et } x+dx) = (1/lpm) * \exp(-x/lpm) * dx$

Fin de la Vie:

La fin de la simulation a lieu lorsque **la particule se colle** sur une paroi (selon la probabilité définie dans la fonction P à modifier dans le code.) **ou qu'elle sort par le trou.**

Types de simulations réalisables à partir de la partie Simulation:

→ 1) La simulation la plus basique consiste à simuler la vie *d'une seule* particule.

Dans ce cas le programme va enregistrer **la durée de la vie de la particule** simulée, le **nombre de rebonds effectués** (paroi et chocs avec autres particules) ainsi qu'un **entier valant 0 si la particule s'est collée et 1 si elle est sortie**. De plus le programme peut facultativement enregistrer et/ou tracer cette trajectoire si désiré

→ 2) Le deuxième type de simulation est la simulation d'un nombre N de vies de particules à partir de *paramètres identiques*.

On peut si on le désire tracer les trajectoires des particules directement et/ou les sauvegarder.

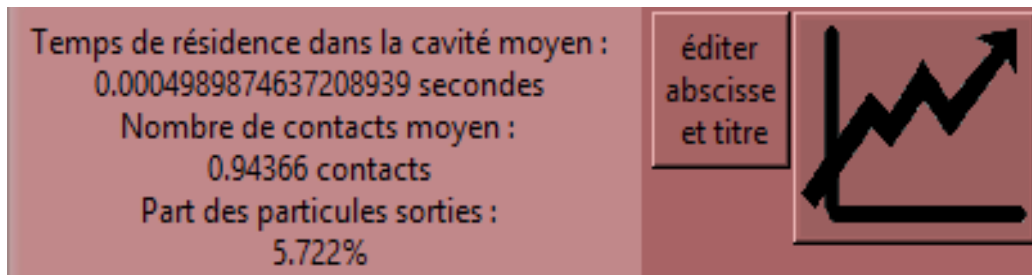
→ 3) Le dernier type de simulation est la simulation de N vies de particules à partir de paramètres identiques puis de *faire varier l'un des paramètres* et de simuler encore N vies etc...

On peut si on le désire sauvegarder les positions des particules mais pas les tracer directement. (Il faut utiliser la partie Graphe de l'interface par la suite pour cela)

Types de résultats sortables à partir de la partie Graphe:

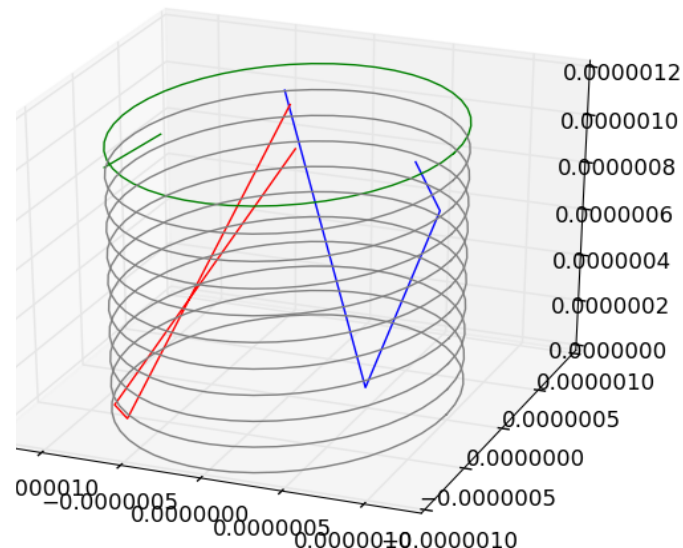
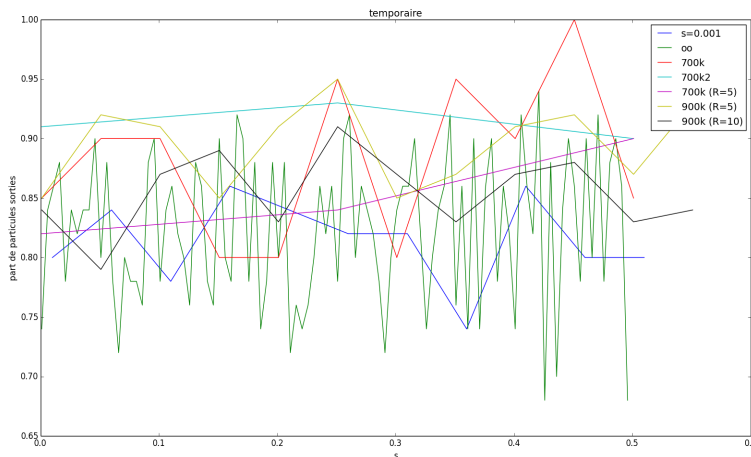
→ 1) Le résultat basique est la moyenne des datas sorties à l'aide d'une simulation de type 2).

Cette liste de moyennes s'affiche à gauche du bouton Graphe :



→ 2) Le deuxième résultat est la visualisation de ces mêmes moyennes mais réalisées sur les vies de particules appartenant à un même *palier de simulation* d'une simulation de type 3):
On a donc un graphe en 2D généré par les fonctions de la bibliothèque Matplotlib qui nous permettent également de sauvegarder ce graphe au format .png)

→ 3) Le troisième résultat est le tracé des trajectoires provenant d'un fichier dans lequel l'enregistrement des positions a été activé. On obtient un Graphe en 3D (on peut le faire tourner) également sauvable sous forme d'une capture au format .png



Organisation des sorties de Datas :

```

legende temporaire natureQ 1 mode2 1 sigma 0.3 r 4.9999 ...(liste écourtée
TempsVie(s) ; Nb contacts ; SortieParTrou?
1e-06 5.166132793477091e-10 0 1
1e-06 1.252998185367141e-09 1 1
1e-06 5.554135373070392e-10 0 1
1e-06 1.5880034424668405e-09 1 1
1e-06 1.2105429189673022e-09 1 1
1e-06 4.404740939337954e-10 0 1
1e-06 1.567818975850679e-09 0 1
1e-06 2.4764880883465263e-09 1 1
1e-06 5.56960965558355e-10 1 0
1e-06 4.843752337304791e-10 0 1
1e-06 2.78110876612243e-10 0 1
1e-06 1.021020078328304e-09 0 1

```

... ..

La première ligne est une longue ligne récapitulant tout les paramètres de la simulation.

La deuxième ligne rappelle à quoi correspondent, et dans quel ordre, *les trois dernières colonnes* de tous les fichiers de data.

Lorsqu'il y a 4 colonnes, c'est que notre simulation est de type 3) et la première colonne correspond alors à la valeur du paramètre variant. (c'est le cas ici et les valeurs de la première colonne sont les mêmes car la capture est située sur un même *palier de simulation*).

Lorsque l'on choisi de sauvegarder les trajectoires des particules, on a en début de ligne plein de valeurs qui correspondent à la liste des coordonnées de la trajectoire (x puis y puis z) et enfin en bout de ligne nos résultats : (paramètre variant) / temps_de_vie / nb_contacts / sortie_ou_pas.

III/ Détails des entrées

Partie Simulation (ligne par ligne, de gauche à droite) :

Note : La valeur **entre crochets** devant chaque description de champs sur l'interface correspond à la **valeur par défaut** utilisée si le champs n'est pas rempli.

Note2 : Les champs se terminant par un '?' attendent que l'on entre soit 0 soit 1 (questions fermées)

Entrée 1 : Choix du type de géométrie de la cavité entre la géométrie Sphérique et Cylindrique. On entre 0 pour la sphère et 1 pour le cylindre.

Entrée 2 : Choix de l'activation ou non (0 ou 1) de l'option de géométrie: Cette option correspond :

- Cas du cylindre : au fait que le cylindre possède deux sorties : en haut et en bas. Si l'option n'est pas activée, le trou est unique et positionné en haut.

- Cas de la Sphère : La cavité de base sphérique est paramétrée comme le cylindre par deux valeurs (cf Entrée 3 et 4). Dans son cas c'est le rayon du trou et le rayon de la sphère. Cependant avec ces deux paramètres, on peut imaginer deux configurations correspondant à une cavité de volume supérieur à une demi sphère de rayon R et un cavité de volume inférieur à cette même demi sphère. On choisit comme origine le centre de la sphère. Le trou est orthogonal à l'axe des z vertical : cette option, si elle est activée, revient à dire que la position selon z du trou est négative et qu'elle est positive si l'option est désactivée.

Entrée 3 : On choisit ici le rayon du trou reliant l'intérieur de la cavité et l'extérieur. Il faut entrer une valeur qui sera ensuite interprétée comme étant en micromètres.

Entrée 4 : On choisit ici la valeur du second paramètre dimensionnel :

- Cas de la Sphère : C'est son rayon R

- Cas du Cylindre : C'est sa hauteur L

Il faut entrer une valeur qui sera ensuite interprétée comme étant en micromètres.

Entrée 5 : Choix de la valeur en Kelvins de la température de(s) particule(s). Si l'on désire une vitesse spécifique (au lieu d'en piocher une dans une distribution de Maxwell-Boltzmann associée à une température) alors il faut entrer à la place une vitesse en accolant le caractère 'v' devant : Exemple, pour une directive en vitesse de 400 m/s, entrer : « v400 »

On peut également entrer une vitesse en ajoutant 'm' devant la valeur d'une température T: cela aura pour effet de donner à la simulation une directive en vitesse correspondant à la vitesse moyenne d'une particule dans un gaz à T.

Exemple : entrer « m100 » revient à entrer « v1450 » pour un atome d'hydrogène.

Entrée 6 : Choix de la valeur en Kelvins de la température de la surface de la cavité.

Entrée 7 : Choix de la provenance des particules : 0 si elles viennent de l'intérieur et 1 si elles arrivent par le trou.

Entrée 8 : Choix (si l'on a choisi une provenance intérieur en Entrée 7) du type d'apparition uniformément aléatoire à l'intérieur: Soit sur la surface de la paroi (entrer 0), soit dans le volume de la cavité (entrer 1)

Entrée 9 : Choix de la masse des particules. La valeur entrée est ensuite multipliée par la masse d'un proton. On doit ainsi entrer 1 pour de l'H, 2 pour du D, 16 pour de l'O etc...

Entrée 10 : On choisit à quel point le rebond sera aléatoire en paramétrant l'écart-type de la densité de probabilité gaussienne utilisée : comme cette modification est apportée aux composantes d'un vecteur unitaire, il s'en suit que pour $\sigma > 0,8$ le rebond devient très très aléatoire...

Entrée 11 : Définit la part de sa vitesse que la particule conserve après son choc sur une paroi : entrer 1 pour une vitesse conservée et 0,5 si la particule perd la moitié de sa vitesse à chaque choc par exemple. La Température est également mise à jour par le programme (elle est multipliée par le carré du coefficient comme la température est proportionnelle à l'Energie Cinétique, elle même proportionnelle au carré de la vitesse)

Entrée 12 : Choix de la valeur du libre parcours moyen entre deux chocs entre particules dans la cavité. Pour désactiver les chocs entre particules il suffit d'entrer 0.

Entrée 13 : Ce champs correspond à une liste de doubles (réels de 8 octets) séparés par un espace qui seront utilisables dans le code par la variable globale 'sticklist' de taille arbitraire et dont les éléments pourront entrer dans la formule délivrée par la fonction P (probabilité de se coller).

Entrée 14 : Simulation de type 2) seulement : Nombre de simulations à effectuer avec les paramètres constants.

Entrée 15 : Sauf pour simulation de type 3) : Active ou pas l'option d'affichage en fin de simulation des trajectoires des particules en 3D (entrer 0 ou 1)

Entrée 16 : Active ou pas l'option de sauvegarde des trajectoires des particules (entrer 0 ou 1)
Entrée 17 : Choix du nom que l'on désire donner au fichier .txt de datas (nom à entrer sans l'extension)
Entrée 18 : Entrer 1 si l'on désire une simulation de type 3) ou 0 si l'on désire une simulation de type 1) ou 2)
Entrée 19 : Seulement pour simulation de type 3) : Choix du paramètre à faire varier parmi :
 -entrer r : faire varier l'Entrée 3)
 -entrer R : faire varier l'Entrée 4)
 -entrer s : faire varier l'Entrée 10)
 -entrer T : faire varier l'Entrée 5) (en terme de température)
 -entrer v : fait varier l'Entrée 5) (en terme de vitesse)
 -entrer m : faire varier l'Entrée 9)
 -entrer Ts : faire varier l'Entrée 6)
 -entrer l : faire varier l'Entrée 12)
Entrée 20 : Seulement pour simulation de type 3) : Valeur initiale du paramètre variant (on tient compte des unités définies dans l'Entrée correspondant au paramètre (exemple si le paramètre est r, on entre les valeurs en micromètres comme précisé dans l'Entrée correspondante)
Entrée 21 : Seulement pour simulation de type 3) : Valeur *Maximale* du paramètre variant (selon le pas choisi, cette valeur sera ou non atteinte)
Entrée 22 : Seulement pour simulation de type 3) : Choix du pas d'évolution du paramètre variant entre chaque *palier de simulation*.
Entrée 23 : Seulement pour simulation de type 3) : Choix du nombre de simulations à effectuer pour chaque valeur prise par le paramètre variant.

Partie Graphe :

Boutons : -Bouton Graphe (avec l'image d'un graphe) : Lance l'affichage

-Bouton *'éditer abscisse et titre'* : Permet de modifier la légende des abscisses ainsi que le titre du Graphe 2D ouvert.

Entrée 24 : Si graphe 2D : Choix du titre du graphe.

Entrée 25 : Choix du nom du fichier dont on souhaite utiliser les données (à entrer sans l'extension '.txt')

Entrée 26 : Choix du contenu que l'on souhaite observer :

-Si les datas sont issues d'une simulation avec sauvegarde des trajectoire (type 1), 2) ou 3) acceptés) : entrer 0 pour les visualiser en 3D.
 -Si datas issues d'une simulation de type 1) ou 2) :
 -laisser vide : Moyennes des 3 champs temps de vie, nb de contacts et sortie ou pas ou juste valeur (affichée à gauche du bouton Graphe)
 -Si datas issues d'une simulation de type 3) :
 -entrer 1 pour visualiser le temps de vie moyen des particules en fonction du paramètre variant de la simulation.
 -entrer 2 pour visualiser le nombre de contacts (choc de particules + choc avec paroi) moyen des particules en fonction du paramètre variant de la simulation.
 -entrer 3 pour visualiser la part moyenne des particules étant sorties en fonction du paramètre variant de la simulation.

De plus, s'il s'agit de tracer quelque chose en fonction du rayon du trou et si l'on se trouve dans le cas de la sphère, on peut tracer en abscisse la distance entre le trou et le fond de la cavité à la place du rayon du trou en ajoutant le caractère 'h' (pour hauteur) devant la valeur souhaitée:

Exemple : entrer 'h3' va tracer le % de sortie en fonction de la hauteur du trou !

Entrée 27 : Indiquez si on doit changer la couleur de la prochaine courbe (deux courbes de la même couleur auront la même légende et elle sera la légende données à la première tracée !)

Entrée 28 : Choix de la Légende à afficher pour les abscisses. La valeur par défaut : la valeur de l'Entrée 19)

Entrée 29 : Note : Si l'on trace un graphe 2D alors qu'un autre est déjà ouvert dans une fenêtre, le tracé a lieu sur la même figure. (ce qui donne d'ailleurs lieu à un bug lorsque l'on trace un graphe 2D sans fermer les graphes 3D précédemment ouverts)

Cette entrée permet de choisir la légende de la courbe que l'on va tracer (sur un graphe déjà existant comme sur un deuxième).

Note2 : Si l'on trace sur un graphe déjà ouvert une nouvelle courbe, le titre et la légende d'abscisse sont mis à jour par rapport à leur nouvelle valeur.

Note3 : La légende en ordonnée est automatique et peut être modifiée dans le code dans la fonction grapheur().