

Le microscope à force atomique (AFM)

Sommaire

I. Présentation

- 1. Introduction page 3
- 2. Le microscope à force atomique page 4

II. Méthodes numériques

- 1. La méthode d'Euler page 5
- 2. La méthode de Runge-Kutta à l'ordre 2 page 6
- 3. La méthode de Runge-Kutta à l'ordre 4 page 6
- 4. Précision des méthodes page 7

III. Application à l'AFM

- 1. Mouvement de la pointe à une dimension
 - a) Position page 8
 - b) Vitesse page 9
 - c) Mouvement saccadé page 10
- 2. Force latérale et force de frottement page 12
- 3. Modèle à deux dimensions
 - a) Surface symétrique page 14
 - b) Surface asymétrique page 16

IV. Conclusion page 18

V. Annexe des programmes page 19

I. Présentation

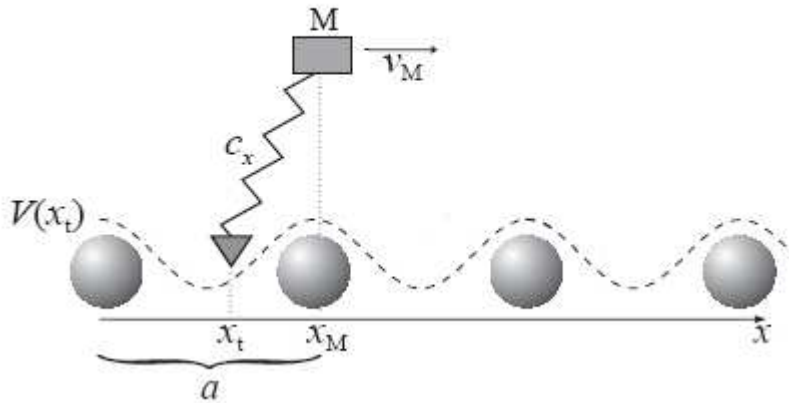
1. Introduction

Le microscope à force atomique (Atomic Force Microscope, AFM), appartient à la famille des microscopes à champ proche développés dans les années 80. Le principe de fonctionnement est le suivant : une sonde (une fine pointe métallique dans notre cas) est montée à l'extrémité d'un levier souple. On déplace ensuite ce levier au dessus d'une surface avec laquelle la pointe réagit. Un laser est utilisé pour détecter les oscillations du levier, qui est mis en mouvement par la sonde interagissant avec la surface, on peut alors reconstruire le profil de la surface balayée à condition que la surface ne conduise pas l'électricité, auquel cas on utilise un microscope à effet tunnel dont le principe de fonctionnement est assez proche de l'AFM.

Notre but est de modéliser d'une manière réaliste le mouvement de cette pointe interagissant avec un potentiel périodique de surface, en faisant intervenir la force de frottement, puis nous allons voir si la force de frottement est indépendante de la vitesse de glisse au niveau microscopique, ce qui est le cas au niveau macroscopique (3ème loi d'Amontons-Coulomb).

2. Le microscope à force atomique.

Pour cela nous allons modéliser le dispositif de la manière suivante : un mobile M qui se déplace au dessus d'une surface à une vitesse constante v_M est relié élastiquement à une pointe via un ressort possédant une constante de raideur C_x suivant la direction x . Cette pointe interagit avec la surface via un potentiel $V(x_t)$ où x_t représente la position de la pointe (x_M est la position du mobile).



On étudiera le cas d'une surface périodique, le potentiel s'écrit donc : $V(x_t) = V_0 \cos\left(\frac{2\pi}{a} x_t\right)$ où a est la périodicité du réseau suivant x (la distance entre les atomes). A cela s'ajoute un terme d'amortissement proportionnel à la vitesse de glissement dû à la dissipation d'énergie.

Trois forces s'exercent donc sur la pointe :

- La force de rappel du ressort qui fait avancer la pointe : $F_x = C_x(x_M - x_t)$
- La force d'amortissement due à la dissipation d'énergie : $F_v = -\gamma_x \dot{x}_t$
- La force qu'exerce le potentiel de la surface sur la pointe : $F_p = -\text{grad}(V(x_t))$

La relation fondamentale de la dynamique nous dicte : $m_x \ddot{x}_t = F_x + F_v + F_p$

Nous avons donc :
$$m_x \frac{d^2 x_t}{dt^2} = C_x (x_M - x_t) - V_0 \frac{2\pi}{a} \sin\left(\frac{2\pi}{a} x_t\right) - \gamma_x \frac{dx_t}{dt} \quad (1)$$

où m_x est la masse effective du système, $x_M = v_M t$ la position d'équilibre du ressort et γ_x la constante d'amortissement.

La solution de cette équation différentielle est la position x_t de la pointe.

Nous définissons $F_x = C_x (x_M - x_t)$ comme étant la force latérale et $\langle F_x \rangle$ moyenne temporelle de F_x comme la force de frottement.

Dans le cas de vitesse du mobile très petit ($v_M \rightarrow 0$) la pointe sera toujours dans sa position d'équilibre, et on résout l'équation pour ($\ddot{x}_t = 0, \dot{x}_t = 0$). Nous allons d'abord montrer que dans ce cas, le mouvement de la pointe va changer dans de grandes proportions lorsque :

$$C_x < - \left[\frac{d^2 V}{dt^2} \right]_{min}$$

L'équation du mouvement nous donne donc :
$$C_x (x_M - x_t) - v_0 \frac{2\pi}{a} \sin\left(\frac{2\pi}{a} x_t\right) = 0$$

Le mouvement varie de façon importante si la variation (c'est à dire la dérivé par rapport à la position) de la force due au potentiel est grande par rapport à la variation de la force due au ressort, puisque sans le potentiel périodique la force de rappel du ressort serait quasi constante et le mouvement serait continu d'où :

$$\frac{d}{dx_t} C_x (x_M - x_t) < \frac{d}{dx_t} \left(V_0 \frac{2\pi}{a} \sin\left(\frac{2\pi}{a} x_t\right) \right) \Leftrightarrow -C_x < V_0 \left(\frac{2\pi}{a}\right)^2 \cos\left(\frac{2\pi}{a} x_t\right) \Leftrightarrow C_x < - \left[\frac{d^2 V}{dt^2} \right]_{min}$$

Lors que cette condition est remplie, le mouvement de la pointe est discontinu, alternant les phases de quasi-immobilité et de déplacement rapide (stick-slip en anglais). Dans cette approche quasi-statique, il n'y a pas de force de frottement qui ne va apparaître que pour des vitesses élevées $v_M \gg 0$ où le mouvement de la pointe sera dominé par le terme d'amortissement, et où en première approximation la force de frottement $F_{fric} \approx \gamma v_M$

Par la suite, nous allons utiliser, pour modéliser ce phénomène, les paramètres suivant :

- $V_0 = 1,0 \text{ eV}$
- $a = 3 \text{ \AA}$
- $C_x = 10,0 \text{ N.m}^{-1}$
- $m_x = 10^{-8} \text{ kg}$
- $\gamma = 10^{-3} \text{ N.s}^{-1}$

II. Méthodes numériques

Pour résoudre une équation différentielle du deuxième ordre, de la forme $\frac{d^2y}{dx^2} + \alpha \frac{dy}{dx} = r(x)$ comme notre équation (1), nous nous ramenons à un système d'équation du premier ordre :

$$\begin{aligned}\frac{dy}{dx} &= z(x) \\ \frac{dz}{dx} &= r(x) - \alpha z(x)\end{aligned}$$

Pour résoudre numériquement ce système, nous allons comparer trois méthodes qui se basent sur approximation du pas d'intégration.

1. La méthode d'Euler

C'est la méthode la plus simple. La série de Taylor nous donne en un point a pour une fonction f :

$$\sum \frac{f^{(n)}}{n!} (x-a)^n$$

On en déduit que $f(x+h) = f(x) + (x+h-x) \frac{f'(x)}{1!} + (x+h-x)^2 \frac{f''(x)}{2!} + \dots + (x+h-x)^n \frac{f^{(n)}}{n!}$

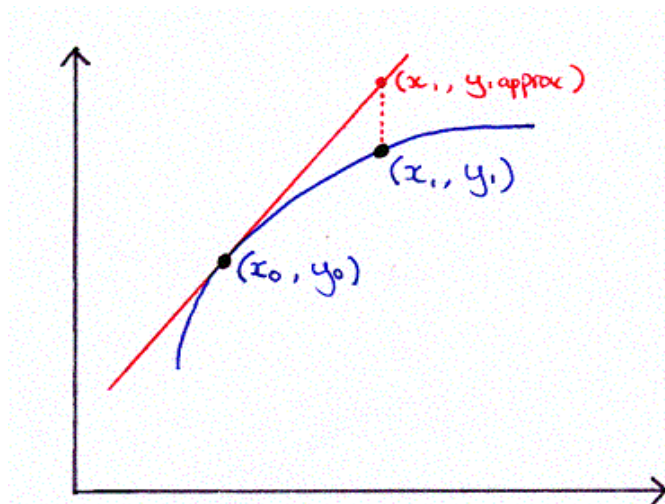
Soit $f(x+h) = f(x) + hf'(x) + o(h^2)$, $o(h^2)$ étant le terme d'erreur à l'ordre 2.

Lorsque nous allons faire notre programme pour résoudre une ou plusieurs équations avec la méthode d'Euler, nous allons négliger le terme d'erreur d'ordre 2.

On calcule alors successivement les valeurs de notre fonction de la manière suivante :

$$y_{i+1} = y_i + hf(x_i, y_i)$$

Si l'on connaît les valeurs initiales $(x_0; y_0)$ on peut alors calculer toutes les valeurs successives y_i avec un pas $h = x_{i+1} - x_i$



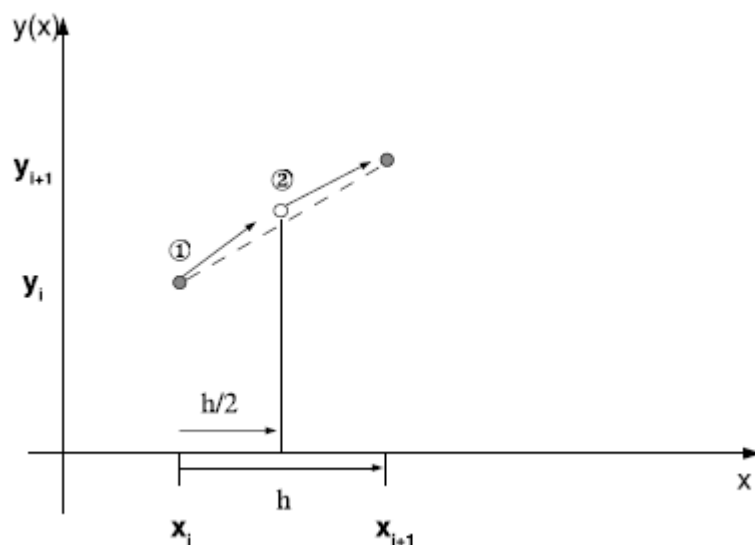
2. La méthode de Runge Kutta à l'ordre 2

La méthode de Runge Kutta à l'ordre 2 (RK2) est basée sur le même principe que la méthode d'Euler, à la différence près que l'on utilise la dérivée au point de départ x_i pour trouver un point intermédiaire. On utilise ensuite la dérivée en ce point pour trouver la valeur suivante de la fonction :

$$k_1 = h \cdot f(x_i, y_i)$$

$$k_2 = h \cdot f(x_i + h/2, y_i + h/2)$$

$$y_{i+1} = y_i + k_2 + o(h^3)$$



C'est une méthode d'ordre 2, l'erreur est donc d'ordre 3. Nous la comparerons ensuite à la méthode d'Euler.

3. La méthode de Runge Kutta à l'ordre 4

La méthode de Runge Kutta à l'ordre 4 (RK4) est semblable à la méthode RK2 mais elle est plus précise puisque l'on utilise quatre points (ou estimations), au lieu de deux points pour la méthode RK2 et un seul pour la méthode d'Euler. L'erreur de cette méthode est donc d'ordre 5.

$$k_1 = h \cdot f(x_i, y_i) \quad (\text{Euler})$$

$$k_2 = h \cdot f(x_i + \frac{h}{2}, y_i + \frac{h}{2}) \quad (\text{RK2})$$

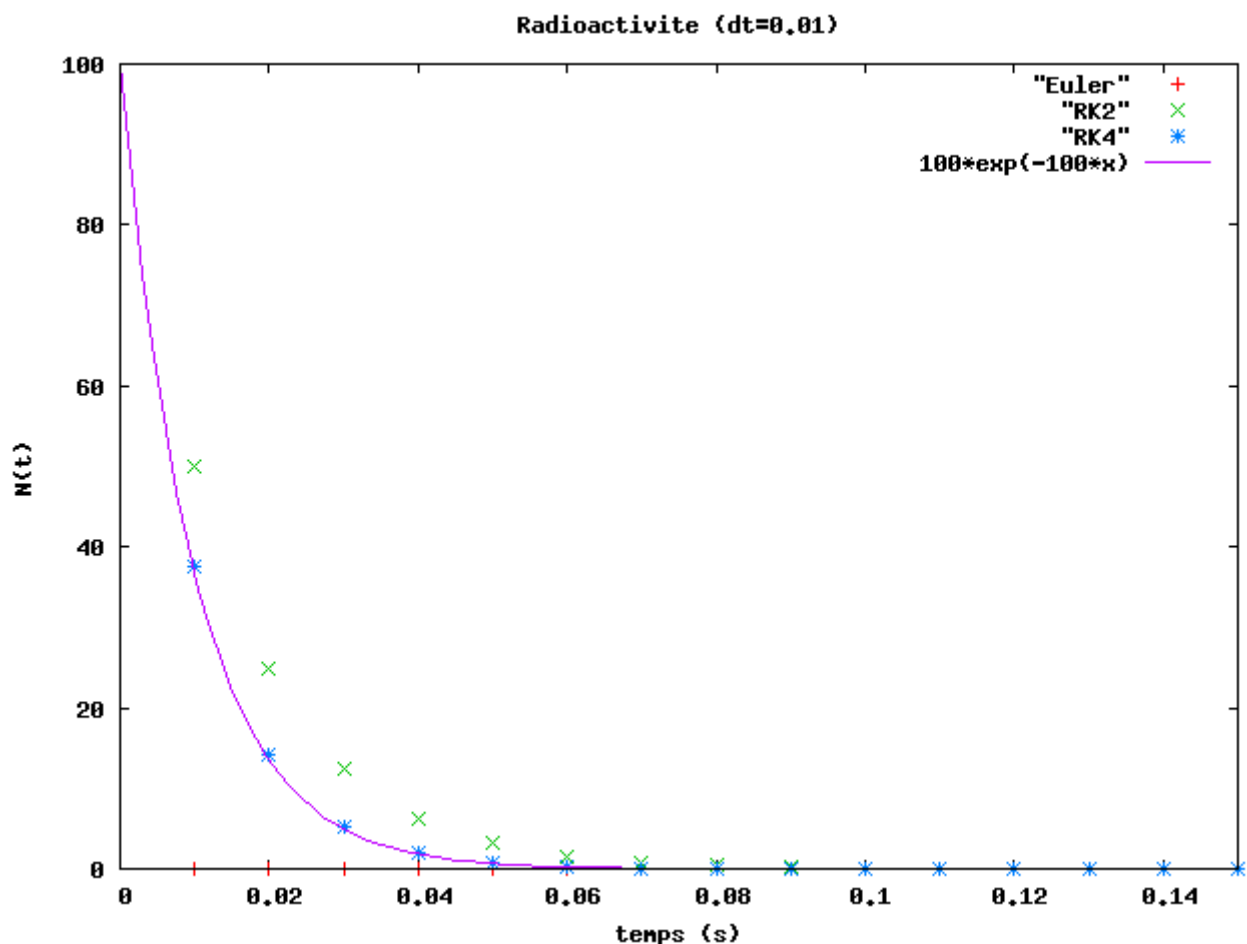
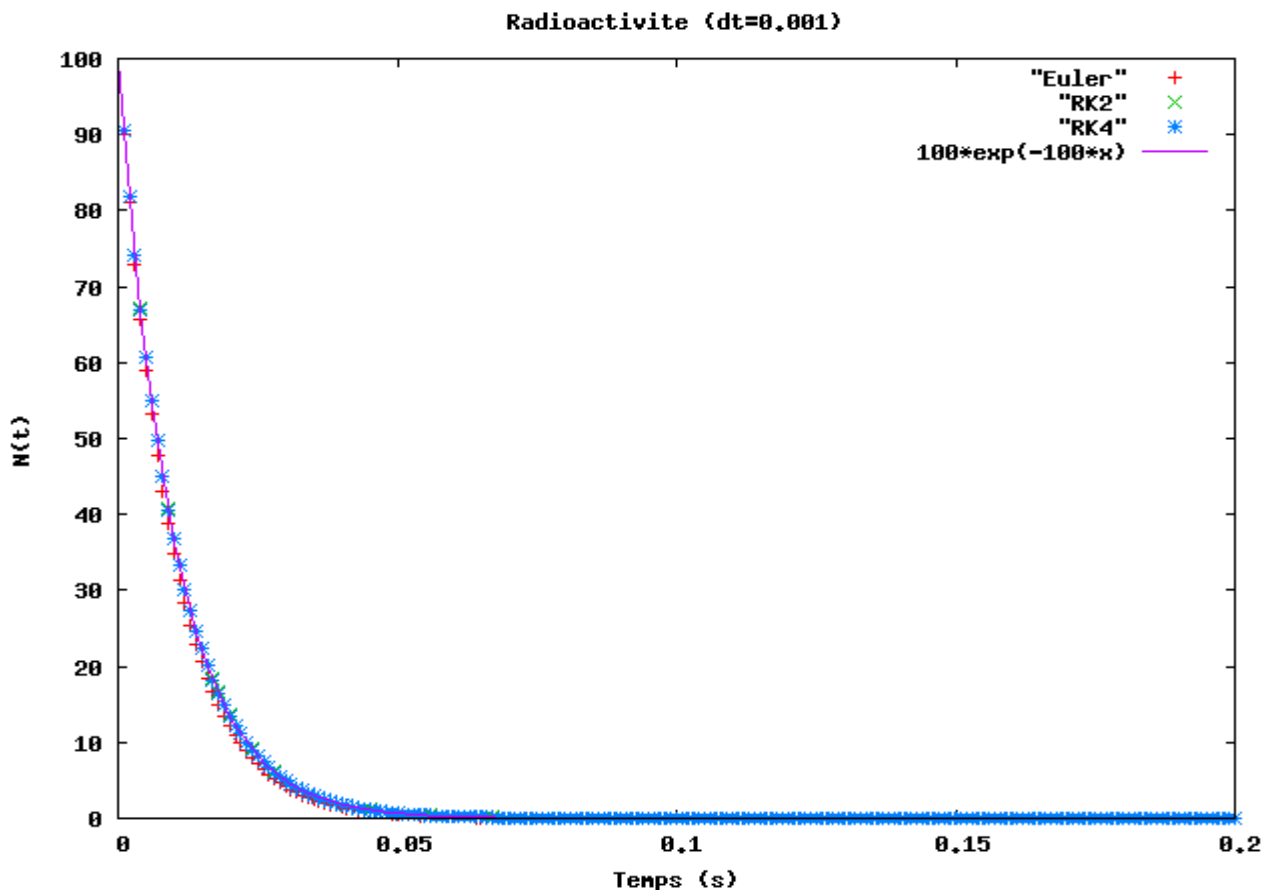
$$k_3 = h \cdot f(x_i + \frac{h}{2}, y_i + \frac{k_1}{2})$$

$$k_4 = h \cdot f(x_i + h, y_i + k_3)$$

$$y_{i+1} = y_i + \frac{k_1}{6} + \frac{k_2}{3} + \frac{k_3}{3} + \frac{k_4}{6} + o(h^5)$$

4. Précision des méthodes

Il est possible de calculer mathématiquement les erreurs des différentes méthodes mais en plus d'être longs, ces calculs ne nous intéressent pas vraiment ici. Nous allons comparer les résultats obtenus avec ces trois méthodes et les comparer au résultat exact sur un exemple simple : la décroissance radioactive. Il s'agit en fait de résoudre une équation différentielle du premier degré.



Nous utilisons les *programmes 1,2 et 3* (voir annexe) pour comparer les trois méthodes. Avec un petit pas, les méthodes RK2 et RK4 sont très proches du résultat exact alors qu'avec un pas un plus grand, la méthode RK4 est la seule méthode précise. C'est celle que nous allons utiliser.

II. Application à l'AFM

1. Mouvement de la pointe à 1 dimension

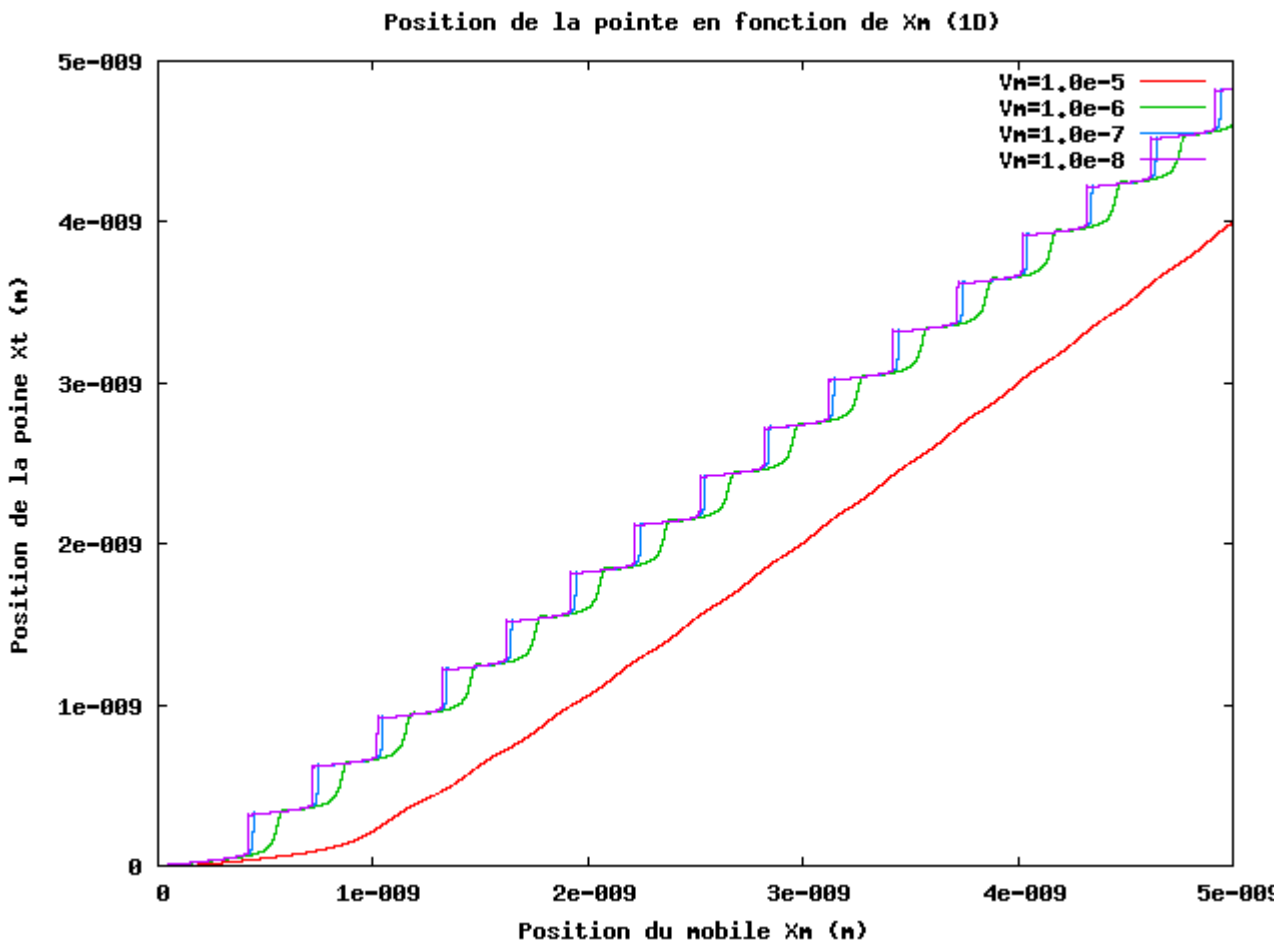
a) Position

Nous allons résoudre l'équation différentielle (1), à l'aide d'un programme utilisant la méthode RK4.

Nous utiliserons un pas $dt=1,0e-7$ secondes.

Le *programme 4* va résoudre l'équation différentielle et afficher la position de la pointe, solution de l'équation différentielle, la vitesse de la pointe, dérivée par rapport au temps de la position de la pointe, ainsi que la position du mobile pour un vitesse de mobile donnée. Nous allons ensuite afficher, grâce au logiciel *gnuplot*, sous forme graphique la position du mobile en fonction de la position du mobile pour différentes vitesses de mobile.

Voici les résultats obtenus :

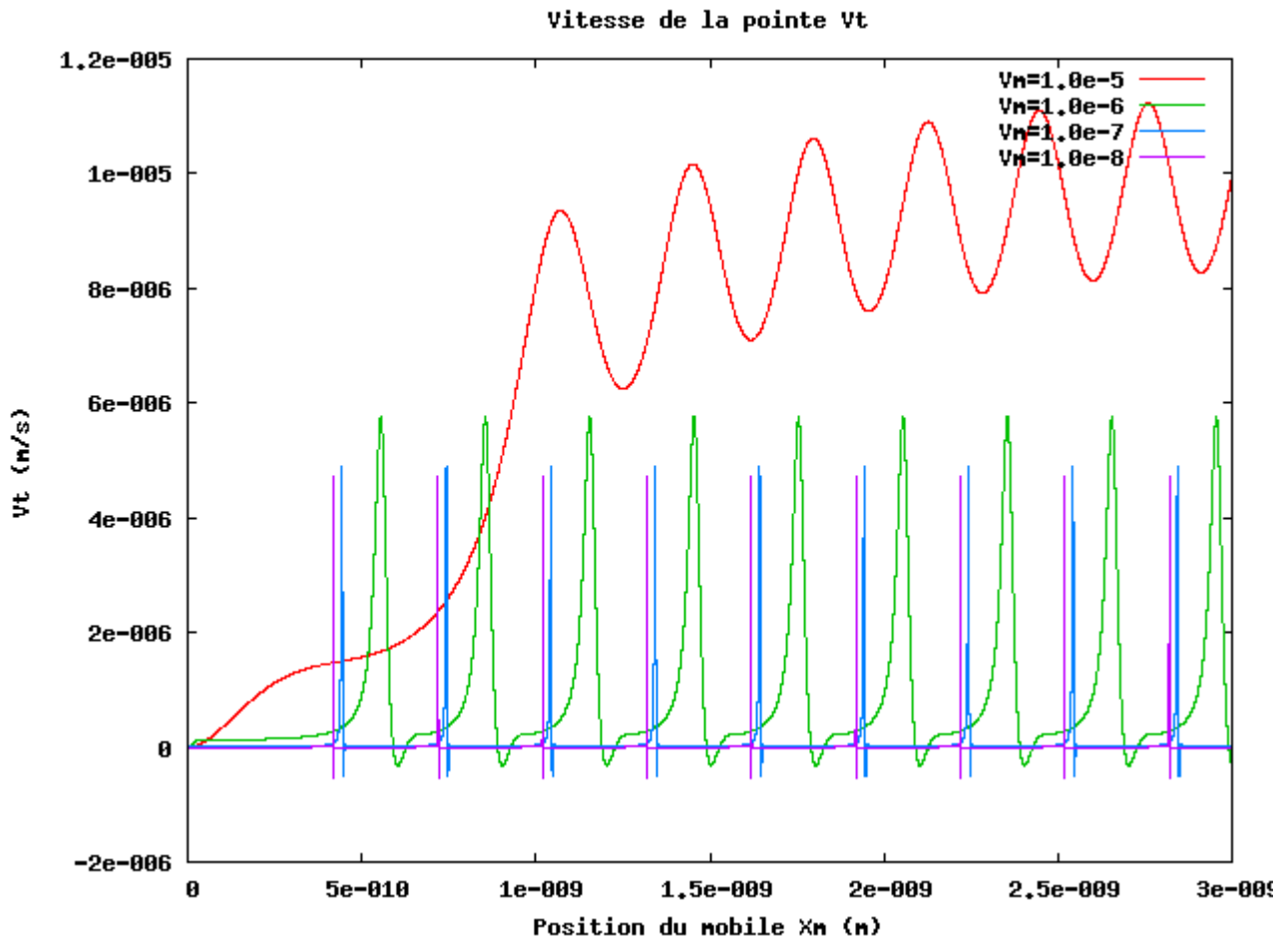


Pour des vitesses de mobile $V_m=1,0e-7$ m/s et inférieures, le mouvement est saccadé et oscille entre phase de quasi immobilité et phase de mouvement très rapide. Pour des vitesses $V_m=1,0e-5$ m/s et supérieures, le mouvement de la pointe est continu, semblable au mouvement du mobile.

Pour $V_m=1,0e-6$ m/s, le mouvement de la pointe est saccadé mais dans une moindre mesure que pour des vitesses de mobile inférieures, nous sommes donc dans un régime transitoire entre mouvement saccadé « Stick-Slip » et mouvement continu.

b) Vitesse

Il est aussi intéressant, afin de mieux comprendre ce mouvement, d'utiliser la vitesse de la pointe donnée par le *programme 4* et l'afficher par rapport à la position du mobile :

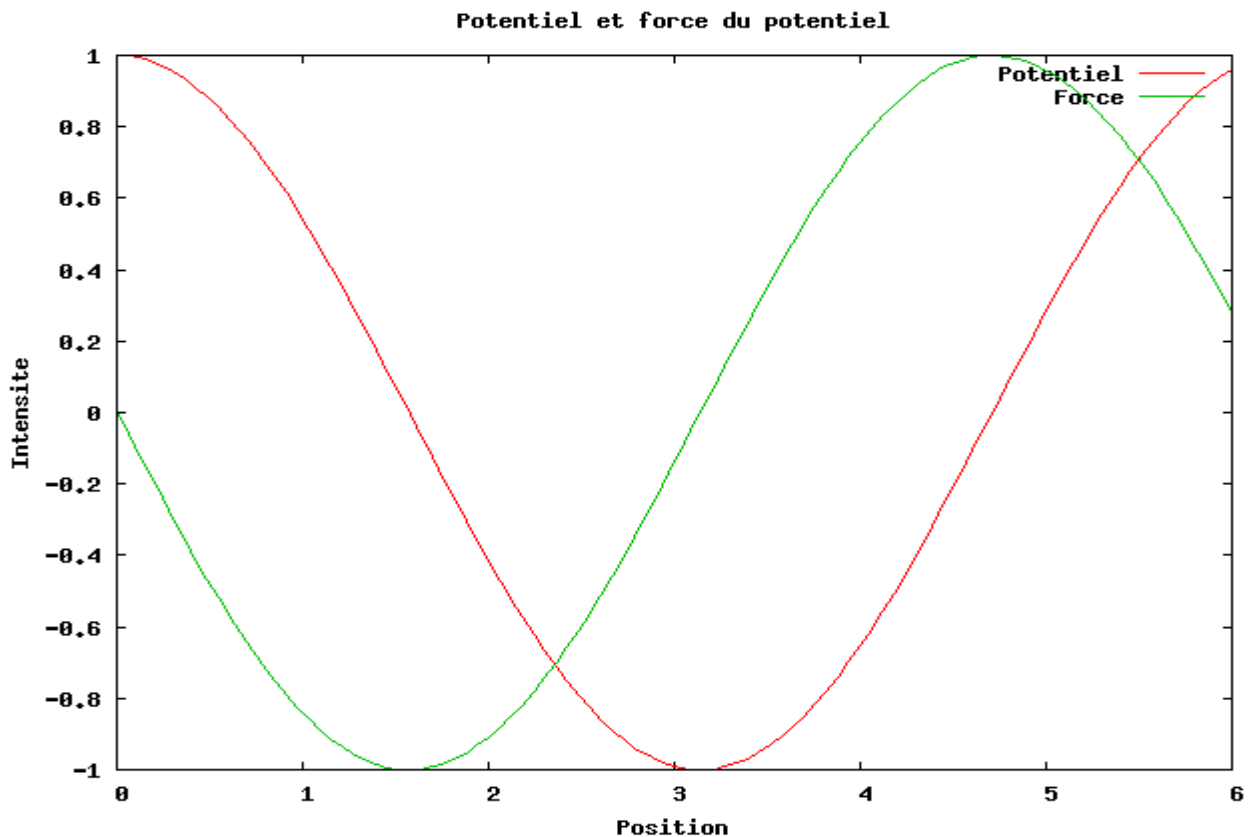


Ce graphe nous permet de comprendre le mouvement saccadé « Stick-Slip » qui alterne entre vitesses élevées et vitesses quasi nulles. Il permet aussi de voir que le mouvement de la pointe pour la vitesse de mobile $V_m = 1,0 \times 10^{-5}$ m/s n'est pas complètement continu comme l'on pouvait le penser en regardant le graphe précédent.

Enfin, ici aussi, à $V_m = 1,0 \times 10^{-6}$ m/s, nous sommes dans un régime transitoire entre un mouvement saccadé et un mouvement quasi continu.

c) Mouvement saccadé

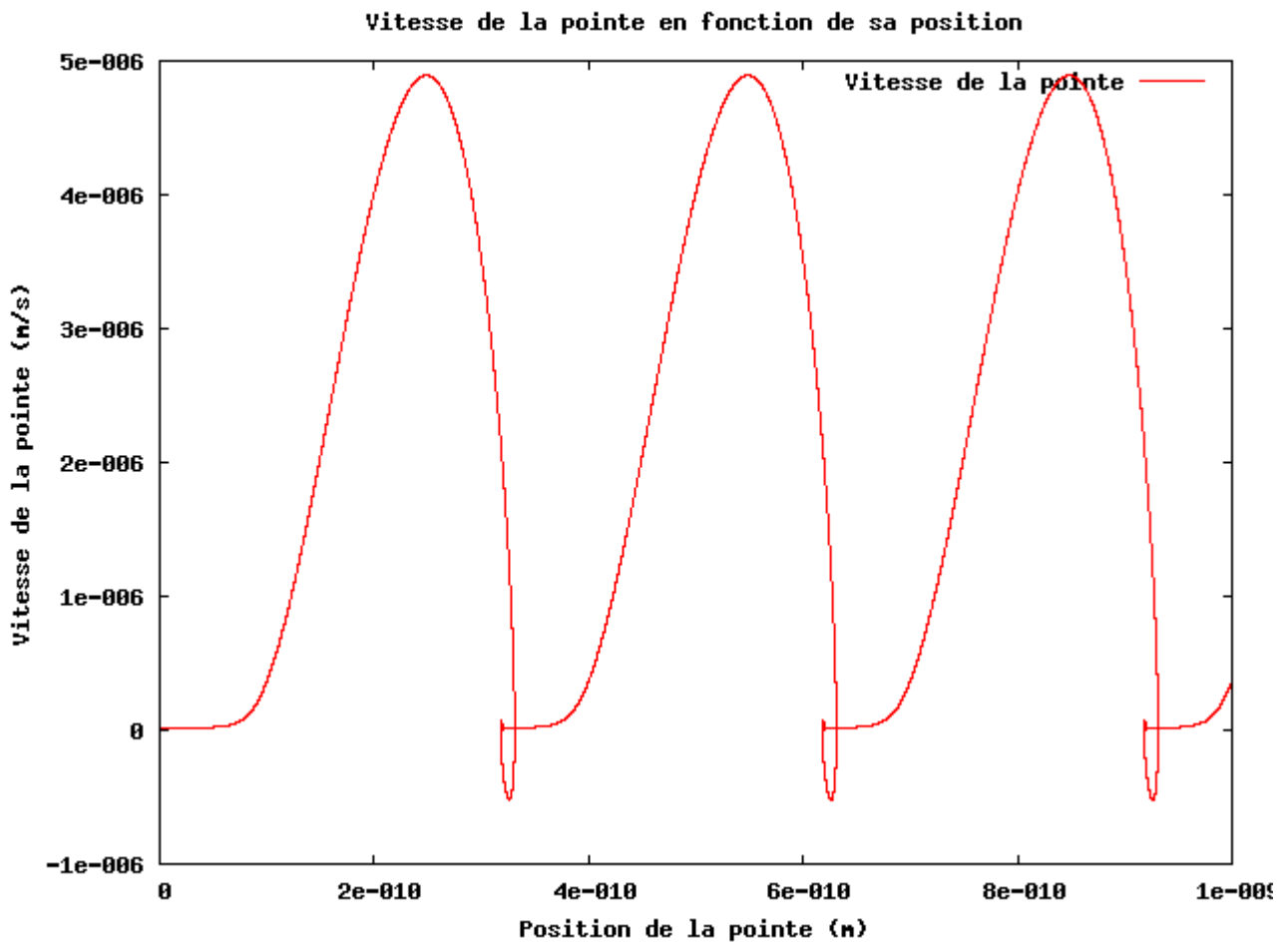
Pour mieux comprendre l'origine du mouvement saccadé, le graphe de la page suivante représente les fonctions **cosinus** pour le potentiel et **-sinus** pour la force due au potentiel.



Lorsque la pointe se trouve dans la première moitié de la distance entre deux atomes, la force du potentiel est négative et s'oppose à la force de rappel du ressort, la pointe va donc se trouver en équilibre aux alentours de $\frac{1}{4}$ de la distance inter-atomique, qui est l'endroit où la force d'opposition au mouvement due au potentiel est maximale. Le mobile continuant son mouvement, la force de rappel du ressort va augmenter, faisant légèrement avancer la pointe ce qui a pour conséquence de diminuer la force de rétention du potentiel puisque la pointe s'éloigne de la zone maximale. La pointe va accélérer brusquement lorsqu'elle va entrer dans la deuxième moitié de la distance inter-atomique, la force du potentiel s'ajoutant alors à la force du ressort, et atteignant son maximum à $\frac{3}{4}$ de la distance entre atomes; c'est à cet endroit là que l'on observe le pic de vitesse visible sur le graphe ci-dessus.

Une fois ce pic passé, la vitesse va décroître aussi brusquement qu'elle augmenta. La distance mobile-pointe ayant diminué et la force du potentiel diminuant aussi avec l'avancement de la pointe, la pointe va retrouver l'état d'équilibre de départ jusqu'au prochain saut. C'est le mouvement saccadé « Stick-Slip ».

L'impulsion donnée par le saut est tellement importante, que la pointe va même dépasser légèrement la position d'équilibre. La force de rappel étant devenu très faible, le potentiel va ramener la pointe dans sa position d'équilibre. C'est ce phénomène qui est à l'origine des vitesses négatives observables dans le cas du mouvement saccadé.



La figure ci-dessus représente la vitesse de la pointe en fonction de la position de celle-ci. Cette figure nous montre bien la zone d'équilibre autour de $\frac{1}{4}$ de la distance qui sépare deux atomes, et le pic de vitesse à $\frac{3}{4}$ de cette même distance

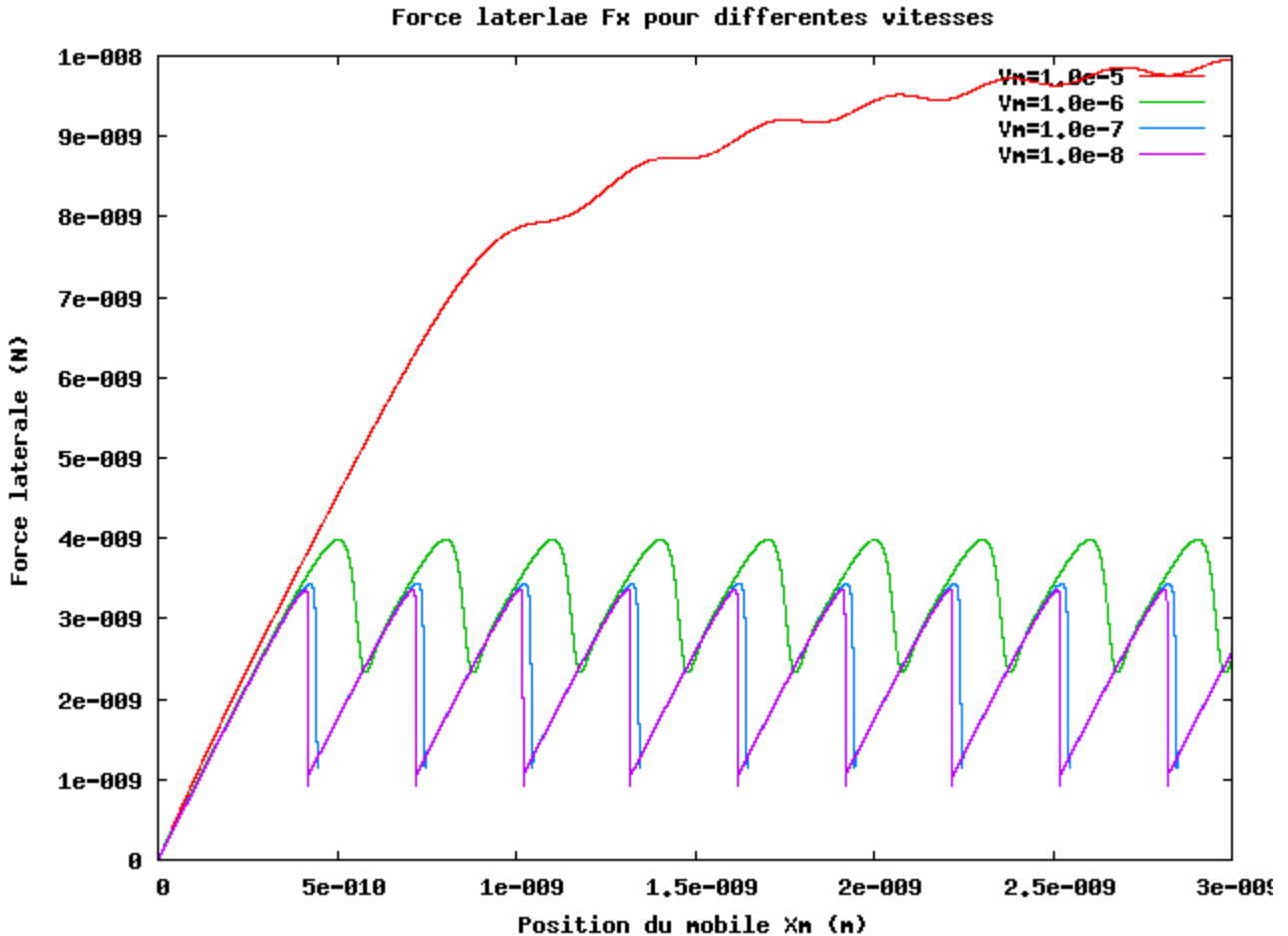
Pourquoi ce mouvement saccadé s'estompe-t-il à des vitesses de mobile plus rapides ?

Il est un peu tôt pour y répondre mais il apparaît que lorsque la vitesse du mobile dépasse la vitesse de glisse, la force de rétention du potentiel n'est plus assez forte pour s'opposer à la force de rappel, elle ne pourra que freiner la pointe, et non plus la maintenir en équilibre : les saccades disparaissent.

Au lieu d'avoir un mouvement alternant phase de quasi immobilité et phase de mouvement extrêmement rapide, la pointe va avoir, une fois amorcé, un mouvement oscillatoire tantôt moins rapide, tantôt plus rapide.

2. Force latérale et force de friction

Nous avons défini dans la présentation les forces latérales F_x et de frottement $F_{fric} = \langle F_x \rangle$. Le programme 5 va calculer la force latérale pour différentes vitesses de mobile V_m .



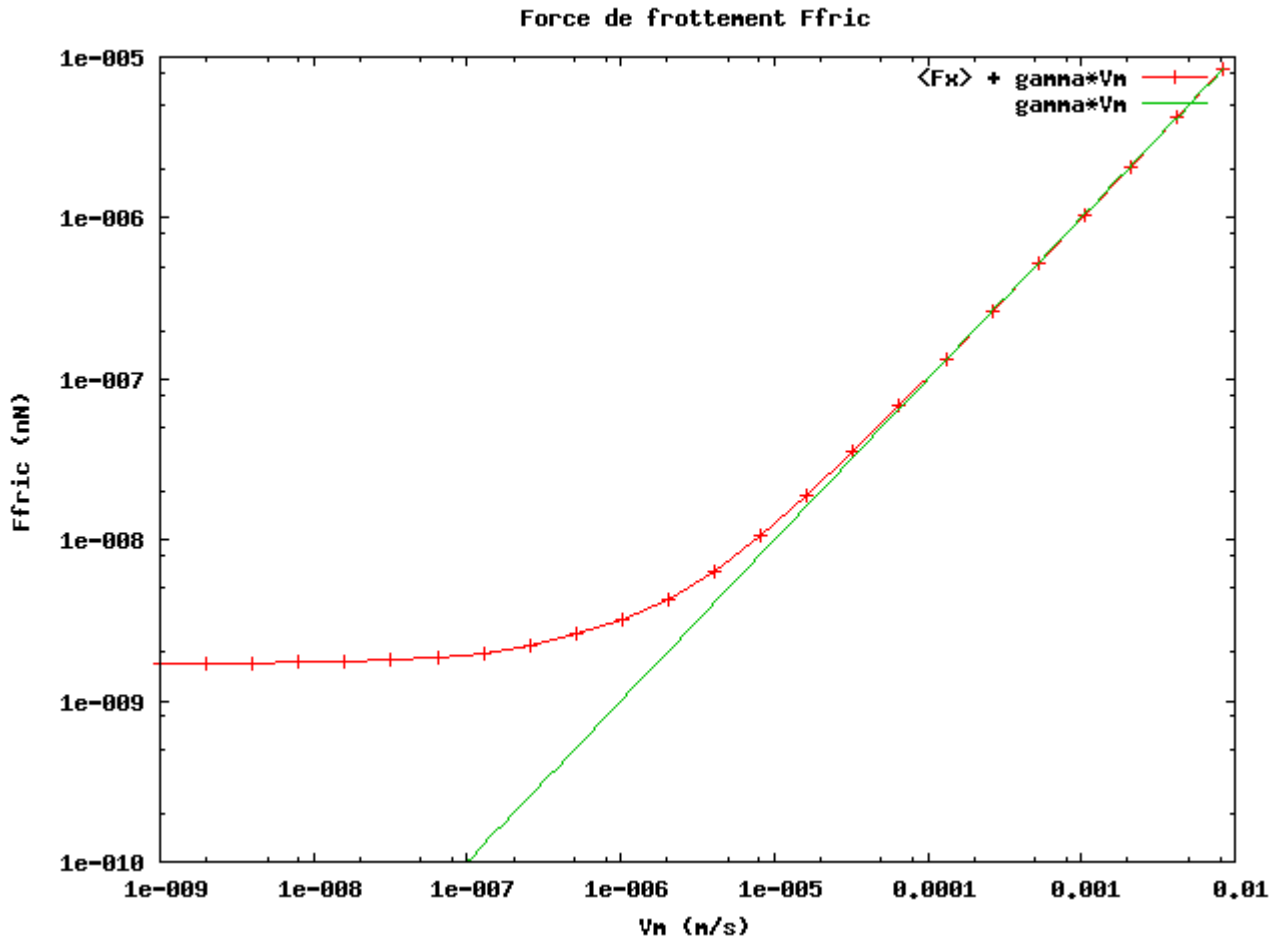
Pour des vitesses de mobile très lentes, inférieures $V_m = 1,0e-6$ m/s, la force latérale varie très peu (pour des vitesses V_m différentes) alors que cette même force augmente significativement pour des vitesses de mobile plus hautes comme $V_m = 1,0e-5$ m/s. Nous allons voir pourquoi.

Pour comprendre ce changement de la force latérale qui s'effectue autour de $V_m = 1,0e-6$ m/s, vitesse à laquelle la vitesse du mobile devient supérieure à la vitesse de glisse, et qui affecte comme nous l'avons vu précédemment le mouvement, la vitesse et maintenant la force latérale de la pointe, nous allons utiliser le programme 6 qui est un peu plus compliqué que les autres.

Ce programme va calculer $\langle F_x \rangle$, moyenne de la force latérale F_x , pour plusieurs valeurs de V_m .

Étant donné que la distance parcourue est égale à $D = V_m * t = V_m * i * dt$, et que la distance parcourue doit être la même pour chaque valeurs de V_m , il faudra que le nombre d'itérations varie avec V_m car la distance D doit rester constante pour ne pas fausser la moyenne.

Voici les résultats obtenus pour 24 valeurs de V_m différentes :



Ce graphe nous permet de voir ce que la vitesse du mobile reste faible, la force de frottement est indépendante de la vitesse du mobile. A partir de $V_m=1,0e-6$ m/s c'est n'est plus le cas, le terme d'amortissement dépendant de la vitesse commence à prendre le dessus sur la force latérale.

La force latérale s'opposant à la force d'amortissement, elle va augmenter exponentiellement avec elle et tendre vers sa valeurs car le potentiel deviendra alors négligeable. Il ne sera plus assez puissant pour maintenir la pointe dans un état d'équilibre. Le mouvement saccadé va alors disparaître comme nous l'avons vu plus haut.

Quelle est l'origine de l'indépendance de la force de frottement par rapport à la vitesse du mobile lorsque cette dernière est inférieure à $1,0e-6$ m/s ?

Comme nous l'avons vu dans c), dans le cas du mouvement saccadé la pointe reste dans un état quasi immobile lorsqu'elle se trouve dans la zone inter-atomique, ne dissipant alors presque pas d'énergie puisque le terme d'amortissement est proportionnel à la vitesse. Par contre, pendant le saut, la pointe atteint une vitesse très importante d'où une dissipation d'énergie mais l'énergie dissipée totale ne change pas significativement tant que la vitesse mobile reste inférieure à la vitesse de la pointe (pour $V_m=1,0e-7$ m/s, $V_{pointe}=5,0e-6$ m/s pendant le saut).

Pour des vitesses plus élevés, le mouvement « Stick-Slip » disparaît, donc la force de frottement devient essentiellement proportionnelle au terme d'amortissement γV_m .

3. Modèle à deux dimensions

Pour aller plus loin, on peut étudier le modèle à 2 dimensions. Nous prendrons une vitesse de mobile $\mathbf{V_m}=1,0\text{e-}7 \text{ m/s}$ qui implique un mouvement saccadé.

Le potentiel périodique est selon x et y : $V(x_t, y_t) = V_0 \cos\left(\frac{2\pi}{a_x} x_t\right) \cos\left(\frac{2\pi}{a_y} y_t\right)$.

Il faut alors résoudre un système de deux équations différentielles :

$$m_x \ddot{x}_t = C_x(x_M - x_t) - \frac{dV(x_t, y_t)}{dx_t} - \gamma_x \dot{x}_t$$

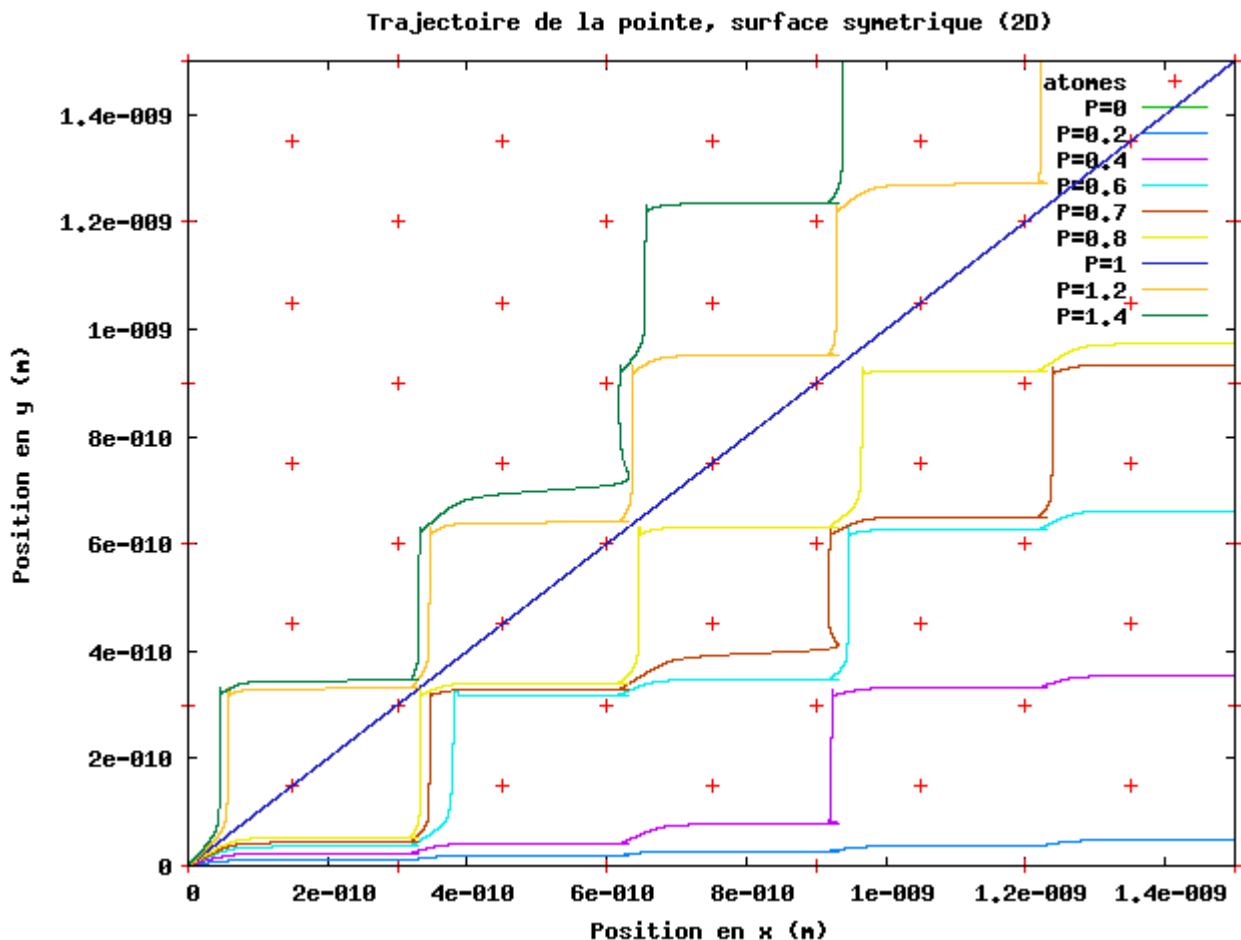
$$m_y \ddot{y}_t = C_y(y_M - y_t) - \frac{dV(x_t, y_t)}{dy_t} - \gamma_y \dot{y}_t$$

a) Surface symétrique

Nous allons d'abord étudier le modèle à deux dimensions dans le cas d'une surface symétrique, c'est à dire que nous avons une surface où $a_x = a_y$.

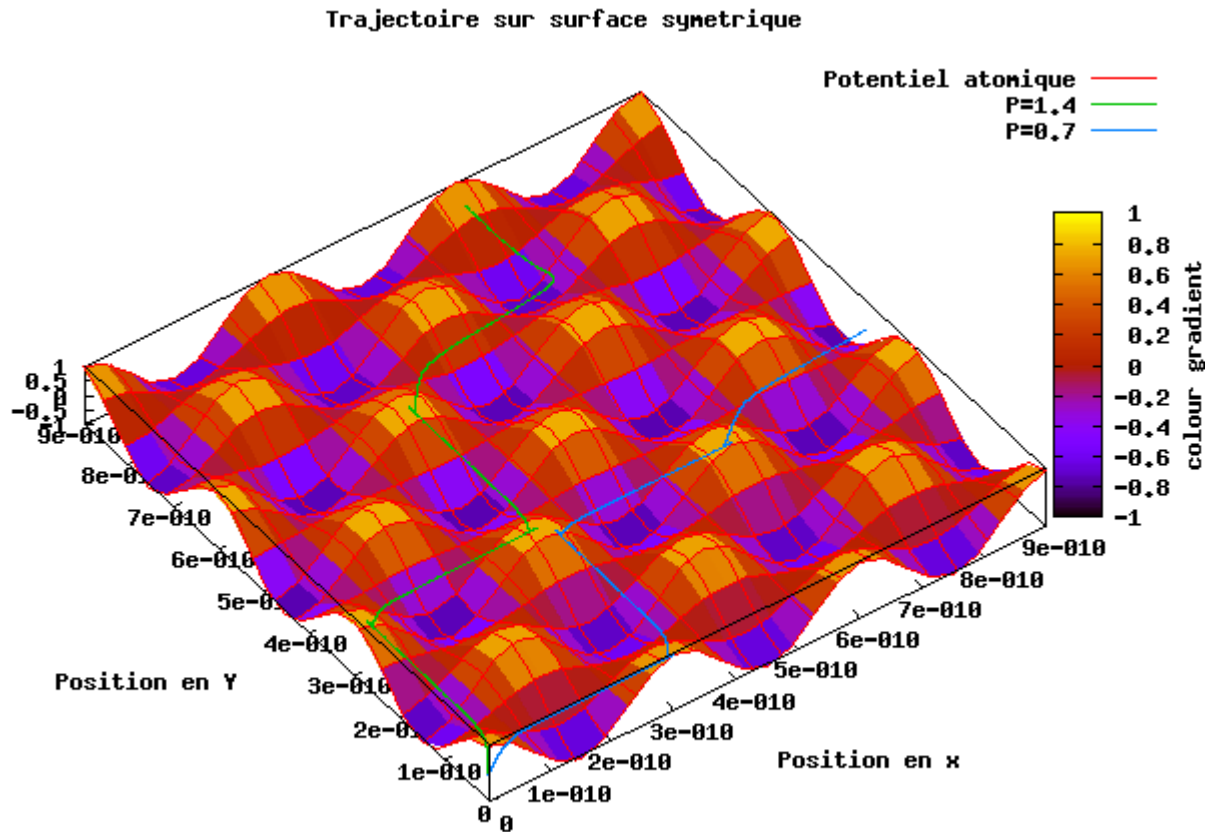
Le programme 7 va calculer la trajectoire de la pointe pour ce modèle à deux dimensions. Afin de montrer dans quelles proportions les trajectoires dépendent des conditions à l'origine, nous allons faire plusieurs simulations avec des vitesses de mobiles selon l'axe y différentes. La vitesse $\mathbf{V_m}$ selon l'axe x restera constante tandis que nous allons faire varier la vitesse $\mathbf{V_m}$ selon l'axe y. Soit P

le rapport $\frac{V_{my}}{V_{mx}}$.



La figure précédente montre que les trajectoires dépendent beaucoup des conditions initiales, pour différentes valeurs de P la trajectoire change complètement. Pour $P=0$ ou 1 , la trajectoire est semblable au modèle à une dimension.

Voici maintenant une représentation en trois dimensions de la trajectoire de la pointe sur la même surface symétrique pour $P=0,7$ et $P=1,4$:



La représentation en trois dimensions sur le logiciel *gnuplot* est quelquefois un peu trompeuse, puisque le potentiel est transparent; le tracé qui se trouve derrière un pic de potentiel peut paraître à l'intérieur de ce pic.

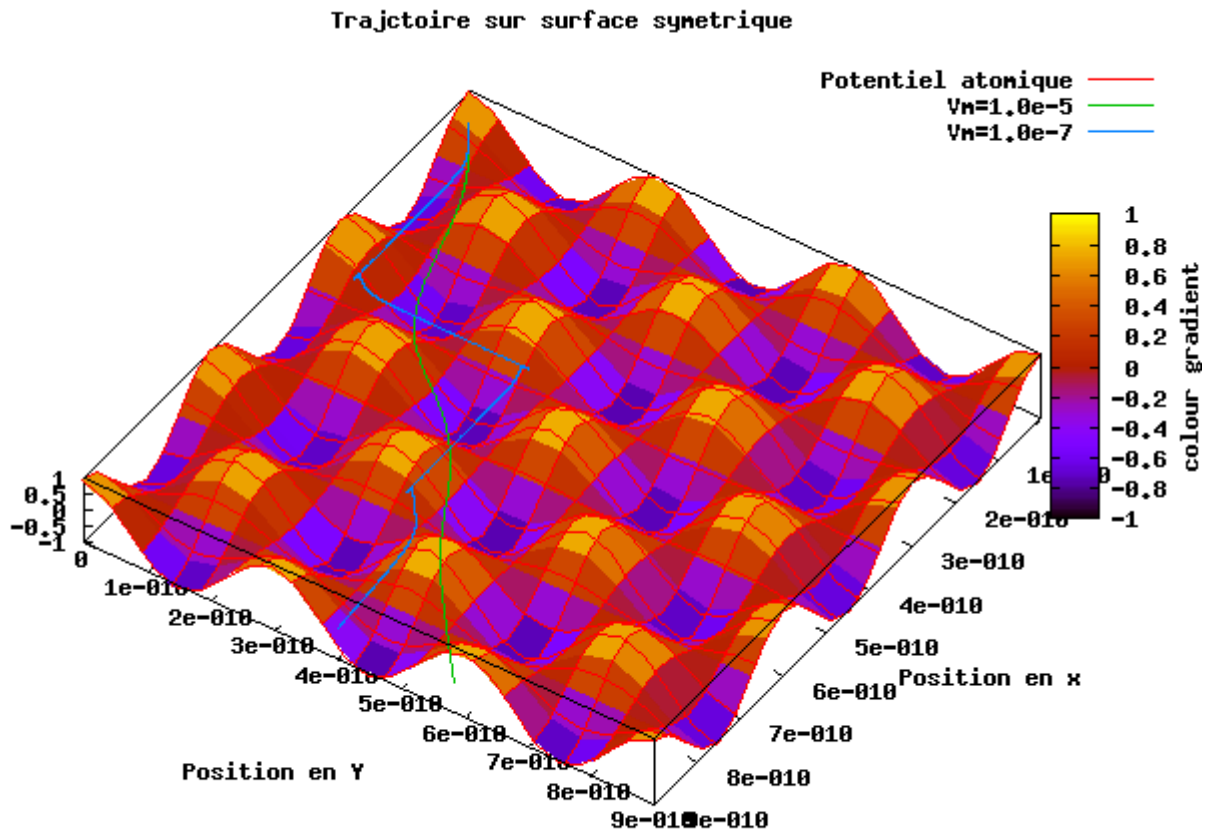
Néanmoins, cette représentation est pratique pour bien visualiser le potentiel et ainsi confirmer ce que nous avons dit pour le modèle à une dimension dans **1. c)**.

Malgré la trajectoire très changeante de la pointe, elle semble toujours tendre vers les zones où le potentiel de surface est proche de 0, à $\frac{1}{4}$ et $\frac{3}{4}$ de la distance inter-atomique.

Dans le modèle à une dimension, la pointe était « retenue » par le potentiel dans une zone autour de $\frac{1}{4}$ de la distance entre les atomes, puis très fortement accélérée dans la zone autour de $\frac{3}{4}$ de la distance a , la force du potentiel s'ajoutant à la force de rappel du ressort.

Ici, la pointe ayant la liberté de se déplacer selon deux axes, elle va naturellement prendre le chemin où la force s'opposant au mouvement est la plus faible, même si elle doit pour cela prendre un chemin plus long et qui s'oppose à la force de rappel qui entraîne la pointe, à la manière d'un cours d'eau rencontrant un obstacle, il lui sera plus facile de le contourner que de le surmonter, tant que la force d'opposition est supérieure à celle qui crée le mouvement, le mouvement sera de type « zig-zag » comme le notre.

Est ce que pour des vitesses de mobile supérieures le mouvement « zig-zag » diminuera comme nous le pensons ? Nous allons le voir en comparant les résultats obtenus pour $V_m=1,0e-7$ m/s et $V_m=1,0e-5$ à $P=0,7$ constant.



C'est le cas ! Pour une vitesse plus élevée, la pointe prend le chemin le plus court, celui « dicté » par le mobile, « ignorant » le potentiel. Comme pour le modèle à une dimension où le mouvement saccadé disparaît lorsque la vitesse du mobile augmente, ici c'est le mouvement « zig-zag » qui tend à disparaître. Ceci pour la même raison : plus la vitesse V_m devient importante, plus la force d'entraînement de la pointe devient importante face à la force du potentiel qui joue alors un rôle moindre dans le mouvement de la pointe.

b) Surface asymétrique

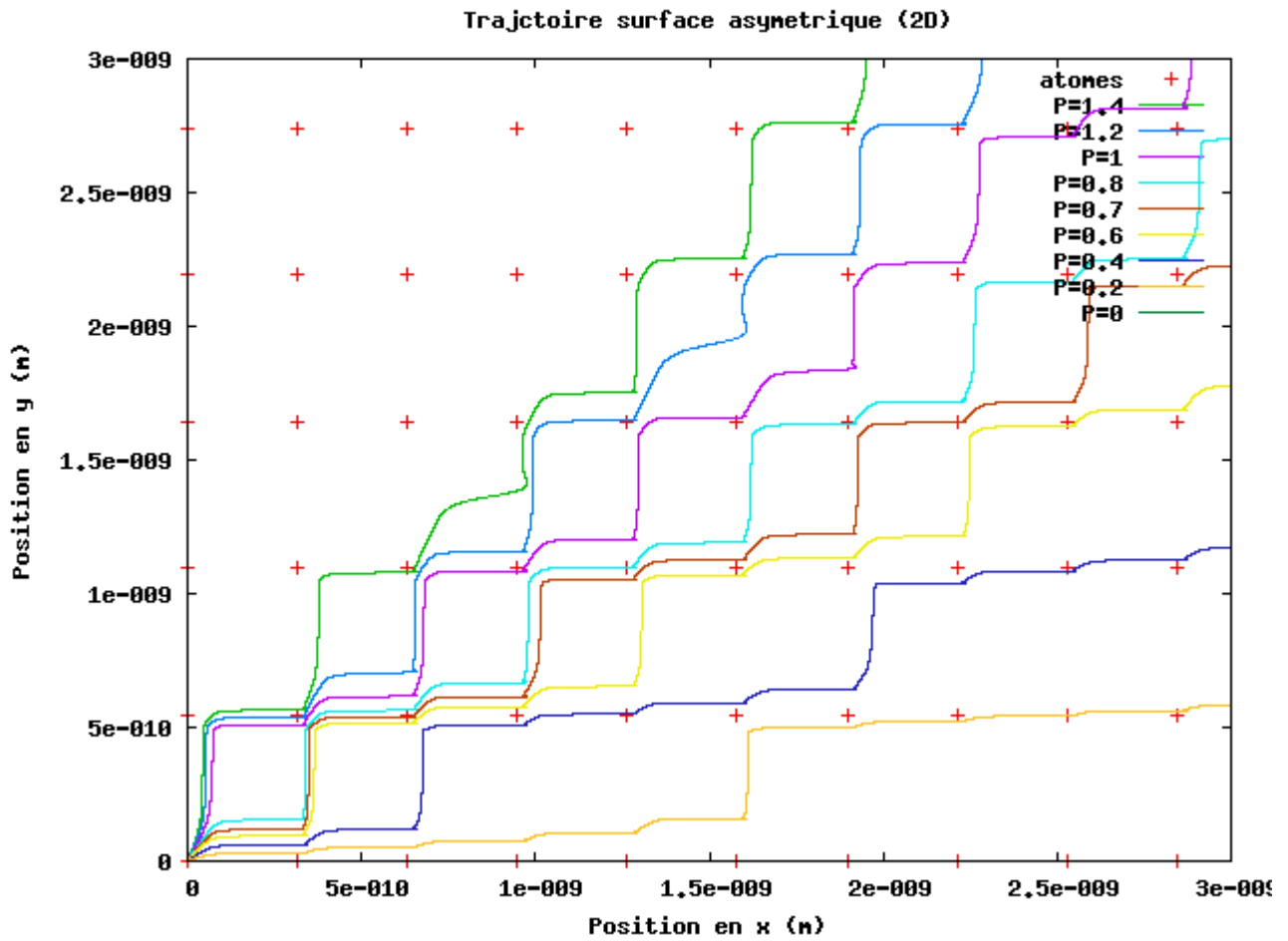
Nous allons calculer dans cette partie la trajectoire de la pointe dans le modèle à deux dimensions, mais pour une surface asymétrique, c'est à dire une surface où la distance entre les atomes selon l'axe x et différente de la distance entre les atomes selon l'axe y, c'est à dire où a_x et a_y sont différents.

Nous prendrons comme valeurs :

$$a_x = 3,16 \text{ Angstrom}$$

$$a_y = 5,48 \text{ Angstrom}$$

Tout les autres paramètres restent les mêmes, et la simulation sera faite pour $V_m=1,0e-7$ m/s.



Dans ce cas, les trajectoires dépendent encore plus des conditions initiales P , et à $P=1$, nous n'avons plus une trajectoire rectiligne semblable au modèle à une dimension.

III. Conclusion

Nous avons modélisé de manière réaliste, grâce à la simulation informatique, le modèle d'un microscope à force atomique d'abord à une puis à deux dimensions.

Il est apparu que pour des vitesses très faibles, le mouvement de la pointe de l'AFM est un mouvement saccadé « Stick-Slip » ou en « zig-zag » pour le cas à deux dimensions.

Pour de tels vitesses, nous avons montré que la force de frottement était indépendante de la vitesse du mobile, validant ainsi la troisième loi d'Amontons-Coulomb au niveau microscopique pour ces vitesses.

Nous avons aussi montré l'origine du phénomène « Stick-Slip », et aussi que ce phénomène existe aussi dans le modèle en deux dimensions sous forme de mouvement « zig-zag » pour les mêmes raisons, et qui disparaît lui aussi lorsque la vitesse du mobile atteint un certain seuil.

Lorsque la vitesse du mobile devient suffisamment grande et dépasse la vitesse de glisse de la pointe, le mouvement saccadé disparaît, ayant pour conséquence une dissipation d'énergie accrue dominée par le terme d'amortissement qui dépend de la vitesse. La force de frottement n'est alors plus indépendante de la vitesse du mobile et la troisième loi d'Amontons-Coulomb n'est plus valide.

Enfin, dans le modèle à deux dimensions, nous avons montré que la trajectoire de la pointe, dépend fortement des conditions à l'origine, surtout dans le cas d'une surface asymétrique.

V. Annexe des programmes

```
// PROGRAMME 1
```

```
#include <stdio.h>
#define k 100
#define dt 0.001 // Pas d'intégration

double fn(double t, double N)
{
double fonction1;
fonction1=-k*N;
return fonction1;
}

int main(void)
{

FILE *fichier; // Export des résultats
fichier=fopen("Euler", "w");

double fn(double t, double N);
int i;
double N, t, K1x, K1v, K2x ,Kx3 , Kx4;
N=100.0;
t=0.0;
for(i=1; i<500; i++)
{
t=i*dt;
K1x=dt*fn(t, N);

N=N+K1x;

printf("%e %e \n", t, N);
fprintf(fichier,"%e %e \n", t, N);

}
return 0;
}
```

```
// PROGRAMME 2
```

```
#include <stdio.h>
#define k 100
#define dt 0.001

double fn(double t, double N)
{
double fonction1;
fonction1=-k*N;
return fonction1;
}

int main(void)
{

FILE *fichier; // Export des résultats
fichier=fopen("RK2", "w");

double fn(double t, double N);
int i;
double N, t, K1x, K1v, K2x ,Kx3 , Kx4;
N=100.0;
t=0.0;
for(i=1; i<500; i++)
{
t=i*dt;
K1x=dt*fn(t, N);

K2x=dt*fn(t+dt/2, N+K1x/2);

N=N+K2x;

printf("%e %e \n", t, N);
fprintf(fichier,"%e %e \n", t, N);

}
return 0;
}
```

```
// PROGRAMME 3
```

```
#include <stdio.h>
#define k 100
#define dt 0.001

double fn(double t, double N)
{
double fonction1;
fonction1=-k*N;
return fonction1;
}

int main(void)
{

FILE *fichier; // Export des résultats
fichier=fopen("RK4", "w");

double fn(double t, double N);
int i;
double N, t, K1x, K1v, K2x ,Kx3 , Kx4;
N=100.0;
t=0.0;
for(i=1; i<500; i++)
{
t=i*dt;
K1x=dt*fn(t, N);

K2x=dt*fn(t+dt/2, N+K1x/2);

Kx3=dt*fn(t+dt/2, N+K2x/2);

Kx4=dt*fn(t+dt, N+Kx3);

N=N+(K1x + 2*K2x +2*Kx3 +Kx4)/6.0;

printf("%e %e \n", t, N);
fprintf(fichier,"%e %e \n", t, N);

}
return 0;
}
```

//PROGRAMME 4

```
#include <stdio.h>
#include <math.h>
#include <stdlib.h>
#define V0 1.6e-19 // J (= 1.0 eV)
#define a 3.0e-10 // m (= 3 Anstrom)
#define dt 5.0e-6 // s
#define Cx 10.0 // N/m
#define Mx 1.0e-8 // kg
#define gamma 1.0e-3 // N/s
#define Vm 1.0e-10 // m/s
#define Pi 3.1415926
#define dim 2 // Modèle en 1D

void derive(double t, double Y[], double dY[])
{
dY[0]=Y[1];
dY[1]=(1/Mx)*(Cx*(Vm*t-Y[0])-V0*(2*Pi/a)*sin(2*Pi*Y[0]/a)-gamma*Y[1]);
}

int main(void)
{
int i,j;
double t;
double Y[dim]={0.0},
dY[dim]={0.0},
K1[dim]={0.0},
K2[dim]={0.0},
K3[dim]={0.0},
K4[dim]={0.0},
Ktemp[dim]={0.0};

Y[1]=0.0;
Y[0]=0.0;
t=0.0;

FILE *fichier; // Export des résultats
fichier=fopen("1D_10", "w");

for (i=0; i<1000000; i++)
{
t=i*dt;
printf("%e %e %e\n",Vm*t , Y[0], Y[1]); // Affiche les résultats
fprintf(fichier, "%e %e %e\n", Vm*t, Y[0], Y[1]); // Ecrit les résultats

derive(t, Y, dY);

for (j=0; j<dim; j++)
{
K1[j]=dt*dY[j];
Ktemp[j]=Y[j]+K1[j]/2.;
}

derive(t+dt/2, Ktemp, dY);

for (j=0; j<dim; j++)
{
K2[j]=dt*dY[j];
Ktemp[j]=Y[j]+K2[j]/2.;
}
}
```

```
derive(t+dt/2, Ktemp, dY);

for (j=0; j<dim; j++)
{
    K3[j]=dt*dY[j];
    Ktemp[j]=Y[j]+K3[j];
}

derive(t+dt, Ktemp, dY);

for (j=0; j<dim; j++)
{
    K4[j]=dt*dY[j];
    Y[j]=Y[j]+(1./6.)*(K1[j]+2*K2[j]+2*K3[j]+K4[j]);
}
}

return 0;
}
```


// PROGRAMME 5

```
#include <stdio.h>
#include <math.h>
#include <stdlib.h>
#define Pi 3.1415926
#define V0 1.6e-19 // J (= 1.0 eV)
#define a 3.0e-10 // m (= 3 Anstrom)
#define dt 5.0e-7 // s
#define Cx 10.0 // N/m
#define Mx 1.0e-8 // kg
#define gamma 1.0e-3 // N/s
#define dim 2 // Modèle en 1D (1 position et 1 vitesse)
#define Vm 1.0e-6 // m/s

void derive(double t, double Y[], double dY[])
{
dY[0]=Y[1];
dY[1]=(1/Mx)*(Cx*(Vm*t-Y[0])-V0*(2*Pi/a)*sin(2*Pi*Y[0]/a)-gamma*Y[1]);
}

int main(void)
{
int i,j,k;
double t,fx;
double Y[dim]={0.0},
dY[dim]={0.0},
K1[dim]={0.0},
K2[dim]={0.0},
K3[dim]={0.0},
K4[dim]={0.0},
Ktemp[dim]={0.0};

Y[1]=0.0;
Y[0]=0.0;
t=0.0;

FILE *fichier; //Export des résultats
fichier=fopen("Fx_1", "w");

for (i=0; i<100000; i++)
{
t=i*dt;

fx=Cx*(Vm*t-Y[0]);

derive(t, Y, dY);

for (j=0; j<dim; j++)
{
K1[j]=dt*dY[j];
Ktemp[j]=Y[j]+K1[j]/2.;
}

derive(t+dt/2, Ktemp, dY);

for (j=0; j<dim; j++)
{
K2[j]=dt*dY[j];
Ktemp[j]=Y[j]+K2[j]/2.;
}
}
```

```

derive(t+dt/2, Ktemp, dY);

for (j=0; j<dim; j++)
{
K3[j]=dt*dY[j];
Ktemp[j]=Y[j]+K3[j];
}

derive(t+dt, Ktemp, dY);

for (j=0; j<dim; j++)
{
K4[j]=dt*dY[j];
Y[j]=Y[j]+(1./6.)*(K1[j]+2*K2[j]+2*K3[j]+K4[j]);
}

printf("%e %e\n",Vm*t,fx);
fprintf(fichier,"%e %e\n",Vm*t,fx);

}

system("PAUSE");

return 0;
}

```

```
// PROGRAMME 6
```

```
#include<stdio.h>
#include <math.h>
#include <stdlib.h>
#define V0 1.6e-19
#define a 3.0e-10
#define dt 5.0e-8
#define Cx 10.0
#define Mx 1.0e-8
#define gamma 1.0e-3
#define Pi 3.1415926
#define dim 4

void derive(double t, double Y[], double dY[], double Vm)
{
dY[0]=Y[1];
dY[1]=(1/Mx)*(Cx*(Vm*t-Y[0])-V0*(2*Pi/a)*sin(2*Pi*Y[0]/a)-gamma*Y[1]);
}

int main(void)
{
int j,k;
double i,l;
double t;
double m,fx,Fric,Vm;
double Y[dim]={0.0},
        dY[dim]={0.0},
        K1[dim]={0.0},
        K2[dim]={0.0},
        K3[dim]={0.0},
        K4[dim]={0.0},
        Ktemp[dim]={0.0};

t=0.0;
Vm=5.0e-10;
l = 10000000;

FILE *fichier; //Export des résultats
fichier=fopen("fric", "w");

for (k=1; k<25; k++)
{
Y[1]=0.0;
Y[0]=0.0;
Vm=Vm*2;
fx=0;

for (i=0; i<l; i++)
{
t=i*dt;
fx=fx+Cx*(Vm*t-Y[0]);
derive(t, Y, dY, Vm);

for (j=0; j<dim; j++)
{
K1[j]=dt*dY[j];
Ktemp[j]=Y[j]+K1[j]/2.;
}
}
}
```

```

derive(t+dt/2, Ktemp, dY, Vm);

for (j=0; j<dim; j++)
{
    K2[j]=dt*dY[j];
    Ktemp[j]=Y[j]+K2[j]/2.;
}

derive(t+dt/2, Ktemp, dY, Vm);

for (j=0; j<dim; j++)
{
    K3[j]=dt*dY[j];
    Ktemp[j]=Y[j]+K3[j];
}

derive(t+dt, Ktemp, dY, Vm);

for (j=0; j<dim; j++)
{
    K4[j]=dt*dY[j];
    Y[j]=Y[j]+(1./6.)*(K1[j]+2*K2[j]+2*K3[j]+K4[j]);
}

}
Fric=fx/l+gamma*Vm;
printf("%e %e\n",Vm,Fric);
fprintf(fichier,"%e %e\n",Vm,Fric);
l=l/2;

}
system("PAUSE");

return 0;
}

```

```
// PROGRAMME 7
```

```
#include <stdio.h>
#include <math.h>
#include <stdlib.h>
#define V0 1.6e-19 // J (= 1.0 eV)
#define ax 3.16e-10 // m (= 3.16 Angstrom)
#define ay 5.48e-10 // m (=5.48 Angstrom)
#define dt 5.0e-7 // s
#define Cx 10.0 // N/m
#define Mx 1.0e-8 // kg
#define gamma 1.0e-3 // N/s
#define Vm 1.0e-7 // m/s
#define Pi 3.1415926
#define dim 4 // Modèle en 2D

void derive(double t, double Y[], double dY[])
{
dY[0]=Y[1];
dY[1]=(1/Mx)*(Cx*(Vm*t-Y[0])-V0*(2*Pi/ax)*sin(2*Pi*Y[0]/ax)-gamma*Y[1]);
dY[2]=Y[3];
dY[3]=(1/Mx)*(Cx*(0.7*Vm*t-Y[2])-V0*(2*Pi/ay)*sin(2*Pi*Y[2]/ay)-gamma*Y[3]);
}

int main(void)
{
int i,j;
double t;
double Y[dim]={0.0},
dY[dim]={0.0},
K1[dim]={0.0},
K2[dim]={0.0},
K3[dim]={0.0},
K4[dim]={0.0},
Ktemp[dim]={0.0};

Y[1]=0.0;
Y[0]=0.0;
t=0.0;

FILE *fichier; // Export des résultats
fichier=fopen("2_7", "w");

for (i=0; i<100000; i++)
{
t=i*dt;
printf("%e %e %e\n",Y[0], Y[2], 0); // Affiche les résultats
fprintf(fichier, "%e %e %e\n", Y[0], Y[2], 0); // Ecrit les résultats

derive(t, Y, dY);

for (j=0; j<dim; j++)
{
K1[j]=dt*dY[j];
Ktemp[j]=Y[j]+K1[j]/2.;
}

derive(t+dt/2, Ktemp, dY);

for (j=0; j<dim; j++)
{
```

```

        K2[j]=dt*dY[j];
        Ktemp[j]=Y[j]+K2[j]/2.;
    }

    derive(t+dt/2, Ktemp, dY);

    for (j=0; j<dim; j++)
    {
        K3[j]=dt*dY[j];
        Ktemp[j]=Y[j]+K3[j];
    }

    derive(t+dt, Ktemp, dY);

    for (j=0; j<dim; j++)
    {
        K4[j]=dt*dY[j];
        Y[j]=Y[j]+(1./6.)*(K1[j]+2*K2[j]+2*K3[j]+K4[j]);
    }
}

return 0;
}

```