

Rapport de projet Master 1

Sujet : Simulation numérique d'un modèle de gaz bidimensionnel

Encadrant :
M. Fabre David

Etudiants :
Diop Birame
Ciss El Hadji Abdoulaye

Année académique : 2016/2017

Remerciements

Nous saluons l'engagement de Monsieur David Fabre qui a su adhérer à l'élaboration de ce présent rapport en ne ménageant aucun effort pour nous apporter son soutien intellectuel. Nous louons également son dynamisme, sa collaboration a été essentielle depuis le début de notre exercice. Nous lui réitérons notre confiance et l'encourageons. Nous adressons aussi particulièrement des remerciements à Monsieur Etienne Mangaud, doctorant du laboratoire IRFMAC.

Sommaire

Avant-Propos

Introduction Générale

Première Partie : Étude théorique

1-Théorie de la Cinétique des gaz parfaits de Maxwell

1.1- Modèle du gaz parfait

1.2- Lois statistiques du chaos moléculaire

2- Distribution maxwellienne des vitesses

- Distribution gaussienne des vitesses

3 -Gaz bidimensionnel

3.1- Description du système simulé

3.2- Gaz parfait sans collisions mutuelle

3.3- Gaz parfait avec collisions entre particule

4- Modèle de collision du gaz étudié

Deuxième Partie : Simulation numérique

1- Kinetics

1.1- La position instantanée des particules

1.2- Histogramme de la position instantanée des particules

Histogramme de la vitesse verticale et horizontale des particules

Température

1.3- Courbe d'évolution de la force de pression instantanée

1.4- Température normalisée à 1

2 -Construction de la boucle d'itérations et de tests pour obtenir la pression optimale

2.1- Itérations

2.2- Test

3 -Construction de la Pression en fonction des paramètres N, T et d

3.1- Construction de la boucle du nombre de particules N

3.2- Construction de la boucle de température T

3.3- Construction de la boucle du diamètre d

Conclusion

Annexes

Avant Propos

L'Université Paul Sabatier connue de tous par son prestige, forme des ingénieurs dans diverses filières qui sont potentielles dans le marché du travail et de la recherche.

C'est ainsi qu'un projet de type recherche couronne la fin de la première année de Master.

Le projet s'inscrit également dans la perspective d'une confrontation des connaissances théoriques acquises et des réalités pratiquées en entreprise ou en laboratoire de recherche.

Ceci permet à l'étudiant de se familiariser avec le milieu du travail et de la recherche car ayant pour vocation de prendre des responsabilités au sein du projet. Dans cet optique, l'étudiant acquiert le maximum d'expérience professionnelle, notamment technique et scientifique, dans l'élaboration d'un travail de recherche.

Il sera mis en évidence l'esprit d'ouverture, le respect mutuel, le souci du travail bien fait en équipe entre autres, participant activement à l'essor effectif du milieu professionnel ou de la recherche.

Un nouvel élan pour l'insertion et source de motivation du savoir.

Introduction

De nos jours, la simulation est une technique largement utilisée en mécanique, dans les laboratoires universitaires ou dans les centres de recherche appliquée. En raison de la puissance de calcul toujours croissante des ordinateurs et des soucis économiques de maîtrise des coûts, elle est devenue un outil d'analyse extraordinaire pour son efficacité et par conséquent indispensable.

Pour cette raison, elle est introduite naturellement dans l'enseignement, notamment en mécanique, physique, car elle offre naturellement un moyen facile d'initier les étudiants à l'analyse de systèmes complexes. En effet, il n'existe pas de méthodes analytiques exactes pour décrire de tels systèmes. Avant les techniques de simulation, le mécanicien ou le physicien réduisait la complexité d'un système en invoquant des hypothèses simplificatrices partiellement justifiées. Bien que nécessaire, cette analyse simplifiée s'avère insuffisante lorsqu'une description minutieuse et sans approximation est exigée. Le recours à la simulation s'impose alors.

La précision n'est pas le seul intérêt de la simulation. La possibilité de changer les conditions du problème, de modifier l'ensemble des paramètres et d'observer le résultat obtenu, apporte un éclairage complémentaire qui contribue à approfondir notre connaissance du système. Plusieurs thèmes thermodynamiques importants sont étudiés par la simulation. Par exemple la simulation d'un modèle de gaz bidimensionnel, qui fait l'objet d'étude de notre projet.

On se propose de simuler un gaz afin de vérifier les résultats de la théorie cinétique des gaz parfaits. Le gaz est assimilé à un ensemble de particules homogènes identiques. Le but est de faire plusieurs essais en faisant varier le diamètre des particules, le nombre de particules, la température dans une optique de mesurer la pression afin de déterminer une loi de la pression en fonction de la température. Ce qui fera intervenir beaucoup plus de paramètres, que l'on détaillera, dans les lignes qui suivent.

1-Théorie de la Cinétique des gaz parfaits de Maxwell

1.1-Modèle du gaz parfait

Le modèle du gaz parfait est défini à partir de trois hypothèses : l'une concernant la géométrie des molécules, la deuxième l'interaction entre molécules et la troisième l'équilibre.

a) Molécules de dimension négligeable

Les molécules d'un gaz parfait sont supposées de dimension négligeable devant la distance moyenne qui les sépare ; il en résulte qu'un gaz réel se comporte comme un gaz parfait lorsqu'il est suffisamment dilué. Dans le cas des gaz monoatomiques, ces molécules peuvent être assimilées à des sphères dures de très faibles diamètres. Lorsque les gaz parfaits sont diatomiques et plus largement polys atomiques, la structure interne des molécules est plus complexe mais de diamètre toujours très faible devant la distance moyenne entre deux molécules.

b) Interaction de courte portée

Les interactions entre molécules sont uniquement de très courte portée, ce qui localise l'interaction, comme dans les collisions. Pour simplifier l'analyse, on suppose en outre que les collisions sont élastiques ; cette dernière hypothèse ne limite en rien la généralité du modèle, car le comportement statistique d'un système résultant d'un grand nombre de collisions entre molécules ne dépend pas de la nature précise de la collision.

c) Chaos moléculaire à l'équilibre

A l'équilibre, les composantes des vecteurs positions et des vecteurs quantités de mouvement des molécules, suivant trois directions orthogonales, sont distribuées au hasard : c'est le chaos moléculaire caractéristique de l'équilibre.

1.2- Lois statistiques du chaos moléculaire

Une fois admis le chaos moléculaire, il reste à préciser la loi statistique qui lui est associée. Pour établir cette loi, Maxwell a proposé trois hypothèses supplémentaires de nature statistique.

a) Uniformité de la répartition des molécules en l'absence de champ extérieur

En l'absence de champ extérieur, il est naturel d'admettre que les molécules se répartissent uniformément dans tout le volume offert ; le nombre de particules par unité de volume est alors, en moyenne, le même en tout point du volume de gaz. On en déduit, pour un nombre de particules N dans un volume V , la concentration suivante :

$$n_v = \frac{N}{V}$$

b) Isotropie des vitesses

L'hypothèse de l'équivalence des différentes directions des vitesses est naturelle aussi. Elle revient à considérer que la vitesse du centre de masse des molécules n'apparaît dans la loi de probabilité que par sa seule norme.

c) Indépendance des composantes des vitesses

La répartition des vitesses du centre de masse des molécules suivant une direction ne dépend que de la vitesse suivant cette direction ; ceci revient à admettre que la loi de probabilité se met sous la forme d'un produit de trois termes analogues, compte tenu de l'isotropie :

$$f_x(V_x^2)f_y(V_y^2)f_z(V_z^2)=f(V_x^2)f(V_y^2)f(V_z^2)$$

2-Distribution maxwellienne des vitesses

Les hypothèses de la théorie cinétique des gaz de Maxwell sont d'une part celles qui définissent le modèle du gaz parfait et ensuite celles de nature statistique qui caractérisent, à l'échelle microscopique, l'état d'équilibre thermodynamique du gaz.

2.1- Distribution gaussienne des vitesses

Les hypothèses précédentes suffisent pour établir rigoureusement le type de fonction $f(V_x)$. En effet, en raison de l'isotropie des vitesses, la quantité suivante, qui est une densité de probabilité P_v , s'écrit

$$\frac{1}{dN} \frac{dN}{dv_x dv_y dv_z} = P_v(v^2) = P_v(v_x^2 + v_y^2 + v_z^2) = f(v_x^2) f(v_y^2) f(v_z^2)$$

En dérivant P_v par rapport à v_x , on trouve:

$$\frac{\partial P_v(v^2)}{\partial v_x} = \frac{dP_v}{dv^2} \times \frac{\partial v^2}{\partial v_x} = 2v_x \frac{dP_v}{dv^2} \quad \text{puisque} \quad \frac{\partial v^2}{\partial v_x} = 2v_x$$

on a aussi

$$\frac{\partial P_v(v^2)}{\partial v_x} = \frac{\partial f(v_y^2) f(v_z^2) f(v_x^2)}{\partial v_x} = 2v_x f(v_y^2) f(v_z^2) \frac{df(v_x^2)}{dv_x^2}$$

On obtient en identifiant les deux expressions précédentes :

$$\frac{dP_v(v^2)}{dv^2} = f(v_y^2) f(v_z^2) \frac{df(v_x^2)}{dv_x^2} \quad \text{soit} \quad \frac{1}{P_v} \frac{dP_v(v^2)}{dv^2} = \frac{1}{f(v_x^2)} \frac{df(v_x^2)}{dv_x^2}$$

En divisant par P_v . De même, pour les autres variables, v_y^2 et v_z^2 :

$$\frac{1}{P_v} \frac{dP_v(v^2)}{dv^2} = \frac{1}{f(v_y^2)} \frac{df(v_y^2)}{dv_y^2} \quad \text{et} \quad \frac{1}{P_v} \frac{dP_v(v^2)}{dv^2} = \frac{1}{f(v_z^2)} \frac{df(v_z^2)}{dv_z^2}$$

Par conséquent, les variables v_x^2, v_y^2, v_z^2 étant indépendantes :

$$\frac{1}{f(v_x^2)} \frac{df(v_x^2)}{dv_x^2} = \frac{1}{f(v_y^2)} \frac{df(v_y^2)}{dv_y^2} = \frac{1}{f(v_z^2)} \frac{df(v_z^2)}{dv_z^2} = \text{cte.}$$

Ces équations s'intègrent facilement : en effet, en désignant par $-B$ la constante précédente, il vient :

$$\frac{df(v_x^2)}{f(v_x^2)} = -B dv_x^2 \quad \text{donne} \quad \ln \left[\frac{f(v_x^2)}{A} \right] = -B v_x^2 \quad \text{d'où}$$

$$\boxed{f(v_x^2) = A \exp(-B v_x^2)}$$

A étant une autre constante. On en déduit la probabilité élémentaire dP_v , c'est à dire le nombre relatif dN/N de particules qui, en moyenne, ont un vecteur vitesse compris entre v et $v+dv$:

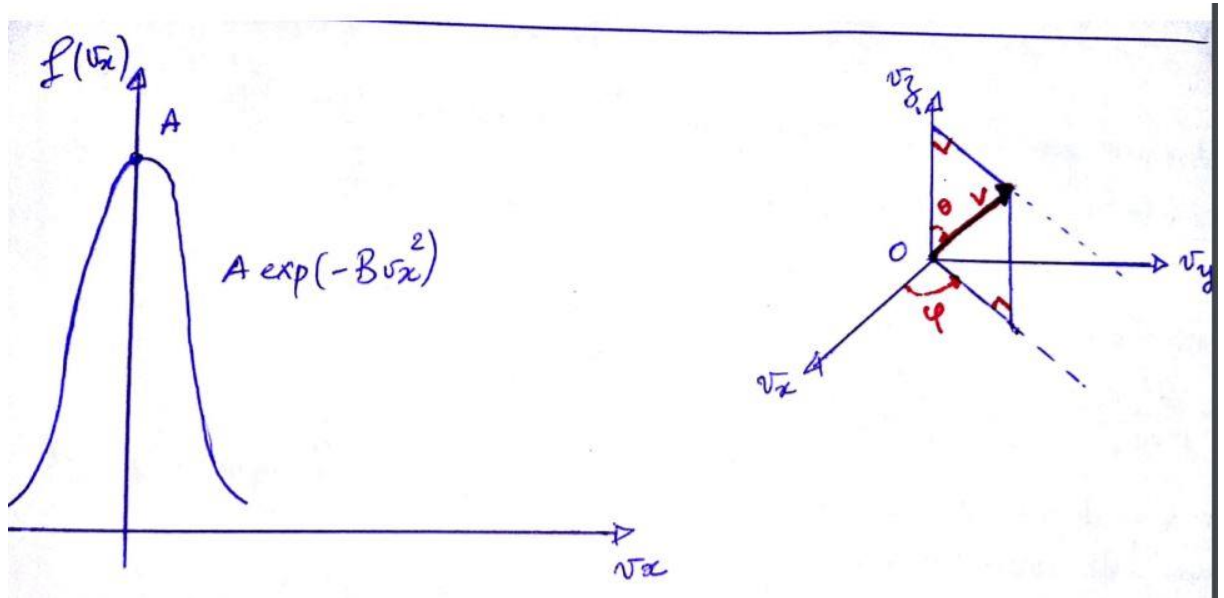
$$\frac{dP_v}{N} = \frac{dN}{N} = A^3 \exp(-B v_x^2) \times \exp(-B v_y^2) \times \exp(-B v_z^2) dv_x dv_y dv_z = A^3 \exp(-B v^2) dv_x dv_y dv_z$$

$$A \text{ et } B \text{ sont reliées par : } \int dP_v = \int \frac{dN}{N} = \int A^3 \exp(-B v^2) dv_x dv_y dv_z = 1.$$

$B > 0$ donc on trouve une loi statistique du type Gaussien.

Une analyse intuitive simple, s'appuyant sur le nombre considérable de collisions que subit une molécule, rend ce résultat prévisible. En effet, le phénomène décrit peut être considéré comme l'issue d'une expérience qui résulte d'un grand nombre d'épreuves (les collisions). Si l'on

s'intéresse à la variable aléatoire qu'est la vitesse V d'une molécule et à sa valeur après un grand nombre de collisions de même nature, on montre que la loi de probabilité obtenue est une loi de Gauss qui ne dépend que du nombre d'épreuves. Comme, en outre, la valeur moyenne d'une composante de la vitesse suivant un axe est nulle, d'après l'hypothèse moléculaire, la fonction de probabilité est une gaussienne centrée. Cette tendance à la loi de Gauss est connue en probabilité sous le nom du théorème de la limite centrale ou de la tendance normale.



3-Le Gaz Bidimensionnel

3.1-Description du système simulé

Considérons un système mécanique plan, comportant N particules de masse m , de diamètre d , enfermés dans une enceinte ABCD de longueur $L = AB = CD$ et de largeur $l = BC = AD$. Comme dans l'étude théorique du gaz parfait, on néglige l'influence de la pesanteur terrestre sur le système et on suppose que les chocs des particules contre les parois sont élastiques. Nous envisageons deux cas :

- Les particules du gaz n'interagissent pas entre elles ; en dehors des collisions sur les parois du récipient, il n'y a pas de collisions qui modifient leurs trajectoires. Cette étude constitue un préalable technique important à l'élaboration progressive de la simulation.
- Les particules interagissent par des collisions élastiques mutuelles.

3.2- Gaz parfait sans collisions mutuelles

Le système subit une succession d'évolutions dites balistiques, interrompues par les collisions. La progression dans le temps se fait en itérant les deux séquences : évolution balistique et collision. Le temporel Δt est variable et vaut à l'instant t :

$$\Delta t = t_p - t$$

Si l'on considère que les collisions contre les parois s'échelonnent régulièrement au cours du temps, on peut estimer alors la valeur moyenne de ce pas temporel, en fonction de la durée moyenne de collision ζ_p , par la quantité : $\langle \Delta t \rangle = \frac{\zeta_p}{N}$

Quant à ζ_p , on calcule en rappelant que dans chaque direction du plan, la vitesse quadratique moyenne des particules est égale, d'après le théorème de l'équipartition de l'énergie :

$$\langle V_x^2 \rangle = \langle V_y^2 \rangle = \frac{\langle V_q^2 \rangle}{2} \quad \text{avec} \quad \langle V_q^2 \rangle = \langle V_x^2 \rangle + \langle V_y^2 \rangle$$

L'enceinte est rectangulaire, avec un rapport de la longueur L sur la largeur l . Lorsque les particules parcourent, suivant la direction Ox , la distance l , elles subissent une collision sur les parois BC et AD . Lorsqu'elles parcourent, suivant la direction Oy , la même distance, elles subissent deux collisions sur les parois AB et CD . La durée $\zeta_p = \frac{L}{V_q}$, appelée durée dynamique, représente la durée mise par une particule pour parcourir une longueur caractéristique du système, ici L .

3.3- Gaz parfait avec collisions entre particules

a) Etat initial

A l'instant initial, les positions et les vitesses des centres d'inertie des disques qui représentent les particules, sont tirées au hasard. Les particules sont réparties uniformément dans l'enceinte et les vecteurs vitesses, de normes identiques, pointent de façon isotrope dans toutes direction du plan.

b) Méthode numérique

Le système subit une succession d'évolutions balistiques interrompues par des collisions. On le fait évoluer dans le temps en itérant les séquences balistiques et les résolutions des collisions selon leur nature : paroi-particule ou particule-particule. Evidemment, apparaît dans le système l'échelle de temps associée à la durée moyenne ζ_c de collision entre les particules. Par analogie au modèle des sphères dures on a :

$$n_s 4R v_{r,m} \zeta_c = 1$$

$n_s = \frac{N}{S}$ désigne le nombre de particules par unité de surface et $v_{r,m} = \sqrt{2} v_m$ la vitesse moyenne entre deux particules, v_m étant la valeur moyenne de la norme de la vitesse des particules.

Etablissons la relation entre la vitesse moyenne v_m et la vitesse quadratique v_q pour un gaz bidimensionnel. On a :

Etablissons une relation entre la vitesse moyenne et la vitesse de quadrature :

$$v_q^2 = \int_0^\infty v^2 dP = \int_0^\infty v^2 A \exp(-Bv^2) 2\pi v dv = \frac{1}{B}$$

$$v_m = \int_0^\infty v dP = \int_0^\infty v A \exp(-Bv^2) 2\pi v dv = \frac{1}{2} \sqrt{\frac{\pi}{B}}$$

donc $\frac{v_m}{v_q} = \frac{\frac{1}{2} \sqrt{\frac{\pi}{B}}}{\sqrt{\frac{1}{B}}} = \frac{\sqrt{\pi}}{2} \Rightarrow v_m = \frac{\sqrt{\pi}}{2} v_q$

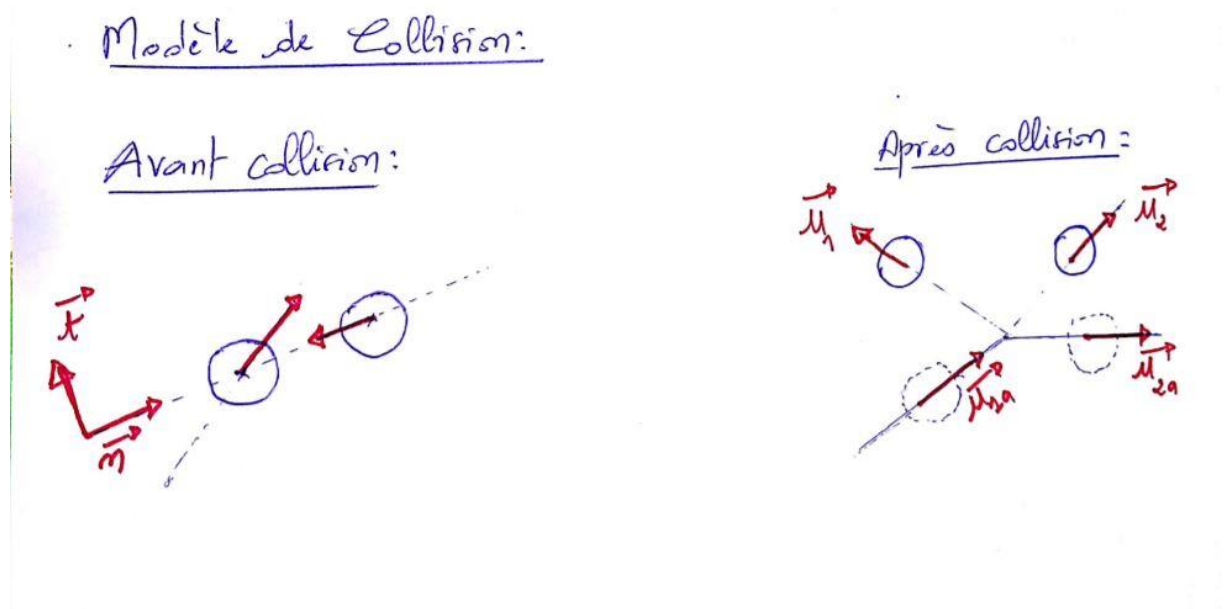
$v_m \approx 0,89 v_q$

Etablissons l'expression du libre parcours moyen l_p des particules. Rappelons que c'est la moyenne de la distance parcourue par une particule, entre deux collisions successives :

$$\zeta_p v_m = \zeta_p v_q = l_p$$

4-Modèle de collision du gaz étudié

Pour notre gaz nous avons un modèle de collision comme suit : (insérer scan)



L'Objectif ici étant de calculer \vec{u}_1 et \vec{u}_2 en fonction de \vec{u}_{1a} et \vec{u}_{2a} (U_1, V_1, U_2, V_2).

Donc on remarque que nous avons besoin de 4 équations car nous avons 4 inconnues. Pour cela, nous allons utiliser la conservation de la quantité de mouvement globale pour déterminer les deux premières équations

$m(\vec{u}_1 + \vec{u}_2) = m(\vec{u}_{1a} + \vec{u}_{2a})$ ce qui implique :

$$U_1 + U_2 = U_{1a} + U_{2a} \quad (1)$$

$$V_1 + V_2 = V_{1a} + V_{2a} \quad (2)$$

Ensuite on applique la conservation de l'énergie Cinétique ; Ce qui nous donne :

$$\frac{1}{2} m (\vec{u}_1^2 + \vec{u}_2^2) = \frac{1}{2} m (\vec{u}_{1a}^2 + \vec{u}_{2a}^2) \text{ donc on a :}$$

$$U_1^2 + V_1^2 + U_2^2 + V_2^2 = U_{1a}^2 + U_{2a}^2 + V_{1a}^2 + V_{2a}^2 \quad (3)$$

Enfin on ajoute une équation supplémentaire, équation de conservation de la composante tangentielle de la vitesse relative :

$$\left(\frac{\rightarrow}{U_1} \frac{\rightarrow}{U_2} \right) \cdot \frac{\rightarrow}{t} = \left(\frac{\rightarrow}{U_{1a}} \frac{\rightarrow}{U_{2a}} \right) \cdot \frac{\rightarrow}{t} \quad (4)$$

Après les calculs on trouve :

$$\begin{aligned} &> EQ1 := U1 + U2 = U1a + U2a; \\ &\quad EQ1 := U1 + U2 = U1a + U2a \quad (1) \\ &= \\ &> EQ2 := V1 + V2 = V1a + V2a; \\ &\quad EQ2 := V1 + V2 = V1a + V2a \quad (2) \\ &= \\ &> EQ3 := U1^2 + U2^2 + V1^2 + V2^2 = U1a^2 + U2a^2 + V1a^2 + V2a^2 \\ &\quad EQ3 := U1^2 + U2^2 + V1^2 + V2^2 = U1a^2 + U2a^2 + V1a^2 + V2a^2 \quad (3) \\ &= \\ &> EQ4 := (U1 - U2) \cdot tx + (V1 - V2) \cdot ty = (U1a - U2a) \cdot tx + (V1a - V2a) \cdot ty; \\ &\quad EQ4 := (U1 - U2) tx + (V1 - V2) ty = (U1a - U2a) tx + (V1a - V2a) ty \quad (4) \\ &= \\ &> solve(\{EQ1, EQ2, EQ3, EQ4\}, \{U1, U2, V1, V2\}); \\ &\quad \{U1 = U1a, U2 = U2a, V1 = V1a, V2 = V2a\}, \left\{ U1 \right. \\ &\quad = \frac{U1a tx^2 + U2a ty^2 + V1a tx ty - V2a tx ty}{tx^2 + ty^2}, U2 \\ &\quad = \frac{U1a ty^2 + U2a tx^2 - V1a tx ty + V2a tx ty}{tx^2 + ty^2}, V1 \\ &\quad = \frac{U1a tx ty - U2a tx ty + V1a ty^2 + V2a tx^2}{tx^2 + ty^2}, V2 = \\ &\quad \left. - \frac{U1a tx ty - U2a tx ty - V1a tx^2 - V2a ty^2}{tx^2 + ty^2} \right\} \\ &> \end{aligned}$$

Deuxième Partie : Simulation numérique

1-Kinetics :

Kinetic.m est un programme qui calcule la valeur de la Pression P par rapport aux paramètres de contrôle du problème numérique :

- * T (la Température),
- * N (le nombre de Particules),
- * d (le Diamètre des Particules).

Il résout l'équation des gaz, suivant le modèle mathématique proposé ci-après :

$$P = N * T / (V - Nb)$$

b=d² volume de la particule,
V=1 volume unité.

L'exécution du programme nous permet de visualiser les résultats suivants :

1-1. La position instantanée des particules

Elles se déplacent dans toutes les directions et viennent impacter les parois de la structure. Elles effectuent aussi des collisions entre elles et des rebonds élastiques.

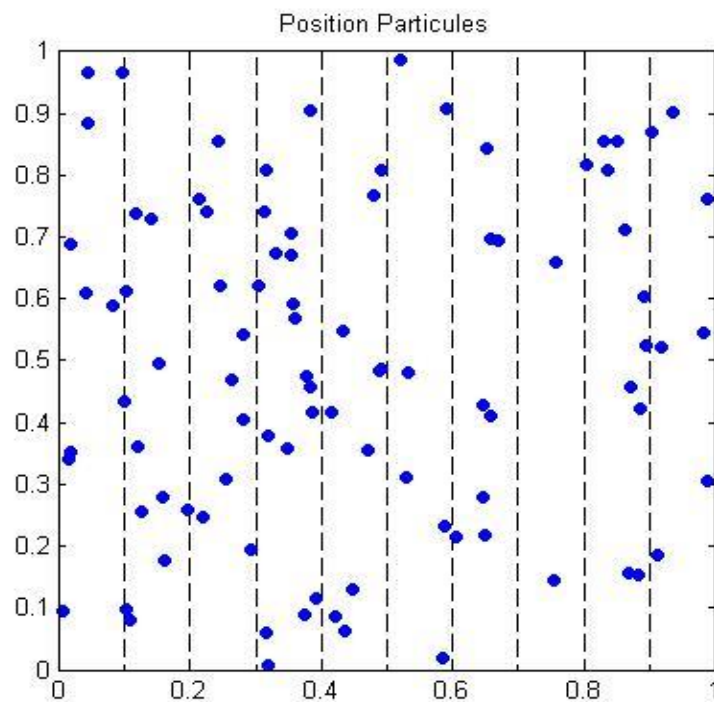


Schéma 1 : position aléatoire des particules

1-2. Histogramme de la position instantanée des particules -Histogramme de la vitesse verticale et horizontale des particules -Température

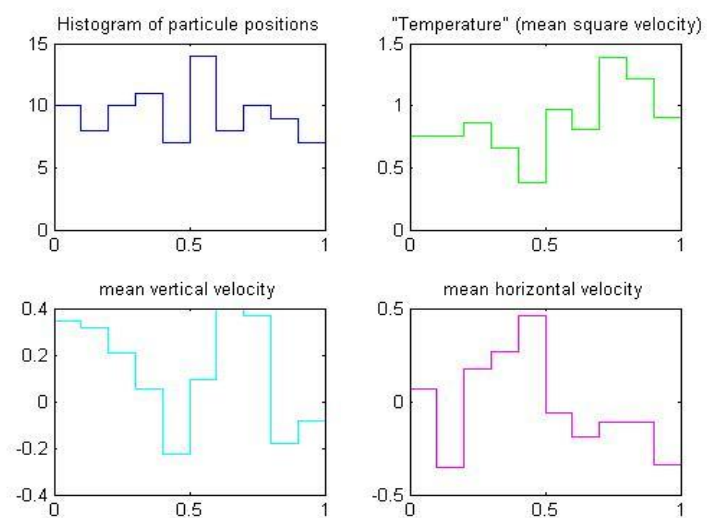


Schéma 2 : histogramme positions et vitesses et température

1-3. Courbe d'évolution de la force de pression instantané -température normalisée à 1

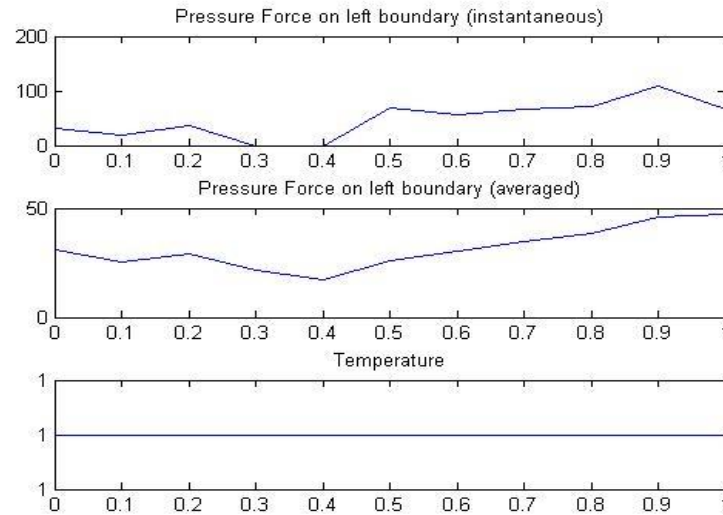


Schéma 3 : pression instantanée et température

Ce programme calcul la **pression moyenne P** obtenue au bout de **10 itérations** sur la position initiale des particules.

Cette position des particules est redistribuée à chaque itération. La configuration du gaz (Position initiale, Vitesse initiale), change après chaque itération.

En effet, la fonction **rand** crée une matrice dont les éléments sont uniformément répartis entre 0 et 1, en l'occurrence, position et vitesse initiale. Aucune région de l'espace n'est privilégiée.

Confère « *annexe 1 Programme kinetics* »

```

38  %% Initialisations : initial positions and velocities of particles
39  Nparticules = Nparticules_1;
40  vq1 = sqrt(T1);
41  X(1:Nparticules_1) = Xlmin+(Xlmax-Xlmin)*rand(1,Nparticules_1);
42  Y(1:Nparticules_1) = rand(1,Nparticules_1);
43  U(1:Nparticules_1) = Um1+vq1/sqrt(2)*randn(1,Nparticules_1);
44  V(1:Nparticules_1) = Vm1+vq1/sqrt(2)*randn(1,Nparticules_1);

```

2 -Construction de la boucle d'itérations et de tests pour obtenir la pression optimale

2-1. Itérations

% Tous les paramètres fixés

```
itermax = 10;
```

```
for iter=1:itermax ;
```

```
    [ ...]
```

```
end
```

```
disp(' iteration numero ');
```

```
iter
```

```
disp('temperature :');
```

```
Temp
```

```
disp('Pression :');
```

```
P;
```

```
Tableau_P(iter) = P;
```

```
end
```

```
Tableau_P
```

```
mean(Tableau_P)
```

Cette suite d'instruction vise à calculer le Tableau de pression après 10 itérations et d'en déduire la pression moyenne : mean(Tableau_P).

La construction de cette boucle d'itération est le point de départ de notre simulation numérique. Elle nous a permis de comprendre les préoccupations du projet qui est de comparer un modèle de gaz réel et celui du parfait.

2-2. Test

```
testmax=10;
```

```
for test = 1:testmax;
```

```
    itermax = 10;
```

```
    for iter=1:itermax
```

```
        [...]
```

```
    end
```

```
end
```

```
%%calcul de la pression moyenne pour 10 réalisations
```

```
k=mean(Tableau_P);
```

```
k_tab=[];
```

```
k_tab = [k_tab k];
```

```
disp('test numero ');
```

```
test;
```

```
disp('mean(Tableau_P): ');
```

```
k;  
Tableau_k(test) = k;  
end  
Tableau_k  
mean(Tableau_k)
```

Cette boucle nous donne la pression moyenne P au bout de 100 réalisations. Elle est de l'ordre de **46** pour (**N=100, T=1, d=0,005**).

Confère «*Annexe 2 : Pression moyenne après 100 réalisations*»

3-Construction de la Pression en fonction des paramètres N, T et d

Dans cette section nous expliquerons comment on a construit les boucles effectuées sur les paramètres de contrôle du problème (N T d). Vous remarquerez qu'elles ont la même structure au vu du rôle commutatif de T et N dans la formule $P = N \cdot T / (V - N \cdot b)$.

Nous verrons plus loin que la construction de la boucle sur le Diamètre nous a permis de déterminer le domaine de validité du modèle de départ.

3-1. Construction de la boucle du nombre de particules N

Le nombre de particules varie de 100 à 1000. Nous cherchons à fixer les autres paramètres du problème (T et d fixés) et à construire la courbe de P(N), T et d fixés.

Construction de la boucle des Particules 100 à 1000

```
T1 = 1;  
d = 0.01;  
for q=1:10;  
q= q*100;  
[...]
```

% calcul de la pression en fonction du nombre de particules

```
r = mean(Tableau_k);  
r_tab=[];  
r_tab = [r_tab r];  
disp(' Nombre de Particule1 ');
```

```

q
disp('moyenne de (Tableau_r)');
r
Tableau_r(q) = r;
end
Tableau_r

```

Tableau_r représente ici le vecteur qui contient les valeurs de la **pression optimale** en fonction du nombre de **particules**.

Exemple : valeurs obtenues pour la pression après simulation numérique

N	100	200	300	400	500	600	700	800	900	1000
P	50.2075	99.9171	151.7779	204.1978	256.9908	313.8151	369.2899	426.1298	477.9690	536.971

Tableau2 : tableau de pression en fonction du nombre de particules

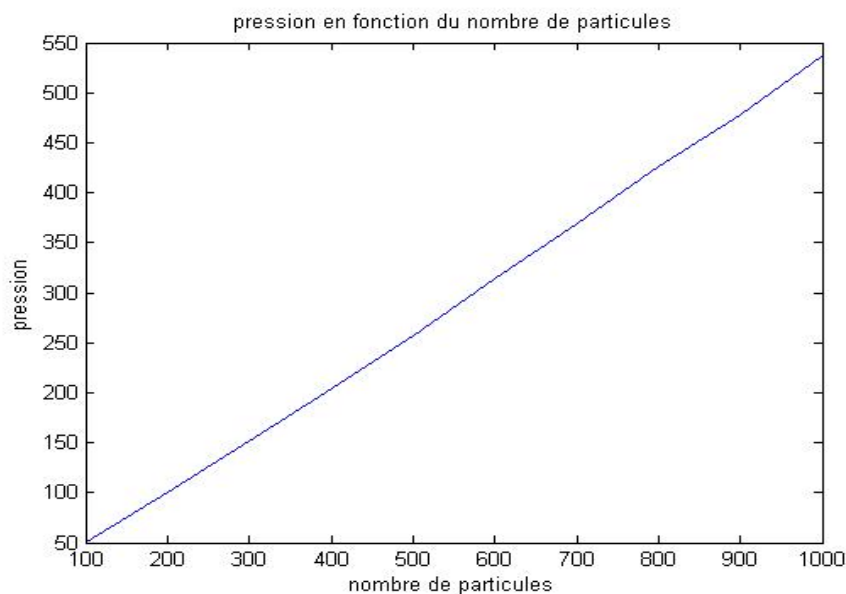


Schéma 4 : pression en fonction du nombre de particules

Confère : « **annexe 3 Tableau de pression par rapport au nombre de particules** »

Remarques :

Nous notons que la pression croît avec le nombre de particules. Les autres paramètres fixés, $d=0,01$ et $T=1$. Ceci n'est possible que dans la limite d'un diamètre critique « **d_critique** » à déterminer ultérieurement.

Nous avons remarqué, qu'en exécutant le programme à plusieurs reprises, les résultats obtenus ne varient pas beaucoup, donc le programme **kinetics** est stable numériquement.

3-2. Construction de la boucle de température T

La température varie linéairement avec la pression. Nous avons construit sa boucle pour chaque nombre de particule fixés :

```
d = 0.01;
for T=1:10;
    for ip=1:10 ;
        N = ip*100 ;
        [...]
    end
end
[...]
end
r = mean(Tableau_k);
r_tab=[];
r_tab = [r_tab; r];
disp('T');
T
disp('moyenne Tableau de Pression:');
r
Tableau_r(:,T)=r;
end
Tableau_r
```

Tableau_r est ici la matrice qui stocke les valeurs de la pression en fonction de la température et pour chaque N

Confère : « *annexe 4 : Tableau de pression par rapport à la température, pour chaque nombre de particules* »

	1	2	3	4	5	6	7	8	9	10	11	12	13
1	46.2719	91.5280	132.2255	174.6550	209.6577	249.8912	286.2198	329.3791	363.8423	397.8959			
2	95.6035	182.6410	266.1795	343.8458	426.8611	509.1551	579.9237	658.6898	738.7479	803.4435			
3	143.3079	273.8841	402.8281	522.6308	641.9249	759.1772	868.0413	985.9039	1.0827e+03	1.2024e+03			
4	190.1469	364.2092	540.2566	694.5378	857.3323	1.0119e+03	1.1655e+03	1.3114e+03	1.4619e+03	1.5988e+03			
5	237.6879	458.8332	664.5595	869.0295	1.0701e+03	1.2595e+03	1.4612e+03	1.6533e+03	1.8271e+03	2.0104e+03			
6	283.9523	552.0062	802.8363	1.0523e+03	1.2943e+03	1.5209e+03	1.7583e+03	1.9683e+03	2.2008e+03	2.4209e+03			
7	332.6692	639.5444	937.4892	1.2154e+03	1.5031e+03	1.7918e+03	2.0507e+03	2.3009e+03	2.5621e+03	2.8245e+03			
8	381.8617	737.7155	1.0703e+03	1.4091e+03	1.7246e+03	2.0424e+03	2.3391e+03	2.6330e+03	2.9322e+03	3.2088e+03			
9	430.5298	827.8994	1.2083e+03	1.5769e+03	1.9390e+03	2.2906e+03	2.6329e+03	2.9635e+03	3.3037e+03	3.6290e+03			
10	476.6063	920.8146	1.3463e+03	1.7611e+03	2.1476e+03	2.5612e+03	2.9305e+03	3.3241e+03	3.6638e+03	4.0441e+03			
11													
12													

Tableau 2 : Pression en fonction de la température pour chaque nombre de particules

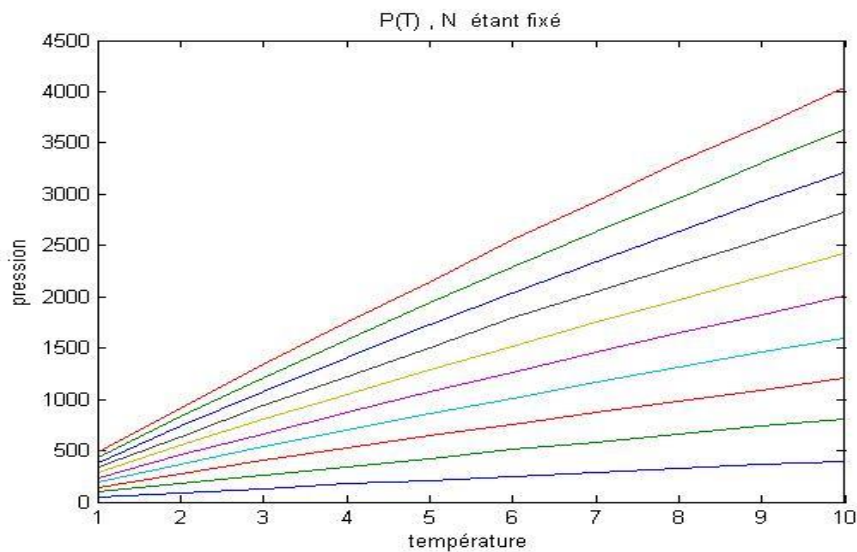


Schéma 5 : pression en fonction de la température pour chaque nombre de particules

Remarque :

Les courbes de la figure ci-dessus représentent la variation de la pression en fonction de température pour chaque nombre de particules. Plus nous augmentons la température, plus la pression augmente. Si de surcroît, nous augmentons le nombre de particules, la pression exercée par le gaz augmente aussi. Ces caractéristiques sont celles d'un gaz parfait. La linéarité conservée.

3-3. Construction de la boucle du diamètre d

```
T=1
for id=1:10; % id indice du diamètre d
d=0.005*id;

    for ip=1:10; % ip indice des particules
        N= ip*100;
        [...]
    end

    mean(Tableau_pression);
    r = mean(Tableau_pression);
    r_tab=[];
    r_tab = [r_tab r];
    disp('id');
    ip;
    disp('moyenne de (Tableau_pression): ');
    r;
    Tableau_r(id)=r;
```

```

end
Tableau_r
t=Tableau_r;
t_tab=[];
t_tab = [t_tab; t];
disp('id');
id
disp('Tableau_r:');
t
Tableau_t(:,id)=t;
end
Tableau_t

```

Tableau_t est une matrice 10*10 stockant les valeurs de la pression par rapport au diamètre.

	1	2	3	4	5	6	7	8	9	10
1	1.0200	0.5400	0.3933	0.3550	0.3660	0.4400	0.5586	0.6750	0.7400	0.7960
2	2.0500	1.1100	0.9367	1.0325	1.3100	1.5717	1.7371	1.8975	2.0500	2.1690
3	3.0400	1.7850	1.6633	2.0600	2.5200	2.8650	3.1786	3.4675	3.7489	3.9850
4	4.1000	2.4950	2.6167	3.2975	3.8860	4.3983	4.8986	5.3413	5.7244	6.0610
5	5.1400	3.3000	3.6933	4.6300	5.4060	6.1650	6.8471	7.4800	7.9200	8.3250
6	6.2700	4.1700	4.9333	6.1025	7.1340	8.1550	9.0271	9.7388	10.2933	10.7260
7	7.3600	5.1700	6.2367	7.6650	9.0220	10.2950	11.3643	12.1600	12.7744	13.1900
8	8.4700	6.1700	7.6367	9.4000	11.0660	12.5700	13.7657	14.6513	15.3367	15.7430
9	9.6000	7.2650	9.1200	11.2325	13.2540	14.9633	16.3071	17.2212	17.8711	18.3450
10	10.7400	8.4650	10.6733	13.2175	15.5860	17.5267	18.9343	19.9050	20.5944	21.0070
11										
12										

Tableau 3 : pression en fonction du diamètre

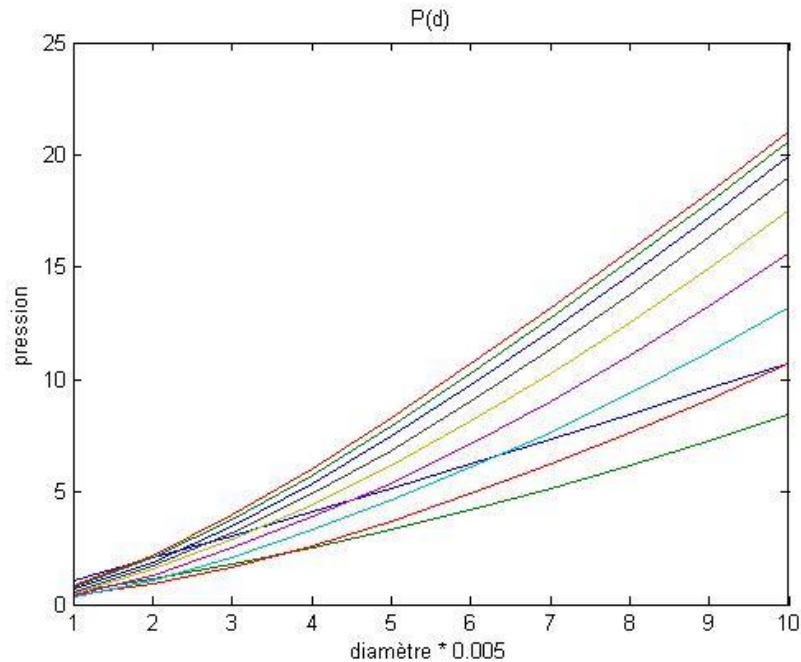


Schéma 6 : pression en fonction du diamètre

Commentaire :

Ce schéma des courbes de la pression par rapport à la température et pour chaque nombre de particules met en évidence l'existence de deux zones :

-diamètre < 0.01 :

Pour les petits diamètres ($d < 0.01$) la Courbe de la pression varie linéairement (sensiblement). L'effet de la taille des particules est négligeable dans une première approximation, même pour un grand nombre de particules. La raison principale est que dans l'équation :

$$P = N \cdot T / (V - N \cdot b),$$

Le terme $N \cdot b$ est un correctif par rapport au volume de l'enceinte qui est normalisé à 1 dans notre problème.

En effet :

$b = d^2$ qui est le volume d'une particule 2D, N étant le nombre de particules, le terme $N \cdot b$ est dérisoire devant V volume de l'enceinte.

L'équation du gaz se réduit à :

$P = N \cdot T / V$, équation du gaz parfait (linéaire en nombre de particules et température et inversement proportionnel au volume).

-diamètre > 0.01

A partir de ce diamètre de particule, l'évolution de la pression n'est plus linéaire. Nous arrivons dans une zone de transition où le modèle n'est plus stable linéairement. L'évolution devient parabolique

En effet :

L'influence du terme N^2b commence à être sensible. Ce terme est appelé covolume propre. Cette correction par rapport au gaz parfait est nécessaire pour déterminer l'équation d'état calorique du gaz réel.

Une autre correction est apportée sur la pression. Il s'agit de l'énergie potentielle d'interaction entre particules. Il faut la soustraire de l'énergie totale, par conséquent la pression finale du gaz est égale à :

$P = P_{\text{int}} - N^2a/V^2$, le terme de pression N^2a/V^2 étant assimilé à une pression entre particules.

Confère « Cours de L2 Mécanique énergétique, David Fabre ».

4-Variation de la pression en fonction des paramètres combinée

Nous avons construit cette boucle pour mettre en évidence l'influence du nombre de particules et du diamètre sur la courbe de pression.

Code

```
T1 = 1;
for id = 1:10;
    d = id*0.01;
    for ip = 1:10;
        N = ip*100;
        [...]
        [...]
        Tableau_k;
        mean(Tableau_k);
        r = mean(Tableau_k);
        r_tab=[];
        r_tab = [r_tab r];
        disp('ip');
        ip;
        disp('moyenne de (Tableau_k): ');
        r;
        Tableau_r(ip)=r;
    end

Tableau_r
t=Tableau_r;
t_tab=[];
t_tab = [t_tab t];
```

```
disp('id');
id
disp('Tableau_r:');
t
Tableau_t(:,id)=t;
end
Tableau_t
Confère : «annexe 5 kinetic9».
```

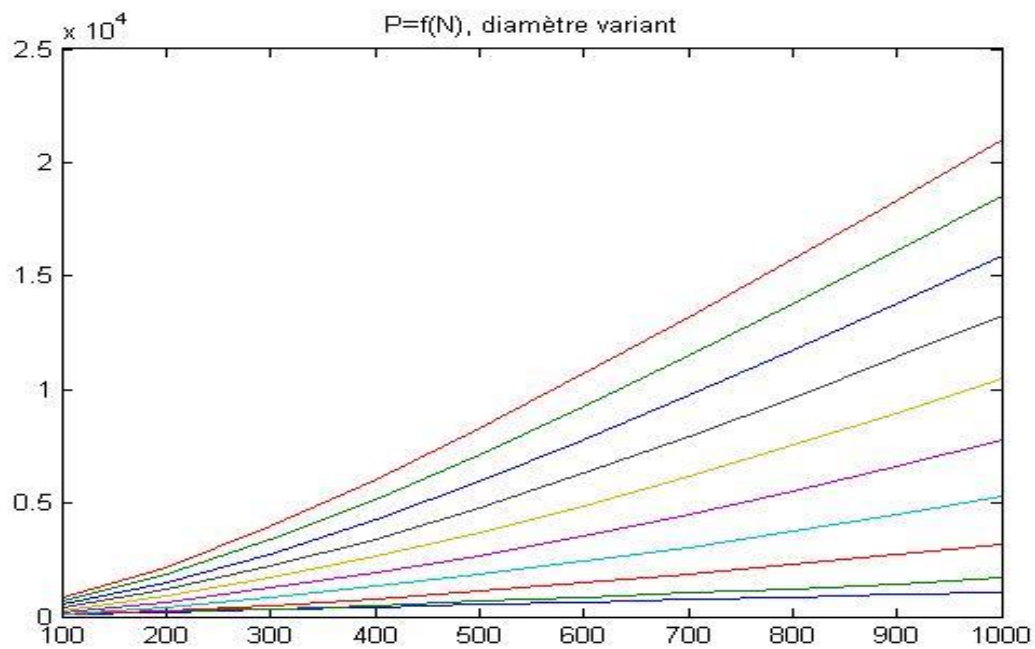


Schéma 7 : pression en fonction du nombre de particules pour chaque diamètre

Nous constatons que la pression augmente en combinant ces deux facteurs de contrôle. Elle atteint des valeurs de l'ordre de **10000** (unité normalisée à 1).

A titre de comparaison avec le **Schéma 4**, la pression en fonction de N et d peut atteindre des valeurs 20 fois plus élevées que si elle est seulement fonction de N .

D'où l'effet de la taille des particules qui, à partir de certaines valeurs, éloigne le modèle étudié du domaine de validité de l'équation des gaz parfaits.

P/TN en fonction de N

Cette fonction est équivalente à $1/(V-Nd^2)$. Elle rend compte un peu plus de la limite du modèle parfait.

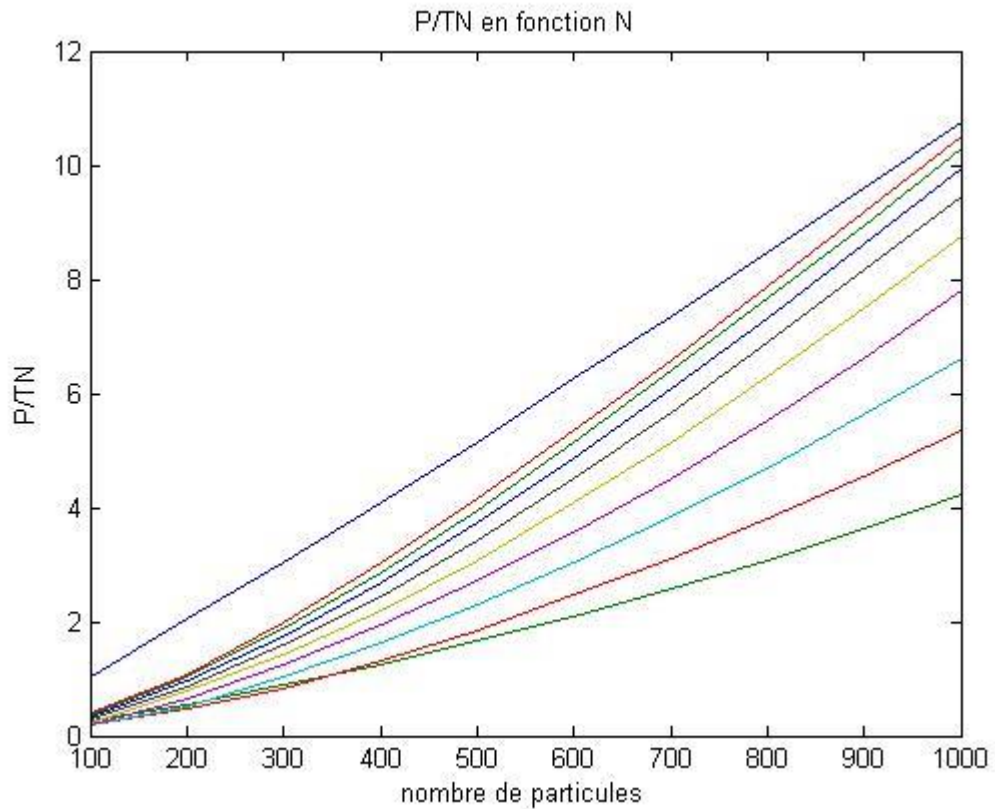


Schéma 8 : Pression normalisée $P/N \cdot T$ en fonction du nombre de particules

P/TN en fonction de d

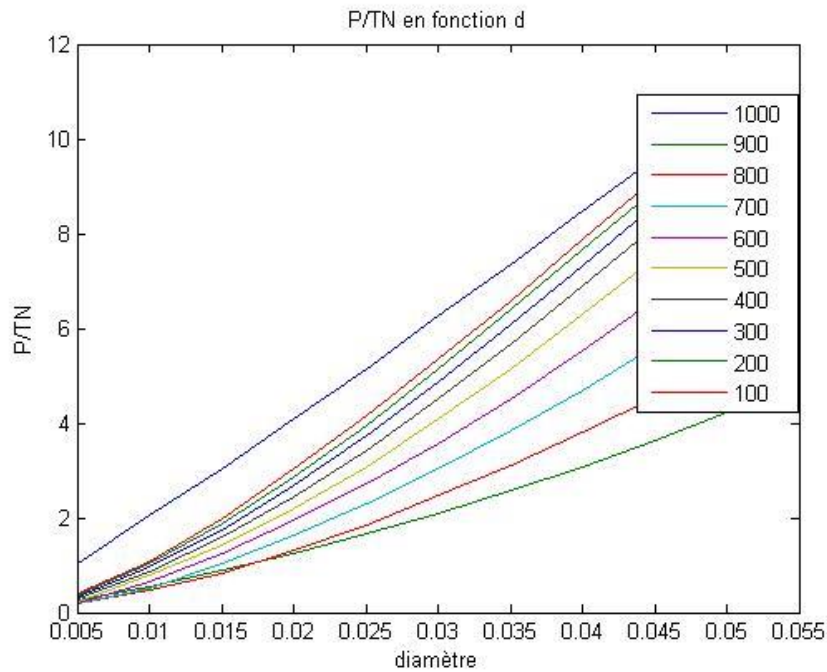
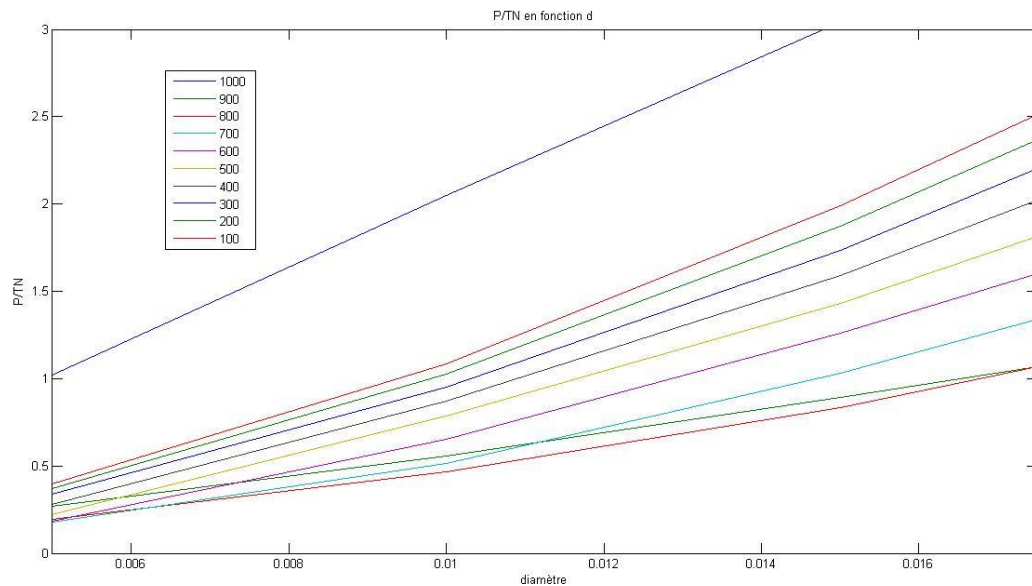


Schéma 11 : pression en fonction du diamètre

Commentaire :



Nous remarquons qu'à partir d'un diamètre critique de 0.01, l'évolution de la pression n'est plus linéaire. Ce résultat n'est pas étonnant car le covolume qui est fonction du diamètre devient plus grand. Donc, la pression qui lui est inversement proportionnelle diverge.

Conclusion :

La simulation d'un gaz parfait bidimensionnel, relativement simple sur le plan numérique, permet de retrouver l'essentiel des résultats de la théorie cinétique des gaz : équation d'état, distribution des vitesses, libre parcours moyen, durée moyenne de collision. Le comportement du gaz simulé est conforme aux prédictions de la théorie statistique.

Annexe 1

le programme Kinetics

```
%% Program Kinetics.m
%%
%% Dynamics of a 2D gas of hard spheres
%% Pedagogical program to illustrate the concepts of thermodynamics
%% and fluid mechanics.
%%
%% Version 1.0, 16/01/2017
%% D. Fabre
%%
%% USAGE :
%% 1/ Set parameters (lines 25-48 of the program)
%% 2/ Launch program
%% The program will open the following figures :
%%     Figure 1 : Instantaneous position of particules
%%     Figure 2 : Statistics of particules in vertical 'bins'
%%     Figure 3 : Force exerted on right boundary due to collision and its
averaged value ("Pressure")
%% 3/ Displace the figures on screen to avoid overlap
%% 4/ press "enter" to run the simulation

close all;
clear all;
Lx = 1;
Ly = 1;

%% PARAMETRES

particulediameter = 0.005;
Nparticules_1 = 100;
X1min = 0; X1max = 1;
T1 = 1;
Um1 = 0; Vm1 = 0;
color_1 = 'b';

Nparticules_2 = 0;
X2min = 0.5; X2max = 1;
T2 = 2;
Um2 = 0; Vm2 = 0;
color_2 = 'r';

Nbin = 10;
stepfig = 10;
fig1 = 1;
fig2 = 1;
fig3 = 1;
fig4 = 1;
fig5 = 1;
stepfig4 = 40;
Tmax = 1;
dt = 0.1;
diameter_for_plots = 345*particulediameter;

testmax=10;
for test = 1:testmax;

itermax = 10;
```

```
for iter=1:itermax

%% CORPS DU PROGRAMME

%% Initialisations : initial positions and velocities of particles
Nparticules = Nparticules_1;
vq1 = sqrt(T1);
X(1:Nparticules_1) = X1min+(X1max-X1min)*rand(1,Nparticules_1);
Y(1:Nparticules_1) = rand(1,Nparticules_1);
U(1:Nparticules_1) = Um1+vq1/sqrt(2)*randn(1,Nparticules_1);
V(1:Nparticules_1) = Vm1+vq1/sqrt(2)*randn(1,Nparticules_1);

vq1_premiertirage = sqrt(mean(U.^2 + V.^2))
U = U/vq1_premiertirage*sqrt(T1);
V = V/vq1_premiertirage*sqrt(T1);
vq1_apresrenormalisation = sqrt(mean(U.^2 + V.^2))

if(Nparticules_2>0)
vq2 = sqrt(T2);
X(Nparticules_1+1:Nparticules) = X2min+(X2max-X2min)*rand(1,Nparticules_2);
Y(Nparticules_1+1:Nparticules) = rand(1,Nparticules_2);
U(Nparticules_1+1:Nparticules) = Um2+vq2/sqrt(2)*randn(1,Nparticules_2);
V(Nparticules_1+1:Nparticules) = Vm2+vq2/sqrt(2)*randn(1,Nparticules_2);
end

for i=1:2:Nbin+1
    Xcage(2*i-1) = (i-1)*Lx/Nbin;
    Xcage(2*i) = Xcage(2*i-1);
    Xcage(2*i+1) = i*Lx/Nbin;
    Xcage(2*i+2) = Xcage(2*i+1);
    Ycage(2*i-1) = -1;
    Ycage(2*i) = Ly+1;
    Ycage(2*i+1) = Ly+1;
    Ycage(2*i+2) = -1;
end

t_tab = [];
F_tab = [];
P_tab = [];
T_tab = [];
A = [];
for i=1:Nbin
    x_bin(2*i-1)=(i-1)/10;
    x_bin(2*i) = i/10;
    x_bin_c(2*i-1)=(i-.5)/10;
    x_bin_c(2*i)=(i-.5)/10;
end

if(fig4==1)
figure(4);
hold off;
end
if(fig5==1)
figure(5);
hold off;
end

for it=0:Tmax/dt
    t = it*dt;
    X = X+dt*U;
    Y = Y+dt*V;
```

```

Force_right_boundary = 0;
for i=1:Nparticules
    if(X(i)<0&U(i)<0)
        U(i)=-U(i);
    end
    if(X(i)>Lx&U(i)>0)
        U(i)=-U(i);
        Force_right_boundary = Force_right_boundary-2*U(i)/dt;
    end
    if(Y(i)<0)
        Y(i)=Y(i)+Ly;
    end
    if(Y(i)>1)
        Y(i)=Y(i)-Ly;
    end
end

for i = 1:Nparticules
    for j=i+1:Nparticules
        distance = sqrt((X(i)-X(j))^2+(Y(i)-Y(j))^2);
        if( (distance<particulediameter) & (X(j)-X(i))*(U(j)-U(i))+(Y(j)-
Y(i))*(V(j)-V(i)) < 0)
            tx = Y(j)-Y(i);
            ty = X(i)-X(j);
            U1a = U(j);
            U2a = U(i);
            V1a = V(j);
            V2a = V(i);
            U(j) = (U1a*tx*tx+U2a*ty*ty+V1a*tx*ty-
V2a*tx*ty)/(tx*tx+ty*ty);
            U(i) = (U1a*ty*ty+U2a*tx*tx-
V1a*tx*ty+V2a*tx*ty)/(tx*tx+ty*ty);
            V(j) = (U1a*tx*ty-
U2a*tx*ty+V1a*ty*ty+V2a*tx*tx)/(tx*tx+ty*ty);
            V(i) = -(U1a*tx*ty-U2a*tx*ty-V1a*tx*tx-
V2a*ty*ty)/(tx*tx+ty*ty);
        end
    end
end

t_tab = [t_tab t];
F_tab = [F_tab Force_right_boundary];
P = mean(F_tab);
P_tab = [P_tab P];
Temp = mean(U.^2+V.^2);
T_tab = [T_tab Temp];

if(mod(t/dt,stepfig)==0)

if(fig1==1)
    figure(1);
    plot(Xcage,Ycage,'k--') % draws the bins
    hold on;
    h = plot(X(1:Nparticules_1),Y(1:Nparticules_1),[color_1
'o'],'MarkerSize',diameter_for_plots);
    set(h,'MarkerFaceColor',color_1)
    h2 =
plot(X(Nparticules_1+1:Nparticules),Y(Nparticules_1+1:Nparticules),[color_2
'o'],'MarkerSize',diameter_for_plots);

```



```

        set(h2,'MarkerFaceColor',color_2)
        axis square;
        axis([0 Lx 0 Ly])
        hold off;
    end

    if (fig2==1)
        figure(2);
        subplot(3,1,1);
        plot(t_tab,F_tab);
        title('Pressure Force on left boundary (instantaneous)');

        subplot(3,1,2);
        plot(t_tab,P_tab);
        title('Pressure Force on left boundary (averaged)');

        subplot(3,1,3);
        plot(t_tab,T_tab);
        title('Temperature');
    end

    for i=1:Nbin

        Vmoy_bin(2*i-1) = sum(V(X>(i-1)/10&X<i/10))/sum(X>(i-1)/10&X<i/10);
        Vmoy_bin(2*i) = Vmoy_bin(2*i-1);
        Umoy_bin(2*i-1) = sum(U(X>(i-1)/10&X<i/10))/sum(X>(i-1)/10&X<i/10);
        Umoy_bin(2*i) = Umoy_bin(2*i-1);
        T_bin(2*i-1) = (sum( (V(X>(i-1)/10&X<i/10)-Vmoy_bin(2*i)).^2 ...
                               + (U(X>(i-1)/10&X<i/10)-
Umoy_bin(2*i)).^2 )/sum(X>(i-1)/10&X<i/10)));
        T_bin(2*i) =T_bin(2*i-1);
        N1_bin(2*i-1) = sum(X(1:Nparticules_1)>(i-
1)/10&X(1:Nparticules_1)<i/10);
        N1_bin(2*i) = N1_bin(2*i-1);
        N2_bin(2*i-1) = sum(X(Nparticules_1+1:Nparticules)>(i-
1)/10&X(Nparticules_1+1:Nparticules)<i/10);
        N2_bin(2*i) = N2_bin(2*i-1);

    end

    if (fig3~=0)
        figure(3);
        subplot(2,2,1);
        if(color_1==color_2)

            plot(x_bin,N1_bin+N2_bin,'b')
        else
            plot(x_bin,N1_bin+N2_bin,'k')
            hold on;
            plot(x_bin,N1_bin,'b')
            plot(x_bin,N2_bin,'r')

        end
        title('Histogram of particle positions')
        hold off;

        subplot(2,2,2);
        plot(x_bin,T_bin,'g');
        title(' "Temperature" (mean square velocity)')
    end
end

```

```
if (fig2==2)
    hold on
end

subplot(2,2,3);
plot(x_bin,Vmoy_bin,'c');

title('mean vertical velocity');
if (fig2==2)
    hold on
end

subplot(2,2,4);
plot(x_bin,Umoy_bin,'m');
title('mean horizontal velocity')

if (fig2==2)
    hold on
end

end

if(fig4==1&&mod(t/dt,stepfig4)==0)
figure(4);
plot(x_bin_c,Nl_bin,'b');

title('repartition of blue particules at successive instants');
hold on;
end

if(fig4==1&&mod(t/dt,stepfig4)==0)
figure(5);
plot(x_bin_c,Vmoy_bin,'c');

title('mean vertical velocity at successive instants');
hold on;
end

if(t==0)
else
    pause(0.001);
end
end

end
disp(' iteration numero ');
iter
disp('temperature :');
Temp
disp('Pression :');
P;
Tableau_P(iter) = P;
end
Tableau_P
mean(Tableau_P)

% travail effectué sur ce Kinetics --> se référer aux autres programmes
modifiés
```

« Annexe 2 : Pression moyenne après 100 réalisations »

```
% Kinetics + calcul de la pression moyenne pour 100 réalisations
close all;
clear all;
Lx = 1;
Ly = 1;

particulediameter = 0.005;
Nparticules_1 = 100;
X1min = 0; X1max = 1;
T1 = 1;
Um1 = 0; Vm1 = 0;
color_1 = 'b';

Nparticules_2 = 0;
X2min = 0.5; X2max = 1;
T2 = 2;
Um2 = 0; Vm2 = 0;
color_2 = 'r';

Nbin = 10;
stepfig = 10;
fig1 = 1;
fig2 = 1;
fig3 = 1;
fig4 = 1;
fig5 = 1;
stepfig4 = 40;
Tmax = 1;
dt = 0.1;
diameter_for_plots = 345*particulediameter;

testmax=10;
for test = 1:testmax;

itermax = 10;
for iter=1:itermax

%% Initialisations : initial positions and velocities of particles
Nparticules = Nparticules_1;
vq1 = sqrt(T1);
X(1:Nparticules_1) = X1min+(X1max-X1min)*rand(1,Nparticules_1);
Y(1:Nparticules_1) = rand(1,Nparticules_1);
U(1:Nparticules_1) = Um1+vq1/sqrt(2)*randn(1,Nparticules_1);
V(1:Nparticules_1) = Vm1+vq1/sqrt(2)*randn(1,Nparticules_1);

vq1_premiertirage = sqrt(mean(U.^2 + V.^2))
U = U/vq1_premiertirage*sqrt(T1);
V = V/vq1_premiertirage*sqrt(T1);
vq1_apresrenormalisation = sqrt(mean(U.^2 + V.^2))

if(Nparticules_2>0)
vq2 = sqrt(T2);
X(Nparticules_1+1:Nparticules) = X2min+(X2max-X2min)*rand(1,Nparticules_2);
Y(Nparticules_1+1:Nparticules) = rand(1,Nparticules_2);
U(Nparticules_1+1:Nparticules) = Um2+vq2/sqrt(2)*randn(1,Nparticules_2);
V(Nparticules_1+1:Nparticules) = Vm2+vq2/sqrt(2)*randn(1,Nparticules_2);
end
```

```
for i=1:2:Nbin+1
    Xcage(2*i-1) = (i-1)*Lx/Nbin;
    Xcage(2*i) = Xcage(2*i-1);
    Xcage(2*i+1) = i*Lx/Nbin;
    Xcage(2*i+2) = Xcage(2*i+1);
    Ycage(2*i-1) = -1;
    Ycage(2*i) = Ly+1;
    Ycage(2*i+1) = Ly+1;
    Ycage(2*i+2) = -1;
end
t_tab = [];
F_tab = [];
P_tab = [];
T_tab = [];
A = [];
for i=1:Nbin
    x_bin(2*i-1)=(i-1)/10;
    x_bin(2*i) = i/10;
    x_bin_c(2*i-1)=(i-.5)/10;
    x_bin_c(2*i)=(i-.5)/10;
end

if(fig4==1)
    figure(4);
    hold off;
end
if(fig5==1)
    figure(5);
    hold off;
end

for it=0:Tmax/dt
    t = it*dt;
    X = X+dt*U;
    Y = Y+dt*V;

    Force_right_boundary = 0;
    for i=1:Nparticules
        if(X(i)<0&U(i)<0)
            U(i)=-U(i);
        end
        if(X(i)>Lx&U(i)>0)
            U(i)=-U(i);
            Force_right_boundary = Force_right_boundary-2*U(i)/dt;
        end
        if(Y(i)<0)
            Y(i)=Y(i)+Ly;
        end
        if(Y(i)>1)
            Y(i)=Y(i)-Ly;
        end
    end

    for i = 1:Nparticules
        for j=i+1:Nparticules
            distance = sqrt((X(i)-X(j))^2+(Y(i)-Y(j))^2);
            if( (distance<particulediameter)&(X(j)-X(i))*(U(j)-U(i))+(Y(j)-Y(i))*(V(j)-V(i)) < 0)
                tx = Y(j)-Y(i);
```

```

        ty = X(i)-X(j);
        U1a = U(j);
        U2a = U(i);
        V1a = V(j);
        V2a = V(i);
        U(j) = (U1a*tx*tx+U2a*ty*ty+V1a*tx*ty-
V2a*tx*ty)/(tx*tx+ty*ty);
        U(i) = (U1a*ty*ty+U2a*tx*tx-
V1a*tx*ty+V2a*tx*ty)/(tx*tx+ty*ty);
        V(j) = (U1a*tx*ty-
U2a*tx*ty+V1a*ty*ty+V2a*tx*tx)/(tx*tx+ty*ty);
        V(i) = -(U1a*tx*ty-U2a*tx*ty-V1a*tx*tx-
V2a*ty*ty)/(tx*tx+ty*ty);
    end
end
end

t_tab = [t_tab t];
F_tab = [F_tab Force_right_boundary];
P = mean(F_tab);
P_tab = [P_tab P];
Temp = mean(U.^2+V.^2);
T_tab = [T_tab Temp];

if(mod(t/dt,stepfig)==0)

if(fig1==1)
    figure(1);
    plot(Xcage,Ycage,'k--') % draws the bins
    hold on;
    h = plot(X(1:Nparticules_1),Y(1:Nparticules_1),[color_1
'o'],'MarkerSize',diameter_for_plots);
    set(h,'MarkerFaceColor',color_1)
    h2 =
plot(X(Nparticules_1+1:Nparticules),Y(Nparticules_1+1:Nparticules),[color_2
'o'],'MarkerSize',diameter_for_plots);
    set(h2,'MarkerFaceColor',color_2)
    axis square;
    axis([0 Lx 0 Ly])
    hold off;
end

if (fig2==1)
figure(2);
subplot(3,1,1);
plot(t_tab,F_tab);
title('Pressure Force on left boundary (instantaneous)');

subplot(3,1,2);
plot(t_tab,P_tab);
title('Pressure Force on left boundary (averaged)');

subplot(3,1,3);
plot(t_tab,T_tab);
title('Temperature');
end

for i=1:Nbin

```

```

Vmoy_bin(2*i-1) = sum(V(X>(i-1)/10&X<i/10))/sum(X>(i-1)/10&X<i/10);
Vmoy_bin(2*i) = Vmoy_bin(2*i-1);
Umoy_bin(2*i-1) = sum(U(X>(i-1)/10&X<i/10))/sum(X>(i-1)/10&X<i/10);
Umoy_bin(2*i) = Umoy_bin(2*i-1);
T_bin(2*i-1) = (sum( (V(X>(i-1)/10&X<i/10)-Vmoy_bin(2*i)).^2 ...
                    + (U(X>(i-1)/10&X<i/10)-
Umoy_bin(2*i)).^2 )/sum(X>(i-1)/10&X<i/10));
T_bin(2*i) =T_bin(2*i-1);
N1_bin(2*i-1) = sum(X(1:Nparticules_1)>(i-
1)/10&X(1:Nparticules_1)<i/10);
N1_bin(2*i) = N1_bin(2*i-1);
N2_bin(2*i-1) = sum(X(Nparticules_1+1:Nparticules)>(i-
1)/10&X(Nparticules_1+1:Nparticules)<i/10);
N2_bin(2*i) = N2_bin(2*i-1);

end

if (fig3~=0)
figure(3);
subplot(2,2,1);
if(color_1==color_2)

plot(x_bin,N1_bin+N2_bin,'b')
else
plot(x_bin,N1_bin+N2_bin,'k')
hold on;
plot(x_bin,N1_bin,'b')
plot(x_bin,N2_bin,'r')

end
title('Histogram of particle positions')
hold off;

subplot(2,2,2);
plot(x_bin,T_bin,'g');
title(' "Temperature" (mean square velocity)')

if (fig2==2)
hold on
end

subplot(2,2,3);
plot(x_bin,Vmoy_bin,'c');

title('mean vertical velocity');
if (fig2==2)
hold on
end

subplot(2,2,4);
plot(x_bin,U moy_bin,'m');
title('mean horizontal velocity')

if (fig2==2)
hold on
end

end

if(fig4==1&&mod(t/dt,stepfig4)==0)
figure(4);

```

```
plot(x_bin_c,Nl_bin,'b');

title('repartition of blue particules at successive instants');
hold on;
end

if(fig4==1&&mod(t/dt,stepfig4)==0)
figure(5);
plot(x_bin_c,Vmoy_bin,'c');

title('mean vertical velocity at successive instants');
hold on;
end

if(t==0)
else
    pause(0.001);
end
end

end
disp(' iteration numero ');
iter
disp('temperature :');
Temp
disp('Pression :');
P;
Tableau_P(iter) = P;
end
Tableau_P
mean(Tableau_P)
std(Tableau_P);

%%calcul de la pression moyenne pour 100 réalisations
k= mean(Tableau_P);
k_tab=[];
k_tab = [k_tab k];
disp('test numero ');
test;
disp('mean(Tableau_P): ');
k;
Tableau_k(test) = k;
end
Tableau_k
mean(Tableau_k)
```

Annexe 3

```
%%Construction du tableau des moyennes de pression en fonction de N=[100 :  
1000]  
  
close all;  
clear all;  
Lx = 1;  
Ly = 1;  
  
particulediameter = 0.005;  
  
for q=1:10  
    q= q*100;  
  
    Nparticules_1 = 100;  
    X1min = 0; X1max = 1;  
    T1 = 1;  
    Um1 = 0; Vm1 = 0;  
    color_1 = 'b';  
  
    Nparticules_2 = 0;  
    X2min = 0.5; X2max = 1;  
    T2 = 2;  
    Um2 = 0; Vm2 = 0;  
    color_2 = 'r';  
  
    Nbin = 10;  
    stepfig = 10;  
    fig1 = 1;  
    fig2 = 1;  
    fig3 = 1;  
    fig4 = 1;  
    fig5 = 1;  
    stepfig4 = 40;  
    Tmax = 1;  
    dt = 0.1;  
    diameter_for_plots = 345*particulediameter;  
  
    Nbin = 10;  
    stepfig = 10;  
    fig1 = 1;  
    fig2 = 1;  
    fig3 = 1;  
    fig4 = 1;  
    fig5 = 1;  
    stepfig4 = 40;  
    Tmax = 1;  
    dt = 0.1;  
    diameter_for_plots = 345*particulediameter;  
  
    testmax=10;  
    for test = 1:testmax;  
  
        itermax = 10;  
        for iter=1:itermax  
  
            %% CORPS DU PROGRAMME
```



```

%% Initialisations : initial positions and velocities of particules
Nparticules = Nparticules_1;
vq1 = sqrt(T1);
X(1:Nparticules_1) = X1min+(X1max-X1min)*rand(1,Nparticules_1);
Y(1:Nparticules_1) = rand(1,Nparticules_1);
U(1:Nparticules_1) = Um1+vq1/sqrt(2)*randn(1,Nparticules_1);
V(1:Nparticules_1) = Vm1+vq1/sqrt(2)*randn(1,Nparticules_1);

vq1_premiertirage = sqrt(mean(U.^2 + V.^2))
U = U/vq1_premiertirage*sqrt(T1);
V = V/vq1_premiertirage*sqrt(T1);
vq1_apresrenormalisation = sqrt(mean(U.^2 + V.^2))

if(Nparticules_2>0)
vq2 = sqrt(T2);
X(Nparticules_1+1:Nparticules) = X2min+(X2max-X2min)*rand(1,Nparticules_2);
Y(Nparticules_1+1:Nparticules) = rand(1,Nparticules_2);
U(Nparticules_1+1:Nparticules) = Um2+vq2/sqrt(2)*randn(1,Nparticules_2);
V(Nparticules_1+1:Nparticules) = Vm2+vq2/sqrt(2)*randn(1,Nparticules_2);
end

for i=1:2:Nbin+1
    Xcage(2*i-1) = (i-1)*Lx/Nbin;
    Xcage(2*i) = Xcage(2*i-1);
    Xcage(2*i+1) = i*Lx/Nbin;
    Xcage(2*i+2) = Xcage(2*i+1);
    Ycage(2*i-1) = -1;
    Ycage(2*i) = Ly+1;
    Ycage(2*i+1) = Ly+1;
    Ycage(2*i+2) = -1;
end

t_tab = [];
F_tab = [];
P_tab = [];
T_tab = [];
A = [];
for i=1:Nbin
    x_bin(2*i-1)=(i-1)/10;
    x_bin(2*i) = i/10;
    x_bin_c(2*i-1)=(i-.5)/10;
    x_bin_c(2*i)=(i-.5)/10;
end

if(fig4==1)
figure(4);
hold off;
end
if(fig5==1)
figure(5);
hold off;
end

for it=0:Tmax/dt
    t = it*dt;
    X = X+dt*U;
    Y = Y+dt*V;

    Force_right_boundary = 0;
    for i=1:Nparticules

```

```

        if(X(i)<0&U(i)<0)
            U(i)=-U(i);
        end
        if(X(i)>Lx&U(i)>0)
            U(i)=-U(i);
            Force_right_boundary = Force_right_boundary-2*U(i)/dt;
        end
        if(Y(i)<0)
            Y(i)=Y(i)+Ly;
        end
        if(Y(i)>1)
            Y(i)=Y(i)-Ly;
        end
    end

    for i = 1:Nparticules
        for j=i+1:Nparticules
            distance = sqrt((X(i)-X(j))^2+(Y(i)-Y(j))^2);
            if( (distance<particulediameter) & (X(j)-X(i))*(U(j)-U(i))+(Y(j)-
Y(i))*(V(j)-V(i)) < 0)
                tx = Y(j)-Y(i);
                ty = X(i)-X(j);
                U1a = U(j);
                U2a = U(i);
                V1a = V(j);
                V2a = V(i);
                U(j) = (U1a*tx*tx+U2a*ty*ty+V1a*tx*ty-
V2a*tx*ty)/(tx*tx+ty*ty);
                U(i) = (U1a*ty*ty+U2a*tx*tx-
V1a*tx*ty+V2a*tx*ty)/(tx*tx+ty*ty);
                V(j) = (U1a*tx*ty-
U2a*tx*ty+V1a*ty*ty+V2a*tx*tx)/(tx*tx+ty*ty);
                V(i) = -(U1a*tx*ty-U2a*tx*ty-V1a*tx*tx-
V2a*ty*ty)/(tx*tx+ty*ty);
            end
        end
    end

    t_tab = [t_tab t];
    F_tab = [F_tab Force_right_boundary];
    P = mean(F_tab);
    P_tab = [P_tab P];
    Temp = mean(U.^2+V.^2);
    T_tab = [T_tab Temp];

    if(mod(t/dt,stepfig)==0)

        if(fig1==1)
            figure(1);
            plot(Xcage,Ycage,'k--') % draws the bins
            hold on;
            h = plot(X(1:Nparticules_1),Y(1:Nparticules_1),[color_1
'0'],'MarkerSize',diameter_for_plots);
            set(h,'MarkerFaceColor',color_1)
            h2 =
plot(X(Nparticules_1+1:Nparticules),Y(Nparticules_1+1:Nparticules),[color_2
'0'],'MarkerSize',diameter_for_plots);
            set(h2,'MarkerFaceColor',color_2)
            axis square;
            axis([0 Lx 0 Ly])
        end
    end
end

```

```

        hold off;
    end

    if (fig2==1)
        figure(2);
        subplot(3,1,1);
        plot(t_tab,F_tab);
        title('Pressure Force on left boundary (instantaneous)');

        subplot(3,1,2);
        plot(t_tab,P_tab);
        title('Pressure Force on left boundary (averaged)');

        subplot(3,1,3);
        plot(t_tab,T_tab);
        title('Temperature');
    end

    for i=1:Nbin

        Vmoy_bin(2*i-1) = sum(V(X>(i-1)/10&X<i/10))/sum(X>(i-1)/10&X<i/10);
        Vmoy_bin(2*i) = Vmoy_bin(2*i-1);
        Umoy_bin(2*i-1) = sum(U(X>(i-1)/10&X<i/10))/sum(X>(i-1)/10&X<i/10);
        Umoy_bin(2*i) = Umoy_bin(2*i-1);
        T_bin(2*i-1) = (sum( (V(X>(i-1)/10&X<i/10)-Vmoy_bin(2*i)).^2 ...
            + (U(X>(i-1)/10&X<i/10)-
Umoy_bin(2*i)).^2 )/sum(X>(i-1)/10&X<i/10));
        T_bin(2*i) =T_bin(2*i-1);
        N1_bin(2*i-1) = sum(X(1:Nparticules_1)>(i-
1)/10&X(1:Nparticules_1)<i/10);
        N1_bin(2*i) = N1_bin(2*i-1);
        N2_bin(2*i-1) = sum(X(Nparticules_1+1:Nparticules)>(i-
1)/10&X(Nparticules_1+1:Nparticules)<i/10);
        N2_bin(2*i) = N2_bin(2*i-1);

    end

    if (fig3~=0)
        figure(3);
        subplot(2,2,1);
        if(color_1==color_2)

            plot(x_bin,N1_bin+N2_bin,'b')
        else
            plot(x_bin,N1_bin+N2_bin,'k')
            hold on;
            plot(x_bin,N1_bin,'b')
            plot(x_bin,N2_bin,'r')
        end

        title('Histogram of particle positions')
        hold off;

        subplot(2,2,2);
        plot(x_bin,T_bin,'g');
        title(' "Temperature" (mean square velocity)')

        if (fig2==2)
            hold on
        end
    end

```

```
subplot(2,2,3);
plot(x_bin,Vmoy_bin,'c');

title('mean vertical velocity');
if (fig2==2)
    hold on
end

subplot(2,2,4);
plot(x_bin,Umoy_bin,'m');
title('mean horizontal velocity')

if (fig2==2)
    hold on
end

end

if(fig4==1&&mod(t/dt,stepfig4)==0)
figure(4);
plot(x_bin_c,Nl_bin,'b');

title('repartition of blue particules at successive instants');
hold on;
end

if(fig4==1&&mod(t/dt,stepfig4)==0)
figure(5);
plot(x_bin_c,Vmoy_bin,'c');

title('mean vertical velocity at successive instants');
hold on;
end

if(t==0)
else
    pause(0.001);
end
end

end
disp(' iteration numero ');
iter
disp('temperature :');
Temp
disp('Pression :');
P;
Tableau_P(iter) = P;
end
Tableau_P
mean(Tableau_P)
disp('moyenne de(Tableau_k):');
mean(Tableau_k);
P_optimale = mean(Tableau_k);

r = mean(Tableau_k);
r_tab=[];
r_tab = [r_tab r];
disp(' Nombre de Particule1 ');
```

```
q
disp('moyenne de (Tableau_r)');
r
Tableau_r(q) = r;
end
```

Tableau_r

Annexe 4:

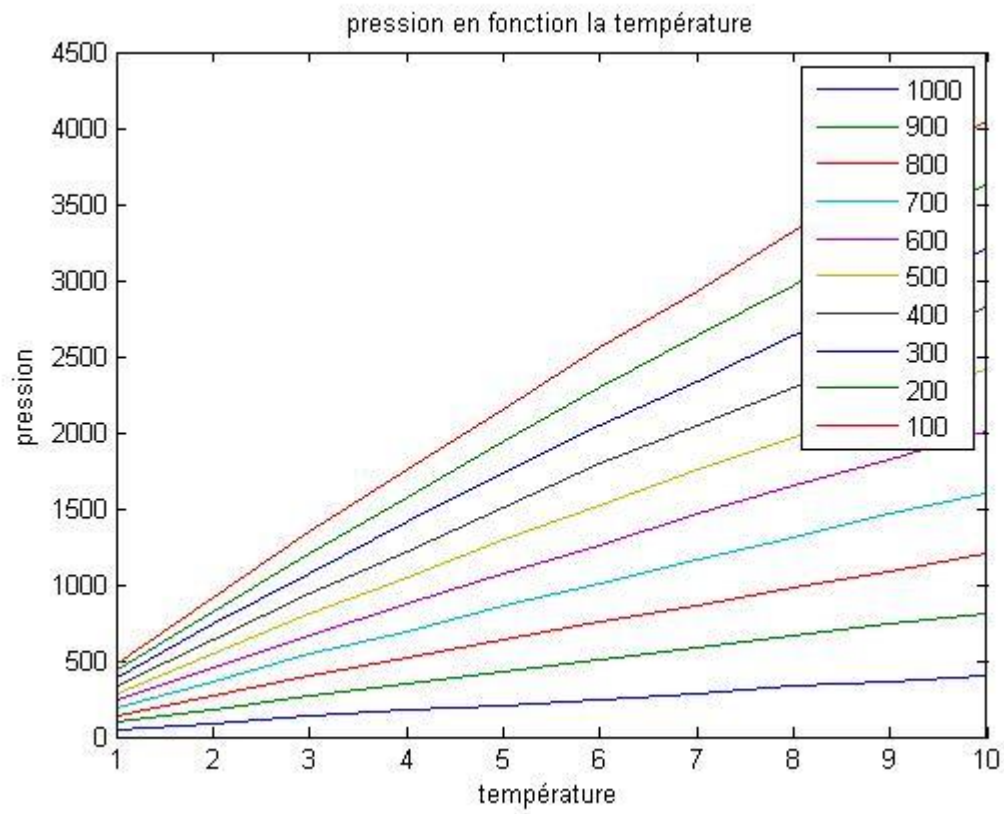
```

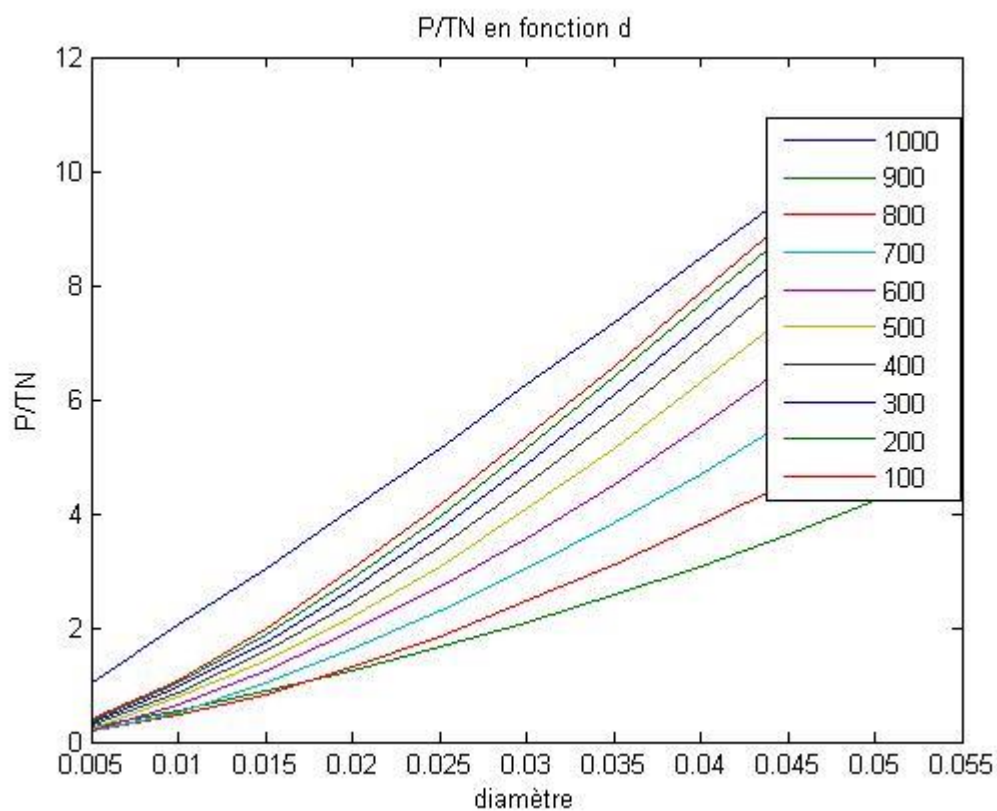
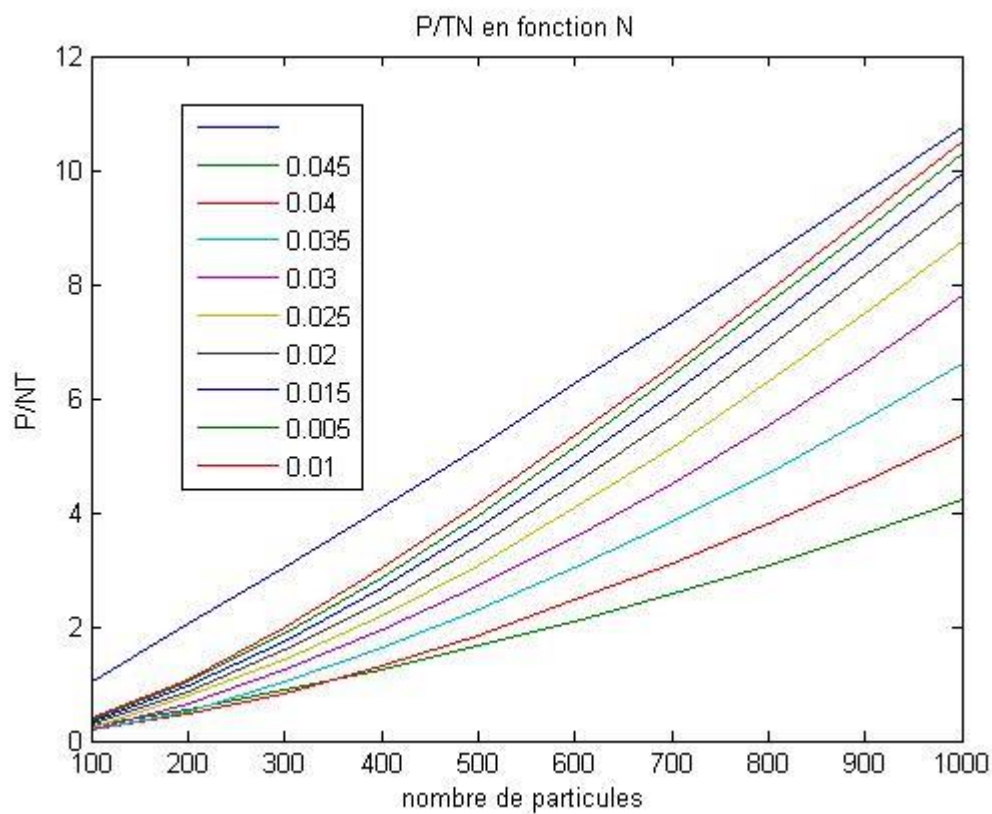
d = 0.01;
for T=1:10;
    for ip=1:10 ;
        N = ip*100 ;
        [ kinetics]

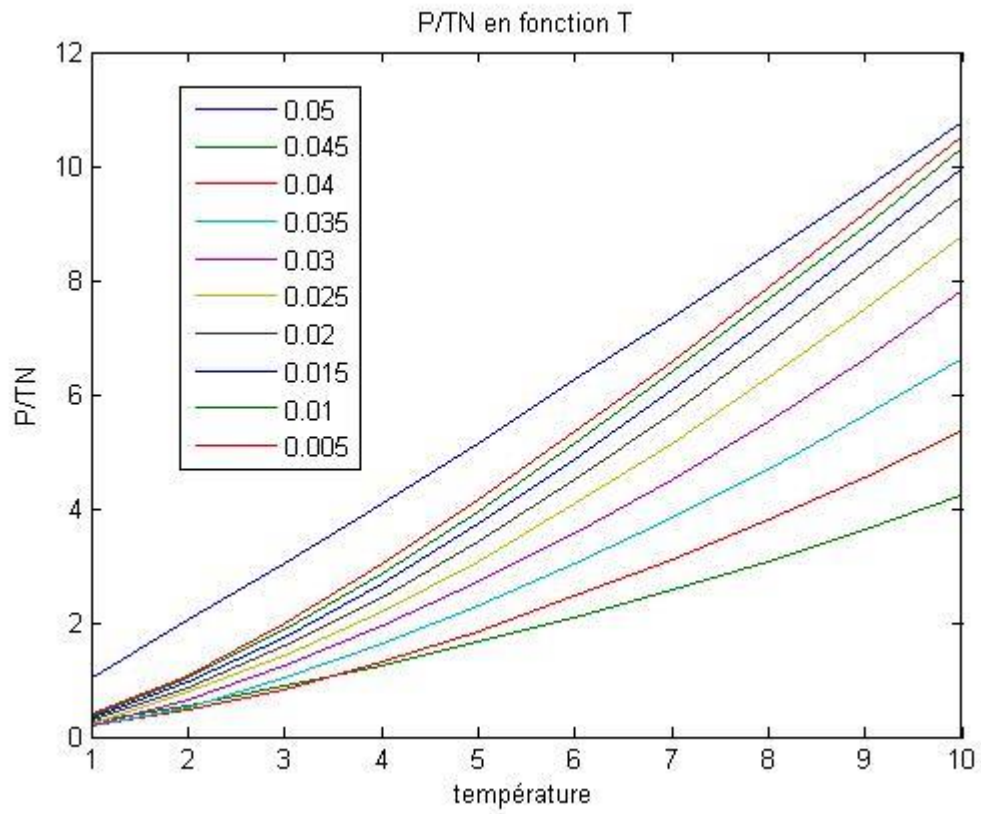
    end
    r = mean(Tableau_k);
    r_tab=[];
    r_tab = [r_tab; r];
    disp('T');
    T
    disp('moyenne Tableau de Pression:');
    r
    Tableau_r(:,T)=r;
end
Tableau_r

```

PT <10x10 double>													
	1	2	3	4	5	6	7	8	9	10	11	12	13
1	46.2719	91.5280	132.2255	174.6550	209.6577	249.8912	286.2198	329.3791	363.8423	397.8959			
2	95.6035	182.6410	266.1795	343.8458	426.8611	509.1551	579.9237	658.6898	738.7479	803.4435			
3	143.3079	273.8841	402.8281	522.6308	641.9249	759.1772	868.0413	985.9039	1.0827e+03	1.2024e+03			
4	190.1469	364.2092	540.2566	694.5378	857.3323	1.0119e+03	1.1655e+03	1.3114e+03	1.4619e+03	1.5988e+03			
5	237.6879	458.8332	664.5595	869.0295	1.0701e+03	1.2595e+03	1.4612e+03	1.6533e+03	1.8271e+03	2.0104e+03			
6	283.9523	552.0062	802.8363	1.0523e+03	1.2943e+03	1.5209e+03	1.7583e+03	1.9683e+03	2.2008e+03	2.4209e+03			
7	332.6692	639.5444	937.4892	1.2154e+03	1.5031e+03	1.7918e+03	2.0507e+03	2.3009e+03	2.5621e+03	2.8245e+03			
8	381.8617	737.7155	1.0703e+03	1.4091e+03	1.7246e+03	2.0424e+03	2.3391e+03	2.6330e+03	2.9322e+03	3.2088e+03			
9	430.5298	827.8994	1.2083e+03	1.5769e+03	1.9390e+03	2.2906e+03	2.6329e+03	2.9635e+03	3.3037e+03	3.6290e+03			
10	476.6063	920.8146	1.3463e+03	1.7611e+03	2.1476e+03	2.5612e+03	2.9305e+03	3.3241e+03	3.6638e+03	4.0441e+03			
11													
12													







Annexe 5 : Kinectic2 (diamètre)

```
%% pression par rapport au diametre
clear all;
close all;
Lx = 1;
Ly = 1;

for id=1:10;
particulediameter=0.01*id;

    for ip=1:10;
        Nparticules_1 = ip*100;
        Xlmin = 0; Xlmax = 1;
        T1 = 1;

        [...Kinetics...]

    end

end

disp(' iteration numero ');
iter
disp('temperature :');
Temp
disp('Pression :');
P;
Tableau_P(iter) = P;
end
Tableau_P
mean(Tableau_P)
std(Tableau_P);
%% k=mean(Tableau_P)
k= mean(Tableau_P);
k_tab=[];
k_tab = [k_tab k];
disp('test numero ');
test
disp('mean(Tableau_P): ');
k;
Tableau_k(test) = k;
end
Tableau_k;
mean(Tableau_k);
r = mean(Tableau_k);
r_tab=[];
r_tab = [r_tab r];
disp('ip');
ip;
disp('moyenne de (Tableau_k): ');
r;
Tableau_r(ip)=r;
end
Tableau_r
t=Tableau_r;
t_tab=[];
t_tab = [t_tab t];
disp('id');
id
disp('Tableau_r: ');
t
```

Rapport de Projet

```
Tableau_t(id)=t;  
end  
Tableau_t  
plot(Tableau_t)
```

Référence Bibliographique : José-Philippe Pérez, Thermodynamique, fondements et applications, 3^e édition
DUNOD

-Internet