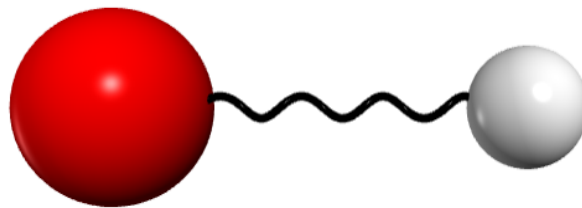


Modélisation et Simulation système masses-ressorts



Étudiants :

Fabio DELL'OVA
Clément LANGOLFF
Idris LESEUR
Yuhao XIE

Enseignant-responsable du projet :

Bernard GLEYSE

Date de remise du rapport : 15/06/2022

Référence du projet : STPI/P6/2022 – 17

Intitulé du projet : Modélisation et système masse-ressort

Type de projet : Modélisation

Objectifs du projet :

- Résoudre des équations de système masses-ressorts avec des masses différentes dans un plan
- Appliquer le modèle de l'oscillateur harmonique sur des molécules

Mots-clés du projet : Molécule, Runge Kutta, Oscillateur hamonique, Python

Table des matières

I	La théorie	5
1	Résolution dans le cas d'un système fixe	6
1.1	Résultats analytiques	6
1.2	Résultats numériques	7
1.3	Influence du rapport des masses	7
2	Résolution dans le cas d'un système libre	9
2.1	Résultats analytiques	9
2.2	Résultats numériques	10
3	Exemples plus complexes	11
3.1	Le triangle	11
3.2	Les formes cycliques	14
II	Application : modélisation vibratoire d'une liaison	16
4	Origine de la vibration	17
5	Les modèles	20
5.1	Le modèle de l'oscillateur harmonique	20
5.2	Le modèle de Morse	20
5.3	Confrontation des modèles et limites	21
5.4	Etude de résistance du ressort	21
6	Exemple de la liaison C-H	23
7	Perspectives d'amélioration des modèles	25

Remerciements

En préambule à ce rapport de projet, nous tenons à remercier l'INSA de proposer la possibilité d'effectuer des projets tel que celui de P6 dans le cadre de notre formation ingénieur.

Plus particulièrement, nous remercions Monsieur Bernard Gleyse, enseignant dans le département Génie Mathématiques de l'école, d'avoir supervisé le projet et de nous avoir apporté son aide lorsque cela était nécessaire.

Introduction

Les molécules sont présentes partout autour de nous. Coudées, triangulaires, quelconques ou simplement linéaires, les molécules se distinguent par leur forme particulière les une des autres. Portant notre regard vers l'infiniment petit, nous pouvons nous demander pourquoi prennent-elles ces formes, mais surtout, quelles interactions existent-ils entre les atomes permettant à la molécule de garder sa forme

En réfléchissant très brièvement, nous pouvons penser que de simples "bâtons" les relient, mais ce modèle n'expliquerait en aucun cas le caractère élastique de la matière. Le ressort, alliant résistance et élasticité, paraît être un modèle de choix pour décrire le mouvement au sein d'une molécule.

Ce projet consistera, dans une première partie, à résoudre des systèmes d'équations d'oscillateur harmonique avec différentes masses pour nous donner plus de liberté dans les choix et la forme de la molécule à décrire. Contrairement aux années précédentes, nous exploiterons le lien entre oscillateur harmonique et vibrations moléculaires et appliquerons ce modèle sur des molécules existantes. Nous verrons finalement les limites de ce modèle et proposerons différentes façons de l'améliorer pour les années futures.

Première partie

La théorie

Chapitre 1

Résolution dans le cas d'un système fixe

1.1 Résultats analytiques

Les études physiques donnent le système d'équation suivant :

$$\begin{cases} m_1 \ddot{z}_1 = -2kz_1 + kz_2 \\ m_2 \ddot{z}_2 = -2kz_2 + kz_1 \end{cases}$$

Auquel nous pouvons associer le système de matrice suivant :

$$\begin{pmatrix} \ddot{z}_1 \\ \ddot{z}_2 \end{pmatrix} = -k \underbrace{\begin{pmatrix} \frac{2}{m_1} & \frac{-1}{m_2} \\ \frac{-1}{m_2} & \frac{2}{m_1} \end{pmatrix}}_A \begin{pmatrix} z_1 \\ z_2 \end{pmatrix}$$

Nous allons chercher à diagonaliser la matrice pour retomber sur les équations de l'oscillateur harmonique. Après résolutions nous trouvons

$$D = \begin{pmatrix} \frac{m_a+m_b-\sqrt{m_a^2+m_b^2-m_a m_b}}{m_a m_b} & 0 \\ 0 & \frac{m_a+m_b+\sqrt{m_a^2+m_b^2-m_a m_b}}{m_a m_b} \end{pmatrix}$$

$$P = \begin{pmatrix} \frac{m_a-m_b+\sqrt{m_a^2+m_b^2-m_a m_b}}{m_a} & \frac{m_a-m_b-\sqrt{m_a^2+m_b^2-m_a m_b}}{m_a} \\ 1 & 1 \end{pmatrix}$$

La matrice diagonale D correspond aux valeurs propres de la matrice A. Le système matricielle dans la base $B' = P$ devient :

$$\begin{pmatrix} \dot{c}_1 \\ \dot{c}_2 \end{pmatrix} = -k \begin{pmatrix} \frac{m_a+m_b-\sqrt{m_a^2+m_b^2-m_a m_b}}{m_a m_b} & 0 \\ 0 & \frac{m_a+m_b+\sqrt{m_a^2+m_b^2-m_a m_b}}{m_a m_b} \end{pmatrix} \begin{pmatrix} c_1 \\ c_2 \end{pmatrix}$$

où $\begin{pmatrix} c_1 \\ c_2 \end{pmatrix}$ correspond aux coordonnées de $\begin{pmatrix} z_1 \\ z_2 \end{pmatrix}$ dans la base B' . On calcule ensuite les coordonnées dans la base usuelle de R^2 par

$$\begin{pmatrix} z_1 \\ z_2 \end{pmatrix} = P \begin{pmatrix} c_1 \\ c_2 \end{pmatrix}$$

Le système d'équation de départ devient :

$$\begin{cases} \ddot{c}_1 = -\omega_1^2 c_1 \\ \ddot{c}_2 = -\omega_2^2 c_2 \end{cases}$$

où $\omega_1 = \sqrt{k \frac{m_a+m_b-\sqrt{m_a^2+m_b^2-m_a m_b}}{m_a m_b}}$ et $\omega_2 = \sqrt{k \frac{m_a+m_b+\sqrt{m_a^2+m_b^2-m_a m_b}}{m_a m_b}}$.

Posons $S = m_a - m_b$ et $R = \sqrt{m_a^2 + m_b^2 - m_a m_b}$

Nous obtenons :

$$z_1 = \frac{(A\cos(\omega_1 t) + B\sin(\omega_1 t))(S + R) + (C\cos(\omega_2 t) + D\sin(\omega_2 t))(S - R)}{m_a}$$

$$z_2 = A\cos(\omega_1 t) + B\sin(\omega_1 t) + C\cos(\omega_2 t) + D\sin(\omega_2 t)$$

où A,B,C et D sont calculées selon les conditions initiales :

$$A = \frac{z_{10}(S + R) + m_b z_{20}}{2R}$$

$$B = \frac{z_{10}'(S + R) + m_b z_{20}'}{2\omega_1 R}$$

$$C = \frac{z_{10}(R - S) - m_b z_{20}}{2R}$$

$$D = \frac{z_{10}'(R - S) - m_b z_{20}'}{2\omega_2 R}$$

1.2 Résultats numériques

Pour simuler la position de nos masses, nous utiliserons la méthode Runge-Kutta 4. Cette méthode consiste à approximer la solution exacte sur un interval de temps très court. Pour cela, nous devons d'abord ramener la situation à un problème de Cauchy :

$$\begin{cases} Y'(t) = f(t, Y(t)) \\ Y(t_0) = Y_0 \end{cases}$$

Puis, avec l'application f, le vecteur Y à l'instant $t + dt$ est calculé par

$$\begin{cases} K_1 = f(Y_t) \\ K_2 = f(Y_t + \frac{K_1}{2}) \\ K_3 = f(Y_t + \frac{K_2}{2}) \\ K_4 = f(Y_t + K_3) \\ Y_{t+dt} = Y_t + \frac{dt}{6}(K_1 + 2K_2 + 2K_3 + K_4) \end{cases}$$

Pour le cas d'un système à deux masses fixe nous avons :

$$f : \begin{pmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{pmatrix} \rightarrow \begin{pmatrix} x_3 \\ x_4 \\ \frac{k}{m_1}(-2x_1 + x_2) \\ \frac{k}{m_2}(-2x_2 + x_1) \end{pmatrix}$$

1.3 Influence du rapport des masses

En prenant les masses telles que $\frac{m_a}{m_b} = 10^n$ avec $n \in \mathbb{Z}$, nous pouvons observer un caractère particulier sur la courbe de position en fonction du temps.

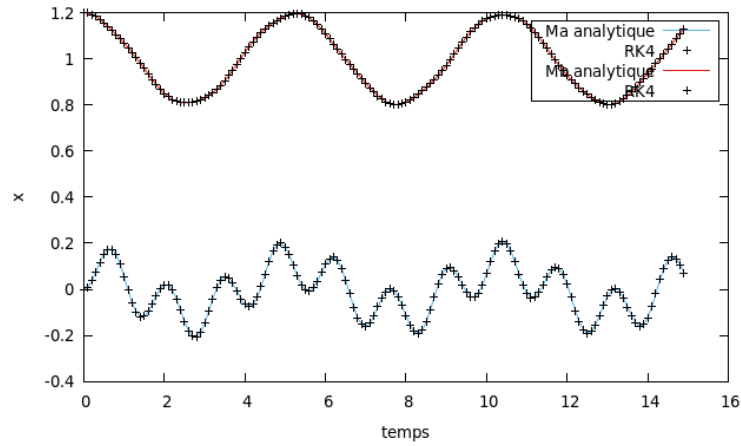


FIGURE 1.1 – Position (unité de distance) en fonction du temps (unité de temps)

conditions initiales	mobile A	mobile B	longueur à vide
position x	0	0,2	$l_{ABx} = 1.0$
position y	0	0	$l_{ABy} = 0$
vitesse x	0	0	
vitesse y	0	0	
masse	0.1	1.0	
$k = 1.0$			

Ici, la masse $m_b = 10m_a$. La position du mobile M_b à l'instant $t = 0$ se trouve à une distance de 0.2 de son point d'équilibre et le mobile M_a est positionné sur son point d'équilibre. Les vitesses des mobiles sont initialisées à 0. Nous observons que le mobile M_b oscille de manière parfaitement sinusoïdale. De plus les deux mobiles possèdent la même période d'oscillation et ont chacun la même amplitude maximale de ± 0.2 .

Chapitre 2

Résolution dans le cas d'un système libre

2.1 Résultats analytiques

Les études physiques donnent le système d'équation suivant :

$$\begin{cases} \ddot{z}_1 = \frac{k}{m_1}(z_2 - z_1) \\ \ddot{z}_2 = \frac{k}{m_2}(z_1 - z_2) \end{cases}$$

Nous disposons de la 3^{ème} équation dû au centre de masse fixe : $m_1 z_1 + m_2 z_2 = 0$

Par conséquent, nous avons $z_2 = \frac{-m_1 z_1}{m_2}$, ce qui nous permet de réécrire le système de la manière suivante :

$$\begin{cases} \ddot{z}_1 = -k z_1 \left(\frac{m_1 + m_2}{m_1 m_2} \right) \end{cases}$$

Nous avons ici l'équation de l'oscillateur harmonique avec $\omega^2 = \frac{k}{\mu}$.

Et donc, après recherche des constantes A et B nous avons :

$$\begin{cases} z_1 = z_{10} \cos(\omega t) + \frac{z_{10}}{\omega} \sin(\omega t) \\ z_2 = \frac{-m_1}{m_2} (z_{10} \cos(\omega t) + \frac{z_{10}}{\omega} \sin(\omega t)) \end{cases}$$

2.2 Résultats numériques

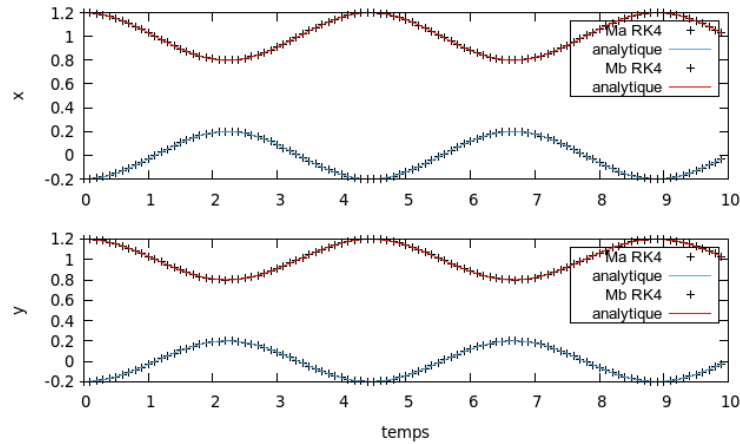


FIGURE 2.1 – Position (unité de distance) en fonction du temps (unité de temps)

conditions initiales	mobile A	mobile B	longueurs à vide
position x	-0.2	0,2	$l_{ABx} = 1.0$
position y	-0.2	0.2	$l_{ABy} = 1.0$
vitesse x	0	0	
vitesse y	0	0	
masse	1.0	1.0	
$k = 1.0$			

Sur ce système, nous avons pu tester que notre programme fonctionne correctement avec le système de coordonnées $z = x + iy$. En effet, en inclinant le ressort de $\frac{\pi}{4}$ par rapport à l'axe x, les deux masses étant égales, nous pouvons être sûr que les trajectoires des deux mobiles seront symétriques par rapport à l'origine, c'est à dire par rapport au centre de masse.

Chapitre 3

Exemples plus complexes

3.1 Le triangle

Les études physiques donnent le système d'équation suivant :

$$\begin{cases} \ddot{z}_1 = \frac{k}{m_1}(z_2 + z_3 - 2z_1) \\ \ddot{z}_2 = \frac{k}{m_2}(z_1 + z_3 - 2z_2) \\ \ddot{z}_3 = \frac{k}{m_3}(z_1 + z_2 - 2z_3) \end{cases}$$

Nous disposons de la 4^{ème} équation dû au centre de masse fixe : $m_1z_1 + m_2z_2 + m_3z_3 = 0$

Par conséquent, nous avons $z_3 = \frac{-m_1z_1 - m_2z_2}{m_3}$, ce qui nous permet de réécrire le système de la manière suivante :

$$\begin{cases} \ddot{z}_1 = -k(z_1(\frac{2}{m_1} + \frac{1}{m_3}) + z_2(\frac{m_2}{m_1m_3} - \frac{1}{m_1})) \\ \ddot{z}_2 = -k(z_1(\frac{m_1}{m_2m_3} - \frac{1}{m_2}) + z_2(\frac{2}{m_2} + \frac{1}{m_3})) \end{cases}$$

Nous savons que le système est oscillant, nous avons, en prenant ω comme valeur propre particulière

$$\begin{cases} z_1 = A\cos(\omega t) + B\sin(\omega t) \\ z_2 = C\cos(\omega t) + D\sin(\omega t) \end{cases} \quad \text{et} \quad \begin{cases} \ddot{z}_1 = -\omega^2(A\cos(\omega t) + B\sin(\omega t)) = -\omega^2z_1 & A, B \in \mathbb{C} \\ \ddot{z}_2 = -\omega^2(C\cos(\omega t) + D\sin(\omega t)) = -\omega^2z_2 & C, D \in \mathbb{C} \end{cases}$$

De plus, nous savons que $\omega^2 = \frac{k}{M}$ où M est une constante homogène à une masse. Ce qui nous permet d'écrire le système sous une forme matricielle de la façon suivante :

$$\begin{aligned} -\omega^2 \begin{pmatrix} z_1 \\ z_2 \end{pmatrix} &= -k \begin{pmatrix} \frac{2}{m_1} + \frac{1}{m_3} & \frac{m_2}{m_1m_3} - \frac{1}{m_1} \\ \frac{m_1}{m_2m_3} - \frac{1}{m_2} & \frac{2}{m_2} + \frac{1}{m_3} \end{pmatrix} \begin{pmatrix} z_1 \\ z_2 \end{pmatrix} \\ -\frac{k}{M} \begin{pmatrix} z_1 \\ z_2 \end{pmatrix} &= -k \begin{pmatrix} \frac{2}{m_1} + \frac{1}{m_3} & \frac{m_2}{m_1m_3} - \frac{1}{m_1} \\ \frac{m_1}{m_2m_3} - \frac{1}{m_2} & \frac{2}{m_2} + \frac{1}{m_3} \end{pmatrix} \begin{pmatrix} z_1 \\ z_2 \end{pmatrix} \\ I_2 &= M \begin{pmatrix} \frac{2}{m_1} + \frac{1}{m_3} & \frac{m_2}{m_1m_3} - \frac{1}{m_1} \\ \frac{m_1}{m_2m_3} - \frac{1}{m_2} & \frac{2}{m_2} + \frac{1}{m_3} \end{pmatrix} \end{aligned}$$

Ce qui nous amène à la conclusion suivante :

$$\text{Det}(I_2) = \text{Det} \left(M \begin{pmatrix} \frac{2}{m_1} + \frac{1}{m_3} & \frac{m_2}{m_1m_3} - \frac{1}{m_1} \\ \frac{m_1}{m_2m_3} - \frac{1}{m_2} & \frac{2}{m_2} + \frac{1}{m_3} \end{pmatrix} \right)$$

Soit

$$\begin{aligned} 1 &= M^2 \left(\left(\frac{2}{m_1} + \frac{1}{m_3} \right) \left(\frac{2}{m_2} + \frac{1}{m_3} \right) - \left(\frac{m_1}{m_2m_3} - \frac{1}{m_2} \right) \left(\frac{m_2}{m_1m_3} - \frac{1}{m_1} \right) \right) \\ &= 3M^2 \underbrace{\frac{m_1 + m_2 + m_3}{m_1m_2m_3}}_{\frac{1}{\mu}} = 1 \\ M^2 &= \frac{\mu}{3} \end{aligned}$$

Ce qui nous donne $M = \pm\sqrt{\frac{\mu}{3}}$. Une masse ne pouvant pas être négative, nous prenons en compte seulement la solution positive, et ainsi $\omega^2 = \frac{k}{M} = k\sqrt{\frac{3}{\mu}}$.
Après recherche des constantes A,B,C et D nous avons dans ce cas

$$z_1 = z_{1_0} \cos(\omega t) + \frac{\dot{z}_{1_0}}{\omega} \sin(\omega t)$$

$$z_2 = z_{2_0} \cos(\omega t) + \frac{\dot{z}_{2_0}}{\omega} \sin(\omega t)$$

$$z_3 = \frac{-m_1 z_{1_0} \cos(\omega t) - m_1 \frac{\dot{z}_{1_0}}{\omega} \sin(\omega t) - m_2 z_{2_0} \cos(\omega t) - m_2 \frac{\dot{z}_{2_0}}{\omega} \sin(\omega t)}{m_3}$$

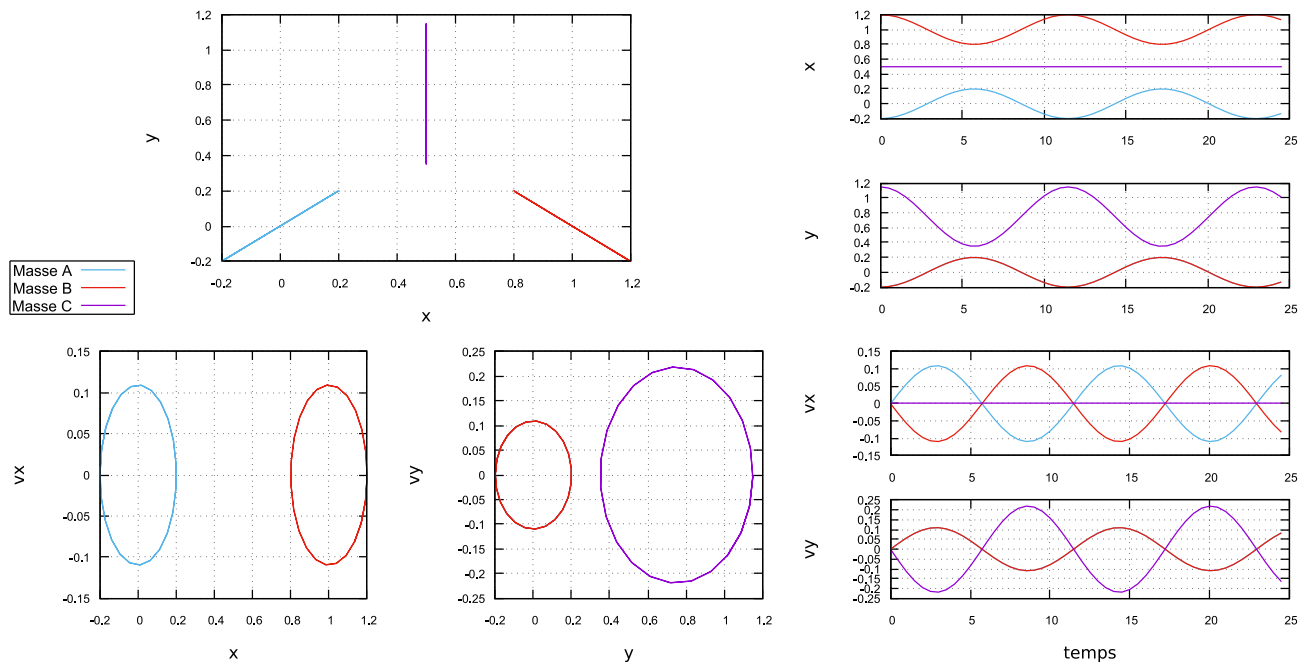


FIGURE 3.1 – Les positions ainsi que les longueurs à vide ont été initialisées de manière à simuler un triangle étiré proportionnellement.

conditions initiales	mobile A	mobile B	mobile C	longueurs à vide
position x	-0.2	0,2	$-\frac{m_a}{m_c} x_a - \frac{m_b}{m_c} x_b$	$l_{ABx} = 1.0 \quad l_{ABy} = 0$
position y	-0.2	-0.2	$-\frac{m_a}{m_c} x_a - \frac{m_b}{m_c} x_b$	$l_{ACx} = 0.5 \quad l_{ACy} = 0.75$
vitesse x	0	0	0	
vitesse y	0	0	0	
masse	1.0	1.0	1.0	
$k = 0.1$				

La position du mobile C ne variant pas sur l'axe des x, il ne possède aucune vitesse x. Et donc, il n'est pas représenté sur le graphe de vitesse x en fonction de la position x. En revanche sa position et sa vitesse en y sont 2 fois plus grandes que celle des autres masses. Nous pouvons voir que le graphe de vitesse en fonction de la position sur l'axe y affichent deux cercles. Le petit cercle correspond à la masse A et B, superposé car les mobiles A et B possèdent la même masse et leur position est symétrique par rapport à l'axe y. Et le grand cercle, 2 fois plus grand, correspondant à la masse C. L'équation de centre de masse fixe est bien respectée.

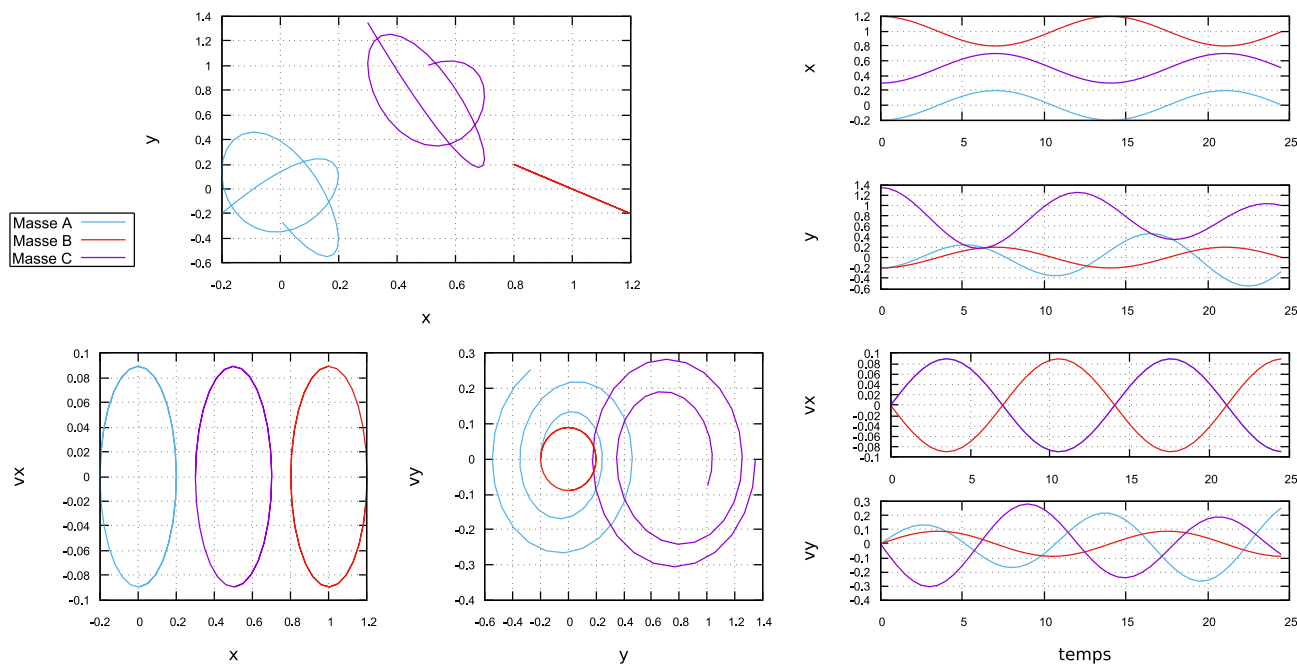


FIGURE 3.2 – Les positions ainsi que les longueurs à vide ont été initialisées de manière à simuler un triangle étiré proportionnellement.

conditions initiales	mobile A	mobile B	mobile C	longueurs à vide
position x	-0.2	0,2	$-\frac{m_a}{m_c} x_a - \frac{m_b}{m_c} x_b$	$l_{ABx} = 1.0 \quad l_{ABy} = 0$
position y	-0.2	-0.2	$-\frac{m_a}{m_c} x_a - \frac{m_b}{m_c} x_b$	$l_{ACx} = 0.5 \quad l_{ACy} = 0.75$
vitesse x	0	0	0	
vitesse y	0	0	0	
masse	1.0	2.0	1.0	

Nous avons pu remarquer que, si une masse était égale à la somme des autres masses alors, cette masse possédait une trajectoire linéaire tandis que les autres masses oscillaient de manière chaotique dans le plan. Cela est vrai seulement si le triangle est étiré proportionnellement, c'est à dire que les masses sont étirer par rapport aux médianes du triangle. Autrement, les graphes sont totalement aléatoire, du moins pour les tests que nous avons eu le temps de faire.

Nous pouvons observer que les trois formes circulaires du graphe vitesse x contre position x semble de même dimension. Une autre observation liée à la précédente peut se faire en remarquant que la courbe de vitesse en fonction du temps du mobile B (celui dont la masse est égale aux deux autres) s'oppose aux deux courbes des masses A et B qui sont superposées. Aucune explication n'a été trouvée pour justifier ce phénomène.

Le carré :

Lors du calcul de ce système nous obtenons comme solution particulière :

$$det(I_3) = 4M^3 \frac{m_1 + m_2 + m_3 + m_4}{\underbrace{m_1 m_2 m_3 m_4}_{\frac{1}{\mu}}}$$

L'hexagone :

$$det(I_5) = 6M^5 \frac{m_1 + m_2 + m_3 + m_4 + m_5 + m_6}{\underbrace{m_1 m_2 m_3 m_4 m_5 m_6}_{\frac{1}{\mu}}}$$

3.2 Les formes cycliques

Nous observons une formule générale pour les systèmes cycliques (très commun chez les molécules). Pour un système à n masses ainsi que n ressorts, nous avons besoin de $n-1$ équations car le centre de masse est supposé fixe ; nous obtenons ainsi une matrice $A \in M_{n-1}(R)$. En suivant le même raisonnement, nous avons alors

$$\det(I_{n-1}) = nM^{n-1} \frac{\sum_{i=1}^n m_i}{\prod_{i=1}^n m_i}$$

Le résultat étant compliqué à montrer, nous allons nous intéresser au cas particulier dans lequel les masses sont égales. Dans le cas suivant, nous développons les équations du PDF, mettons sous forme matricielle l'ensemble et obtenons :

$$A = \frac{1}{m} \begin{pmatrix} 3 & 0 & 1 & \dots & 1 \\ -1 & 2 & -1 & & \\ & \ddots & \ddots & \ddots & \\ & & -1 & 2 & -1 \\ 1 & \dots & 1 & 0 & 3 \end{pmatrix} \xrightarrow{DET} \frac{1}{m^{n-1}} n^2$$

Lorsque nous écrivons les équations du PFD et que nous passons à la forme matricielle nous enlevons une ligne et une colonne associée à une des masses. Cependant, si nous gardons le système original sans tenir compte de l'équation relative au centre de masse fixe, nous obtenons la matrice $B \in M_n(R)$ de la forme :

$$B = \frac{1}{m} \begin{pmatrix} 2 & -1 & & & -1 \\ -1 & 2 & -1 & & \\ & \ddots & \ddots & \ddots & \\ & & -1 & 2 & -1 \\ -1 & & & -1 & 2 \end{pmatrix}$$

Ce type de matrice est très particulier car nous pouvons remarquer que le motif écrit en première ligne est décalé d'un cran vers la droite à la ligne du dessous. Ces matrices portent le nom de matrices circulantes et, en théorie des graphes, sont associées à des graphes cycles ce qui est l'équivalent de notre système de base (avec n masses).

De plus, nous savons également que si la matrice est diagonalisable alors :

$$\begin{aligned} \det(A) &= \det(PDP^{-1}) \\ &= \det(P)\det(D)\det(P^{-1}) \\ &= \det(D)\det(I_n) = \det(D) * 1 = \det(D) \end{aligned}$$

Lorsque nous calculons les valeurs propres des deux matrices pour les diagonaliser nous nous apercevons que les deux matrices possèdent les mêmes valeurs propres à l'exception qu'il y a une valeur propre de plus pour B puisque la dimension de la matrice est n et non $n - 1$. Cependant la matrice B traduit un graphe et au cours de mon projet mathématiques j'ai pu démontrer que le spectre de la matrice (rangemet des valeurs propres dans l'ordre croissant) avait comme première valeur propre 0 puisqu'elle témoigne de la connexité du graphe et celle ci possède une multiplicité égale au nombre de composantes connexes du graphe (ici 1). La matrice A possède des valeurs propres difficiles à calculer, cependant au cours de mon projet mathématiques, j'ai pu remarquer que les valeurs propres d'une matrice circulante étaient égales aux racines énièmes de l'unité i.e. $v_k = 2 - 2\cos(\frac{2k\pi}{n})$ où $k \in \{0, 1, \dots, n-1\}$ et n correspondant à la taille de la matrice. Nous avons désormais la matrice diagonale associée à A qui s'écrit de la forme :

$$D = \begin{pmatrix} v_0 & & & & \\ & v_1 & & & \\ & & v_2 & & \\ & & & \ddots & \\ & & & & v_{n-1} \end{pmatrix}$$

Finalement, il est simple de calculer le déterminant de D puisque cela revient à multiplier toutes les composantes de la diagonale (excepté la première qui vaut $v_0 = 0$). Nous obtenons

$$\det(A) = \frac{1}{m^{n-1}} \prod_{k=1}^{n-1} v_k$$

$$\begin{aligned}
 \prod_{k=1}^{n-1} v_k &= \prod_{k=1}^{n-1} 2(1 - \cos(\frac{2\pi k}{n})) \\
 &= 2^{n-1} \prod_{k=1}^{n-1} (1 - \cos(\frac{2\pi k}{n})) \\
 &= 2^{n-1} \prod_{k=1}^{n-1} \operatorname{Re}(e^{i0} - e^{\frac{2i\pi k}{n}}) \\
 &= 2^{n-1} \prod_{k=1}^{n-1} \operatorname{Re}(2ie^{\frac{i\pi k}{n}} \sin(\frac{-\pi k}{n})) \\
 &= 2^{n-1} \prod_{k=1}^{n-1} \operatorname{Re}(-2i(\cos(\frac{k\pi}{n}) + i\sin(\frac{k\pi}{n}))\sin(\frac{\pi k}{n})) \\
 &= 2^{2n-2} \prod_{k=1}^{n-1} \sin^2(\frac{\pi k}{n})
 \end{aligned}$$

Observons le produit des $\sin(\frac{\pi k}{n})$

$$\begin{aligned}
 \prod_{k=1}^{n-1} \sin(\frac{\pi k}{n}) &= \prod_{k=1}^{n-1} \left(\frac{e^{\frac{i\pi k}{n}} - e^{-\frac{i\pi k}{n}}}{2i} \right) \\
 &= \frac{1}{(2i)^{n-1}} \prod_{k=1}^{n-1} (e^{\frac{i\pi k}{n}} - e^{-\frac{i\pi k}{n}}) \\
 &= \frac{1}{(2i)^{n-1}} \prod_{k=1}^{n-1} e^{-\frac{i\pi k}{n}} (e^{\frac{2i\pi k}{n}} - 1) \\
 &= \frac{1}{(2i)^{n-1}} e^{-\frac{i\pi}{n} \sum_{k=1}^{n-1} k} \prod_{k=1}^{n-1} (e^{\frac{2i\pi k}{n}} - 1) \\
 &= \frac{1}{(2i)^{n-1}} e^{-\frac{i\pi(n-1)}{2}} \prod_{k=1}^{n-1} (e^{\frac{2i\pi k}{n}} - 1) \\
 &= \frac{1}{(2i)^{n-1}} (-i)^{n-1} (-1)^{n-1} \prod_{k=1}^{n-1} (1 - e^{\frac{2i\pi k}{n}}) \\
 &= \frac{1}{2^{n-1}} \prod_{k=1}^{n-1} (1 - e^{\frac{2i\pi k}{n}})
 \end{aligned}$$

Posons le polynôme $P_n = X^n - 1 = \prod_{k=0}^{n-1} (X - e^{\frac{2i\pi k}{n}})$

On a $X^n - 1 = (X - 1) \sum_{k=0}^{n-1} X^k$, donc $(X - 1) \prod_{k=1}^{n-1} (X - e^{\frac{2i\pi k}{n}}) = (X - 1) \sum_{k=0}^{n-1} X^k \Leftrightarrow \prod_{k=1}^{n-1} (X - e^{\frac{2i\pi k}{n}}) = \sum_{k=0}^{n-1} X^k$

On évalue en 1 et on obtient :

$$\prod_{k=1}^{n-1} (1 - e^{\frac{2i\pi k}{n}}) = \sum_{k=0}^{n-1} 1^k = \sum_{k=0}^{n-1} 1 = n$$

Finalement

$$\prod_{k=1}^{n-1} \sin(\frac{\pi k}{n}) = \frac{n}{2^{n-1}}$$

Donc

$$\det(D) = 2^{2n-2} \prod_{k=1}^{n-1} \sin^2(\frac{\pi k}{n}) = 2^{2n-2} \left(\prod_{k=1}^{n-1} \sin(\frac{\pi k}{n}) \right)^2 = 2^{2n-2} \left(\frac{n}{2^{n-1}} \right)^2 = n^2$$

Finalement, nous obtenons la formule voulue :

$$\det(A) = \frac{1}{m^{n-1}} n^2$$

Deuxième partie

Application : modélisation vibratoire d'une liaison

Chapitre 4

Origine de la vibration

Les propriétés de la lumière telles que la réflexion, réfraction et interférence peuvent s'interpréter par le caractère ondulatoire de la lumière.

La lumière se propage sous forme d'onde dans le vide à la vitesse $c = 3.10^8 m.s^{-1}$. La période temporelle T correspond à une fréquence $\nu = \frac{1}{T}$ en Hz .

La période spatiale λ , appelée longueur d'onde, est exprimée en mètre et est liée à la période temporelle T par $\lambda = cT = \frac{c}{\nu}$. On note aussi $\sigma = \frac{1}{\lambda}$ en m^{-1} .

Le domaine des ondes électromagnétiques couvre une large gamme de fréquences.

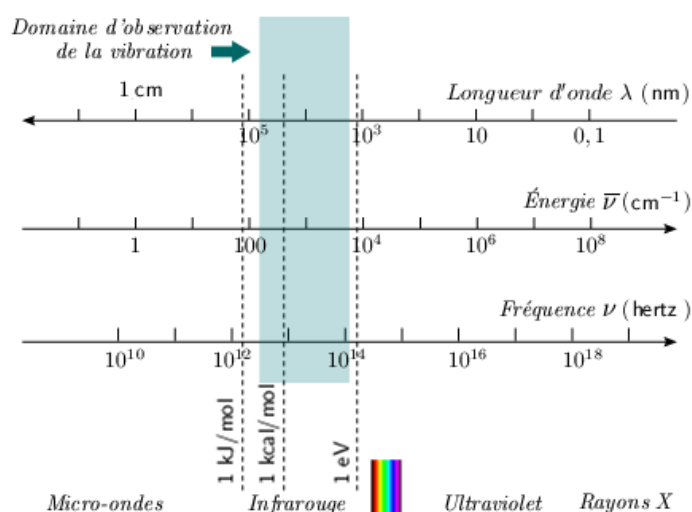


FIGURE 4.1 – Spectre électromagnétique

Nous pouvons associer chacune des fréquences à une vibration de molécule. Un autre spectre est plus adapté pour décrire la vibration moléculaire, il s'agit de la spectroscopie infrarouge [4]. Cette spectroscopie est largement utilisée pour déterminer les types de liaison dans une molécule. Nous nous baserons par la suite sur ce graphe pour traduire la vibration moléculaire au sein d'une liaison.

Les fréquences de vibration des molécules sont reliées à l'énergie par $E = h\nu$ où $h = 6,626.10^{-34} J.s^{-1}$ est la constante de Planck.

Cependant, les molécules ne vibrent pas n'importe comment. Les vibrations sont basées sur des niveaux d'énergie E_n calculé à partir de l'équation de Schrödinger. Ainsi lors du passage d'un niveau supérieur à inférieur, la molécule rejette son énergie sous forme de photons. A l'inverse, nous pouvons apporter de l'énergie à la molécule (essentiellement sous forme de chaleur met aussi sous forme d'onde comme pour construire le spectre infrarouge). L'énergie apportée doit dépasser un certain seuil pour pouvoir casser la molécule. Cette énergie de cassure peut se trouver dans les tables (déterminer grâce à la fonction d'enthalpie). Les molécules ne se trouve jamais sans énergie, leur niveau minimal est toujours E_0 .

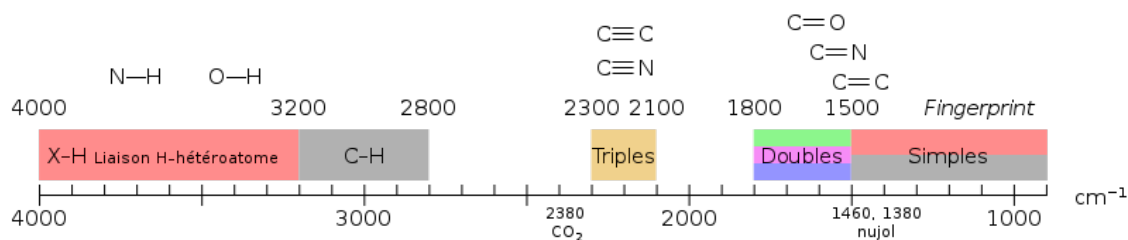


FIGURE 4.2 – Spectre infrarouge de liaisons courantes

Le principe de la spectroscopie infrarouge se base sur l’absorption de la lumière par les molécules dans la région de l’infrarouge du spectre électromagnétique. Cette absorption est convertit en vibration moléculaire. Nous pouvons observer sur le schéma ci dessus, qu’il n’existe pas une seule fréquence propre de vibration pour chaque liaison. En effet, il s’agit d’une bande d’absorption. Si nous agrandissons cette bande, nous pourrions observer une infinité de pics correspondant aux niveaux d’énergie de la molécule. Prenons l’exemple d’une liaison se trouvant à une énergie E_{n^*}

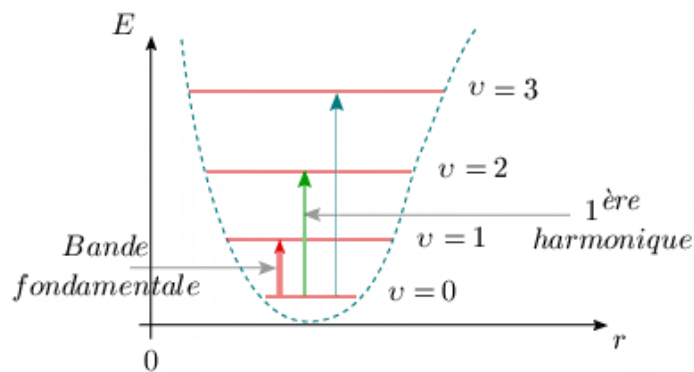


FIGURE 4.3 – Schéma de transition d’énergie d’une liaison

Nous appelons la bande fondamentale, la quantité d’énergie nécessaire pour passer une liaison d’un niveau d’énergie n^* à un niveau d’énergie $n^* + 1$. Nous nommons intensités harmoniques la différence d’énergie (ΔE) entre un niveau k et $k + 1$ partant d’un niveau d’énergie n^* . Contrairement à la représentation du schéma, l’intensité des harmoniques tend vers 0 lorsque k tend vers l’infini. Ces harmoniques partant du même niveau d’énergie n^* , seront séparées d’une distance multiple de λ sur un graphe de transmittance. C’est pourquoi nous n’observons pas de pics uniques sur les spectres IR. Nous visualisons en fait les différences d’énergie entre chaque niveau d’énergie et donc un large pic dû à la précision du graphe.

Il existe plusieurs sortes de vibration au sein d’une molécule, ce nombre de vibration est proportionnel au nombre d’atomes présents dans la molécule. De manière générale, il a été noté que les molécules linéaires possédaient $3N-5$ modes de vibrations différentes contre $3N-6$ pour les molécules non linéaires. Par exemple, le dioxyde de carbone possède quatre modes de vibration [6] : symétrique, asymétrique et deux modes de torsion (plan et hors plan). Pour notre projet, nous nous sommes focaliser sur des liaisons simples. Ainsi, nous considérons la liaison comme une molécule linéaire composée de 2 atomes. Nous avons donc un unique mode de vibration possible : le mode symétrique. Le graphe de spectroscopie infrarouge se base sur un changement de moment dipolaire, c’est pourquoi le graphe IR nous montrera les fréquences de vibration pour les modes assymétriques et de torsion. La vibration symétrique ne faisant pas varier le moment dipolaire dans une molécule linéaire, aucun pic concernant ce mode apparaîtra sur ce graphe. Cependant, il existe une autre méthode de spectroscopie appelée spectroscopie de Raman ne se basant pas sur la variation d’un moment dipolaire et donc faisant apparaître le pic donnant les informations sur la vibration symétrique[2]. Les graphes de Raman seront donc nos principales sources d’information pour étudier la vibration symétrique d’une liaison de 2 atomes.

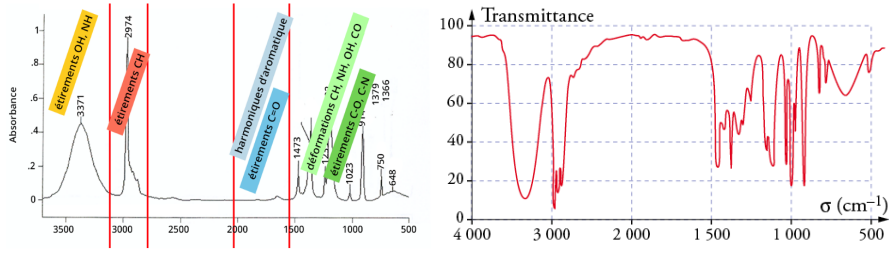


FIGURE 4.4 – Spectre de Raman et infrarouge du Butan-1-ol

Chapitre 5

Les modèles

5.1 Le modèle de l'oscillateur harmonique

En absorbant et émettant les radiations électromagnétiques dans une zone proche de l'infrarouge, les molécules diatomiques réagissent de la même façon qu'un oscillateur harmonique. Nous savons que l'expression de la fréquence d'un oscillateur harmonique est donnée par $T = 2\pi\sqrt{\frac{m}{k}}$. Pour notre modèle d'oscillateur harmonique, nous considérerons les liaisons diatomiques libres, c'est à dire sans autres forces extérieures. m aura ainsi la valeur du centre de masse du système (désignée par μ). [10, 8, 9, 4]
Nous pouvons finalement exprimer la constante élastique k propre à chaque liaison :

$$T = \frac{1}{c\sigma}$$

d'où

$$k = (2\pi c\sigma)^2 \mu$$

avec c la célérité de la lumière en $m.s^{-1}$, σ en m^{-1} , μ en kg et donc k en $N.m^{-1}$ (ou $kg.s^{-1}$).

Nous pouvons ainsi représenter l'énergie potentielle de la molécule diatomique par l'énergie potentielle de l'oscillateur harmonique $E_{el} = \frac{k}{2}(r - r_0)^2$.

Selon la mécanique quantique, le mouvement de la particule peut être décrit par l'équation de Schrödinger :

$$\frac{d^2\Psi}{dr^2} + \frac{8\pi^2\mu}{h^2} \left[E - \frac{k}{2}(r - r_0)^2 \right] \Psi = 0$$

où E est l'énergie totale de la molécule et Ψ la fonction d'onde et h la constante de Planck.

La solution de cette équation fournit les valeurs de chaque niveau d'énergie et donc des fréquences d'oscillation de la molécule.

$$E_n = \frac{h}{2\pi} \sqrt{\frac{k}{\mu}} \left(n + \frac{1}{2} \right)$$

5.2 Le modèle de Morse

Le potentiel de Morse est un modèle plus précis pour représenter l'énergie entre 2 atomes. Il s'exprime de la manière suivante :

$$V(r) = D_e(1 - e^{-\alpha(r-r_0)})^2$$

où

$$\alpha = \sqrt{\frac{k_e}{2D_e}}$$

où D_e correspond à l'énergie de dissociation de la liaison que nous pouvons trouver sur les tables (Table de Dissociation Wikipedia).

La solution à l'équation de Schrödinger s'exprime de la sorte :

$$E_n = h\nu_0 \left(n + \frac{1}{2} \right) - \frac{(h\nu_0)^2}{4D_e} \left(n + \frac{1}{2} \right)^2$$

où $\nu_0 = c\sigma$ est la fréquence du niveau d'énergie E_0 .

5.3 Confrontation des modèles et limites

Nous pouvons noter une différence importante entre les 2 modèles. Les différences d'énergie entre deux paliers ne diminuent pas pour le modèle de l'oscillateur harmonique, contrairement au modèle de Morse où les différences tendent vers 0 jusqu'à la limite de cassure de la liaison. Le modèle de l'oscillateur harmonique reste cependant tout à fait convenable lorsque l'apport en énergie dépasse peu le palier E_0 .

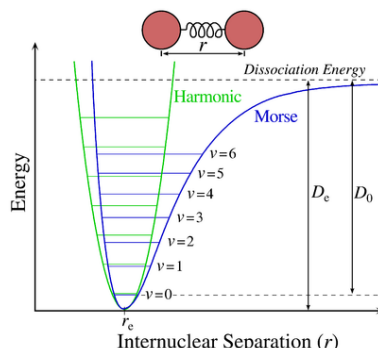


FIGURE 5.1 – Potentiel de Morse et élastique

Nous pouvons noter que le potentiel élastique ne possède aucune limite d'énergie. Ainsi, selon ce modèle, la liaison de la molécule ne se dissociera jamais. Nous avons ainsi besoin de définir une limite de cassure pour notre modèle de l'oscillateur harmonique.

5.4 Etude de résistance du ressort

Le but de cette partie est de savoir à partir de quel effort il y aura rupture du ressort. Puis, quelle est la longueur maximale d'allongement du ressort.[1]

Nous étudierons les sollicitations dans un ressort hélicoïdale à fil rond.

Les caractéristiques d'un ressort à fil rond sont les suivantes :

- D** (en mm) le diamètre d'enroulement de l'hélice moyen ;
- d** (en mm) le diamètre du fil
- n** le nombre de spires utilisées

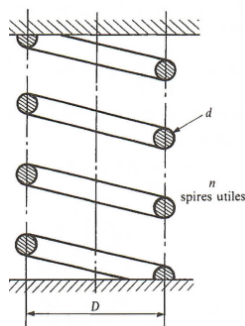


FIGURE 5.2 – Schéma d'un ressort hélicoïdal

Comme vu précédemment pour un ressort, il y a proportionnalité entre la charge axiale notée P (en N) et la déformation correspondante $l - l_0$ (en mm), l étant la longueur sous la charge P et l_0 la longueur à vide. Ce coefficient de proportionnalité est noté k (en N/mm), c'est la rigidité du ressort. Cette rigidité peut être exprimé de la manière suivante :

$$k = \frac{G * d^4}{8 * D^3 * n}$$

avec G le module de Coulomb (en MPa).

Le cas d'un système libre à 2 masses peut se traduire par un ressort en équilibre stable sous l'action de 2 torseurs de liaison se réduisant à deux torseurs glisseurs.

Étudions les efforts en traction dans un premier temps.

$$\{\vec{\tau}\} = \left\{ \begin{matrix} \vec{P} \\ \vec{0} \end{matrix} \right\}$$

et

$$\{\vec{\tau}'\} = \left\{ \begin{matrix} \vec{P}' \\ \vec{0} \end{matrix} \right\}$$

Effectuons maintenant une section fictive S normale au fil . Soit G le centre de surface de la section du fil. Soit P la norme de \vec{P}

$$\{\vec{\tau}_{coh}\} = -\{\vec{\tau}\} = \begin{pmatrix} -P * \sin(\alpha) & P * \cos(\alpha) * \frac{D}{2} \\ P * \cos(\alpha) & P * \sin(\alpha) * \frac{D}{2} \\ 0 & 0 \end{pmatrix}$$

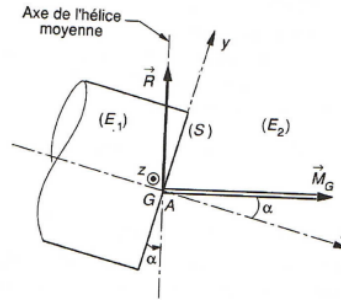


FIGURE 5.3 – Schéma de coupe d'un fil sur un ressort

Pour un ressort α vaut 6° . Nous allons donc faire l'approximation suivante $\cos(\alpha) \approx 1$ et $\sin(\alpha) \approx 0$. Le torseur de cohésion devient donc

$$\{\vec{\tau}_{coh}\} = \begin{pmatrix} 0 & P * \frac{D}{2} \\ P & 0 \\ 0 & 0 \end{pmatrix}$$

La contrainte en torsion s'exprime de la manière suivante $\tau_1 = \frac{M_{tmax}}{(\frac{I_0}{\nu})} = \frac{8 * P * D}{\pi * d^3}$ avec M_{tmax} le moment de torsion maximal et en notant ν la distance maximale entre G et M nous avons $\frac{I_0}{\nu} = \frac{\pi * d^3}{16}$ (ce rapport est appelé module de torsion, I_0 a cette valeur dans le cas d'une section droite circulaire de diamètre d).

La contrainte de flexion s'exprime de la manière suivante $\tau_2 = \frac{T}{S} = \frac{4 * P}{\pi * d^2}$ avec T l'effort tranchant. Ce qui nous donne comme contrainte totale

$$\tau_{tot} = \tau_1 + \tau_2 = \frac{8 * P * D}{\pi * d^3} * (1 + \frac{d}{2D}) \approx \frac{8 * P * D}{\pi * d^3}$$

En effet, mis à part pour les ressorts utilisés dans les véhicules (avec de très grande rigidité) on peut négliger le rapport $\frac{d}{D}$ (pour un ressort de rigidité moyenne $D \approx 6d$). Cette approximation entraîne seulement 8% d'erreur.

La condition de résistance sous charge maximale est $\tau_{tot} \leq \tau_e$ donc $\frac{8 * P_{max} * D}{\pi * d^3} \leq \tau_e$.

Nous pouvons donc connaître la force maximale applicable au ressort

$$\frac{\pi * \tau_e * d^3}{8 * D} \geq P_{max}$$

Or $P = k * (l - l_0)$ et comme déjà écrit précédemment $k = \frac{G * d^4}{8 * D^3 * n}$.

Donc $\frac{\pi * \tau_e * D^2 * n}{G * d} \geq l - l_0$.

Avec la même méthode ,nous obtenons un résultat identique (au signe près) pour les efforts en compressions. Nous pouvons donc en conclure que les masses vont osciller au maximum entre $\pm \frac{\pi * \tau_e * D^2 * n}{G * d}$ (par rapport à leur position d'origine).

Chapitre 6

Exemple de la liaison C-H

Liaison C-H.

$$r_0 = 109 \text{ pm (picomètre, } 10^{-12} \text{ mètre)}$$

$$m_{\text{carbone}} = 12,04u \iff 2,0 \cdot 10^{-26} \text{ kg } (1u \iff 1,66 \cdot 10^{-27} \text{ kg})$$

$$m_{\text{hydrogène}} = 1,007u \iff 1,67 \cdot 10^{-27} \text{ kg}$$

$$\mu = 1,54 \cdot 10^{-23} \text{ kg}$$

$$\sigma = 2974 \text{ cm}^{-1} = 2974 \cdot 10^2 \text{ m}^{-1}$$

$$k_{el} = (2\pi c\sigma)^2 \mu = 493 \text{ N} \cdot \text{m}^{-1}$$

Quelques valeurs pour les 4 premiers niveaux d'énergie :

	Potentiel harmonique	Potentiel de Morse
E_0	$3 \cdot 10^{-20} \text{ J} = 0.1844 \text{ eV}$	$2,95 \cdot 10^{-20} \text{ J} = 0.1844 \text{ eV}$
E_1	$9,10 \cdot 10^{-20} \text{ J} = 0.5632 \text{ eV}$	$8,66 \cdot 10^{-20} \text{ J} = 0.5409 \text{ eV}$
E_2	$1,5 \cdot 10^{-19} \text{ J} = 0.9388 \text{ eV}$	$1,41 \cdot 10^{-19} \text{ J} = 0.8812 \text{ eV}$
E_3	$2,1 \cdot 10^{-19} \text{ J} = 1.3143 \text{ eV}$	$1,93 \cdot 10^{-19} \text{ J} = 1.2056 \text{ eV}$

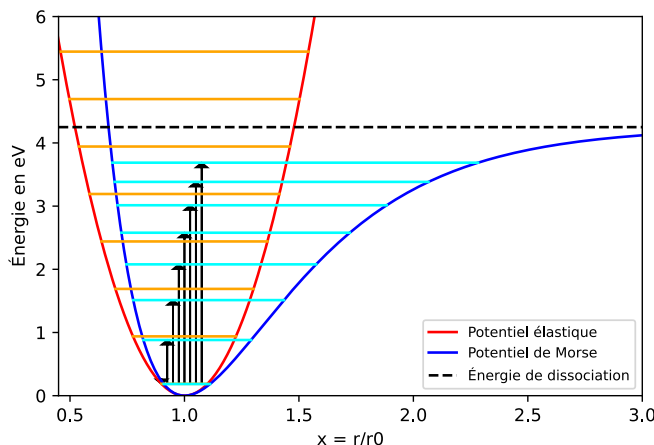


FIGURE 6.1 – Graphe d'énergie de la liaison C-H

Étude de la vibration au niveau d'énergie E_3

Si la vitesse initiale du système est nulle (voir annexe “ressort et moment angulaire” 7), les racines de l'équation $g(x) = E_n - E_p = 0$ seront nos conditions initiales pour évaluer la vibration de la liaison. Rappelons que l'abscisse de nos fonctions E_n et E_p étant normalisées, les 2 racines de la fonction $g(x)$ seront de la forme $x_1 = \frac{r_{min}}{r_0} < x_2 = \frac{r_{max}}{r_0}$ où r_{min} et r_{max} correspondent respectivement à la contraction maximale et l'extension maximale du ressort.

Par choix arbitraire, nous prendrons comme condition initiale une contraction du ressort, nous choisirons donc r_{min} .

Le programme Pascal exige une distance initiale par rapport au point d'équilibre de chaque masse. Nous calculons donc ces positions grâce à l'équation du centre de masse, r_a sera notre atome de carbone et r_b notre

atome d'hydrogène.

$$\Delta r_a = \frac{m_b}{m_a + m_b} (r_{min} - r_0)$$

$$\Delta r_b = \frac{-m_a}{m_a + m_b} (r_{min} - r_0)$$

Nous trouvons $\Delta r_a = 3pm$ et $\Delta r_b = -36pm$

Par simplification, nous considérerons la liaison seulement sur l'axe x, c'est à dire que nous n'attribuerons aucune partie imaginaire.

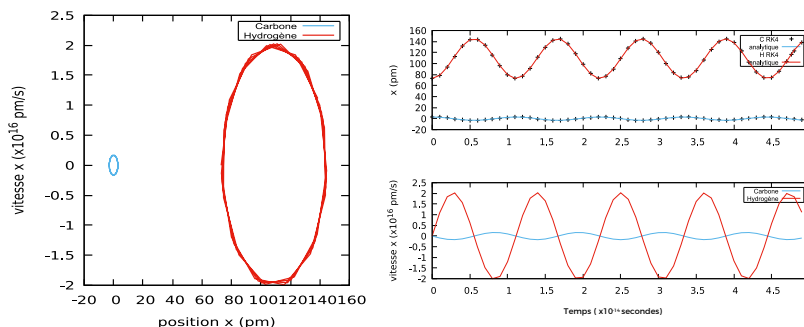


FIGURE 6.2 – Graphe de la liaison C-H un dt de 10^{-15} secondes sur un interval de 5.10^{-14} secondes

conditions initiales	Carbone	Hydrogène	longueurs à vide
position x (pm)	3	36	$l_{CHx} = 109$
position y (pm)	0	0	$l_{CHy} = 0$
vitesse x (pm)	0	0	
vitesse y (pm)	0	0	

L'atome d'hydrogène étant plus de 10 fois moins massif que l'atome de carbone, il aura beaucoup plus de facilité à se déplacer. Nous observons bien ce phénomène sur les graphes ci-dessus.

Nous allons maintenant essayer de modéliser cette liaison C-H par les valeurs d'un ressort 5.4. Pour ce faire, nous allons considérer le système comme deux ressorts reliés entre eux au niveau du centre de masse G de la liaison.

Nous fixons $d = 1pm$ (ce paramètre doit être de l'ordre de grandeur de l'oscillation). Nous partons des amplitudes trouvées dans le paragraphe ci-dessus, soient $\Delta r_a = 3.10^{-12}m$ et $\Delta r_b = -36.10^{-12}m$. Nous prenons $k_{el} = 493N.m^{-1}$ correspondant à la constante élastique qui modélise la liaison.

$$\tau_e = \frac{48k\Delta l}{\pi d^2}$$

En réalisant l'application numérique, nous obtenons une contrainte de $2,26 * 10^{16}Pa$ pour l'atome d'hydrogène et une contrainte de $2,71 * 10^{17}Pa$ pour l'atome de carbone. Nous remarquons que ces valeurs sont bien éloignées des valeurs de contraintes de ressorts que nous pouvons trouver à notre échelle.

Chapitre 7

Perspectives d'amélioration des modèles

Les modèles vus précédemment peuvent encore être améliorés. Nous avons abordé seulement les vibrations de liaisons simples. D'autres vibrations sont présentes comme la vibration de rotation ou encore de torsion. Ces modèles restent cependant très coûteux en terme de calcul et d'énergie de calcul comparés à notre simple modèle [5, 7, 3].

L'énergie totale dans un système chimique peut s'exprimer de la sorte :

$$E_{totale} = E_{liée} + E_{non-liée}$$

où les composantes des contributions covalentes et non-covalentes sont données par les sommes suivantes :

$$E_{liée} = E_{liaison} + E_{angle} + E_{torsion}$$

$$E_{non-liée} = E_{VanDerWalls} + E_{électrostatique}$$

Nous pouvons choisir la méthode AMBER pour calculer l'énergie dans notre système. Cette méthode est utilisée par les professionnels pour simuler des molécules complexes.

(Méthode AMBER)

L'énergie de liaison est modélisée par le potentiel harmonique, efficace aux alentours de l'équilibre :

$$E_{liaison} = \sum_{liaisons} \frac{1}{2} k_{liaison} (l - l_0)^2$$

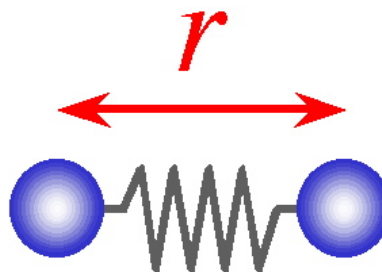


FIGURE 7.1 – Energie de liaison linéaire

Energie d'angle aussi modélisée par un potentiel harmonique :

$$E_{angle} = \sum_{angles} \frac{1}{2} k_a (\theta - \theta_0)^2$$

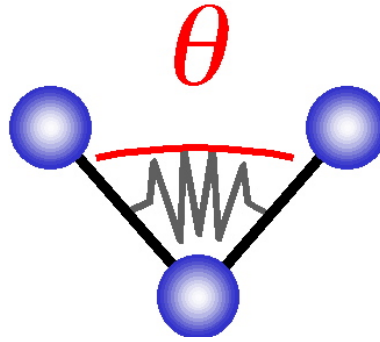


FIGURE 7.2 – Energie d'angle

Energie de torsion de la liaison (pour modéliser les liaisons doubles, triples)

$$E_{torsion} = \sum_{torsions} \frac{1}{2} v_n [1 + \cos(nw - \gamma)]$$

où v_n est la constante de torsion, w l'angle dièdre et γ l'angle de phase.

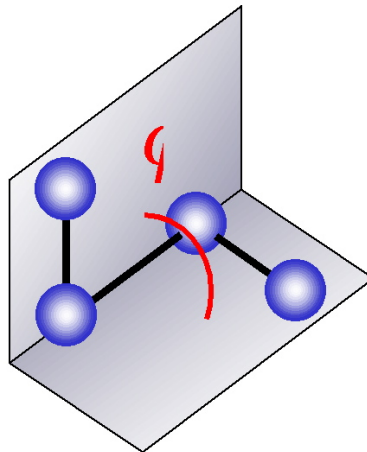


FIGURE 7.3 – Energie de torsion

Energie des liaisons de Van der Walls modélisée par le potentiel de Lennard-Jones

$$E_{VdW} = \sum_{j=1}^{N-1} \sum_{i=j+1}^N \{4\varepsilon_{i,j} [(\frac{\sigma_{ij}}{r_{ij}})^{12} - (\frac{\sigma_{ij}}{r_{ij}})^6]\}$$

Avec σ le nombre d'onde et $\varepsilon_{i,j}$ le niveau d'énergie le plus bas de la liaison.

Energie électrostatique

$$E_{electrostatique} = \sum_i \sum_{j>i} \frac{q_i q_j}{4\pi\varepsilon_0 r_{ij}}$$

où q_i et q_j sont les charges des atomes et ε_0 la permittivité du vide.

Toutes ces énergies proviennent de forces conservatives que nous pouvons obtenir avec le gradient. Les positions peuvent ensuite être simulées grâce au PFD et à la méthode Runge-Kutta 4 vue précédemment.

Ressentis et implications

— Yuhao

Pendant le projet p6, ce fût un peu difficile de communiquer avec les camarades en français, parfois nous échangeons en anglais ce qui fut plus pratique, je pouvais comprendre leurs pensées. De plus, nous avons fait une partie des calculs théoriques ensemble, et le professeur m'a suggéré d'utiliser le langage C pour simuler le mouvement du ressort afin d'obtenir les mêmes données que langage pascal. Comme je n'avais jamais appris le langage C, j'ai commencé à chercher des manuels sur internet pour apprendre le langage par moi-même. Lorsque nous avons besoin de simuler le mouvement de molécules chimiques, j'ai commencé à apprendre le cours de dynamique moléculaire, qui est un cours très difficile. Avec la coopération de mes camarades de projet, nous avons complété le modèle du ressort moléculaire. J'ai finalement poussé mes recherches dans la modélisations des molécules en étudiant et proposant les énergies à prendre en compte dans une molécule. Puis, j'ai fini sur des recherches bibliographiques pour comprendre le fonctionnement et la mise en place des spectroscopies infrarouges et de Raman

— Clément

Le projet de physique fut une expérience intéressante, je trouve malheureusement dommage le fait d'être restait longtemps bloqué sur des problèmes de résolution à la main, alors que des langages pouvait faire les calculs en quelques lignes. Ce temps aurait pu nous permettre d'approfondir sur des formes plus complexes et d'étudier beaucoup plus de cas. Mon travail au sein du groupe était assez omniprésent, je m'occupais d'abord de la mise en oeuvre des programmes Runge-Kutta en Pascal. La programmation s'est présenté comme une difficulté puisqu'il était difficile de savoir qui du programme ou des solutions analytiques affichaient les bons résultats, c'est pourquoi je me suis aussi penché sur des méthodes de résolutions analytiques et ainsi apporter mon aide à Fabio. Une fois les résultats cohérents, j'ai participé à la recherche bibliographique pour la réflexion et l'application du modèle à des molécules. Le module C1 vu en première année s'est révélé être d'une grande aide pour nous apporter des informations complémentaires, qui nous a permis de mieux comprendre ce qui était proposé dans les livres ou en ligne. Je suis ensuite parvenu à établir un lien entre vibration et oscillateur et à l'aide des courbes énergétiques, j'ai pu à l'aide de python déterminer toutes les conditions initiales nécessaires pour simuler la positions des masses sur Pascal et afficher les résultats sur gnuplot. J'ai ensuite mis en place le code python présenté en annexe pour illustrer nos propos et avoir des courbes d'énergie en fonction de nos donnés. Ce programme m'a permis d'être non seulement plus à l'aise mais aussi plus admiratif de ce langage en voyant toute les possibilités de calculs physiques et de graphiques rendu possible en quelques lignes. J'ai enfin grandement participé à la rédaction de ce projet et à la mise en place des fichiers Pascal et Gnuplot qui, j'espère, permettront aux années futures d'approfondir et d'observer toutes les choses que nous n'avons pas eu le temps d'observer.

— Fabio

Pour ma part, le projet P6 fut une expérience enrichissante qui s'est déroulée en plusieurs étapes. Tout d'abord, au début du projet, j'ai repris les exemples des systèmes masses-ressorts que nous avons pu traiter en première année. J'ai alors commencé à traiter le système avec deux masses fixées aux bords. Pour résoudre le système d'équations différentielles j'ai dû utiliser la diagonalisation pour connaître les pulsations d'oscillation puis les trajectoires des masses. J'ai également travaillé sur ces mêmes systèmes mais avec trois masses. J'obtenais alors des polynômes du troisième degré. J'ai donc utilisé la méthode de Tschirnhaus suivi de la méthode de Cardan pour trouver les solutions des pulsations. En revanche, ces résolutions sont très mathématiques et peuvent donc être négatives, complexes... et la détermination des solutions "physiques" à conserver est nettement plus compliquée à obtenir. Après ce premier système, je me suis concentré sur les systèmes libres avec l'hypothèse d'un centre de masse fixe. Lors de la résolution analytique de ces systèmes, j'ai essayé plusieurs méthodes de résolution à commencer par la diagonalisation comme précédemment, puis en utilisant la conservation de l'énergie. Mais cela compliquait beaucoup les résolutions. Puis finalement, avec l'aide de Clément nous avons

exploité le fait que nous connaissons déjà une partie de la pulsation. En effet, la pulsation dépend de la raideur k et d'une masse M . Ainsi en supposant connu la constante de raideur, il nous suffisait juste de déterminer M . Grâce à cette méthode, il a été bien plus facile de trouver les équations des trajectoires des masses en posant les conditions initiales. Enfin, en étudiant divers systèmes je me suis intéressé plus spécifiquement aux systèmes cycliques qui semblaient donner des relations répétitives. C'est alors que j'en ai empiriquement déduit une première formule et fait la démonstration de ce résultat pour le cas particulier des masses égales. Grâce à cela, nous pouvions donc déduire directement la pulsation suivie des trajectoires des masses d'un système cyclique. Après avoir effectué cette partie analytique, j'ai porté mon aide à Clément sur la partie chimique où nous avons passé du temps à reprendre le cours de C1 pour regarder les énergies qui intervenaient dans un système diatomique, pour pouvoir l'appliquer sur un exemple. Nous avons donc étudié les modèles de l'oscillateur harmonique puis celui de Morse en prenant l'exemple de la liaison CH. Cette partie applicative au domaine de la chimie fut très intéressante car elle nous a permis de nous rendre compte de l'utilité des simulations que nous avons faites bien que, sur des systèmes plus complexes, d'autres paramètres seraient à prendre en compte.

— Idris

De mon point de vue, le projet P6 a été un projet de groupe formidable mais extrêmement complexe. J'ai au début du projet commencé par reprendre ce que j'avais étudié lors des années précédentes concernant les systèmes masses ressorts. J'ai eu des difficultés à comprendre comment poser les équations à l'aide de la notation complexe et de comprendre l'hypothèse du centre de masse fixe; Clément m'a bien expliqué et j'ai pu l'épauler dans la résolution du système libre à 2 masses. Suite à ce premier système j'ai aidé Fabio sur les autres systèmes libres. Il m'a permis de mieux appréhender les matrices et la résolution à partir du déterminant. Après avoir essayé différentes géométries pour ces systèmes nous nous sommes rendus compte d'une certaines récurrences dans la forme du déterminant et j'ai voulu trouver une corrélation en fonction de la géométrie de la molécule (ce qui n'a rien donné et n'a donc pas servi). A ce moment là du projet, mes camarades étaient sur la simulation ou la modélisation chimique. Ces deux domaines (numérique et chimie) n'étant pas mes tasses de thé j'ai décidé de m'intéresser à la mécanique et à la résistance du ressort et comment celui-ci travaillais sous des efforts de traction ou de compression. La plus grande difficulté a été pour moi de comprendre la résolution car dans le cadre du cours de P9, nous ne voyons que le cas de la poutre et un ressort ne peut pas être assimilé à cette catégorie d'objet. Je me suis donc renseigné sur le sujet et ce fut un réel plaisir que d'apprendre par moi-même. Finalement, ce projet était un défi pour moi car j'ai habituellement du mal à comprendre et travailler par moi-même, en particulier si je ne sais pas quoi faire précisément. J'ai fini par prendre part à ce projet et je suis fier d'avoir pu y participer.

— Groupes

La répartition des tâches s'est faite naturellement selon les options du semestre 4 et les capacités de chacun. Yuhao était responsable de la partie Chimie, Clément de la partie Informatique, Fabio de la partie Mathématique et Idris de la partie Mécanique. Notre sujet et nos objectifs ont fait que nous avons pu nous répartir correctement les tâches au début du projet. Puis, les difficultés sont apparues et nous nous sommes rendu compte qu'avancer ensemble et s'aider entre nous a permis de terminer ce rapport de manière complète et précise. Les connaissances dans chacun des domaines et les points de vue de chacun ont beaucoup joué sur le bon fonctionnement de ce groupe.

Conclusion

Ce projet a permis de nous rendre compte qu'à partir d'un système très simple, nous pouvions déduire un grand nombre d'informations sur des choses invisibles à l'oeil et à peine perceptibles par des machines. Les perspectives pour ce projet sont immenses, le sujet offre un large champ d'études et d'observations possibles dû fait de ses libertés d'application. En partant d'un simple concept de ressort, nous avons pu modéliser le déplacement des atomes grâce aux outils informatiques mais surtout grâce à tous les concepts physiques établis durant ce siècle.

Disposant des bases de résolutions mathématiques et des codes pour simuler les atomes, nous espérons aux années futures de poursuivre notre projet et de se rendre compte du vaste champ possible d'étude à faire sur le sujet.

Annexes

Le spectre IR et le spectre de Raman

1. La spectroscopie infrarouge (IR)

Le principe d'une spectroscopie IR est d'envoyer des radiations infrarouges sur une molécule, une liaison chimique, un système... Certaines longueurs d'onde sont alors absorbées par les liaisons chimiques des molécules se trouvant dans l'échantillon. L'interprétation du spectre mettant en évidence la transmittance par rapport au nombre d'onde (inverse de la longueur d'onde), consiste à faire correspondre les bandes d'absorption avec les liaisons chimiques et les groupes caractéristiques correspondants de la molécule. Une spectroscopie infrarouge peut ainsi avoir différentes finalités :

- Identifier et caractériser une molécule inconnue, en référençant ses liaisons.
- Tester la présence ou l'absence d'une molécule dans un échantillon.
- Procéder à un dosage d'une espèce chimique

2. La spectroscopie Raman

La spectroscopie Raman est une technique de spectroscopie moléculaire qui repose sur l'interaction de la lumière avec la matière pour obtenir des informations sur la composition ou les caractéristiques de la matière étudiée. La spectroscopie Raman est une technique qui utilise la lumière diffusée pour observer les modes de vibrations d'une molécule. La spectroscopie Raman mesure le changement de couleur qui est renvoyé lorsque la lumière se disperse après avoir rencontré une molécule. Grâce à ce phénomène (effet Raman) on obtient un spectre traduisant l'intensité (récupérée par les photons dispersés) par rapport au nombre d'onde. De cette manière, nous obtenons d'une autre façon des caractéristiques sur les liaisons et les caractéristiques de l'échantillon.

3. Similarités et différences

La spectroscopie Raman, comme l'infrarouge, permet d'accéder aux niveaux vibrationnels des molécules. En revanche, les informations obtenues par spectroscopie Raman résultent d'un processus de diffusion de la lumière, alors que la spectroscopie IR repose sur l'absorption de lumière.

De plus les transitions vibrationnelles en infrarouge ne sont possible seulement si la molécule possède un moment dipolaire. Les vibrations des molécules diatomiques (O_2 ou H_2 par exemple) ne sont donc pas observables sur un diagramme IR, elles sont, au contraire, observées sur le diagramme de Raman[?].

Ressort et moment angulaire

Nous considérons 2 masses m_1, m_2 libres soumises à la force élastique du ressort les reliant. Le système possède un centre de masse noté G . Nous considérons un repère polaire I dont l'origine est G . $I\{\vec{G}, \vec{u}_r, \vec{u}_\theta, \vec{u}_z\}$, on suppose le repère $I\{\vec{G}, \vec{u}_r, \vec{u}_\theta, \vec{u}_z\}$ galiléen.

On note $\vec{r}_1 = r_1 \vec{u}_r$ la position de la masse 1 par rapport à I et $\vec{r}_2 = -r_2 \vec{u}_r$, la position de la masse 2.

On note $\vec{v}_1 = \dot{r}_1 \vec{u}_r + r_1 \dot{\theta}_1 \vec{u}_\theta$ la vitesse de la masse 1 par rapport à I et $\vec{v}_2 = \dot{r}_2 \vec{u}_r + r_2 \dot{\theta}_2 \vec{u}_\theta$ la vitesse de la masse 2.

Ainsi

$$m_1 \vec{r}_1 + m_2 \vec{r}_2 = 0$$

L'expression de l'énergie cinétique totale du système est donnée par $E_c = \frac{m_1+m_2}{2m_2} m_1 (r_1 \dot{\ }^2 + r_1^2 \dot{\theta}_1^2)$

L'expression de l'énergie potentielle de la force élastique totale du système est donnée par $E_p = \frac{k}{2} (\frac{m_1+m_2}{m_2} (r_1 - r_{10}))^2$

Le moment angulaire total du système est donné par $\Gamma = r_1^2 \dot{\theta}_1 (m_1 + m_2) \frac{m_1}{m_2}$.

Or le système n'est soumis qu'à des forces centrales, il n'y a donc aucun moment. Donc d'après le théorème du moment cinétique $\Gamma = \Gamma_0$, le moment angulaire du système à l'état initial. Ainsi nous pouvons substituer la valeur de $\dot{\theta}_1$ dans l'expression de l'énergie cinétique.

Finalement nous avons l'expression de l'énergie totale du système :

$$E = E_c + E_p = \underbrace{\frac{1}{2} m_1 \frac{m_1 + m_2}{m_2} r_1 \dot{\ }^2}_{E_c} + \underbrace{\frac{k}{2} (\frac{m_1 + m_2}{m_2} (r_1 - r_{10}))^2}_{E_{pel}} + \underbrace{\frac{\Gamma_0^2}{2m_1 (m_1 + m_2)} \frac{m_2}{r_1^2}}_{E_{peff}}$$

Nous avons donc un terme E_{peff} qui apparait s'il y a un moment cinétique, c'est à dire s'il y a une vitesse initiale non nul.

Code Python du diagramme de Morse et calcul des conditions initiales pour simuler la liaison avec Pascal

Exécuté sous jupyter notebook avec python 3

```
import matplotlib.pyplot as plt
import numpy as np
ma = 2 * 10**-26 #masse carbone
mb = 1.671 * 10**-27 #masse hydrogene
mu = (ma * mb) / (ma + mb)
sigma = 3000 * 10**2
c = 3.0 * 10**8
k = (2 * np.pi * sigma * c)**2 * mu #constante elastique
x0 = 109 * 10**-12 #longueur à vide
De = 4.25 #energie de dissociation en eV
DeJ = 6.8 * 10**-19 #energie de dissociation en Joules
nu = 9.0 * 10**13 #frequence de vibration à l'etat E0 en s^-1
alpha = np.sqrt(k / (2 * DeJ))

h = 6.63 * 10**-34 #constante de Plank
def f(x):
    return 0.5 * k * (x0*(x - 1))**2 / (1.6 * 10**-19) #potentiel elastique

def g(x):
    return De * (1 - np.exp(-alpha * (x0 * (x - 1))))**2 #potentiel de Morse

const = (0.5 * k * x0**2)/(1.6 * 10**-19)
E0 = (h / (2 * np.pi) * np.sqrt(k / mu) * (0.5)) / (1.6 * 10**-19)
I = np.linspace(0,5,num = 300)
a = 0.45
b = 3.0
dx = 0.05
plt.xlim(a,b)
plt.ylim(0,6)
plt.plot(I , f(I) , color='red' , label = 'Potentiel_élastique')
plt.plot(I , g(I) , color='blue' , label = 'Potentiel_de_Morse')
plt.axhline(y = De, linestyle = '--',color = 'black' , label = 'Énergie_de_dissociation')

#E_n de l'oscillateur harmonique
for i in range(8):
    En = (h / (2 * np.pi) * np.sqrt(k / mu) * (0.5 + 2 * i)) / (1.6 * 10**-19)
    coeff = [const * 1, -2 * const, const - En]
    t = np.roots(coeff)
    plt.axhline(y = En,xmin = (t[1] - a) / (b - a) , xmax = (t[0] - a) / (b - a) , color = 'orange')
    #a + xmin*d = r1
    #a + xmax*d = r2

#E_n de Morse
for i in range(8):
    En = ((h * nu) * (0.5 + 2 * i) - (h * nu * (0.5 + 2 * i))**2 / (4 * DeJ)) / (1.6 * 10**-19)
    r1 = - np.log(np.sqrt(En / De) + 1) / (alpha * x0) + 1
    r2 = - np.log(1 - (np.sqrt(En / De))) / (alpha * x0) + 1
    plt.axhline(y = En,xmin = (r1 - a) / (b - a) , xmax = (r2 - a) / (b - a) , color = 'aqua')
    plt.arrow((1 - 0.15) + dx , E0 , 0 , En - E0 , width = 0.001 , head_width = 0.05 ,
    color = 'black',length_includes_head = True)
    dx = dx + 0.025

plt.xlabel('x_ = r/r0')
plt.ylabel('Énergie_en_eV')
plt.legend(fontsize = 8 , loc = 4)
```

```
plt.show()
#plt.savefig('morse.svg')

i = 3
En = (h / (2 * np.pi) * np.sqrt(k / mu) * (0.5 + 2 * i)) / (1.6 * 10**-19)
coeff = [const * 1, -2 * const, const - En]
t = np.roots(coeff) #range les racines dans l'ordres croissant
rmin = t[0] * x0 #si on prend une compression, on prendra rmax pour une extension
deltaRa = mb * (rmin - x0) / (ma + mb)
deltaRb = -ma * (rmin - x0) / (ma + mb)
print(deltaRa, deltaRb)
```

Codes Pascal

Classe Complexe et Vecteur (à adapter suivant le nombre de masse)

```

type complexe = record
  re : Double;
  im : Double;
end;

type vecteur4 = record
  za : complexe;
  zb : complexe;
  vza : complexe;
  vzb : complexe;
end;

function addcomplexe(z1, z2 : complexe) : complexe;
begin
  addcomplexe.re := z1.re + z2.re;
  addcomplexe.im := z1.im + z2.im;
end;

function multcomplexe(z : complexe ; a : double) : complexe;
begin
  multcomplexe.re := a * z.re;
  multcomplexe.im := a * z.im;
end;

function addvecteur(v1, v2 : vecteur4) : vecteur4;
begin
  addvecteur.za := addcomplexe(v1.za, v2.za);
  addvecteur.zb := addcomplexe(v1.zb, v2.zb);
  addvecteur.vza := addcomplexe(v1.vza, v2.vza);
  addvecteur.vzb := addcomplexe(v1.vzb, v2.vzb);
end;

function multvecteur(v : vecteur4 ; a : double) : vecteur4;
begin
  multvecteur.za := multcomplexe(v.za, a);
  multvecteur.zb := multcomplexe(v.zb, a);
  multvecteur.vza := multcomplexe(v.vza, a);
  multvecteur.vzb := multcomplexe(v.vzb, a);
end;

```

Procédure de la méthode Runge Kutta 4

```

procedure rk4(var Y : Vecteur4 );
var Yb , Yc , Yd : Vecteur4;
      K1 , K2 , K3 , K4 , K : Vecteur4;
begin
  Yb := Y;
  Yc := Y;
  Yd := Y;

  K1 := applyf(Y);

  Yb := addvecteur(multvecteur(multvecteur(K1, dt), 0.5), Y);
  K2 := applyf(Yb);

  Yc := addvecteur(multvecteur(multvecteur(K2, dt), 0.5), Y);
  K3 := applyf(Yc);

  Yd := addvecteur(multvecteur(K3, dt), Y);
  K4 := applyf(Yd);

  K := addvecteur(K1, multvecteur(K2, 2.0));
  K := addvecteur(K, multvecteur(K3, 2.0));
  K := addvecteur(K, K4);
  K := multvecteur(K, dt / 6.0);

  Y := addvecteur(Y, K);
end;

```

Programme pour deux masses fixées

```

program deuxmassesfixe;

var ma , mb : double;
    t , k , wa_sqr , wb_sqr : double;
    W1, W2 : double;
    S,Q : double;
    Yn , Yanalytique : vecteur4;
    i : word;
    F : Text;
    za_0 , zb_0 , vza_0 , vzb_0 , l0 : complexe;
    A, B, C, D : complexe;

const dt = 0.1;

function applyf( A : Vecteur4) : Vecteur4; // équations différentielles
begin
    applyf.za := A.vza;
    applyf.zb := A.vzb;
    applyf.vza := multcomplexe(addcomplexe(A.zb , multcomplexe(A.za , -2.0)), wa_sqr);
    applyf.vzb := multcomplexe(addcomplexe(A.za , multcomplexe(A.zb , -2.0)), wb_sqr);
end;

procedure resultat(Y ,Yan : vecteur4 ; t : Double);
begin

    write(F,t:0:2); //temps

    write(F, ' ');
    write(F,Y.za.re:0:5); //RK4 position x
    write(F, ' ');
    write(F,l0.re + Y.zb.re:0:5);

    write(F, ' ');
    write(F,Y.za.im:0:5); //RK4 position y
    write(F, ' ');
    write(F,l0.im + Y.zb.im:0:5);

    write(F, ' ');
    write(F,Y.vza.re:0:5); //RK4 vitesse x
    write(F, ' ');
    write(F,Y.vzb.re:0:5);

    write(F, ' ');
    write(F,Y.vza.im:0:5); //RK4 vitesse y
    write(F, ' ');
    write(F,Y.vzb.im:0:5);

    write(F, ' ');
    write(F,Yan.za.re:0:5); //analytique position x
    write(F, ' ');
    write(F,l0.re + Yan.zb.re:0:5);

    write(F, ' ');
    write(F,Yan.za.im:0:5); //analytique position y
    write(F, ' ');
    writeln(F,l0.im + Yan.zb.im:0:5);
end;

function analytique(t : Double) : vecteur4;
begin
    analytique.za.re := A.re * cos(W1 * t) + B.re * sin(W1 * t) + C.re * cos(W2 * t) + D.re * sin(W2 * t);
    analytique.zb.re := -(C.re * cos(W2 * t) + D.re * sin(W2 * t))
        * (S + Q) / mb - (A.re * cos(W1 * t) + B.re * sin(W1 * t)) * (S - Q) / mb ;
    analytique.za.im := A.im * cos(W1 * t) + B.im * sin(W1 * t) + C.im * cos(W2 * t) + D.im * sin(W2 * t);
    analytique.zb.im := -(C.im * cos(W2 * t) + D.im * sin(W2 * t))
        * (S + Q) / mb - (A.im * cos(W1 * t) + B.im * sin(W1 * t)) * (S - Q) / mb ;
end;

procedure initialisation ();
begin
    ma := 0.1;
    mb := 1.0;

    k := 1.0;

```

```

za_0.re := 0;
za_0.im := 0;           //position par rapport à leur equilibre respective
zb_0.re := 0.2;
zb_0.im := 0;
vza_0.re := 0;         // vitesse initiale
vza_0.im := 0;
vzb_0.re := 0;
vzb_0.im := 0;

l0.re := 1;
l0.im := 0;

wa_sqr := k / ma;
wb_sqr := k / mb;

Q := sqrt(ma * ma + mb * mb - ma * mb);

W1 := sqrt(k * (ma + mb - Q) / (ma * mb));
W2 := sqrt(k * (ma + mb + Q) / (ma * mb));

S := ma - mb;

A := multcomplexe(addcomplexe(multcomplexe(za_0,Q+S), multcomplexe(zb_0,mb)), 1/(2*Q));
C := multcomplexe(addcomplexe(multcomplexe(za_0,Q-S), multcomplexe(zb_0,-mb)), 1/(2*Q));
B := multcomplexe(addcomplexe(multcomplexe(vzb_0,mb), multcomplexe(vza_0,Q+S)), 1/(2*W1*Q));
D := multcomplexe(addcomplexe(multcomplexe(vzb_0,-mb), multcomplexe(vza_0,Q-S)), 1/(2*W2*Q));

Yn.za := za_0;
Yn.zb := zb_0;
Yn.vza := vza_0;
Yn.vzb := vzb_0;

Yanalytique.za := za_0;
Yanalytique.zb := zb_0;
Yanalytique.vza := vza_0;
Yanalytique.vzb := vzb_0;
end;

begin
initialisation();
t := 0;
Assign(F, 'resultat.txt');
Rewrite(F);
for i := 1 to 150 do
begin
Yanalytique := analytique(t);
resultat(Yn, Yanalytique, t);
rk4(Yn);
t := t + dt;
end;
close(F);
end.

Programme deux masses libres

program deuxmasseslibre;

uses math;

var ma , mb : double;
k, wa_sqr, wb_sqr, l0x, l0y : double;
Yn , Yanalytique : vecteur4;
i : word;
t , mu, W : Double;
F : Text;
za_0 , zb_0, vza_0 , vzb_0 , A , B : complexe;
dt : double;

function applyf( A : Vecteur4) : Vecteur4; //equations différentielles
begin
applyf.za := A.vza;
applyf.zb := A.vzb;
applyf.vza := multcomplexe(addcomplexe(A.zb, multcomplexe(A.za, -1.0)), wa_sqr);
applyf.vzb := multcomplexe(addcomplexe(A.za, multcomplexe(A.zb, -1.0)), wb_sqr);
end;

```

```

procedure resultat(Y ,Yan: vecteur4 ; t : Double);
begin
  write(F,t);
  write(F,'_');

  write(F,Y.za.re);
  write(F,'_'); //RK4 position x
  write(F,10x + Y.zb.re);
  write(F,'_');

  write(F,Y.za.im);
  write(F,'_'); //RK4 poition y
  write(F,10y + Y.zb.im);
  write(F,'_');

  write(F,Y.vza.re);
  write(F,'_'); //RK4 vitesse x
  write(F,Y.vzb.re);
  write(F,'_');

  write(F,Y.vza.im);
  write(F,'_'); //RK4 vitesse y
  write(F,Y.vzb.im);
  write(F,'_');

  write(F,Yan.za.re);
  write(F,'_'); //analytique position x
  write(F,10x + Yan.zb.re);
  write(F,'_');

  write(F,Yan.za.im);
  write(F,'_'); //analytique poition y
  writeln(F,10y + Yan.zb.im);
end;

function analytique(t : Double ) : vecteur4;
begin
  analytique.za.re := A.re * cos(W * t) + B.re * sin(W * t);
  analytique.za.im := A.im * cos(W * t) + B.im * sin(W * t);
  analytique.zb.re := - (ma / mb) * analytique.za.re;
  analytique.zb.im := - (ma / mb) * analytique.za.im;
end;

procedure initialisation();
var c, sigma : double;
begin
  ma := 12.0; // en unité u (poids atomique)
  mb := 1.0;
  sigma := 3000.0 * power(10,2); //en mètre-1
  10x := 109.0; //en picomètre longueur à vide liaison
  10y := 0.0;

  za_0.re := 3.0; //en picomètre
  za_0.im := 0.0; //position par rapport à leur équilibre respective
  zb_0.re := - (ma / mb) * za_0.re;
  zb_0.im := - (ma / mb) * za_0.im;

  vza_0.re := 0.0;
  vza_0.im := 0.0; //vitesse initiale en pm/s
  vzb_0.re := 0.0;
  vzb_0.im := 0.0;

  dt := power(10,-15); // interval de calcul entre les points
  //((doit être de l'ordre_de_la_période)
  mu_:=_(ma*_mb)^(ma+_mb);
  c_:=_3.0*_power(10,8);
  k_:=_power(2*_pi*_c*_sigma,2)^(*_mu);

  wa_sqr_:=_k/ma;
  wb_sqr_:=_k/mb;
  W_:=_sqrt(k/mu);

  A_:=_za_0;
  B_:=_multcomplexe(vza_0_,_1/W);

```

```

Yn.za_:=_za_0;
Yn.zb_:=_zb_0;
Yn.vza_:=_vza_0;
Yn.vzb_:=_vzb_0;

Yanalytique.za_:=_za_0;
Yanalytique.zb_:=_zb_0;
Yanalytique.vza_:=_vza_0;
Yanalytique.vzb_:=_vzb_0;
end;

begin
initialisation();
t_:=_0;
Assign(F,'resultat.txt');
Rewrite(F);
for i_:=_1_ to _50_ do
begin
Yanalytique_:=_analytique(t);
resultat(Yn,Yanalytique,t);
rk4(Yn);
t_:=_t_+_dt;
end;
close(F);
end.

```

Programme triangle

```

program troismasseslibre;

uses math;

type complexe = record
  re : Double;
  im : Double;
end;

type vecteur6 = record
  za : complexe;
  zb : complexe;
  zc : complexe;
  vza : complexe;
  vzb : complexe;
  vzc : complexe;
end;

var ma , mb , mc , l12x , l13x , l12y , l13y : double;
  k,wa_sqr,wb_sqr,wc_sqr : Single;
  Yn , Yanalytique : vecteur6;
  i : word;
  t , mu ,M , W: Double;
  F : Text;
  za_0 , zb_0, zc_0, vza_0 , vzb_0, vzc_0 ,A,B,C,D : complexe;
const dt = 0.5;

function addcomplexe(z1,z2 : complexe) : complexe;
begin
addcomplexe.re := z1.re + z2.re;
addcomplexe.im := z1.im + z2.im;
end;

function multcomplexe(z : complexe ; a : double) : complexe;
begin
multcomplexe.re := a * z.re;
multcomplexe.im := a * z.im;
end;

function addvecteur(v1,v2 : vecteur6) : vecteur6;
begin
addvecteur.za := addcomplexe(v1.za,v2.za);
addvecteur.zb := addcomplexe(v1.zb,v2.zb);
addvecteur.zc := addcomplexe(v1.zc,v2.zc);
addvecteur.vza := addcomplexe(v1.vza,v2.vza);
addvecteur.vzb := addcomplexe(v1.vzb,v2.vzb);
addvecteur.vzc := addcomplexe(v1.vzc,v2.vzc);
end;

```



```

function multvecteur(v : vecteur6 ; a : double) : vecteur6 ;
begin
multvecteur.za := multcomplexe(v.za , a) ;
multvecteur.zb := multcomplexe(v.zb , a) ;
multvecteur.zc := multcomplexe(v.zc , a) ;
multvecteur.vza := multcomplexe(v.vza , a) ;
multvecteur.vzb := multcomplexe(v.vzb , a) ;
multvecteur.vzc := multcomplexe(v.vzc , a) ;
end ;

function applyf( A : Vecteur6) : Vecteur6 ; // equation differentiel
begin
applyf.za := A.vza ;
applyf.zb := A.vzb ;
applyf.zc := A.vzc ;
applyf.vza := multcomplexe(addcomplexe(A.zc , addcomplexe(A.zb , multcomplexe(A.za , -2.0))) , wa_sqr) ;
applyf.vzb := multcomplexe(addcomplexe(A.zc , addcomplexe(A.za , multcomplexe(A.zb , -2.0))) , wb_sqr) ;
applyf.vzc := multcomplexe(addcomplexe(A.zb , addcomplexe(A.za , multcomplexe(A.zc , -2.0))) , wc_sqr) ;
end ;

procedure rk4(var Y : Vecteur6 ) ;
var Yb , Yc , Yd : Vecteur6 ;
    K1 , K2 , K3 , K4 , K : Vecteur6 ;
begin
Yb := Y ;
Yc := Y ;
Yd := Y ;

K1 := applyf(Y) ;

Yb := addvecteur(multvecteur(multvecteur(K1 , dt) , 0.5) , Y) ;
K2 := applyf(Yb) ;

Yc := addvecteur(multvecteur(multvecteur(K2 , dt) , 0.5) , Y) ;
K3 := applyf(Yc) ;

Yd := addvecteur(multvecteur(K3 , dt) , Y) ;
K4 := applyf(Yd) ;

K := addvecteur(K1 , multvecteur(K2 , 2.0)) ;
K := addvecteur(K , multvecteur(K3 , 2.0)) ;
K := addvecteur(K , K4) ;
K := multvecteur(K , dt / 6.0) ;

Y := addvecteur(Y , K) ;
end ;

procedure resultat(Y , Yan : vecteur6 ; t : Double) ;
begin
write(F , t : 0 : 2) ;
write(F , ' ') ;

write(F , Y.za.re : 0 : 5) ;
write(F , ' ') ;
write(F , 112x + Y.zb.re : 0 : 5) ; //rk4 position x
write(F , ' ') ;
write(F , 113x + Y.zc.re : 0 : 5) ;
write(F , ' ') ;

write(F , Y.za.im : 0 : 5) ;
write(F , ' ') ;
write(F , 112y + Y.zb.im : 0 : 5) ; //rk4 position y
write(F , ' ') ;
write(F , 113y + Y.zc.im : 0 : 5) ;
write(F , ' ') ;

write(F , Y.vza.re : 0 : 5) ;
write(F , ' ') ;
write(F , Y.vzb.re : 0 : 5) ; //rk4 vitesse x
write(F , ' ') ;
write(F , Y.vzc.re : 0 : 5) ;
write(F , ' ') ;

write(F , Y.vza.im : 0 : 5) ;
write(F , ' ') ;

```

```

write(F,Y.vzb.im:0:5);           //rk4      vitesse y
write(F,' ');
write(F,Y.vzc.im:0:5);
write(F,' ');

write(F,Y.za.re:0:5);
write(F,' ');
write(F,l12x + Y.zb.re:0:5);     //analytique      position y
write(F,' ');
write(F,l13x + Y.zc.re:0:5);
write(F,' ');

write(F,Y.za.im:0:5);
write(F,' ');
write(F,l12y + Y.zb.im:0:5);     //analytique      position y
write(F,' ');
writeln(F,l13y + Y.zc.im:0:5);
end;

function analytique(t : Double) : vecteur6;
begin
analytique.za.re := A.re * cos(W * t) + B.re * sin(W * t);
analytique.za.im := A.im * cos(W * t) + B.im * sin(W * t);
analytique.zb.re := C.re * cos(W * t) + D.re * sin(W * t);
analytique.zb.im := C.im * cos(W * t) + D.im * sin(W * t);
analytique.zc.re := - (ma / mc) * analytique.za.re - (mb / mc) * analytique.zb.re;
analytique.zc.im := - (ma / mc) * analytique.za.im - (mb / mc) * analytique.zb.im;
end;

procedure initialisation ();
begin
ma := 1;
mb := 1;
mc := 1;
k := 0.1;
l12x := 1.0;
l12y := 0;
l13x := 0.5;
l13y := 0.75;

za_0.re := -0.2;
za_0.im := -0.2;           //position par rapport à leur equilibre respective
zb_0.re := 0.2;
zb_0.im := -0.2;
zc_0.re := - (ma / mc) * za_0.re - (mb / mc) * zb_0.re;
zc_0.im := - (ma / mc) * za_0.im - (mb / mc) * zb_0.im;;
vza_0.re := 0;
vza_0.im := 0;
vzb_0.re := 0;
vzb_0.im := 0;
vzc_0.re := 0;
vzc_0.im := 0;

wa_sqr := k/ma;
wb_sqr := k/mb;
wc_sqr := k/mc;

mu := (ma * mb * mc) / (ma + mb + mc);
M := sqrt(mu / 3);
W := sqrt(k / M);

A := za_0;
B := multcomplexe(vza_0 , 1/W);
C := zb_0;
D := multcomplexe(vzb_0 , 1/W);

Yn.za := za_0;
Yn.zb := zb_0;
Yn.zc := zc_0;
Yn.vza := vza_0;
Yn.vzb := vzb_0;
Yn.vzc := vzc_0;

Yanalytique.za := za_0;
Yanalytique.zb := zb_0;

```

```

Yanalytique.zc := zc_0;
Yanalytique.vza := vza_0;
Yanalytique.vzb := vzb_0;
Yanalytique.vzc := vzc_0;
end;

begin
initialisation ();

t := 0;
Assign(F, 'resultat.txt');
Rewrite(F);
for i := 1 to 50 do
begin
Yanalytique := analytique(t);
resultat(Yn, Yanalytique, t);
rk4(Yn);
t := t + dt;
end;
close(F);
end.

```

Quelques informations complémentaires concernant l'échelle du code et le choix du langage de programmation

Pour ne pas avoir à travailler avec des puissances de 10 proche de 0 et pour avoir des graphes plus plaisant à regarder, nous avons décidé de se mettre à l'échelle de l'atome, c'est à dire de transmettre au programme les longueurs en picomètre et les masses en unité atomique. Cependant, les principales liaisons de nos molécules vibrent dans l'infrarouge. Autrement dit, la période du signal est très courte. Ainsi, pour la méthode Rung Kutta, il faut faire attention à choisir un intervalle dt suffisamment petit, d'un ordre inférieur à la période de vibration pour ne pas poser de problème informatique. Pour notre liaison C-H, nous calculons une période T de l'ordre de 10^{-14} secondes. Ainsi, un dt inférieur à T , de l'ordre de 10^{-15} secondes est un bon choix pour ne provoquer aucune erreur de dépassement float lors de l'exécution du programme.

Concernant le code, nous avons d'abord choisis de programmer en Pascal pour produire des données. Pascal est un langage facile à appréhender et rapide à l'exécution une fois que le programme est compilé. Pour afficher les données sur un graphe, nous avons choisis Gnuplot pour sa simplicité, Gnuplot est un puissant logiciel pour afficher des données brutes. Cependant, Python s'est révélé plus efficace et concis que Pascal et Gnuplot associés pour calculer et afficher des graphes plus complexes. De plus, Python peut être utilisé avec des bibliothèques de calculs formels, c'est à dire sans donner de valeur à une variable. Ce qui a permis de gagner en temps et en efficacité pour les résolutions analytiques.

Bibliographie

- [1] P Agati, F Lerouge, and N Mattéra. *Mécanique appliquée*. Dunod. 5.4
- [2] Yvan Boulanger. Spectroscopie vibrationnelle : Infrarouge et Raman. 4
- [3] CEA. Brève introduction à la mécanique ({MM}) et à la dynamique moléculaires (DM), January 2021. 7
- [4] Guy Colin and Marc Couture. Chimie / physique - Module 2 - Théorie. 4, 5.1
- [5] Wikipedia Foundation. Potentiel interatomique, November 2020. 7
- [6] Wikipedia Foundation. Vibration moléculaire, October 2020. 4
- [7] Wikipedia Foundation. Champ de force (chimie), November 2021. 7
- [8] Wikipedia Foundation. Dynamique moléculaire, March 2022. 5.1
- [9] KhanAcademyFrancophone. Modélisation des liaisons par des ressorts. 5.1
- [10] Claude Raclot. La spectrométrie infrarouge, modèle harmonique de la liaison de valence concours général 2009. 5.1