

## **Rapport de Stage**

Emmanuel Martins Seromenho  
Master 1 Physique Fondamentale et Applications

---

Modélisation de la dynamique de nuage d'ions et de leurs oscillations plasma

---

Sous la direction du Professeur Laurent Hilico

Laboratoire Kastler Brossel  
Sorbonne Université campus Pierre et Marie Curie

# Remerciements

Je souhaite remercier mon maître de stage le Professeur Laurent Hilico de m'avoir accepté comme stagiaire et aussi de m'avoir consacré beaucoup de temps pour m'aider pendant ce stage. Je remercie aussi Jean-Philippe Karr de m'avoir mis en relation avec le Professeur Hilico.  
Je remercie aussi Thomas Louvradoux pour m'avoir guidé pour les écritures des programmes Python.

## Table des matières

<b>Introduction</b>	<b>3</b>
<b>Sujet</b>	<b>3</b>
<b>Outils</b>	<b>4</b>
<b>1. Piège radio-fréquence</b>	<b>4</b>
1.1. Piège de Paul linéaire	5
1.2. Le potentiel de piégeage	5
1.3. Les équations de Mathieu	6
<b>2. Le nuage d'ions</b>	<b>8</b>
2.1. Simulation numérique de ce nuage	8
2.2. Pression de radiation sur le nuage	8
2.3. Modes de vibration	9
2.4. Addition d'un mode de perturbation	10
2.5. Simulation des fréquences de résonance	11
<b>3. Résultats</b>	<b>12</b>
3.1 Interprétation des graphiques	12
3.2. Les fréquences de résonance de chaque mode plasma	13
<b>Conclusion</b>	<b>14</b>
<b>Bibliographie</b>	<b>15</b>
<b>Annexes</b>	<b>16</b>
Annexe 1 : Résolution numérique des domaines de stabilité	16
Annexe 2 : Résolution numérique des équations de Mathieu	17
Annexe 3 : Perturbations sur les vitesses pour chaque mode plasma étudié	18
Annexe 4 : Constante pour chaque mode plasma	18
Annexe 5 : Schéma montrant le nuage d'ions soumis à la pression de radiation due au laser	19
Annexe 6 : Détermination des positions d'équilibre	19
Annexe 7 : Programme pour obtenir le spectre des moments multipolaires	20

# Introduction

Ce stage a été effectué au sein du laboratoire Kastler Brossel situé à Sorbonne Université campus Pierre et Marie Curie. Le groupe qui m'a accueilli est le groupe d'« Ions piégés ». L'activité de ce groupe se concentre sur la métrologie de haute précision en utilisant des ions froids piégés. Ils étudient particulièrement l'ion moléculaire  $H_2^+$  que ce soit dans l'aspect théorique ou expérimental en utilisant une méthode de spectroscopie à haute résolution. Le but de leurs recherches est de tester les théories fondamentales et d'améliorer la précision des constantes physiques fondamentales. La motivation principale de leurs travaux est de contribuer à la détermination du rapport entre la masse du proton sur la masse de l'électron ( $\mu = m_p/m_e$ ). Le groupe fait aussi partie de la collaboration internationale GBAR (étude du comportement gravitationnel de l'antimatière). Les modélisations de nuage d'ion que j'ai faites sont liées à ce dernier sujet. Pour GBAR, un ion  $\bar{H}^+$  composé d'un antiproton et de deux positrons devra être capturé et refroidi par un nuage de  $Be^+$  refroidi par laser.

Ce groupe est composé du Professeur Laurent Hilico, qui est mon maître de stage, l'enseignante chercheuse Albane Douillet et de l'enseignant chercheur Jean-Philippe Karr. Il y a aussi la présence d'un post doctorant, Thomas Louvradoux et d'un doctorant, Mohammad Haidar.

## Sujet

Le stage de Master initialement proposé portait sur le piégeage et le refroidissement laser d'ions pour la spectroscopie à haute résolution d'ions moléculaires. Le stage a commencé début avril, pendant le confinement, et d'un commun accord, nous avons redéfini un sujet autour de la modélisation de la dynamique de nuage d'ions et de leurs oscillations plasma.

Pour faire la spectroscopie à haute résolution, l'ion moléculaire  $H_2^+$  doit être refroidi. Le refroidissement par laser de  $H_2^+$  est impossible. Par conséquent l'ion moléculaire doit être refroidi par refroidissement sympathique c'est-à-dire que d'autres particules de nature différente vont refroidir les ions  $H_2^+$ . L'ion Béryllium  $Be^+$  sera ainsi utilisé pour refroidir  $H_2^+$  car cet ion est facile à créer. Des ions  $Be^+$  sont créés dans le piège par impact électronique sur un jet effusif de  $Be$  neutre. Le piège oblige les ions à rester ensemble. Le laser va refroidir le nuage formé d'ions Béryllium en retirant l'énergie cinétique de celui-ci. Le nuage d'ions  $Be^+$  est maintenant prêt pour refroidir sympathiquement l'ion  $\bar{H}^+$ . Pour que  $\bar{H}^+$  soit refroidi, il doit passer dans le nuage d'ions. En passant dans le nuage,  $\bar{H}^+$  va donc le perturber et le plasma va commencer à osciller.

Un article de Leaf Tuner [8] nous montre qu'un plasma situé dans un piège avec un certain potentiel aura une forme qui dépendra du potentiel effectif. Dans le piège de Paul linéaire qu'on utilise, le potentiel appliqué est un potentiel radio-fréquence quadripolaire. L'observation montre que la forme du nuage d'ions  $Be^+$  a une forme sphéroïdale. La théorie de Leaf Tuner nous confirme bien cette forme.

Cette forme sphéroïdale nous permet de faire un lien avec l'article [5]. En effet, l'article nous expose l'idée qu'un plasma froid magnétisé de forme sphéroïdale dans un piège de Penning et un plasma froid non magnétisé de même forme dans un piège de Paul linéaire (notre cas pendant ce stage) suivent la même équation de fréquences plasma. Ainsi nous connaissons la théorie qui permet de déterminer les fréquences plasma. De plus, cette théorie nous montre qu'il existe des modes plasmas c'est-à-dire des configurations spécifiques de vibration. Notre but va être de retrouver cette théorie expérimentalement (*i.e.* numériquement dans notre cas).

Les articles de Daniel H. E. Dubin (*cf.* [6] et [7]), nous exposent la théorie décrivant les modes d'oscillation d'un plasma sphéroïdal en utilisant des coordonnées elliptiques, et donne les formules permettant d'exciter et de détecter un mode plasma donné dans les simulations numériques. Il suffit juste d'appliquer le gradient à ces équations pour un mode plasma voulu pour obtenir les équations de perturbation.

Le but de ce stage sera d'identifier numériquement les résonances plasma après avoir perturbé le nuage avec un certain mode  $(l, m)$ .

Les résultats obtenus pendant ce stage auront pour but de savoir si un mode d'un plasma de  $Be^+$  peut avoir une fréquence propre égale à la fréquence d'oscillation d'un ion de masse  $1u$  dans la piège, afin d'avoir un bon couplage et donc un bon refroidissement de l'ion léger. L'idée sera d'utiliser un couplage résonant pour le refroidissement sympathique d'un ion de masse  $1u$  (*i.e.* un proton ou  $\bar{H}^+$ ) par des ions  $Be^+$  refroidis par

laser. Ainsi les modes  $(l, m = 0)$  ne conviendront pas pour la suite. En effet il faut considérer des modes  $(l, m = 1)$ . Le travail sera d'identifier numériquement les résonances plasma avant d'essayer numériquement le refroidissement sympathique.

## Outils

Pour répondre à la problématique, nous n'avons utilisé que des outils informatiques. En effet, le stage s'est articulé autour de la modélisation numérique du nuage d'ions et du traitement numérique des résultats. Les outils utilisés ont été :

- Le langage informatique Fortran. En effet, le programme qui modélisait le nuage d'ions dans le piège de Paul linéaire était déjà écrit en ce langage. J'ai dû donc apprendre ce langage pendant le stage.
- Le langage Python. De nombreux programmes ont été écrits en Python car cela nous a permis de tracer les courbes liées aux équations théoriques et d'exploiter les fichiers de données obtenues à l'aide du programme Fortran.
- Les calculateurs du laboratoire. Grâce à l'émulateur de terminal Putty j'ai pu utiliser ces calculateurs à distance. Cela m'a permis d'utiliser un nouvel outil de compilation : `gfortran`. Nous avons eu aussi l'utilité de la librairie de parallélisation `openmp`.
- Nous avons aussi utilisé Slurm qui est un outil de gestion de queues de calcul. Le script `slurm` se lance avec l'instruction `sbatch` (fichier `batch` est un terme de jargon qui désigne un petit fichier qui contient des paramètres utilisés par une application, ici `sbatch` va lire quoi faire dans `dyn-molec-script-para`)
- Les visioconférences. Compte tenu de la situation de confinement, nous avons dû trouver de nouveaux moyens de communication. Alors nous avons fait régulièrement des visioconférences pour partager les idées concernant le stage.

## 1. Piège radio-fréquence

Considérons un potentiel RF (Radio-Fréquence) quadripolaire idéal de forme générale :

$$\Phi_{RF}(x, y, z, t) = \frac{U(t)}{r_0^2} (Ax^2 + By^2 + Cz^2), \quad (1)$$

avec  $U(t)$  le potentiel radio-fréquence,  $r_0$  une dimension caractéristique du piège considéré et  $A, B$  et  $C$  des constantes adimensionnées. Ce potentiel doit obéir à l'équation de Laplace  $\Delta\Phi_{RF} = 0$  si le champ est de nature électrostatique et varie dans l'ARQS (Approximation des Régimes Quasi Stationnaire) où la taille du circuit (ici le piège,  $\sim 3,5$  à  $12$  mm) est très petite devant la longueur d'onde associée à  $U(t)$  (ici,  $RF = 13$  MHz et  $\lambda \approx 20$  m). Ainsi, on doit avoir :

$$A + B + C = 0. \quad (2)$$

Cette équation montre l'impossibilité de confiner des particules chargées dans les trois directions de l'espace car le potentiel est nécessairement déconfinant dans une des trois directions de l'espace c'est-à-dire qu'au moins une des constantes de l'équation (2) doit être négative pour satisfaire l'équation.

En 1958, le physicien allemand Wolfgang Paul a eu l'idée d'utiliser un potentiel électrique  $U(t)$  changeant de signe périodiquement en fonction du temps pour confiner des ions dans un potentiel quadripolaire. Grâce à cette idée, il est colauréat avec Hans Georg Dehmelt du prix Nobel de physique en 1989.

Lorsqu'une direction de l'espace est déconfinant à un instant donné, la demi-période suivante, cette même direction de l'espace devient confinante et ainsi de suite en fonction du temps.

Il existe différentes configurations possibles mais 2 sont couramment utilisées. Il y a le piège de Paul hyperbolique avec  $(A = 1, B = 1, C = -2)$  et le piège de Paul linéaire avec  $(A = 1, B = -1, C = 0)$ .

Dans la suite, nous allons seulement nous intéresser au piège de Paul linéaire.

## 1.1. Piège de Paul linéaire

Le principe de ce piège consiste à supprimer le champ radio-fréquence sur l'une des directions de l'espace, notée  $z$ . On a donc  $C = 0$ .

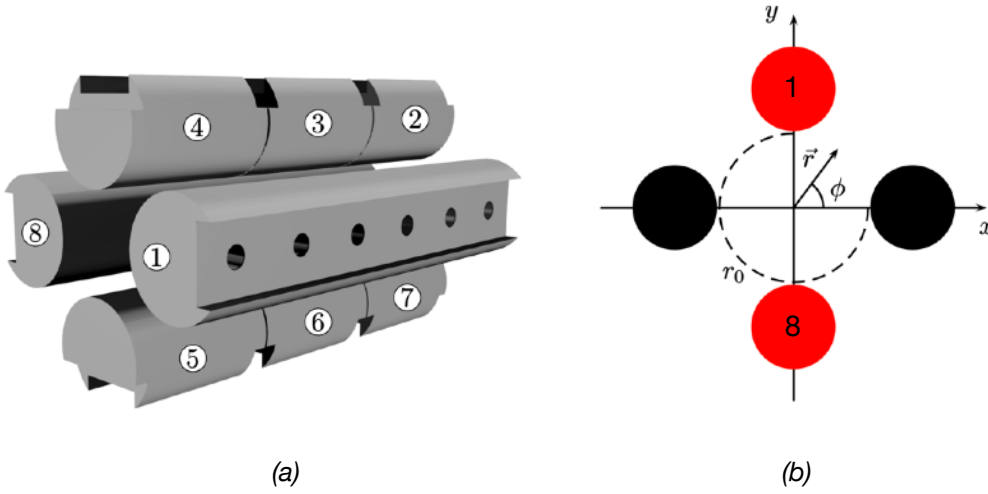


Figure 1 - (a) Représentation d'un piège de Paul linéaire, tirée de [3].  
(b) Schéma d'un piège de Paul linéaire dans le plan radial, tirée de [2].

Sur la *Figure 1*, **1** et **8** sont les électrodes RF. **2**, **4**, **5** et **7** sont les électrodes endcap. **3** et **6** sont les électrodes centrales. Les électrodes RF sont alimentées avec une tension  $V_{RF} = U_0 + V_0 \cos(\Omega t)$  et les électrodes endcap sont alimentées avec un voltage constant  $U_1 \approx 10V$ .

## 1.2. Le potentiel de piégeage

En connaissant le potentiel nous pouvons en déduire le champ électrique, puis la force et ainsi les équations du mouvement qui régissent le système.

Pour des électrodes d'une longueur  $L$  ( $L \gg r_0$ ), le potentiel radio-fréquence quadripolaire au voisinage du centre du piège peut s'écrire :

$$\Phi_{RF}(x, y, t) = \frac{x^2 - y^2 + r_0^2}{2r_0^2} (U_0 + V_0 \cos(\Omega t)). \quad (3)$$

On obtient le champ électrique en appliquant le gradient à  $\Phi_{RF}$  tel que  $\vec{E} = -\vec{\nabla} \Phi_{RF}$ . Le champ électrique s'écrit :

$$\vec{E} = \frac{U_0 + V_0 \cos(\Omega t)}{r_0^2} (-x\vec{x} + y\vec{y}). \quad (4)$$

La force exercée sur les ions de charge  $Q$  par le piège est :  $\vec{F} = Q\vec{E}$ . En utilisant la loi de Newton  $\vec{F} = m\vec{a}$ , on obtient deux équations différentielles (non couplées) du second ordre sans second membre :

$$Q \frac{U_0 + V_0 \cos(\Omega t)}{r_0^2} \begin{bmatrix} -x \\ y \end{bmatrix} = m \begin{bmatrix} \ddot{x} \\ \ddot{y} \end{bmatrix}. \quad (5)$$

Ainsi les équations du mouvement d'un ion soumis à la force  $\vec{F}$  dans le plan  $xy$  sont données par :

$$\ddot{x} = -\frac{Q}{mr_0^2} (U_0 + V_0 \cos(\Omega t)) x, \quad (6)$$

$$\ddot{y} = \frac{Q}{mr_0^2} (U_0 + V_0 \cos(\Omega t)) y. \quad (7)$$

### 1.3. Les équations de Mathieu

On peut réécrire les équations (6) et (7) sous la forme d'équations de Mathieu (i.e.  $\frac{d^2x}{dt^2} + G(t)x = 0$  avec  $G(t)$  fonction périodique) :

$$\frac{d^2x}{d\tau^2} + (a_x + 2q_x \cos(\Omega t)) x = 0, \quad (8)$$

$$\frac{d^2y}{d\tau^2} + (a_y + 2q_y \cos(\Omega t)) y = 0. \quad (9)$$

On définit ainsi des paramètres adimensionnés  $(a_x, q_x)$  et  $(a_y, q_y)$  pour chaque composante, et un temps réduit  $\tau$  tels que :

$$\tau = \frac{\Omega t}{2}, \quad (10)$$

$$a_x = -a_y = \frac{4QU_0}{m\Omega^2 r_0^2}, \quad (11)$$

$$q_x = -q_y = \frac{2QV_0}{m\Omega^2 r_0^2}. \quad (12)$$

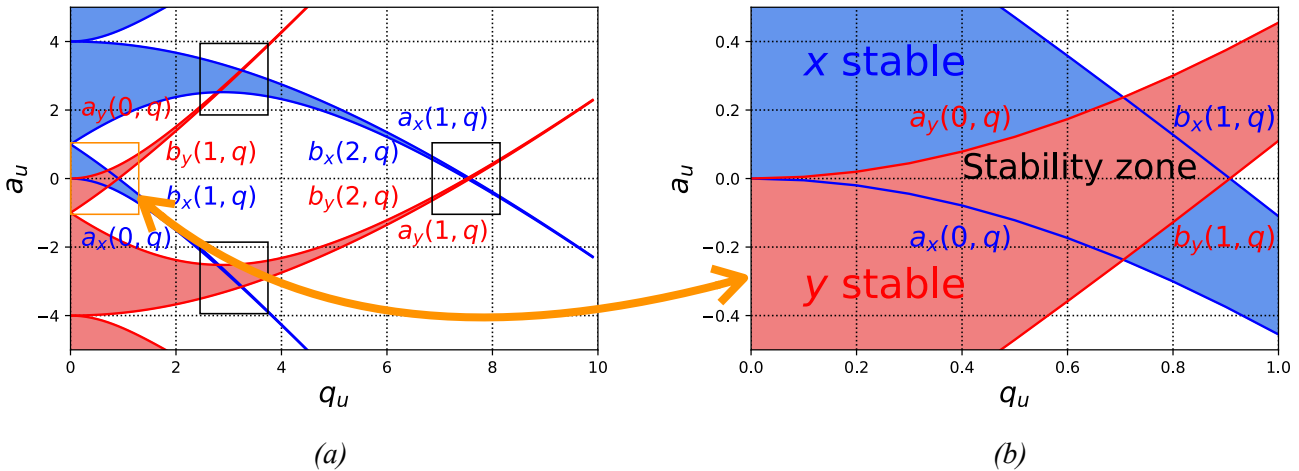
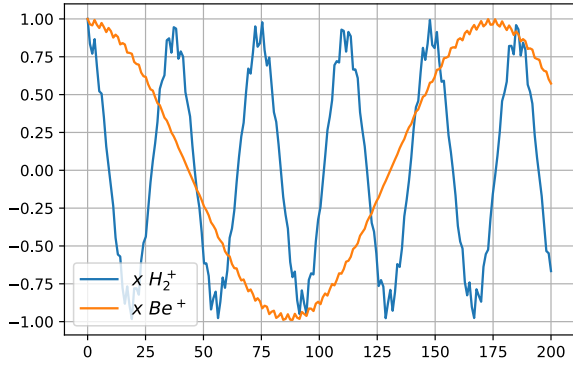


Figure 2 - (a) Diagramme de stabilité des solutions  $x$  (bleu) et  $y$  (rouge).  
(b) Première région de stabilité.

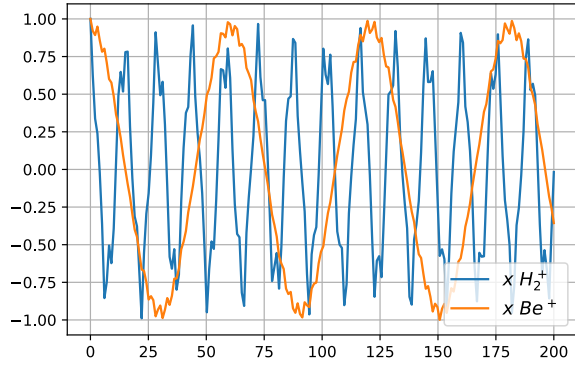
Les équations (8) et (9) sont très complexes à résoudre. Cependant, la stabilité des solutions de  $x(\tau)$  et  $y(\tau)$  est fonction des paramètres  $a_u$  et  $q_u$  (ici  $u \in \{x, y\}$ ). Lorsqu'on trace le domaine de stabilité dans le plan  $(q_u, a_u)$ , nous obtenons la Figure 2. Ces deux courbes ont été obtenues en utilisant les fonctions Python `mathieu_a` et `mathieu_b` de la bibliothèque `scipy.special` (cf. Annexe 1).

On remarque que les domaines de stabilité en  $x$  et  $y$  sont symétriques. Confiner physiquement une particule est possible à l'intersection entre les domaines de stabilité du mouvement en  $x$  et en  $y$ . En choisissant un point dans la zone de stabilité, cela caractérisera un fonctionnement spécifique du piège radio-fréquence. Habituellement, nous choisissons un point dans la zone de stabilité la plus proche de l'origine car c'est ce qui est le plus facilement atteignable physiquement. En effet, les tensions à appliquer sont plus raisonnables que si on prenait un point éloigné de l'origine. De plus, on voit que si  $q_u > 0,9$  (cf. Figure 2 (b)) le mouvement de l'ion devient instable.

En utilisant les paramètres de la Table 1, nous allons résoudre les 2 équations différentielles (8) et (9). Pour ce faire nous avons utilisé un outil numérique c'est-à-dire la bibliothèque `scipy.integrate` du langage Python avec la fonction `odeint` (cf. Annexe 2).



(a)



(b)

Figure 3 - Trajectoire suivant  $x$  d'un ion  $H_2^+$  et d'un ion  $Be^+$ , avec  $\Omega = 2\pi \times 13,5 \text{ MHz}$ .

(a)  $H_2^+$  avec  $a_x = 2,36 \cdot 10^{-4}$  et  $q_x = 0,236$ ,  $Be^+$  avec  $a_x = 5,02 \cdot 10^{-5}$  et  $q_x = 0,050$ .

(b)  $H_2^+$  avec  $a_x = 0,0212$  et  $q_x = 0,5312$ ,  $Be^+$  avec  $a_x = 4,722 \cdot 10^{-3}$  et  $q_x = 0,11$ .

D'après la Figure 3 on remarque que le mouvement de l'ion  $H_2^+$  est bien plus rapide que l'ion  $Be^+$ . Cela vient du fait que l'ion  $Be^+$  est environ 4,5 fois plus lourd que l'ion  $H_2^+$  (masse  $Be^+ \approx 9 u$  et masse  $H_2^+ \approx 2 u$  avec  $1u \approx 1,66054 \cdot 10^{-27} \text{ kg}$ ). De plus, on peut observer que le mouvement d'un ion dans le plan  $xy$  comporte la superposition de 2 mouvements. Ainsi le mouvement de l'ion est composé d'un mouvement séculaire (lent) et d'un micro-mouvement (rapide) de fréquence  $\Omega$ .

Nous supposons donc que le mouvement selon  $x$  peut s'écrire sous la forme :

$$x(t) = \underbrace{X(t)}_{\text{lent}} + \underbrace{\xi(t)}_{\text{rapide}} . \quad (13)$$

En dérivant l'équation (13), on obtient  $\dot{x} = \dot{X} + \dot{\xi}$ . Lorsqu'on a  $\dot{\xi}$  grand devant  $\dot{X}$  on peut considérer l'approximation adiabatique pour laquelle les solutions des équations (8) et (9) sont stables et les solutions peuvent être mises sous la forme (cf. [1]) :

$$x(t) = x_0 \cos(\omega_x t + \varphi_x) \left( 1 + \frac{q_x}{2} \cos(\Omega t) \right), \quad (14)$$

$$y(t) = y_0 \cos(\omega_y t + \varphi_y) \left( 1 + \frac{q_y}{2} \cos(\Omega t) \right). \quad (15)$$

$$\text{avec } \omega_x = \frac{\Omega}{2} \sqrt{\frac{q_x^2}{2} + a_x} \text{ et } \omega_y = \frac{\Omega}{2} \sqrt{\frac{q_y^2}{2} + a_y}.$$

Le mouvement séculaire sera de fréquence  $\omega_x$  ou  $\omega_y$  selon l'axe considéré.

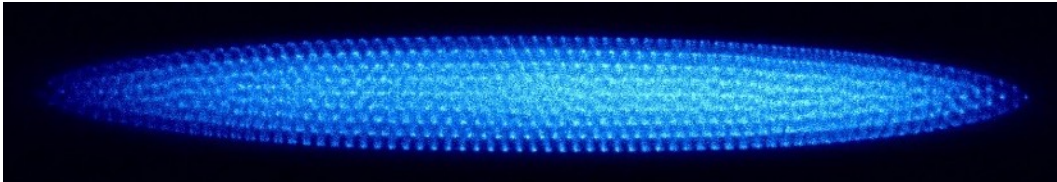
Maintenant que l'on sait comment évolue un ion dans le piège de Paul linéaire et quels sont les paramètres optimaux à appliquer au système, on doit s'intéresser à un ensemble d'ions qui forme un plasma.

Après une brève description du plasma, nous allons étudier les fréquences de résonance plasma ainsi que la réaction de ce plasma lorsqu'on lui applique une perturbation.

## 2. Le nuage d'ions $Be^+$

### 2.1. Simulation numérique de ce nuage

Lorsqu'on place de nombreux ions Béryllium  $Be^+$  dans un piège de Paul linéaire, le piège oblige les ions à rester ensemble. La répulsion coulombienne tend à les éloigner les uns des autres. De plus le refroidissement laser retire de l'énergie cinétique au système. Finalement, les ions forment un plasma c'est-à-dire un nuage d'ions  $Be^+$ . Ce plasma est de forme sphéroïdale car c'est la configuration nécessitant le moins d'énergie (cf. [8]) comme on peut le voir sur la *Figure 4* enregistrée par l'équipe d'ions piégés.



*Figure 4 - Nuage d'environ un millier d'ions  $Be^+$  refroidis par laser à 313 nm  
Chaque point brillant est dû à la fluorescence des ions  $Be^+$ .*

La longueur du nuage est d'environ 1 mm et le diamètre est approximativement de 0,11 mm. La distance entre deux ions avoisine les 14  $\mu m$ .

Pour construire numériquement ce nuage, nous utiliserons un code écrit en langage Fortran. Ce code a été écrit par le Professeur Laurent Hilico. Ce programme simule le comportement du nuage d'ions soumis aux forces du piège de Paul linéaire, à la répulsion coulombienne entre ions et à l'interaction du laser avec ce nuage. Ma contribution a été de compléter ce code pour pouvoir appliquer une perturbation à ce nuage d'ions et voir comment il réagit. En utilisant ce code, nous avons accès à la position, la vitesse et l'accélération pour chaque ion. Pour perturber le nuage nous agirons seulement sur la vitesse et la position des ions.

### 2.2. Pression de radiation sur le nuage

Le code nous a permis de construire un nuage d'ions. Afin de refroidir ces ions, nous allons utiliser un refroidissement par laser.

On s'intéresse ici à un atome au repos placé dans l'axe du laser. Cet atome peut être dans 2 états : état fondamental ou état excité. Quand celui-ci est dans un état excité, il peut se désexciter par émission spontanée ou stimulée pour revenir vers l'état fondamental. Ce processus d'absorption et d'émission se répète ainsi de nombreuses fois. Lorsqu'un atome absorbe un photon, son impulsion change de  $\hbar \vec{k}_L$  (impulsion du laser). Le bilan d'échange d'impulsion pendant un temps  $T$  montre que l'impulsion varie de  $n \hbar \vec{k}_L$  ( $n$  est le nombre de cycle d'absorption/émission). Donc l'atome subit une force moyenne, qu'on appelle pression de radiation, qui peut s'écrire (cf. [4]):

$$P_R = \frac{\hbar \vec{k}_L \Gamma}{2} \frac{s}{s+1}, \quad (16)$$

avec  $\Gamma$  l'inverse de la durée de vie du niveau excité et  $s = \frac{\Omega_R^2/2}{\delta^2 + \Gamma^2/4}$  le paramètre de saturation.  $\delta$  est le désaccord entre la fréquence de résonance de l'atome et la fréquence du laser ( $\delta = \omega_L - \omega_A$ ).

$\Omega_R$  est la fréquence de Rabi qui s'exprime sous la forme :

$$\Omega_R^2 = \Gamma^2 \frac{I}{2I_{sat}}, \quad (17)$$

où  $I$  est l'intensité et  $I_{sat}$  l'intensité de saturation du faisceau laser.

Il va donc falloir considérer la pression de radiation lors de l'écriture du code pour les modes de vibration (cf. 2.5.1). En effet, la pression de radiation décale le nuage de l'origine et on verra plus tard que cela joue



un rôle sur l'interprétation des résultats. De plus, le laser change de puissance brutalement au cours du temps dans nos simulations et il faudra alors considérer ce changement de puissance.

## 2.3. Modes de vibration

Les modes de vibration plasma froid d'ions  $Be^+$  ont été identifiés grâce à des expériences d'imagerie par vélocimétrie (mesure de vitesses) Doppler dans un piège de Penning. Ces expériences ont permis de montrer que les modes sont corrélés à des modes spécifiques  $(l, m)$  théoriquement prédits pour des liquides de forme sphéroïdale uniformément chargés.

Nous utilisons un piège de Paul linéaire mais cela ne pose aucun problème ici car les fréquences correspondantes sont proches de celles des modes  $(l, m)$ . Les fréquences pour un plasma sphéroïdal sont données par [5] :

$$\omega_{(l,m)} = \frac{\omega_p}{\sqrt{1 - \frac{P_l^m Q_l^m}{Q_l^m P_l^m}}}, \quad (18)$$

où  $\omega_p$  est la fréquence plasma,  $P_l^m = P_l^m \left( \frac{1}{\sqrt{1-\alpha^2}} \right)$  et  $Q_l^m = Q_l^m \left( \frac{1}{\sqrt{1-\alpha^2}} \right)$  sont les fonctions de Legendre du premier et du second type avec les indices cylindriques  $(l, m)$ .  $\alpha$  est le rapport d'aspect du plasma avec  $\alpha = \frac{R}{L}$ , où  $L$  est la demi-longueur et  $R$  le rayon de la sphère.

Nous allons considérer  $(l, m = 0)$  car tous les modes avec  $m = 0$  n'ont qu'une variation selon l'axe  $z$  et cela nous facilitera l'écriture du code. Nous allons nous restreindre à étudier des modes  $l = 1, 2, 3, 5$ .

En utilisant la bibliothèque `scipy.special` du langage Python et les fonctions `lpnm` et `lqmn`, nous avons pu, en utilisant l'équation (18), tracer la *Figure 5*. Ces courbes permettent de voir l'évolution des fréquences de résonance pour quelques modes en fonction du rapport d'aspect  $\alpha$ . On remarque que plus  $\alpha$  augmente (*i.e.* plus le nuage devient allongé), plus la fréquence de résonance augmente aussi.

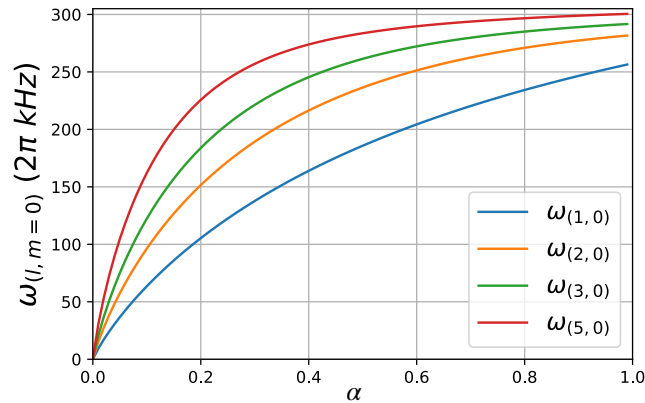


Figure 5 - Fréquences de résonance pour  $l \in \{1,2,3,5\}$  et  $m = 0$  en fonction du rapport d'aspect  $\alpha$ .

Dans notre cas, nous allons tracer les fréquences de résonance données par l'équation (18) en fonction de la fréquence de piégeage  $f_z$ . Pour cela, nous allons ajouter une sous-routine dans le programme Fortran qui va nous permettre d'appliquer la perturbation avec l'un des modes souhaités (ici  $l = 1, 2, 3, 5$  et  $m = 0$ ), on pourra ainsi essayer d'observer les fréquences de résonance plasma pour chacun des modes.

## 2.4. Addition d'un mode de perturbation

Précédemment nous avons émis l'idée que nous avons seulement besoin des positions et des vitesses des ions pour pouvoir appliquer une perturbation au nuage. D'après [6], il est possible d'exciter les différentes résonances plasma en appliquant au nuage une perturbation donnée par :

$$\vec{x}_i(t=0) = \vec{x}_i^{(0)} + \delta \vec{x}_i, \quad (19)$$

$$\vec{v}_i(t=0) = \vec{v}_i^{(0)} + \delta \vec{v}_i, \quad (20)$$

$$\delta \vec{v}_i = C_{lm} \vec{\nabla} \psi_{lm}(x_i^{(0)}), \quad (21)$$

avec  $C_{lm}$  constante. Nous pouvons considérer  $\delta x_i = 0$  car nous recentrons le nuage d'ions (cf. 2.5.1). D'après [7], en utilisant les coordonnées elliptiques  $(\xi_1, \xi_2, \phi)$  on a :

$$\psi_{lm} = P_l^m \left( \frac{\xi_1}{d} \right) P_l^m(\xi_2) \exp(im\phi), \quad (22)$$

avec  $\xi_1$  et  $\xi_2$  coordonnées elliptiques définies ci-dessous et  $\phi$  l'angle azimutal.  $d = \sqrt{L^2 - R^2}$  est le diamètre du plasma.

Les coordonnées elliptiques  $(\xi_1, \xi_2, \phi)$  sont liées aux coordonnées cylindriques  $(\rho, \phi, z)$  par :

$$z = \xi_1 \xi_2, \quad (23)$$

$$\rho = \sqrt{(\xi_1^2 - d^2)(1 - \xi_2^2)}. \quad (24)$$

En utilisant ces paramètres (équations (23) et (24)) et l'équation (22) nous pouvons ainsi obtenir la *Table 2*. Cette table donne l'expression des moments multipolaires. Les moments (1,0), (2,0) et (3,0) ont été tirés de [6], nous avons déterminé le moment (5,0). Grâce à ces moments nous allons pouvoir calculer les  $\delta \vec{v}_i$  (cf. équation (21)) et donc pouvoir appliquer les perturbations aux vitesses. Tous les  $\delta \vec{v}_i$  pour chaque mode sont représentés dans l'*Annexe 3*.

$(l, m)$	$\psi_{lm}$
(1,0)	$z$
(2,0)	$6z^2 - 3\rho^2 - 2d^2$
(3,0)	$z(10z^2 - 15\rho^2 - 6d^2)$
(5,0)	$\frac{z}{64d^2} (945(x^4 + y^4) + 1890x^2y^2 + 840d^2(x^2 + y^2) - 2520z^2(x^2 + y^2) + 504z^4 - 560d^2z^2 + 120d^4)$

*Table 2 - Dépendance spatiale du potentiel  $\psi_{lm}$  en fonction de  $(l, m)$  pour un plasma sphéroïdal, avec  $\rho^2 = x^2 + y^2$ .*

La constante  $C$  n'est pas une constante globale pour tous les modes. Il faut que pour chaque mode une constante  $C_{lm}$  soit appliquée.

Les constantes  $C_{lm}$  vont permettre de faire vibrer plus ou moins fort le nuage. En effet, si on applique un  $C_{lm}$  élevé on peut faire énormément vibrer le nuage au risque de le détruire totalement. Si on applique une constante avec une valeur faible, le risque sera que le nuage ne vibre pas assez et qu'on ne puisse pas observer la fréquence de résonance du mode que l'on a appliqué. Il faut donc essayer de trouver un ordre de grandeur pour chaque  $C_{lm}$  et affiner la valeur en lançant plusieurs exécutions du programme avec différentes valeurs de  $C_{lm}$  (cf. *Annexe 4*).

## 2.5. Simulation des fréquences de résonance

### 2.5.1. Recentrage du nuage

Nous avons vu dans la section 2.2. qu'il existe une pression de radiation qui s'applique sur le nuage d'ions qui décale celui-ci du centre du piège selon l'axe  $z$ . De plus, on utilise une forte puissance laser pour refroidir les ions et préparer le nuage. Après on baisse la puissance laser quand on excite la perturbation pour éviter de trop l'amortir et ainsi pouvoir l'observer. Lorsque le laser est en train de refroidir le nuage avec une certaine puissance  $P$ , nous devons recentrer le nuage d'un certain  $z_{eq}$ . Lorsque le laser passe de la puissance  $P$  à une puissance  $P'$  (où  $P \gg P'$ ), il faut ainsi rajouter encore une correction  $z'_{eq}$  (cf. Annexe 5).

La correction  $z_{eq}$  s'obtient en additionnant toutes les coordonnées en  $z$  de chaque ion puis en divisant par le nombre total d'ions refroidi par laser, c'est-à-dire :

$$z_{eq} = \frac{\sum_i z_i}{n_{tot}}. \quad (25)$$

En utilisant les équations (16) et (17), nous pouvons en déduire  $z'_{eq}$  (cf. Annexe 6 pour les détails de calcul) :

$$z'_{eq} = z_{eq} \frac{P'}{P} \frac{1 + \beta \frac{P}{P_{sat}}}{1 + \beta \frac{P'}{P_{sat}}}. \quad (26)$$

$$\Leftrightarrow z_{eq} = z'_{eq} \frac{P}{P'} \frac{1 + \beta \frac{P'}{P_{sat}}}{1 + \beta \frac{P}{P_{sat}}}. \quad (27)$$

La force de rappel du piège est linéaire en  $z$ , donc le centre du piège se déplace proportionnellement à la force de pression de radiation. Pour vérifier que  $z_{eq}$  suit bien l'équation (27), nous avons tracé  $z_{eq}$  en fonction de  $P/P_{sat}$  sur la Figure 6. Nous avons aussi exécuté le programme (qui utilise l'équation (25)) en faisant varier  $P/P_{sat}$  pour obtenir  $z_{eq}$  expérimental. Ces quelques points ont été reportés sur la Figure 6. Nous remarquons ainsi que les points issus de la modélisation numérique suivent bien la courbe théorique de l'équation (27). Par conséquent cette équation est valable et nous allons donc pouvoir utiliser  $z_{eq}$  et  $z'_{eq}$  (cf. équations (25) et (26)) pour recentrer le nuage.

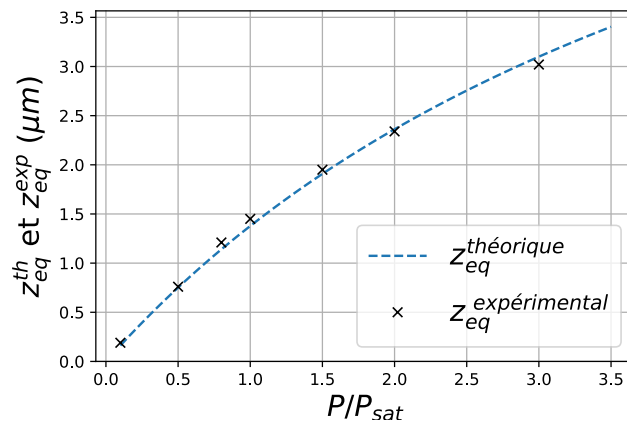


Figure 6 -  $z_{eq}^{exp}$  et  $z_{eq}^{th}$  en fonction de la puissance  $P/P_{sat}$  du laser

L'intensité de saturation est de  $0,83 \text{ mW/mm}^2$ .

Le but de recentrer le nuage est d'éviter d'avoir des modes parasites qui apparaissent lorsqu'on veut faire vibrer le nuage avec un mode précis. Le mode (1,0) est un mode où le nuage se déplace en bloc (mode du centre de masse) selon l'axe  $z$ . Si on ne rectifiait pas le décalage du nuage lié à la pression de radiation du

laser, le nuage oscillerait selon l'axe  $z$ . Cette oscillation, si elle est trop forte masque les petits signaux attendus pour détecter l'excitation des autres modes.

### 2.5.2. Détails du code

Le code Fortran écrit pour perturber le nuage peut être résumé en quelques grandes étapes :

- Préparation du nuage d'ions (initialisation des ions, piégeage des ions, refroidissement par laser).
- On calcule  $z_{eq}$  et  $z'_{eq}$  en fonction de la puissance  $P/P_{sat}$  et  $P'/P_{sat}$  appliquée sur le nuage.
- On applique la perturbation souhaitée (dans notre cas  $l = 1, 2, 3, 5$  et  $m = 0$ ) au nuage en utilisant les  $\delta \vec{v}$  de l'Annexe 3. Chaque position  $z$  des ions est décalée de :  $z_i - z_{eq} + z'_{eq}$ .
- On calcule le moment multipolaire de la perturbation appliquée en faisant la somme des dépendances spatiales du potentiel  $\psi_{lm}$  (cf. Table 2) pour chaque ion c'est à dire :  $\mathcal{M}_{lm}(t) = \sum_i \psi_i(t)$ . Ce moment multipolaire est calculé pour chaque  $t$ .
- Chaque moment est écrit dans un fichier .multi que l'on doit traiter par transformée de Fourier pour obtenir les fréquences de résonance de chaque mode.

## 3. Résultats

Notre but va être de tracer les fréquences de résonance de chaque mode plasma en fonction de la fréquence  $f_z$  du piège selon l'axe  $z$ .

### 3.1 Interprétation des graphiques

Afin de pouvoir traiter les fichiers .multi et déterminer les fréquences de résonance, nous avons écrit un programme Python qui nous permet d'afficher la courbe représentant le moment multipolaire, d'appliquer une Transformée de Fourier (TF) sur le domaine physique qui nous intéresse et de trouver la fréquence principale de cette TF (cf. Annexe 7). La fréquence principale de la TF c'est ce qu'on recherche car elle représente la fréquence de résonance pour un mode  $(l, m)$  en fonction d'une fréquence de piégeage  $f_z$ .

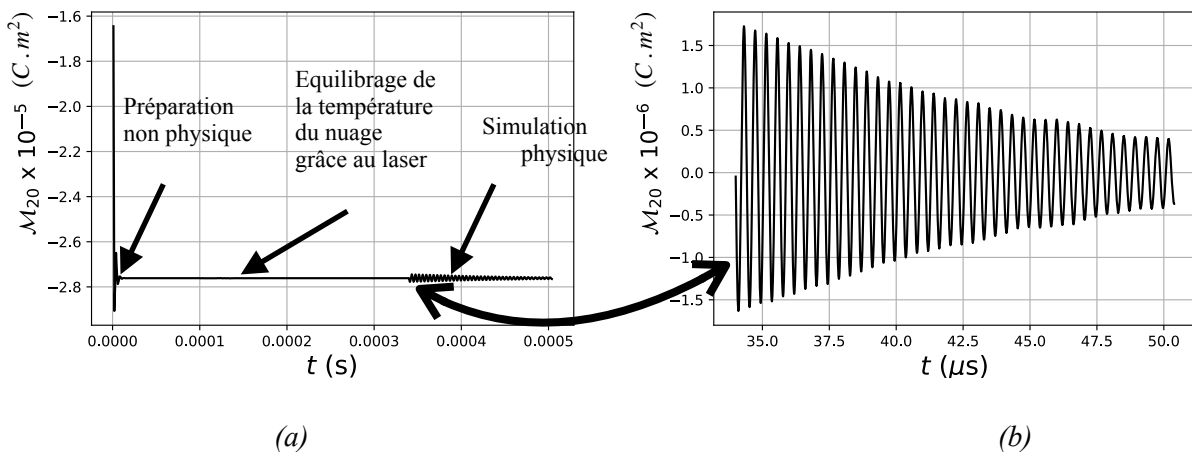


Figure 7 - Moment  $\mathcal{M}_{20}$  en fonction du temps avec :  $f_z = 0,16$  MHz,  $P = 1,5 P_{sat}$  et  $P' = 0,01 P_{sat}$   
 (a) Moment  $\mathcal{M}_{20}$  pendant toute la durée de la simulation  
 (b) Moment  $\mathcal{M}_{20}$  centré en 0 pendant la simulation physique

Regardons le mode (2,0) en utilisant comme paramètres de simulation physique :  $f_z = 0,16$  MHz,  $P = 1,5 P_{sat}$  et  $P' = 0,01 P_{sat}$ . Nous pouvons voir sur la Figure 7 le moment  $\mathcal{M}_{20}$  en fonction du temps. On

peut remarquer qu'il y a trois étapes. La préparation non physique c'est le moment où le nuage est fabriqué et que le laser se met en fonctionnement. L'équilibrage de la température c'est là que le laser refroidit le nuage d'ions. La simulation physique démarre au moment où on applique la perturbation. Le nuage commence à osciller en mode (2,0).

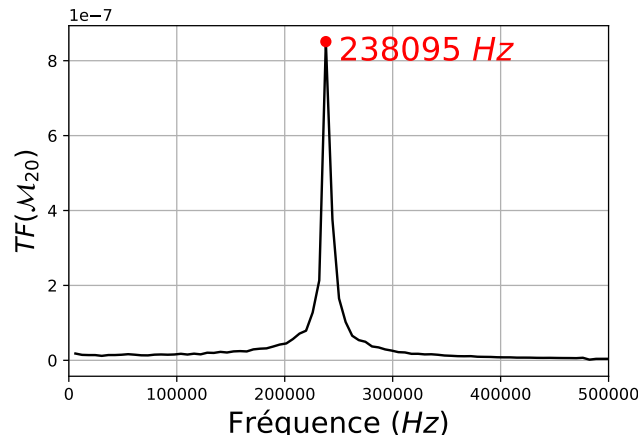


Figure 8 - Transformée de Fourier du moment multipolaire  $\mathcal{M}_{20}$  avec  $f_z = 0,16$  MHz

Nous allons appliquer une TF sur la partie simulation physique car l'information que l'on souhaite extraire se situe à cet endroit. De plus, nous avons recentré verticalement la partie simulation physique (cf. Figure 7 (b)) en 0 car lorsque nous allons faire la TF cela permettra d'éviter les pics parasites. Nous obtenons ainsi la Figure 8 et nous remarquons qu'un pic ressort. Ce pic correspond ainsi à la fréquence de résonance du mode (2,0) avec une fréquence de piégeage de l'ion  $Be^+$  de  $f_z = 0,16$  MHz. Il suffit juste de réitérer la simulation en faisant varier  $f_z$  pour pouvoir tracer les fréquences de résonance du mode (2,0) et des autres modes.

### 3.2. Les fréquences de résonance de chaque mode plasma

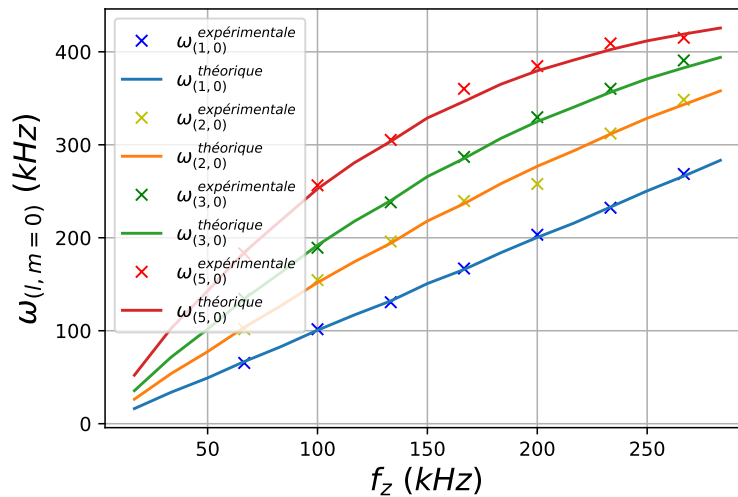


Figure 9 - Fréquences de résonance des modes plasma  $(l, m)$  en fonction de  $f_z$

En ayant fait varier  $f_z$  pour chaque mode  $(l, m = 0)$  (avec  $l \in \{1,2,3,5\}$ ) nous avons ainsi obtenu la Figure 9. Sur cette figure nous pouvons voir les  $\omega_{(l,m=0)}^{exp}$  qui sont les fréquences de résonance obtenues grâce au programme Fortran et les  $\omega_{(l,m=0)}^{th}$  qui sont les courbes théoriques de ces fréquences. Les  $\omega_{(l,m=0)}^{th}$  ont été

tracées à l'aide d'un programme Python (cf. *Annexe*) écrit par Thomas Louvradoux, post doctorant dans l'équipe ions piégés.

## Conclusion

Après avoir vu la théorie entourant le piège de Paul linéaire et le piégeage du nuage d'ions  $Be^+$  dans celui-ci, nous nous sommes intéressés à ce plasma en regardant comment il réagirait à une perturbation.

Nous avons vu via les articles étudiés que le nuage perturbé oscille avec une fréquence de résonance spécifique et nous avons donc essayé de comprendre comment faire vibrer ce nuage en lui appliquant une perturbation avec certains modes de vibration. Nous avons ajouté au programme Fortran une subroutine qui nous permet de perturber ce nuage avec le mode souhaité. Puis il a fallu essayer d'observer ces fréquences de vibration. Notre idée a été d'observer les moments multipolaires du mode appliqué et de faire une transformée de Fourier à l'instant qui nous intéresse (cf. *3.1 simulation physique*) pour pouvoir observer la fréquence de résonance.

Cependant il a fallu prendre en compte un autre problème physique : la pression de radiation due au laser. En effet, au début nous n'avions pas pris en compte cette pression appliquée sur le nuage et lorsque nous observions la transformée de Fourier, nous obtenions des fréquences parasites liées à d'autres modes non souhaités qui apparaissaient dans le spectre. Il a donc fallu prendre en compte la pression de radiation et nous avons donc rectifié le code pour recentrer le nuage sur l'origine et ainsi éviter l'apparition de ces fréquences parasites. A partir de ce moment-là, nous avons pu observer correctement les spectres de chaque mode et ainsi pouvoir tracer la *Figure 8* qui était notre but final.

Personnellement ce stage m'a permis d'avoir une première approche sur l'étude des plasmas. J'ai pu aussi découvrir de nouveaux outils comme les outils numériques (langage Fortran par exemple) ou même savoir utiliser des calculateurs à distance. J'ai dû aussi avoir un esprit critique sur les résultats obtenus et comprendre leur signification. Compte tenu de la situation particulière de ce stage, j'ai aussi appris à m'adapter et à être encore plus autonome dans mon travail. Le professeur Laurent Hilico a toujours été présent pour m'aider et m'accompagner pour la bonne compréhension et le bon déroulement de ce stage. J'ai approfondi un domaine de la physique que je méconnaissais et j'ai trouvé ce domaine de recherche très passionnant.

Pour conclure je me suis épanoui pendant ce stage, j'ai commencé à découvrir le monde de la recherche expérimentale et j'ai surtout appris énormément de nouvelles connaissances et de nouvelles méthodes qui pourront me servir plus tard.

## Bibliographie

- [1] MBARDI, Abdessamad. *Simulation d'une image d'un nuage d'ions  $Be^+/H_2^+$  piégés*. Rapport de stage M1, Sorbonne Université campus Pierre et Marie Curie, Paris, France (2019). Consulté à l'adresse <http://www.lkb.upmc.fr/iontrap/wp-content/uploads/sites/20/2019/07/rapport-stage-M1-Abdessamad-Mbardi.pdf>
- [2] MARCIANTE, Mathieu. *Dynamique d'ions en pièges radio-fréquences*. Thèse de doctorat, Université de Provence, Marseille, France (2011). p. 7-10. Consulté à l'adresse <https://tel.archives-ouvertes.fr/tel-00641606/document>
- [3] HEINRICH, Johannes. *A  $Be^+$  Ion Trap for  $H_2^+$  Spectroscopy*. Thèse de doctorat, Sorbonne Université, Paris, France (2018). p. 9-15. Consulté à l'adresse <https://tel.archives-ouvertes.fr/tel-01889833/document>
- [4] LOUVRADOUX, Thomas. *Spectroscopie à haute résolution de  $H_2^+$  : Production et refroidissement sympathique d'ions piégés*. Thèse de doctorat, Université de Paris, Paris, France (2019). p. 65-68.
- [5] A. Dantan, J. P. Marler, M. Albert, D. Guénot, and M. Drewsen. *Noninvasive Vibrational Mode Spectroscopy of Ion Coulomb Crystals through Resonant Collective Coupling to an Optical Cavity Field*. Physical review letters 105, Septembre 2010.
- [6] Daniel H. E. Dubin. *Normal modes of cold confined one-component plasmas*. Physical Review E, Mai 1996, Volume 53, Numéro 5, p. 5249-5267.
- [7] Daniel H. E. Dubin. *Effect of correlations on the thermal equilibrium and normal modes of a non-neutral plasma*. Physical Review E, Mai 1996, Volume 53, Numéro 5, p. 5268-5290.
- [8] Leaf Turner, *Collective effects on equilibria of trapped charged plasmas*. The physics of Fluids, Octobre 1987, Volume 30, Numéro 10, p.3196-3203.

## Annexes

### Annexe 1 : Résolution numérique des domaines de stabilité

```
#Bibliothèques utiles
import matplotlib.pyplot as plt
import numpy as np
from scipy.special import mathieu_a
from scipy.special import mathieu_b
#Création des vecteurs utiles
a_x = np.zeros(100, float)
b_x = np.zeros(100, float)
a_y = np.zeros(100, float)
b_y = np.zeros(100, float)
x = np.arange(0,100,1)
#Boucle sur les indices m
for m in [0,1,2]:
    #Boucle les q_u
    for q in range(0,100):
        a_x[q] = mathieu_a(m,q/10)
        b_x[q] = mathieu_b(m+1,q/10)
        a_y[q] = - a_x[q]
        b_y[q] = - b_x[q]
    #Affichage du graphique
    plt.plot(x/10, a_x, 'b')
    plt.plot(x/10, b_x, 'b')
    #Remplissage des couleurs entre les courbes
    plt.fill_between(x/10, a_x, b_x, color='cornflowerblue')
    plt.plot(x/10, a_y, 'r')
    plt.plot(x/10, b_y, 'r')
    plt.fill_between(x/10, a_y, b_y, color='lightcoral')
#Nommer les axes
plt.xlabel('$q_u$', fontsize=18)
plt.ylabel('$a_u$', fontsize=18)
#Limiter la fenêtre graphique
plt.xlim(0,10)
plt.ylim(-5,5)
#Placer les notes sur le graphique
plt.annotate('$a_x(0,q)$', xy=(2, 1), xytext=(0.2, -2), color='blue',
    fontsize=16)
plt.annotate('$a_y(0,q)$', xy=(2, 1), xytext=(0.2, 1.8), color='red',
    fontsize=16)
plt.annotate('$a_x(1,q)$', xy=(2, 1), xytext=(6.2, 1.5), color='blue',
    fontsize=16)
plt.annotate('$a_y(1,q)$', xy=(2, 1), xytext=(6.2, -1.8), color='red',
    fontsize=16)
plt.annotate('$b_x(1,q)$', xy=(2, 1), xytext=(1.8, -0.8), color='blue',
    fontsize=16)
plt.annotate('$b_y(1,q)$', xy=(2, 1), xytext=(1.8, 0.5), color='red',
    fontsize=16)
plt.annotate('$b_x(2,q)$', xy=(2, 1), xytext=(4.5, 0.5), color='blue',
    fontsize=16)
plt.annotate('$b_y(2,q)$', xy=(2, 1), xytext=(4.5, -0.8), color='red',
    fontsize=16)
```



```

#Placer les carrés sur le graphique
plt.text(0.65, 0, r' ', {'color': 'black', 'fontsize': 3.5, 'ha':
'center', 'va': 'center', 'bbox': dict(boxstyle="square", fc="None",
ec="darkorange", pad=6)})
plt.text(7.5, 0, r' ', {'color': 'black', 'fontsize': 3.5, 'ha':
'center', 'va': 'center', 'bbox': dict(boxstyle="square", fc="None",
ec="black", pad=6)})
plt.text(3.1, 2.9, r' ', {'color': 'black', 'fontsize': 3.5, 'ha':
'center', 'va': 'center', 'bbox': dict(boxstyle="square", fc="None",
ec="black", pad=6)})
plt.text(3.1, -2.9, r' ', {'color': 'black', 'fontsize': 3.5, 'ha':
'center', 'va': 'center', 'bbox': dict(boxstyle="square", fc="None",
ec="black", pad=6)})
#Afficher le quadrillage
plt.grid(color='black', linestyle=':', linewidth=1)
plt.show()

```

## Annexe 2 : Résolution numérique des équations de Mathieu

```

#Bibliothèques utiles
import matplotlib.pyplot as plt
import numpy as np
from scipy.integrate import odeint
##Paramètres
Omega = 2*np.pi*13.5e6
#H2+
ax1 = 0.0212
qx1 = 0.5312
#Be+
ax2 = 4.722e-3
qx2 = 0.11
n = 2000 #Nombre de points dans l'intervalle
t = np.linspace(0, 200, n)
# L'équation différentielle sous forme de fonction
def equation(X1,t):
    # On décompose notre X en (x,dx) :
    (x,dx)=X1
    # On renvoie ce que vaut X' :
    return [dx,-(ax1+2*qx1*np.cos(Omega*t*(2/Omega)))*x]
# Pour que odeint renvoie séparément les valeurs de X et de X', il faut
rajouter .T à la fin
X1,dX1=odeint(equation, [1,0], t).T
def equation2(X2,t):
    (x,dx)= X2
    return [dx,-(ax2+2*qx2*np.cos(Omega*(t*(2/Omega))))*x]
X2,dX2=odeint(equation2, [1,0], t).T
plt.plot(t, X1, label='$x\ H_2^{+}$')
plt.plot(t, X2, label='$x\ Be^{+}$')
plt.legend(loc='best', fontsize=12)
plt.grid()
plt.show()

```

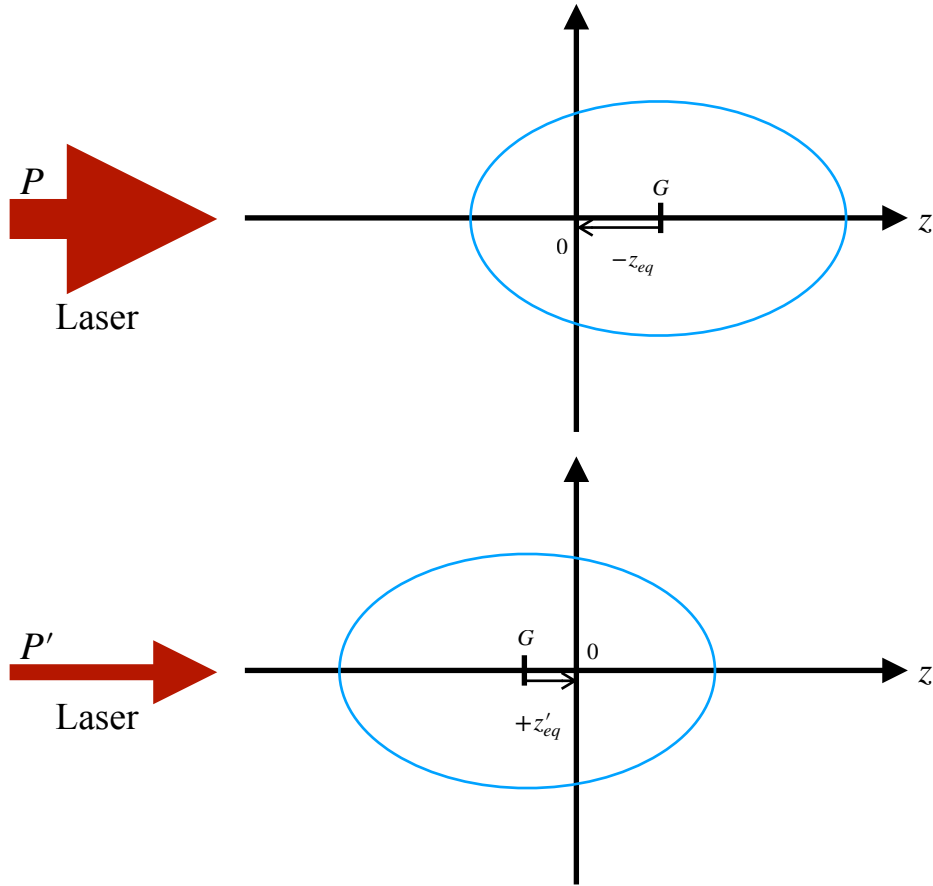
### Annexe 3 : Perturbations sur les vitesses pour chaque mode plasma étudié

$(l, m)$	$\delta \vec{v} = C_{lm} \vec{\nabla} \psi_{lm}$
(1,0)	$\delta v_x = 0$
	$\delta v_y = 0$
	$\delta v_z = C_{10}$
(2,0)	$\delta v_x = -6C_{20} x$
	$\delta v_y = -6C_{20} y$
	$\delta v_z = 12C_{20} z$
(3,0)	$\delta v_x = -30C_{30} xz$
	$\delta v_y = -30C_{30} yz$
	$\delta v_z = 30C_{30} \left( z^2 - \frac{x^2 + y^2}{2} - \frac{d^2}{5} \right)$
(5,0)	$\delta v_x = 420C_{50} xz (9(x^2 + y^2) + 4d^2 - 12z^2)$
	$\delta v_y = 420C_{50} yz (9(x^2 + y^2) + 4d^2 - 12z^2)$
	$\delta v_z = 15C_{50} (63(x^4 + y^4) + 126x^2y^2 + 56d^2(x^2 + y^2) - 504z^2(x^2 + y^2) + 168z^4 - 112d^2z^2 + 8d^4)$

### Annexe 4 : Constante pour chaque mode plasma

$(l, m)$	$C_{lm}$ déterminée	$C_{lm}$ affinée (utilisée pour la modélisation)	Unité
(1,0)	$1.10^3$	0	$m \cdot s^{-1}$
(2,0)	$9.10^5$	$1.10^5$	$s^{-1}$
(3,0)	$4.10^9$	$1.10^5$	$m^{-1} \cdot s^{-1}$
(5,0)	$1.10^{19}$	$1.10^5$	$m^{-3} \cdot s^{-1}$

## Annexe 5 : Schéma montrant le nuage d'ions soumis à la pression de radiation due au laser



## Annexe 6 : Determination des positions d'équilibre

Dans le piège un atome subit une force moyenne, qu'on appelle pression de radiation, qui peut s'écrire (*tirée de [4]*):

$$P_R = \frac{\hbar \vec{k}_L \Gamma}{2} \frac{s}{s+1}, \quad (1)$$

avec  $\Gamma$  l'inverse de la durée de vie du niveau excité et  $s = \frac{\Omega_R^2/2}{\delta^2 + \Gamma^2/4}$  le paramètre de saturation.  $\delta$  est le désaccord entre la fréquence de résonance de l'atome et la fréquence du laser ( $\delta = \omega_L - \omega_A$ ).  $\Omega_R$  est la fréquence de Rabi qui s'exprime sous la forme :

$$\Omega_R^2 = \Gamma^2 \frac{I}{2I_{sat}}, \quad (2)$$

où  $I$  est l'intensité et  $I_{sat}$  l'intensité de saturation du faisceau laser.

Considérons :  $s = \frac{P}{P_{sat}} = \frac{I}{I_{sat}}$ .

On peut réécrire  $s$  en utilisant l'équation (2) tel que :  $s = \frac{\frac{\Gamma^2}{2} \frac{I}{2I_{sat}}}{\delta^2 + \frac{\Gamma^2}{4}} = \frac{\Gamma^2}{4} \frac{1}{\delta^2 + \frac{\Gamma^2}{4}} \frac{I}{I_{sat}}$

Avec  $\beta = \frac{1}{4\frac{\delta^2}{\Gamma^2} + 1}$ . On obtient donc :  $s = \beta \frac{P}{P_{sat}}$ .

On remplace  $s$  dans l'équation (1), on trouve :

$$P_R = \frac{\hbar \vec{k}_L \Gamma}{2} \frac{\beta \frac{P}{P_{sat}}}{\beta \frac{P}{P_{sat}} + 1}. \quad (3)$$

La force du piège peut d'écrire :  $-m\omega_z^2 z$ .  
Ainsi on obtient :

$$m\omega_z^2 z = \frac{\hbar \vec{k}_L \Gamma}{2} \frac{\beta \frac{P}{P_{sat}}}{\beta \frac{P}{P_{sat}} + 1}, \quad (4)$$

$$\Leftrightarrow z = \frac{\hbar \vec{k}_L \Gamma}{2} \frac{1}{\underbrace{m\omega_z^2}_{\eta}} \frac{\beta \frac{P}{P_{sat}}}{\beta \frac{P}{P_{sat}} + 1}. \quad (5)$$

Grâce à la forme de la pression de radiation trouvée précédemment, on obtient ainsi  $z_{eq}$  et  $z'_{eq}$  sous la forme :

$$z_{eq} = \eta \frac{\beta \frac{P}{P_{sat}}}{1 + \beta \frac{P}{P_{sat}}}, \quad (6)$$

$$z'_{eq} = \eta \frac{\beta \frac{P'}{P_{sat}}}{1 + \beta \frac{P'}{P_{sat}}}. \quad (7)$$

Pour finir on fait le rapport entre  $z_{eq}$  et  $z'_{eq}$ , et on trouve :

$$\frac{z_{eq}}{z'_{eq}} = \frac{P}{P'} \frac{1 + \beta \frac{P'}{P_{sat}}}{1 + \beta \frac{P}{P_{sat}}}. \quad (8)$$

## Annexe 7 : Programme pour obtenir le spectre des moments multipolaires

**#Bibliothèques utiles**

import numpy as np

import matplotlib.pyplot as plt

from scipy.signal import find\_peaks

from scipy import optimize

**#Chargement du fichier à traiter**

fichier1 = "run\_200Be\_12342\_fz\_0.5MHZ\_I=1.5-0.01\_l=2\_multi"

data1 = np.loadtxt(fichier1).T

n = 2 **#Permet de récupérer la colonne de données souhaitées**

**#Afficher la partie physique centrée en 0**

time = data1[0][number:] **#Récupérer la partie physique**

datalim1 = data1[n][number:]

data\_mean1 = np.mean(datalim1)

datalim1 = datalim1 - data\_mean1 **#Recentrer en 0**

plt.plot(time, datalim1)

plt.xlabel('\$t\$ (\$\mu s\$)', fontsize=18)

plt.ylabel('\$\mathcal{M}\_{20}\$', fontsize=16)

plt.grid()

plt.show()

**#Transformée de Fourier**

fft\_datalim1 = np.fft.fft(datalim1) **#Transformée de Fourier du domaine physique**

N = len(time) **#Nombre de points de l'échantillon de temps**

```
dT = time[1] - time[0] #Le pas de temps
freq = np.linspace(0, 1.0/(2.0*dT), int(N/2)) #Calcul des fréquences
Y1 = 2.0/N * np.abs(fft_datalim1[0:int(N/2)]) #Recentrage de la TF à 0
freq = freq[1:] #Enlève le premier point inutile de la TF
Y1 = Y1[1:]
pic_found, _ = find_peaks(Y1, height = np.max(Y1) ) #Affiche la fréquence
max du spectre
print(freq[pic_found])
plt.plot(freq, Y1, 'k')
plt.plot(freq[pic_found], Y1[pic_found], 'or')
plt.grid()
plt.show()
```