Simulations numériques de l'équation de Vlasov à l'aide d'outils parallèles

Julien Peyroux

▶ **To cite this version:**

Julien Peyroux. Simulations numériques de l'équation de Vlasov à l'aide d'outils parallèles. Autre [cond-mat.other]. Université Henri Poincaré - Nancy 1, 2005. Français. NNT : 2005NAN10114 . tel-01748366

HAL Id: tel-01748366
https://hal.univ-lorraine.fr/tel-01748366

Submitted on 29 Mar 2018

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

AVERTISSEMENT

Ce document est le fruit d'un long travail approuvé par le jury de soutenance et mis à disposition de l'ensemble de la communauté universitaire élargie.

Il est soumis à la propriété intellectuelle de l'auteur. Ceci implique une obligation de citation et de référencement lors de l'utilisation de ce document.

D'autre part, toute contrefaçon, plagiat, reproduction illicite encourt une poursuite pénale.

Contact : ddoc-theses-contact@univ-lorraine.fr

LIENS

Code de la Propriété Intellectuelle. articles L 122. 4
Code de la Propriété Intellectuelle. articles L 335.2- L 335.10
http://www.cfcopies.com/V2/leg/leg_droi.php
http://www.culture.gouv.fr/culture/infos-pratiques/droits/protection.htm

Sc N2005 114B

U.F.R. Sciences et Techniques de la Matière et des Procèdes
École Doctorale EMMA

Service Commun de la Documentation
de l'Université Henri Poincaré - Nancy 1
B.I.U. Sciences
Rue du Jardin Botanique
54600 VILLERS-LES-NANCY (FRANCE)

Thèse

Présentée pour l'obtention du titre de

Docteur de l'Université Henri Poincaré, Nancy I

en PHYSIQUE DES PLASMAS

Par Julien PEYROUX

SIMULATIONS NUMERIQUES DE L'EQUATION DE VLASOV A L'AIDE D'OUTILS PARALLELES

Soutenance publique le 7 Novembre 2005

Membres du Jury :

Rapporteurs :	M. Eric Sonnendrücker	Professeur, Université L.Pasteur, Strasbourg 1
	M. Henry Habrias	Professeur, Université de Nantes, Nantes
Examinateurs ::	M. Eric Sonnendrücker	Professeur, Université L.Pasteur, Strasbourg 1
	M. Henry Habrias	Professeur, Université de Nantes, Nantes
	M. Pierre Bertrand	Professeur, Université H.Poincaré, Nancy 1
	M. Dominique Méry	Professeur, Université H.Poincaré, Nancy 1

Laboratoire de Physique des Milieux Ionisés
Univers NDOEUVRE-LES-NANCY

BIBLIOTHEQUE SCIENCES NANCY 1
D 095 171580 0

FUSION NUCLEAIRE CONTROLEE **9**

POURQUOI ? 9
INTRODUCTION 9
1. REALISER UNE FUSION NUCLEAIRE. 10
2. LA FUSION PAR FAISCEAUX (FUSION INERTIELLE) 11
3. LES BOITES MAGNETIQUES 11
4. CONCLUSION 12

PARTIE I: GENERALITES SUR LES PLASMAS **13**

1. QUELQUES PROPRIETES 13
2. LES OSCILLATIONS DU PLASMA 14
3. ONDES ELECTROMAGNETIQUES 16
4. INTERACTIONS ENTRE PARTICULES 17
4.1 Ecrantage de Debye 17
4.2 Collisions 20
4.3 Effets collectifs et collisions 24
5. ÉQUATIONS CINETIQUES ET EQUATION DE VLASOV 26
5.1 le grand espace des phases 26
5.2 L'équation de Vlasov 27
5.3 Le système Vlasov-Maxwell 28
6. L'INTERACTION RESONANTE ONDE-PARTICULES 29
6.1 Le système Vlasov-Poisson et l'effet Landau 29
6.2 Interaction onde-particules : particules piégées 33
6.3 Le système Vlasov-onde 36
6.4 Remarque : l'approximation de quasi-neutralité 38
7. CODES NUMERIQUES POUR L'EQUATION DE VLASOV 40
7.1. Introduction 40
7.2. Codes pic vs codes vlasov 43
7.3. Time-splitting et schéma lagrangien 44

PARTIE II : INFORMATIQUE **47**

1. CONCEPTION DE PROGRAMMES PARALLELES 47
1.1 Langages de programmation parallèle 48
1.2 Architectures parallèles 49
1.3 Modèles de programmation parallèle 50
1.4 Méthodologie de programmation parallèle 51
1.5 Synthèse 52
2. ETUDE DE L'EQUATION DE VLASOV 52
2.1 Spécification du code et décomposition modulaire 52
2.2 Analyse du code 55
3. PARALLELISATION 56
3.1 MPI 56
3.2 OpenMP 57
3.3 PVM 58
4. PERSPECTIVES D'UNE BIBLIOTHEQUE POUR LA REUTILISATION DU CODE 60

CONCLUSION **62**

REFERENCES **63**

RESULTATS NUMERIQUES **67**

ANNEXE A : **68**

ANNEXE B : **71**

Liste des principaux symboles utilisés

Pour le champ électromagnétique :

λ	longueur d'onde
$(\omega,\vec{k})$	pulsation et vecteur d'onde
$\vec{E}$	champ électrique
$\vec{B}$	champ magnétique
ϕ	potentiel scalaire
c	vitesse de la lumière dans le vide
ε_0	permittivité du vide
μ_0	perméabilité du vide

Pour le plasma :

e	charge de l'électron
m_e	masse de l'électron
M_i	masse ionique
n_0	densité à l'équilibre (densité ionique neutralisante)
$\mathrm{n_e}$	densité électronique
$\mathrm{n_i}$	densité ionique
$\omega_{pe}=\sqrt{\frac{ne^2}{m_e\varepsilon_0}}$	fréquence (pulsation) plasma
$\mathrm{f_{e,i}}(\boldsymbol{r},\boldsymbol{p},t)$	fonction de distribution électronique, ionique
$x,\vec{r}$	position
$\vec{v},\vec{p}$	vitesse et moment cinétique
g	facteur de grain
λ_D	longueur de Debye
x_L	longueur du plasma
$T_{e,i}$	température électronique, ionique
v_{th}	vitesse thermique
H	hamiltonien de l'électron

k_B	constante de Boltzmann
U_k, U_p, U	énergie cinétique, énergie potentielle, énergie totale

INTRODUCTION

L'état de la matière le plus présent dans l'univers est l'état plasma qui se définit comme étant un gaz chargé globalement neutre. Cet état se divise en 2 catégories : plasma froid et plasma chaud... Celui qui nous intéresse est le second. Pour les scientifiques, le but est de créer de l'énergie à moindres frais et avec composants durables et recyclables. Pour chauffer le plasma, on procède soit par effet Joule soit par effet Compton (rayons laser) ou encore par injection d'ondes HF (10-200GHz) par des antennes sur les tokamaks. Jusqu'à ce jour, on est loin du rendement escompté avec un taux de productivité tournant autour de 10% seulement. Pour garder le plasma à l'intérieur des tores, ce dernier doit être soumis à un confinement magnétique ou inertiel. Produire de l'énergie peut se faire de 2 manières :

-La fission : envoi d'un neutron sur un noyau lourd (type ^{235}U) afin de le casser en 2 noyaux moins lourds ; ce qui provoque un dégagement de chaleur. L'inconvénient est que cette méthode libère des éléments radioactifs difficilement recyclables.

-La fusion : unification de 2 éléments (généralement des noyaux légers) pour donner un élément lourd et un neutron.

La fusion sera développée dans un petit préambule...

Les effets collectifs dans les plasmas dits "chauds" doivent être décrits par les équations de Vlasov pour chaque espèce de particules, couplées aux équations de Maxwell. La complexité de ce système non linéaire qui, dans le cas général, utilise un espace des phases à six dimensions, a tout naturellement conduit à rechercher des modèles plus simples.

Dans tous les cas ce type de physique implique l'étude de plasma de Vlasov que cela soit un modèle simplifié ou fortement non linéaire et relativiste. Pour modéliser ce plasma, il faut s'intéresser à l'interaction onde-particules (résonnante).

Le recours à des outils de simulation performants sur des calculateurs parallèles est nécessaire. Deux types de philosophie existent à l'heure actuelle :

a) La plus populaire est basée sur l'utilisation de codes particulaires (P.I.C. i.e. Particles In Cell), dans lesquels l'équation de Vlasov est remplacée par les équations du mouvement de n superparticules supposées remplacer les N particules réelles du plasma (N 10^{19}-10^{20} et avec, bien entendu, $n<<N$). Le nombre de particules rapides restant faible par rapport au nombre total de particules, ces codes ne permettent pas d'en suivre la dynamique de manière précise.

Or ce sont ces particules relativistes qui sont la clé de la compréhension des mécanismes d'interaction ondes-particules.

b) L'autre philosophie que nous avons contribué à développer est basée sur la résolution des caractéristiques de l'équation de Vlasov. Cette méthode semi-lagrangienne permet de résoudre directement le système des équations de Vlasov avec les équations de Poisson (modèle électrostatique) ou Maxwell (modèle électromagnétique), d'où le nom de code Vlasov.

L'intérêt d'utiliser un code Vlasov tient essentiellement à ce qu'il combine les avantages des méthodes particulaires et des méthodes eulériennes sans certains de leurs inconvénients. Comme on résout les équations des caractéristiques, donc finalement les équations du mouvement des particules, on n'est pas limité par des conditions de type CFL (Courant-Friedrich-Lewy) qui affectent les méthodes aux différences finies explicites sur une grille eulérienne et qui sont si pénalisantes du point de vue de la taille des pas de temps. Par ailleurs le caractère eulérien des codes Vlasov (du fait de l'utilisation d'une grille fixe) est particulièrement bien adapté au parallélisme, chaque processeur « garde ses particules » au cours de l'évolution temporelle.

La physique de base pour modéliser les phénomènes collectifs dans les plasmas est bien connue : On utilise la description cinétique, où chaque composante du plasma est décrite par une fonction de distribution f(r, v, t). Celle-ci correspond à une moyenne statistique sur un grand nombre de réalisations du système physique considéré. Le produit fdrdv est le nombre moyen de particules de l'espèce considérée dont la position et la vitesse se trouvent respectivement dans un élément de volume dr de l'espace ordinaire et dans un élément de volume dv de l'espace des vitesses, centrés respectivement autour des valeurs r et v. La fonction de distribution contient une information bien supérieure à celle contenue dans les grandeurs fluides, puisqu'elle inclut tous les détails de la répartition des vitesses des particules. L'emploi de la description cinétique est particulièrement nécessaire dans ces plasmas collectifs où la fonction de distribution s'écarte sensiblement de la courbe gaussienne correspondant à l'équilibre thermodynamique, appelée fonction de distribution de Maxwell Boltzmann (ou maxwellienne). L'évolution spatio-temporelle de la fonction de distribution d'une espèce de particules est gouvernée par une équation cinétique. Dans la limite où les effets collectifs sont prédominants sur les collisions binaires l'équation cinétique correspondante est l'équation de Vlasov,

$$\frac{\partial f}{\partial t} + \vec{v}.\frac{\partial f}{\partial x} + \frac{q}{m}\left(\vec{E} + \vec{v} \wedge \vec{B}\right)\frac{\partial f}{\partial v} = 0$$

qui traduit simplement la conservation de f le long des trajectoires de particules soumises aux champs moyens E et B. L'équation de Vlasov doit être couplée aux équations de Maxwell qui régissent l'évolution des champs électromagnétiques en présence de la densité de charge moyenne et du courant moyen, calculés de manière auto-cohérente à partir des fonctions de distribution. La résolution numérique du système Vlasov-Maxwell tridimensionnel présente une difficulté considérable du fait de la taille immense du problème. En effet, le système Vlasov-Maxwell est posé dans l'espace des phases et dépend donc de sept variables (trois variables d'espace, trois variables de vitesse ainsi que le temps (x,y,z,v_x,v_y,v_z,t)) pour chaque espèce de particules. Aussi bien pour le calcul que pour la visualisation, il va donc falloir des ressources informatiques considérables.

De ce fait, ce problème représente un champ d'application naturel pour la mise au point d'une grille de calcul mettant en commun des ressources distribuées. En outre pour l'analyse des résultats, on est amené à traiter une masse considérable de données et à développer de nouveaux concepts de visualisation pour un problème posé dans un espace à six dimensions.

Ce travail sera découpe en trois parties : en premier lieu, nous nous intéresserons à la physique des plasmas de manière générale. Ensuite, nous introduirons les équations cinétiques pour aboutir sur l'équation de Vlasov que nous étudierons d'un point de vue purement théorique. Enfin nous nous intéresserons aux différentes techniques de programmation parallèle afin de les appliquer à ce code Vlasov. Mais tout d'abord, un petit préambule sur la fusion nucléaire contrôlée

Préambule :

Fusion nucléaire contrôlée

Pourquoi ?

La consommation annuelle d'énergie dans le monde est de l'ordre de 400 10^{18} Joules soit en moyenne 80 GJ par personne, et ce chiffre ne peut guère n'évoluer qu'à la hausse. Pour fixer les idées, si 5 milliards d'habitants vivaient en consommant autant d'énergie qu'un européen, ce chiffre devrait doubler. A l'échelle de un ou deux siècles, ce paysage va sans doute considérablement évoluer à cause de la raréfaction du pétrole et du gaz et de considérations sur l'impact des gaz à effet de serre sur le climat. Si l'on essaie d'avoir une vue à long terme et que l'on étudie quelles sources d'énergie sont compatibles avec un maintien sur quelques millénaires d'une consommation du niveau de l'actuel, on ne peut guère envisager que quatre scénarios :

- les énergies renouvelables, à condition d'y consacrer de grandes surfaces, la puissance produite variant entre 1 (barrages) et 200 W/m^2 (solaire, éolien). Pour produire les 80 GJ/an du " terrien moyen " il faut entre 12 et 2500 m^2.
- la fission à condition qu'elle consomme des noyaux fertiles abondants comme l'uranium 238 ou le thorium 232 (d'où la nécessité de neutrons rapides)
- la fusion thermonucléaire deutérium-tritium qui, comme on le verra plus tard, consomme principalement du lithium.
- les hybrides fusion-fission, qui permettraient de brûler des noyaux fertiles abondants tout en minimisant les déchets radioactifs à long terme.

introduction

Le mécanisme du rayonnement des étoiles s'apparente à celui de la fusion nucléaire. En effet, au sein des étoiles, les noyaux légers fusionnent et produisent des noyaux plus lourds. Au

cours de cette réaction de fusion, la masse du noyau produit est inférieure à la somme des masses des noyaux légers d'origine. La différence de masse, en vertu de la célèbre relation d'Einstein, $E=mc^2$, est alors convertie en énergie. Bien que l'énergie libérée par la fusion nucléaire soit considérable, les réactions de fusion ne se produisent pas spontanément, du moins dans les conditions de température et de pression auxquelles nous sommes habitués. Ainsi, la probabilité d'observer une réaction de fusion entre deux noyaux d'hydrogène à la surface de la terre est quasiment nulle. En effet, pour fusionner, les noyaux, qui sont chargés positivement, doivent d'abord vaincre leur tendance naturelle à se repousser. Ceci est possible lorsque la matière est dans des conditions extrêmes comme au cœur du Soleil (pression énorme et température de plusieurs millions de degrés).

L'intérêt de la fusion contrôlée est double : d'une part, le deutérium est abondant dans la nature (eau lourde des océans) et le tritium peut être facilement obtenu à partir du lithium (dont le minerai est également abondant) ; ainsi les réserves énergétiques estimées pour la fusion couvriraient nos besoins sur des millénaires; d'autre part, cette forme d'énergie est non polluante, les produits résultant des fusions contrôlées n'étant pas radioactifs.

1. Réaliser une fusion nucléaire.

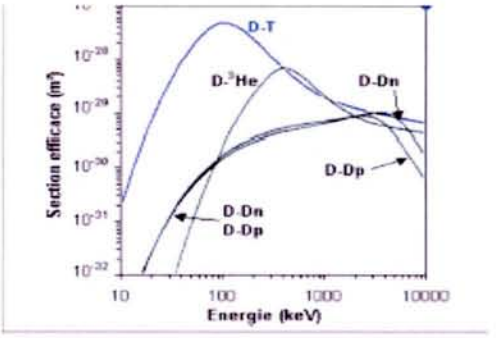
À l'instar de la bombe H qui fut une première réussite en matière de fusion, l'objectif est de canaliser l'énergie produite afin de l'utiliser a des fins industrielles. Dans un plasma chaud (soit plusieurs millions de degrés), les noyaux et leurs électrons se séparent permettant, alors, la fusion de 2 noyaux légers. Ce principe demande biensur la présence d'un champ magnétique toroïdal intense pour confiner le plasma. Actuellement, la réaction qui semble la plus facile à mettre en œuvre dans les tokamaks est la fusion deuterium-tritium :

$D + T \Rightarrow {}^4He + n + 17{,}6\,MeV$

Cela dit, d'autres types de réactions sont aussi possibles comme :

$$D + D \Rightarrow \begin{cases} {}^3He + n + \mathbf{3{,}3MeV} \\ T + p + \mathbf{4MeV} \end{cases}$$

$$D + {}^3He \Rightarrow {}^4He + p + \mathbf{18{,}3MeV}$$



Mais continuons de nous intéresser à la fusion D-T, et faisons le bilan de puissance du plasma.

Pour que le plasma soit stationnaire, il faut que la source d'énergie compense les pertes.

La variation temporelle de l'énergie du plasma $^{dW}/_{dt} = P_{\alpha} + P_{ext} - P_{pertes}$ doit être positive pour que la réaction soit rentable... Pour le moment, ce rendement Q est inférieur à 1.

Soit un gaz ayant une température T, on sait que la distribution de vitesse des particules se définit par l'équation statistique de Maxwell-Boltzman :

$$f(v) = \left(\frac{m}{2\pi kT}\right)^{3/2} e^{-mv^2/2kT} . v^2 dv$$

Ce qui est la probabilité pour que la vitesse de la particule soit entre v *et* $v + dv$. Il faut que la température soit de l'ordre de quelques keV pour un plasma.

Décrivons, maintenant, brièvement, les deux types de fusion…

2. La fusion par faisceaux (fusion inertielle)

On comprime un plasma de quelques milligrammes à des densités de l'ordre de 10 tonnes par mètre cube ($6\ 10^{30}\ m^{-3}$), les réactions de fusion les portant à des températures de 150 millions de degrés, soit une pression de 10^{16} Pascal. Il n'y a pas de confinement, la cible explosant à la vitesse des ions en moins d'une nanoseconde. Pour comparer avec la fusion magnétique, le libre parcours moyen d'un ion pour une réaction de fusion est inférieur au millimètre. L'énergie de fusion dégagée par l'explosion d'une cible est de quelques centaines de MJ.

3. Les boîtes magnétiques

On maintient par un champ magnétique des plasmas chauds. On vise des plasmas de 10^{20} particules par mètre cube (250 000 fois moins que dans l'air ambiant) à 150 millions de degrés soit une pression de 0.2 MPa. Le temps de confinement de l'énergie est de quelques secondes. La puissance fusion est de l'ordre de quelques mégawatts par mètre cube. Compte tenu de la

valeur de la section efficace et de la densité il faut à un ion deutérium environ 50 secondes pour subir une réaction de fusion pendant lesquelles il parcourt environ 50000 km.

4. conclusion

En physique des plasmas, les problèmes sont liés aux faisceaux de particules et à la fusion thermo-nucléaire. La fonction à déterminer par les équations de Vlasov est la densité de probabilité de trouver une particule au point x avec la vitesse v, cette fonction a deux variables (x,v) où chaque variable est à une, deux ou trois dimensions1. Pour capturer les phénomènes physiques on doit mettre suffisamment de points dans chacune des directions d'espace, ce qui entraîne des problèmes gourmands en mémoire et donc en temps processeur. Ces codes sont parallélisés par des méthodes de décomposition de domaine soit en utilisant des bibliothèques de passage de messages telle que MPI, soit par directives avec OpenMp.

Partie I

Généralités sur les plasmas

En 1928, le physicien Langmuir a introduit le terme de plasma pour désigner un gaz ionisé électriquement neutre dans les tubes à décharge. Par extension, on a défini le plasma comme un « nouvel » état de la matière. Mais, la description de ce type de gaz a été faite pour la première fois au début du XX^{e}s par Heaviside qui étudiait la propagation des ondes à grande distance. Depuis ce moment, la physique des plasmas n'a fait que se développer et joue un rôle important dans l'astrophysique et la cosmologie ainsi que pour les recherches sur la fusion nucléaire contrôlée (projet ITER).

1. Quelques propriétés

Dans les plasmas, les densités d'ions et d'électrons sont équivalentes ; car des forces électromagnétiques apparaissent lorsque ce n'est plus le cas ; elles ont pour but de ramener la neutralité. Ce rapport appelé degré d'ionisation se note α.

$$\alpha = \frac{n}{n + n_0}$$

où n_0 est la densité (nombre de particules par unité de volume) des neutres et n celle des électrons (ou des ions positifs)

En pratique, cette valeur varie entre 10^{-10} et 1. Chaque gaz admet une valeur critique α_0 telle que

$\alpha<\alpha_0$: Gaz faiblement ionisé (la fréquence des collisions entre électrons et neutres ou entre ions et neutres est supérieure à la fréquence électrons-électrons ou électrons-ions) ; donc on a cette inégalité : $\nu_{e0} > \nu_{ee}, \nu_{ei}$.

$\alpha > \alpha_0$: Gaz fortement ionisé (moins de collisions entre électrons et neutres qu'entre les particules chargées)

Le gaz est totalement ionisé si $\alpha=1$.

De plus, dans un plasma dont la température est supérieure à 1eV[1], on dit que le gaz (complètement ionisé) constitue un plasma. Si on chauffe un gaz à cette température, l'énergie moyenne (3kT/2) peut devenir égale à l'énergie d'ionisation E_i de l'une des molécules présentes et dès qu'une collision se produit, il y a ionisation de celle-ci.

En équilibre thermodynamique, l'ionisation par collisions engendre des processus de recombinaison électrons-ions qui ont été décrits par M. N. Saha en 1920. L'équation qui en résulte s'écrit sous la forme :

$$\frac{n_e n_i}{n_0} = \frac{\prod^e \prod^i}{\prod^0} \left(\frac{\sqrt{2\pi k T m_e}}{h} \right)^3 .e^{-E_i/kT} \quad (1.1)$$

2. Les oscillations du plasma

Cet équilibre peut être, à tout moment rompu par un excès de charge électrique positive ou négative. Le plasma tend alors à revenir à son équilibre initial : cette perturbation engendre des phénomènes d'oscillation pendulaire autour de son état d'équilibre : oscillations électroniques du plasma. Ces oscillations sont principalement dues aux électrons beaucoup plus mobiles que les ions. La fréquence de ces oscillations électroniques est :

$$f_{pe} = \frac{\omega_{pe}}{2\pi} \text{ avec } \omega_{pe} = \left(\frac{n_{e0} q_e^2}{m_e \varepsilon_0} \right)^{1/2} \quad (2.1)$$

Elles ont été observées pour la première fois par F. M. Penning et détaillées par L. Tonks et I. Langmuir en 1938.

[1] Énergie égale au produit kT : 1eV~11600 K

Ce problème s'étudie de manière quantitative, en posant les équations générales d'une oscillation de charge électronique avec les hypothèses ci-dessous :

-Les ions sont immobiles ;
-Les collisions sont négligeables ;
-L'agitation thermique est négligeable ;
-Les oscillations sont de faible amplitude ;
-Il n'y a pas de champs électriques ou magnétiques extérieurs.

On peut donc écrire les équations hydrodynamiques des électrons :

- Équation de conservation des particules :

$$\frac{\partial n_e}{\partial t} + \vec{\nabla}.n_e\vec{v}_e = 0 \quad (2.2)$$

- Équation de conservation de la quantité de mouvement :

$$n_e m_e\left(\frac{\partial}{\partial t} + \vec{v}_e.\vec{\nabla}\right)\vec{v}_e = n_e q_e \vec{E} \quad (2.3)$$

Compte tenu des hypothèses ci-dessus, dans cette dernière équation, on a négligé les termes de pression cinétique et de collisions ; n_e devient

$$n_e(\vec{r},t) = n_{e0} + n_{e1}(\vec{r},t). \quad (2.4)$$

où n_{e1} est une petite perturbation.

En considérant également, $\vec{v}_e = \vec{v}_{e1}$ *et* $\vec{E}_1 = \vec{E}$ comme étant des petites perturbations d'ordre 1 et ω une fréquence donnée, cette équation ci-dessus se décompose en 2 équations complexes :

$$\begin{cases} -i\omega n_{e1} + n_{e0}\vec{\nabla}.\vec{v}_{e1} = 0 \\ -i\omega n_{e0} m_e \vec{v}_{e1} = n_{e0} e\vec{E}_1 \end{cases} \quad (2.5)$$

En tenant compte de la neutralité du plasma dans un milieu uniforme ($n_{e0}q_e + n_{i0}q_i = 0$), on en déduit :

$$\vec{v}_{e1} = \frac{-q_e}{i\omega m_e}\vec{E}_1$$
$$n_{e1} = \frac{n_{e0}q_e}{\omega^2 m_e}\vec{\nabla}.\vec{E}_1 \quad (2.6)$$

soit, en introduisant la fréquence plasma :

$$(\vec{\nabla}.\vec{E}_1)\left(1 - {\omega_{pe}^2}/{\omega^2}\right) = 0 \quad (2.7)$$

Après avoir linéarisé ces équations pour de petites perturbations, aboutissant a un système d'équations aux dérivées partielles, ce système se réduit à un système linéaire admettant des solutions que si son déterminant est nul.

$$D(\omega,\vec{k}) = 0 \quad (2.8)$$

Elle est appelée équation de dispersion du plasma.

Toutefois, les oscillations dépendent, aussi, de l'agitation thermique ; on remplace la relation $\omega=\omega_p$ dans l'équation de dispersion par la relation de *Bohm-Gross*

$$\omega^2 = \omega_{pe}^2 + 3k^2 v_{T_e}^2 \quad (2.9)$$

où k est le nombre d'onde et $v_{T_e} = \sqrt{{kT_e}/{m_e}}$ vitesse moyenne d'agitation thermique électronique.

Ce résultat montre que les oscillations de plasma se propagent à une vitesse de phase plus grande que la vitesse d'agitation thermique

3. Ondes électromagnétiques

Le plasma peut également supporter des ondes électromagnétiques transverses.

En supposant un champ électrique alternatif $\vec{E}(\omega)$ appliqué localement, on obtient la vitesse des électrons par intégration de l' équation:

$$m_e \frac{d\vec{v}_e}{dt} = q_e\vec{E}_1 \Leftrightarrow \vec{v}_{e1} = \frac{q_e}{m_e}\int \vec{E}_1 dt = \frac{i\omega q_e}{m_e\omega^2}\vec{E}_1 \quad (3.1)$$

dans laquelle le terme de perturbation de deuxième ordre a été négligé.

De plus, on sait que le vecteur courant électrique se note :

$$\vec{J}_{e1} = n_{e0}q_e\vec{v}_{e1} = i\omega\varepsilon_0\frac{\omega_{pe}^2}{\omega^2}\vec{E}_1 \quad (3.2)$$

Il est en quadrature de phase avec le champ E_1. De même, il existe un courant ionique J_{i1} :

$$\vec{J}_{i1} = i\omega\varepsilon_0\frac{\omega_{pi}^2}{\omega^2}\vec{E}_1 \quad (3.3)$$

où ω_{pi} est la fréquence de plasma ionique.

Pour avoir le courant total J_1, on doit ajouter un courant de déplacement J_{D1} :

$$\vec{J}_{D1} = \varepsilon_0 \frac{\partial \vec{E}_1}{\partial} = -i\omega\varepsilon_0 \vec{E}_1 \quad (3.4)$$

La somme de tous les courants nous donne le courant total J_1 :

$$\vec{J}_1 = \vec{J}_{D1} + \vec{J}_{e1} + \vec{J}_{i1} = -i\omega\varepsilon_0 \left(1 - \frac{\omega_p^2}{\omega^2}\right)\vec{E}_1 \quad (3.5)$$

où la fréquence plasma s'obtient en faisant la somme des deux fréquences (électronique + ionique)

$$\omega_p = \sqrt{\omega_{pe}^2 + \omega_{pi}^2} \quad (3.6)$$

Cette théorie ne s'applique qu'à des plasmas isotropes : où la permittivité est égale à l'indice de réfraction N au carré selon la relation:

$$K = N^2 = \left(\frac{kc}{\omega}\right)^2 = 1 - \left(\frac{\omega_p^2}{\omega^2}\right). \quad (3.7)$$

Cela montre que ces ondes ne peuvent se propager que pour des fréquences supérieures à la fréquence plasma ($\omega > \omega_p$).

4. Interactions entre particules

4.1 Ecrantage de Debye

Chaque particule du plasma se déplace avec une trajectoire propre qui dépend de l'interaction avec ses voisines. La distance minimale pour que deux électrons aient leur énergie potentielle identique à l'énergie cinétique d'agitation thermique est la longueur de Landau définie par r_0

$$kT = \frac{e^2}{4\pi\varepsilon_0 r_0} \Leftrightarrow r_0 = \frac{e^2}{4\pi\varepsilon_0 kT} \quad (4.1)$$

qui intervient dans les phénomènes de collisions.

Cependant, il existe, aussi, des interactions collectives, à grande distance, que l'on ne doit pas négliger dans la dynamique du plasma. Pour le montrer, supposons un plasma homogène et infini ; l'énergie d'interaction entre électron-électrons se calcule par l'intégrale suivante :

$$\langle u_{ee} \rangle = \int_0^\infty \frac{e^2 n_e r}{\varepsilon_0} dr \quad (4.2)$$

qui diverge fortement.

De même, l'énergie moyenne d'interaction électrons-ions tend également vers l'infini. De ce fait, l'interaction de l'électron avec le plasma reste nulle.

Cela nous montre que les interactions lointaines sont plus importantes que les autres ; on mesure cela grâce au potentiel électrique défini par l'équation de Poisson :

$$\nabla^2\Phi = -\rho/\varepsilon_0 \quad (4.3)$$

La charge totale étant $\rho = \rho_i + \rho_e$.

La longueur critique d'interaction collective est nommée longueur de Debye. Selon les conditions aux limites (x infini), l'équation de Poisson est du type :

$$\frac{d^2\Phi}{dt^2} + n_e q_e/\varepsilon_0 = 0 \quad (4.4)$$

dont la solution générale s'écrit :

$$\Phi(x) = a + bx - \frac{n_e q_e}{2\varepsilon_0} x^2 \quad (4.5)$$

où a,b sont 2 valeurs à déterminer en x=0 pour $\Phi(0)$ et $d\Phi(0)/dx$.

En remplaçant cette forme du potentiel dans l'équation de Poisson, on obtient la longueur de Debye :

$$\lambda_D = \sqrt{\frac{\varepsilon_0 kT}{n_e e^2}} \quad (4.6)$$

Cette longueur joue un grand rôle dans les corrélations de position dans le plasma. Nous allons étudier cette théorie en commençant par les corrélations ions-électrons.

La neutralité électrique ne s'applique que sur la totalité du plasma. Intuitivement, il paraît normal qu'au voisinage d'un ion, il y ait plus d'électrons que d'ions et vice-versa. Nous voulons trouver une distribution de cet équilibre soit une valeur moyenne de la densité électronique notée n_{e0} et l'on admettra que la densité ionique reste constante et égale à $-n_{e0}q_e/q_i$. Au voisinage d'ions qui génèrent un puit de potentiel, la densité électronique fluctue un peu et satisfait l'équation suivante :

$$n_e(r) = n_{e0}.e^{-q_e\Phi/kT} \underset{dl}{\approx} n_{e0}\left(1 - q_e\Phi/kT\right) \quad (4.7)$$

Si on développe, maintenant, l'équation de Poisson que Φ doit satisfaire

$$\nabla^2\Phi = -{}^{q_e(n_e-n_{e0})}\!/_{\varepsilon_0} = \frac{n_{e0}q_e^2}{\varepsilon_0 kT}\Phi \quad (4.8)$$

le potentiel admet une symétrie sphérique autour de l'ion : d'où l'on tire une équation différentielle de second ordre

$$\frac{1}{r^2}\frac{d}{dr}\left(r^2\frac{d\Phi}{dr}\right) = \frac{\Phi}{\lambda_D^2} \quad (4.9)$$

En faisant le changement de variable $\Phi = \frac{u}{r}$, l'équation s'intègre facilement et nous donne la solution générale :

$$\Phi = \frac{1}{r}\left(Ae^{-{}^{1}\!/_{\lambda_D}} + Be^{{}^{1}\!/_{\lambda_D}}\right) \quad (4.10)$$

En tenant compte des conditions aux limites (potentiel nul à grande distance et potentiel ionique pour r tendant vers 0), on trouve donc :

$$\Phi = \frac{q_i}{4\pi\varepsilon_0 r}e^{-{}^{1}\!/_{\lambda_D}} \quad (4.11)$$

En remplaçant le potentiel dans la densité électronique perturbée par cette expression, on a :

$$n_e(r) = n_{e0}\left(1 + Z_i\frac{r_0}{r}e^{-{}^{1}\!/_{\lambda_D}}\right) \quad (4.12)$$

Ces deux formules ne sont valables que si les corrélations sont faibles ($r_0 << r$). À courte distance, il est facile de calculer ces perturbations.

Pour ce qui est des corrélations entre ions, les formules sont pratiquement analogues à celles des électrons [4.7-4.12]. Si on est dans le cas de faibles corrélations,

$$n_i(r) = n_{i0}(1 - {}^{q_i\Phi}\!/_{kT}) \quad (4.13)$$

La résolution de cette équation est identique que précédemment [4.7-4.13], mais la longueur critique est, cette fois, appelée longueur d'écran exacte :

$$\lambda_s^2 = \frac{\varepsilon_0 kT}{n_{e0}q_e^2 + n_{i0}q_i^2} = \frac{\lambda_D^2}{1+Z_i} \quad (4.14)$$

ce qui nous donne pour le potentiel et les densités perturbées :

$$\begin{aligned}\Phi &= \frac{q_i e^{-{}^{1}\!/_{\lambda_s}}}{4\pi\varepsilon_0 r}\\ n_e(r) &= n_{e0}\left(1 + Z_i\frac{r_0}{r}e^{-{}^{1}\!/_{\lambda_s}}\right) \quad (4.15)\\ n_i(r) &= n_{i0}\left(1 + Z_i^2\frac{r_0}{r}e^{-{}^{1}\!/_{\lambda_s}}\right)\end{aligned}$$

En général, pour décrire correctement les corrélations, il convient d'utiliser les densités doubles :

$$n_{ie}(\vec{r}_1,\vec{r}_2) = n_{e0}n_{i0}\left(1 + Z_i \frac{r_0}{r} e^{-r/\lambda_s}\right) \quad (4.16)$$

où r est la distance entre les 2 points.

4.2 Collisions

Dans le plasma, les particules sont en mouvement perpétuel et, du fait de leurs vitesses et positions, celles-ci s'entrechoquent modifiant, ainsi, leurs vitesses, leurs trajectoires et leurs énergies. Une collision est élastique si les particules en jeu ne changent pas d'état interne (il y a seulement déviation avec échange d'énergie cinétique et de quantité de mouvement).

Par contre, si une au moins des particules change d'état ou se crée ou se détruit : c'est une collision inélastique.

Dans la suite, on se propose d'étudier les caractéristiques de ces collisions (élastiques et inélastiques).

a. Collisions élastiques

Pour débuter, on va s'intéresser à la collision entre deux particules de masses m_1 et m_2 et deux positions arbitraires r_1 et r_2 . La force qui s'exerce de la particule m_1 sur la particule m_2 est $\vec{X}_{21} = -\vec{X}_{12}$

On voit que ces forces s'opposent. C'est le principe d'action-réaction. Ces forces d'interaction dérivent du potentiel φ . Ces forces entraînent un centre de gravité sur le segment m_1m_2 . Les positions de ces particules peuvent êtres déterminées à partir de ce centre

$$\vec{r}_1 = \vec{r}_G + \frac{m_2}{m_1 + m_2}\vec{r}$$
$$\vec{r}_2 = \vec{r}_G - \frac{m_1}{m_1 + m_2}\vec{r} \quad (4.17)$$

avec $\vec{r}_G$ le vecteur position du centre de gravité. Avec ça, on peut établir le mouvement relatif à l'aide de l'équation de la dynamique, on obtient alors

$$\vec{X}_{12} = \frac{m_1 m_2}{m_1 + m_2}\frac{d^2}{dt^2}(\vec{r}_1 - \vec{r}_2) \quad (4.18)$$

Cette interaction échange de l'énergie. Les deux formes d'énergie échangées sont l'énergie cinétique (u_K) et l'énergie potentielle (u_H). De plus, on sait que l'énergie cinétique s'exprime de la façon suivante :

$$u_K = \frac{1}{2}m_1\omega_1^2 + \frac{1}{2}m_2\omega_2^2 \quad (4.19)$$

Calculer l'énergie cinétique des deux particules revient à calculer l'énergie cinétique du centre de gravité, en prenant la valeur relative des vitesses ($g = |\vec{\omega}_1 - \vec{\omega}_2|$)

$$u_R = \frac{1}{2}\frac{m_1 m_2}{m_1 + m_2}g^2 \quad (4.20)$$

Pour décrire le phénomène de collision, on se doit de définir la section efficace angulaire. Soit un faisceau de particules identiques ayant un flux ϕ_1 , on peut alors compter le nombre de particules passant par une section choisie (centre de diffusion) le paramètre d'impact est uniforme sur la cible. On appelle dN_1/dt le nombre de particules diffusées par unité de temps pour une petite section $d\Omega$ dans une direction donnée. On peut calculer ce nombre de particules grâce au potentiel et à la section ; cela nous donne

$$dN_1/dt = \Phi_1\sigma(\chi,\varphi)d\Omega \quad (4.21)$$

où φ est le nombre azimutal et χ est la déviation.

σ est appelée section efficace différentielle de collision élastique et a la même unité qu'une surface. On admettra que cette section efficace ne dépend que de la déviation. Ce qui s'écrit

$$\frac{dN_1}{dt} = \Phi_1\sigma(\chi)2\pi\sin\chi d\chi \quad (4.22)$$

Le nombre de particules est entre χ et $\chi + d\chi$, on peut déduire la section efficace à partir de la déviation et du paramètre d'impact p. Pour évaluer l'importance des collisions élastiques sur le phénomène macroscopique, on va définir la section totale de collision élastique

$$\sigma_0 = \int \sigma(\chi) d\Omega \quad (4.23)$$

qui permet de calculer la probabilité qu'une particule soit diffusée, mais, en général, on ne peut pas calculer σ_0 sous cette forme car l'intégrale est infinie.

Dans quelques cas particuliers, on peut avoir des résultats :

- Potentiel de Coulomb. Il faut alors résoudre

$$\sigma_0 = \int_0^{\pi} \frac{A}{\sin^4 \chi/2} 2\pi \sin\chi d\chi \quad (4.24)$$

cette équation diverge en π.

- Sphères impénétrables. On a alors

$$\sigma_0 = \int \frac{D^2}{4} d\Omega = \pi D^2 \quad (4.25)$$

Du fait de la description des collisions de manière classique, on ne peut pas avoir des résultats satisfaisants, il faudrait passer par une description quantique. Mais, ce n'est pas notre but. On va revenir à des interactions coulombiennes et montrer la divergence de la section efficace en introduisant le paramètre d'impact dans la formule précédente.

$$\sigma(\chi) = \left(\frac{p_0}{2}\right)^2 \frac{1}{\sin^4 \chi/2} \quad (4.26)$$

On voit toujours que si la déviation est faible la section efficace tend vers l'infini. On verra dans le prochain chapitre un peu plus en détail l'effet de ces collisions sur le milieu gazeux.

Nous allons continuer ce chapitre en regardant l'effet des collisions inélastiques.

b. *Collisions inélastiques*

On rappelle que ces collisions peuvent créer des nouvelles particules ou en faire disparaître. Les particules peuvent être dans un état fondamental (E_i=0) ou dans des états excités. Ces niveaux d'énergie supérieure sont en général instables. On note l'énergie interne de la molécule de la manière suivante :

$$u_p = E_e + E_v + E_r \quad (4.27)$$

où E_e définit l'énergie cinétique, E_v l'énergie de vibrations et E_r l'énergie de rotation.

On montre que

$$E_e >> E_v >> E_r \quad (4.28)$$

En général, le niveau de l'énergie de vibration est nul. Certaines collisions peuvent donner des ions négatifs stables, c'est le phénomène de fixation. La transformation physique des molécules nécessite de l'énergie. Chaque particule a une énergie cinétique et potentielle

$$U = U_K + U_P \quad (4.29)$$

avec

$$U_K = \sum_i \frac{1}{2} m_i \omega_i^2 \quad (4.30)$$

$$U_p = \sum_i up_i \quad (4.31)$$

où *i* définit le nombre de particules participant à la collision. Après la collision, les énergies se modifient, car les particules échangent de l'énergie entre elles. Il y a deux types de réactions : la réaction endothermique (q< 0) et la réaction exothermique (q> 0). Pour avoir une réaction endothermique, il faut que l'énergie cinétique soit supérieure à l'énergie fournie au système, en fait, une partie seulement de l'énergie cinétique peut servir à la réaction.

Soit ω_G la vitesse du centre de masse des particules, on a alors

$$\vec{\omega}_G = \frac{\sum m_i \vec{\omega}_i}{M} \quad (4.32)$$

où ω_i est la vitesse des particules. On sait que la vitesse totale se décompose en deux vitesses une relative et une du centre de gravité. La somme des vitesses relatives est nulle. Donc, on peut remplacer ces vitesses dans l'énergie cinétique.

$$U_K = U_G + U_R \quad (4.33)$$

avec

$$U_G = \frac{1}{2} M\omega_G^2$$
$$U_R = \sum_i \frac{1}{2} m_i \omega_{ri}^2 \quad (4.34)\ (4.36)$$

On a dit que dans les collisions inélastiques, il y avait changement d'état des particules c'est-à-dire qu'il peut y avoir un nombre de particules différent de celui initial.

Ces collisions entrent dans la description microscopique des fluides que l'on va détailler maintenant.

4.3 Effets collectifs et collisions

C'est la présence des effets collectifs (oscillations de plasmas par exemple) qui différencie le plus un plasma d'un gaz formé de particules neutres. Décrire les effets collectifs suppose que l'on reste à une échelle spatiale dans laquelle les fluctuations peuvent être ignorées, tout en sachant que le plasma n'est pas un continuum mais un milieu discret de particules (de charge e de masse m et de densité n en ne considérant que les électrons ; dans ce qui suit on négligera les mouvements des ions du fait de leur masse considérablement plus grande). C'est ce caractère discret qui implique l'existence de collisions qui deviennent fortes si la distance entre deux particules devient plus petite que la longueur de Landau définie ci-dessus.

On obtient ainsi une estimation du libre parcours moyen

$$\ell \approx n^{-1}(e^2/\varepsilon_0\theta)^{-2} \quad (4.37)$$

où θ désigne la température en unité d'énergie ($\theta = kT$), et de la fréquence de collisions

$$\nu \approx (\theta/m)^{1/2}/\ell \quad (4.38)$$

Une condition pour que les effets collectifs (qui ont une échelle de temps de l'ordre de l'inverse de la pulsation plasma) dominent les effets individuels est que l'on ait

$$\nu/\omega_p \ll 1 \quad (4.39)$$

D'après la formule ci-dessus donnant la fréquence de collisions l'on obtient

$$\nu/\omega_p \approx 1/nD^3 \quad (4.40)$$

où D désigne la longueur de Debye et la condition sur les effets collectifs est valide pour

$$nD^3 \gg 1 \quad (4.41)$$

Pour aller plus loin, cherchons un paramètre sans dimension, noté *g*, mesurant le caractère plus ou moins « discret » du plasma. Imaginons l'expérience de pensée suivante : chaque particule (e,m) est divisée en deux particules, chacune avec une charge *e*/2 et une masse *m*/2 et ainsi de suite. Dans ce processus de dichotomie, la longueur de Debye et l'inverse de la pulsation plasma sont invariants. La densité de particules *n* tend vers l'infini. En revanche, il est clair que le paramètre g (le facteur de grain) doit être divisé par deux à chaque dichotomie, ce qui permet d'affirmer que

$$g = \frac{1}{nD^3} \quad (4.42)$$

À titre d'exemple, les plasmas de fusion ont un facteur de grain de l'ordre de 10^{-8}.

Les effets collectifs dominent donc les propriétés des plasmas sur une échelle de temps de l'ordre de $(\omega_p g)^{-1}$ et sur des échelles spatiales de l'ordre de *D*/*g*.

Donc trouver le modèle mathématique décrivant les effets collectifs d'un plasma revient à chercher une équation invariante par rapport à ce paramètre g. C'est l'équation de Vlasov, que nous allons étudier ci-dessous.

5.Équations cinétiques et équation de Vlasov

5.1 le grand espace des phases

Pour étudier un gaz (décrire son comportement, sa vitesse, sa densité…), on peut introduire l'hydrodynamique ; mais les plasmas sont rarement en équilibre thermodynamique, il est donc nécessaire d'introduire la *fonction de distribution.* On partira d'une description fine de l'état fluide en définissant la densité D dans l'espace des phases avec laquelle nous établirons l'équation d'évolution (équation de Liouville) qui, après intégration, nous donnera les équations cinétiques avec les évolutions des fonctions de distribution de f_1 .

Tout d'abord, on introduit la densité dans l'espace des phases à la dimension 6N+1 c'est-à-dire un espace position un espace vitesse et le temps. On note la densité de probabilité :

$$D(\vec{r}_1,\vec{r}_2,...,\vec{r}_N,\vec{\omega}_1,\vec{\omega}_2,...,\vec{\omega}_N,t) = D \quad (5.1)$$

On note la probabilité de la phase soit à un instant t donné à l'intérieur d'un élément de volume à 6N dimensions situé au voisinage du point ($\vec{r}_1,\vec{r}_2,...,\vec{r}_N,\vec{\omega}_1,\vec{\omega}_2...,\vec{\omega}_N$). Ce petit volume sera noté $\underline{dr_1 d\omega_1}$ pour la particule 1 etc… La somme de ces probabilités est évidemment égale à 1, elle est notée

$$\int D\underline{dr_1}\underline{dr_2}...\underline{dr_N}\underline{d\omega_1 d\omega_2}...\underline{d\omega_N} = 1 \quad (5.2)$$

On voit que la fonction D est symétrique car on peut permuter l'ordre des éléments de volume et de vitesse. De plus, si les particules ne sont pas en corrélation, la densité totale est un produit de N densités. Toutes les équations cinétiques dérivent de cette densité. Pour faciliter la notation, on prendra comme espace des phases le vecteur position q_i le vecteur quantité de mouvement p_i. L'état du système est donc décrit par la fonction

$$D(q_i, p_i, t)d\Omega \quad (5.3)$$

avec $d\Omega$ le produit des $dq_i dp_i$. Pour continuer ce modèle, on doit introduire l'hamiltonien du système H (p_i , q_i), il permet de déterminer les équations de mouvement de ce système

$$\frac{dq_i}{dt} = \frac{\partial H}{\partial p_i}$$
$$\frac{dp_i}{dt} = -\frac{\partial H}{\partial q_i} \quad (5.4)$$

Le théorème de Liouville nous dit qu'il y a conservation du volume de l'espace des phases lors d'une évolution du système, cela se note :

$$D(q_i', p_i', t')d\Omega' = D(q_i, p_i, t)d\Omega \quad (5.5)$$

Si les volumes sont identiques, cela signifie que les densités de probabilité le sont aussi, on dit que D se conserve suivant le mouvement donc la somme des dérivées partielles à t donné est nulle

$$\frac{\partial D}{\partial t} + \sum_{i=1}^{3N} \frac{dq_i}{dt}\frac{\partial D}{\partial q_i} + \sum_{i=1}^{3N} \frac{dp_i}{dt}\frac{\partial D}{\partial p_i} = 0 \quad (5.6)$$

On sait que l'hamiltonien est la somme des énergies cinétiques et potentielles, on peut donc l'écrire sous la forme

$$H = \sum_i \frac{p_i^2}{2m} + \sum_i \varphi_i + \sum\sum_{i<j} \varphi_{ij} \quad (5.7)$$

avec $p_i = m\omega_i$. Le premier terme est celui de l'énergie cinétique, le suivant, celui de l'énergie potentielle de la particule i soumise à des forces extérieures, et le dernier terme, celui de l'énergie potentielle d'interaction entre particules.

5.2 L'équation de Vlasov

Revenons maintenant à notre description des effets collectifs. Il est clair que les corrélations entre particules sont un effet d'ordre g où g est le paramètre de grain défini plus haut. L'équation de Vlasov va donc être la réduction de l'équation pour la fonction D dans le grand espace des phases, à l'ordre zéro en g.

On peut donc écrire que les particules ne sont pas corrélées ce qui revient à factoriser la fonction D dans le grand espace des phases en un produit de fonctions de distribution

$$D(q_1, p_1, q_2, p_2, ..., q_N, p_N, t) = f(q_1, p_1, t) f(q_2, p_2, t) ... f(q_N, p_N, t) \quad (5.8)$$

et l'équation pour D ci-dessus se ramène à l'équation pour f(q,p,t) (fonction de distribution à une particule)

$$\frac{\partial f}{\partial t} + \frac{p}{m} \cdot \frac{\partial f}{\partial q} - \frac{\partial \Phi}{\partial q} \cdot \frac{\partial f}{\partial p} = 0 \quad (5.9)$$

où Φ est le potentiel dans l'approximation de champ moyen donné par l'équation de Poisson auto-cohérente

$$\frac{\partial^2 \Phi}{\partial q^2} + \frac{e}{\varepsilon_0}\left(\int f dp - n_0\right) = 0 \quad (5.10)$$

où n_0 désigne la densité ionique (dans notre cas, on a supposé que les ions forment un milieu continu homogène neutralisant). On peut vérifier sans peine que le système Vlasov-Poisson est bien invariant par rapport à g.

5.3 Le système Vlasov-Maxwell

Le système Vlasov-Poisson ci-dessus peut se généraliser sans peine au cas électromagnétique. On obtient alors le système Vlasov-Maxwell. Soit α un indice désignant chaque espèce de particules présentes dans le plasma. On définit la fonction de distribution de l'espèce α comme $f_\alpha(r, p_\alpha, t)$ et l'on peut alors écrire les différentes équations Vlasov-Maxwell

$$\begin{aligned}
&\frac{\partial}{\partial \vec{r}} \times \vec{E} = -\frac{\partial \vec{B}}{\partial t} \\
&\frac{\partial}{\partial \vec{r}} \cdot \vec{E} = \frac{\rho}{\varepsilon_0} \\
&\frac{\partial}{\partial \vec{r}} \times \vec{B} = \mu_0\left(\vec{j} + \varepsilon_0 \frac{\partial \vec{E}}{\partial t}\right) \qquad (5.11) \\
&\frac{\partial}{\partial r} \cdot \vec{B} = 0 \\
&\frac{\partial f_\alpha}{\partial t} + \frac{\vec{p}_\alpha}{m_\alpha} \cdot \frac{\partial f_\alpha}{\partial \vec{r}} + q_\alpha\left(\vec{E} + \frac{\vec{p}_\alpha}{m_\alpha} \times \vec{B}\right) \cdot \frac{\partial f_\alpha}{\partial \vec{p}_\alpha} = 0
\end{aligned}$$

Les champs sont calculés à partir du système d'équations de Maxwell. Le problème est de savoir la valeur du champ sur tous les points de la fonction de distribution. On sait que ce système est auto-cohérent, ce qui nous permet d'écrire

$$\rho = \sum q \int f d\vec{p}$$
$$j = \sum q \int \frac{\vec{p}}{m} f d\vec{p} \quad (5.12)(5.13)$$

Il est intéressant à remarquer qu'en intégrant l'équation de Vlasov sur l'espace des vitesses, on obtient l'équation de conservation des particules

$$\frac{\partial n}{\partial t} + div\left(n\frac{\mathrm{p}}{m}\right) = 0 \quad (5.14)$$

Par une intégration sur p , on obtient l'équation du mouvement

$$m\left(\frac{\partial \vec{u}}{\partial t} + (\vec{u}\cdot\nabla)\vec{u}\right) = e\left(\vec{E} + \vec{u}\times\vec{B}\right) - \frac{1}{n}\nabla \mathrm{P} \quad (5.15)$$

P étant la pression cinétique du milieu isotrope.

On peut remarquer que cette équation fait intervenir un moment d'ordre supérieur. C'est le problème bien connu de la fermeture des équations hydrodynamiques. Nous y reviendrons ci-dessous (paragraphe 6.2).

6. L'interaction résonante onde-particules

6.1 Le système Vlasov-Poisson et l'effet Landau

Si on se restreint à un système purement électrostatique, l'équation de Vlasov devient

$$\frac{\partial f_\alpha}{\partial t} + v\frac{\partial f_\alpha}{\partial x} + \frac{q_\alpha}{m_\alpha} E \frac{\partial f_\alpha}{\partial v} = 0 \quad (6.1)$$

et dans le cas unidimensionnel, on a, pour le champ électrique :

$$E_x = -\frac{\partial \Phi}{\partial x}$$
$$\frac{\partial E_x}{\partial x} = \frac{\rho}{\varepsilon_0} \quad (6.2)(6.3)$$

Du fait de sa non-linéarité, on ne peut que résoudre cette équation pour des petites perturbations, donc, on a, pour chaque espèce de particules,

$$\begin{aligned} f_\alpha &= f_{\alpha 0} + f_{\alpha 1}(v,x,t); |f_{\alpha 1}| << f_{\alpha 0} \\ \rho &= \rho_0 + \rho_1(x,t); |\rho_1| << \rho_0 \end{aligned} \quad (6.4)(6.5)$$

Cette perturbation crée un champ électrostatique E = E_1(x ,t). En remplaçant ce champ dans l'équation de Vlasov, on obtient les équations linéarisées suivantes :

$$\begin{aligned} &\frac{\partial f_{\alpha 1}}{\partial t} + v\frac{\partial f_{\alpha 1}}{\partial x} + \frac{q_\alpha}{m_\alpha} E \frac{\partial f_{\alpha 0}}{\partial v} = 0 \\ &\rho_1 = \sum_\alpha q_\alpha \int_{-\infty}^{+\infty} f_{\alpha 1} dv \qquad (6.6)(6.7)(6.8) \\ &\varepsilon_0 \frac{\partial E_1}{\partial x} - \rho_1 = \rho_{ext} \end{aligned}$$

Par une transformation de Fourier, on remplace x et t par ω et k, ce qui nous conduit au système linéaire

$$\begin{aligned} &-i(\omega - kv) f_{\alpha 1} - g_{\alpha 1}(v,k) \frac{q_\alpha}{m_\alpha} E_1 f'_{\alpha 0} = 0 \\ &\rho_1 = \sum_\alpha q_\alpha \int_{-\infty}^{+\infty} f_{\alpha 1} dv \qquad (6.9)(6.10)(6.11) \\ &ik\varepsilon_0 E_1 - \rho_1 = \rho_{ext} \end{aligned}$$

g étant la transformée de Fourier de f.

A t=0, la fonction de distribution perturbée s'écrit

$$f_{\alpha 1} = -i\frac{q_\alpha}{m_\alpha}\frac{f'_{\alpha 0} E_1}{\omega - kv} + i\frac{g_{\alpha 1}}{\omega - kv} \quad (6.12)$$

En utilisant le système linéaire ci-dessus, on peut introduire la densité de charge perturbée qui a pour équation

$$\rho_1 = \sum_\alpha -i\frac{q_\alpha^2}{m_\alpha}\int_{-\infty}^{+\infty}\frac{f'_{\alpha 0} dv}{\omega - kv} E_1 + \sum_\alpha i\frac{q_\alpha}{m_\alpha}\int_{-\infty}^{+\infty}\frac{g_{\alpha 1} dv}{\omega - kv} \quad (6.13)$$

Lorsque $\omega \approx kv$ l'interaction devient résonante. On peut remarquer que la densité de charge est complexe, mais que, seule la première partie dépend du champ électrique E_1, elle est appelée densité de charge collective. L'autre partie dépend des conditions initiales de $g_{\alpha 1}$. En reportant la densité de charge perturbée dans l'équation de Poisson, on a

$$ik\varepsilon_0 E_1 - \rho_{coll} = \rho_{init} + \rho_{ext} \equiv \rho_{exc} \quad (6.14)$$

où ρ_{exc} est la densité de charge des différentes excitations.

L'équation de Vlasov qui lie le champ électrique à une densité de charge collective permet de définir la susceptibilité longitudinale $\chi_L(k,\omega)$ à l'aide de la formule

$$\rho_{coll} = -ik\varepsilon_0 \chi_L E_1 \quad (6.15)$$

Dans la première partie, on a vu l'expression de la fréquence plasma pour une espèce, mais nous pouvons élargir cette expression pour α espèces, cela nous donne

$$\omega_{p\alpha} = \sqrt{\left(q_\alpha^2 n_\alpha / m_\alpha \varepsilon_0\right)} \quad (6.16)$$

On peut en déduire la forme de la susceptibilité longitudinale qui est

$$\chi_L(k,\omega) = \sum_\alpha \frac{\omega_{p\alpha}^2}{k} \int_{-\infty}^{+\infty} \frac{f'_{\alpha 0} dv}{\omega - kv} \quad (6.17)$$

Avec l'équation de Poisson, on peut également déterminer la permittivité longitudinale :

$$\varepsilon_L(k,\omega) = 1 + \chi_L(k,\omega) = 1 + \sum_\alpha \chi_{L\alpha}(k,\omega) \quad (6.18)$$

La permittivité longitudinale fait écran au champ d'excitation extérieur. Celle-ci varie en fonction du vecteur d'onde et de la pulsation. Cette variation fait que le plasma est dispersif. Si les fonctions de distribution des particules du plasma sont du type maxwellien,

on peut avoir de manière explicite la relation de dispersion des ondes. On définit deux catégories d'ondes :

- les ondes haute-fréquence ($\omega > \omega_p$) : les ondes de plasma électroniques
- les ondes basse-fréquence ($\omega < \omega_p$) : les ondes acoustiques

On note la fonction de distribution à l'équilibre des électrons :

$$f_{0e}(v) = \frac{e^{-v^2/v_{th}^2}}{\sqrt{\pi} v_{th}} \quad (6.19)$$

avec $v_{th} = \sqrt{2K_B T / m_e}$ la vitesse thermique électronique.

En utilisant les formules ci-dessus, la permittivité s'exprime alors

$$\varepsilon_L(k,\omega) = 1 + \frac{1 - \Phi(z) + i\sqrt{\pi} z e^{-z^2}}{k^2 \lambda_D^2} \quad (6.20)$$

avec $\Phi(z) = 2ze^{-z^2} \int_0^z e^{t^2} dt$.

Le point important est que ω peut être complexe pour k réel. En se restreignant au cas d'une distribution électronique dans un milieu ionique continu homogène, et en supposant que la partie imaginaire de ω reste petite devant la partie réelle la résolution de l'équation de dispersion ci-dessus conduit à

$$\mathrm{Im}(\omega) = \frac{\pi}{2} \frac{\omega_p^2 \mathrm{Re}(\omega)}{k^2} \left(\frac{df_{0e}}{dv} \right)_{v = \frac{\mathrm{Re}(\omega)}{k}} \quad (6.21)$$

La partie réelle de ω garde la valeur de Bohm-Gross donnée à la fin du paragraphe 2 ci-dessus. On voit que s'il y a suffisamment de particules autour de la vitesse de phase $\mathrm{Re}(\omega)/k$ la partie imaginaire de ω prendra une valeur finie ; si la distribution est une fonction décroissante de la vitesse alors cette valeur sera négative, conduisant au phénomène bien connu *d'amortissement Landau.*

6.2 Interaction onde-particules : particules piégées.

Le calcul ci-dessus suppose que l'on ait linéarisé le système Vlasov-Poisson. On peut montrer que cette hypothèse suppose que l'on étudie l'évolution d'une petite perturbation sur des temps courts ($t \approx \omega_p^{-1}$). Sur des temps plus longs, apparaissent les effets non linéaires qui ne peuvent être traités de manière analytique. Il faut recourir à la *simulation numérique*.

La figure ci-après [p.37] extraite du papier « Vlasov models for laser-plasma interaction » [2] montre une représentation de la fonction de distribution électronique dans l'espace des phases (x-p_x) en présence d'une onde plasma électronique. En rouge pour $f > 510^{-3}$ et en couleurs de plus en plus froides pour $f < 5.10^{-3}$. On voit apparaître une distribution de particules piégées dans le puits de potentiel de l'onde plasma autour de la vitesse de phase. En trait noir sont représentées les trajectoires séparatrices entre particules circulantes et particules piégées.

Ces résultats ont été obtenus par résolution numérique du système Vlasov-Maxwell. Nous reviendrons sur les problèmes de la simulation numérique dans la partie suivante.

On a vu au paragraphe 5.3 ci-dessus qu'il était possible de remplacer l'équation cinétique de Vlasov pour la fonction de distribution f dans un espace des phases en général à 6 dimensions (trois variables de position, trois variables de vitesse) par une hiérarchie d'équations fluides pour les divers moments successifs de f dans un espace à trois dimensions seulement.

Mais pour que ces équations soient utilisables, il faut pouvoir stopper la hiérarchie en introduisant une relation de fermeture c'est-à-dire une relation d'état entre les quantités fluides. Or, en l'absence de *collisions, il n'existe aucun mécanisme pour garantir la*

[2] *Vlasov models for laser-plasma interaction*, Pierre Bertrand, M. Albrecht-Marc, T. Réveillé and A. Ghizzo, à paraître dans Statistical Physics and Transport Theory, 2005

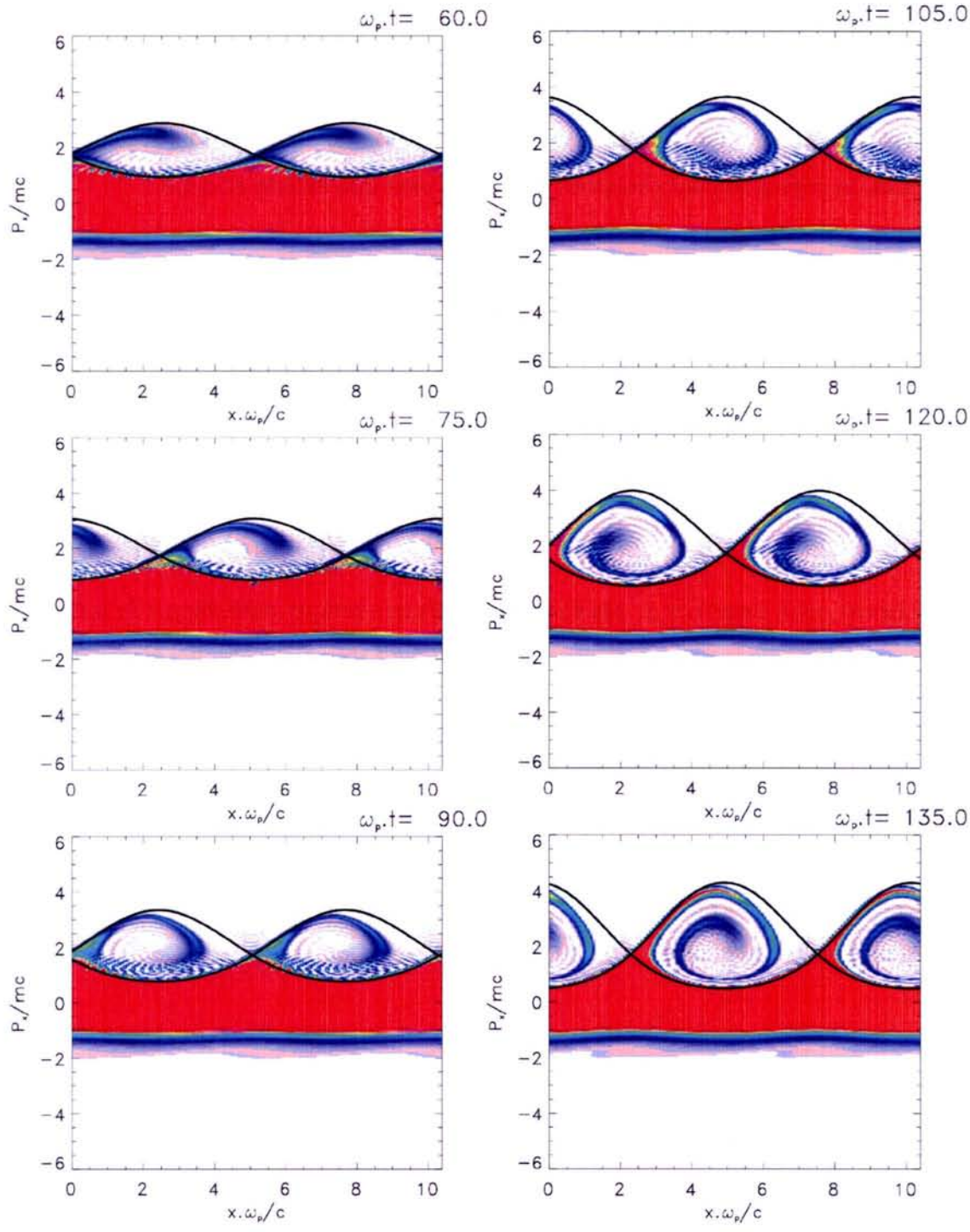
relaxation de f vers une situation d'équilibre thermodynamique local. Ce problème a largement été abordé dans un travail récent [3].

Une approche physique simple permet de bien cerner le problème. Les phénomènes collectifs dans un plasma étant de nature essentiellement ondulatoire, nous introduisons la vitesse de phase $v_\varphi = \omega/k$ du phénomène ondulatoire que l'on veut étudier et nous le comparons à la vitesse thermique v_{th} des électrons. Plusieurs cas limites se présentent :

- $v_\varphi >> v_{th}$: le phénomène n'affecte que la queue de la fonction de distribution et l'on suppose que l'équilibre thermodynamique local (distribution maxwellienne locale) est conservé ; le processus ondulatoire est rapide ; il est décrit par l'équation adiabatique de fermeture $P/n^3 =$ cste (la puissance trois correspond à un mouvement à un degré de liberté, ce qui est le cas des oscillations longitudinales de plasma).
- De plus si $v_\varphi \to \infty$ la fonction de distribution f apparaît comme une distribution de Dirac ; l'équation de Vlasov est alors équivalente au système fermé des équations des équations fluides à pression négligeable, telles que nous les avons utilisées au paragraphe 2 ci-dessus lorsque nous avons décrit les oscillations de plasma.
- $v_\varphi << v_{th}$: cette fois les processus sont lents et l'on peut supposer T constant (processus isotherme).

Bien entendu, le cas général ne correspond pas aux cas limites ci-dessus. Sur la figure ci-dessous, il est clair que l'on ne peut plus supposer que la fonction de distribution reste maxwellienne. La fermeture des équations est donc en général impossible : on ne peut pas utiliser une description fluide et il faut impérativement utiliser le modèle Vlasov-Poisson.

[3] *Numerical study on hydrodynamic and quasi-neutral approximations for collisionless two-species plasmas*, S. Labrunie, A. Carillo and P. Bertrand, Journal of Computational Physics, **200**, 267-298 (2004) et références incluses.


ω_p.t= 60.0
ω_p.t= 105.0
ω_p.t= 75.0
ω_p.t= 120.0
ω_p.t= 90.0
ω_p.t= 135.0
P_x/mc
x.ω_p/c

6.3 Le système Vlasov-onde

On peut aller plus loin et remarquer que, en présence d'interaction résonante onde-particules, la population des particules du plasma peut être divisée en deux :

- les *particules non résonantes* : elles correspondent à la partie en rouge sur la figure ci-dessus. Leur distribution reste maxwellienne et leur contribution peut être décrite par la fonction diélectrique D habituellement obtenue à partir d'un modèle fluide;

- les *particules (quasi) résonantes* : elles seules nécessitent en fait un traitement complètement cinétique.

Considérons pour simplifier le cas d'un plasma électrostatique périodique (de longueur L) en présence d'une onde plasma monochromatique. Soit $\Phi(x,t)$ le potentiel électrostatique

$$\Phi(x,t) = \phi_k(t)\exp\big(i(kx-\omega t)\big) + \text{C.C.} \quad (6.22)$$

où k et ω satisfont à la relation de dispersion $D(k,\omega)=0$ du plasma non perturbé (particules non résonantes). Considérons maintenant la population des N électrons résonants. Leur densité s'écrit

$$\sigma(x,t) = \frac{n_0 L}{N}\sum_{\ell=1}^{N}\delta\big(x-x_\ell(t)\big) \quad (6.23)$$

n_0 étant la densité des électrons et $x_\ell(t)$ la position d'un de ces N électrons résonants. Quant à la contribution des électrons non résonants, elle n'apparaît que par l'intermédiaire de $D(k,\omega)$. On peut donc écrire les équations couplées

$$\begin{aligned}\frac{d\phi_k}{dt} &= \frac{in_0 e}{\varepsilon_0 k^2 N}\beta\sum_{\ell=1}^{N}\exp\big(-i(kx_\ell-\omega t)\big)\\ \frac{d^2x_\ell}{dt^2} &= \frac{iek}{m}\phi_k\exp\big(-i(kx_\ell-\omega t)\big)+\text{C.C.}\end{aligned} \quad (6.24)(6.25)$$

où $\beta = \left(\frac{\partial D}{\partial \omega}\right)^{-1}$

Pour simplifier les notations, on utilise les grandeurs normalisées

$$\begin{aligned} \tau &= \alpha t \\ \xi_\ell &= k x_\ell - \omega t \\ V &= \frac{e k^2}{m \alpha^2} \phi_k \\ \alpha^3 &= \omega_p^2 \beta \end{aligned} \qquad (6.26)$$

Service Commun de la Documentation
de l'Université Henri Poincaré - NANCY 1
B.I.U. Sciences
Rue du Jardin Botanique
54600 VILLERS-LES-NANCY (FRANCE)

Les équations dynamiques ci-dessus qui décrivent l'évolution couplée des électrons et de l'amplitude complexe de l'onde s'écrivent dans ce système normalisé

$$\begin{aligned} \frac{dV}{d\tau} &= \frac{i}{N} \sum_{\ell=1}^{N} \exp(-i\xi_\ell) \\ \frac{d^2\xi_\ell}{d\tau^2} &= iV \exp(i\xi_\ell) - iV^* \exp(-i\xi_\ell) \end{aligned} \qquad (6.27)(6.28)$$

et dérivent du hamiltonien auto-cohérent

$$H(\xi_\ell, p_\ell, V, V^*) = \sum_{\ell=1}^{N} \left(\frac{p_\ell^2}{2} - V \exp(i\xi_\ell) - V^* \exp(-i\xi_\ell \right) \quad (6.29)$$

La construction de cet hamiltonien se décompose en trois parties

-mouvement des électrons résonants

-propagation de l'onde

-couplage onde-particules

On vérifie bien que ce système dynamique conserve bien l'énergie H et le moment $P = \sum_\ell p_\ell + N|V|^2$. (6.30)

On s'intéresse alors à la limite cinétique $N \to \infty$ la distribution discrète σ des électrons résonants est remplacée par la fonction de distribution f_{res} et le système dynamique pour les N électrons résonants tend vers le système *Vlasov-onde*.

On normalise enfin selon $k=\frac{2\pi}{L}, L=2\pi, \omega=\omega_p=1$ et l'on revient aux grandeurs réelles ; le système final Vlasov onde s'écrit:

$$\begin{aligned}
&\left[\partial_t + p\partial_x + \sqrt{n_{res}/n_0}(X\sin x + Y\cos x)\partial_p\right] f = 0 \\
&\dot{X} = Y + \sqrt{n_{res}/n_0}\iint f(x,p)\sin x\, dx\, dp \qquad (6.31) \\
&\dot{Y} = -X + \sqrt{n_{res}/n_0}\iint f(x,p)\cos x\, dx\, dp
\end{aligned}$$

n_{res} étant la densité des particules résonantes.

L'intérêt d'une telle description est double:

- l'effort de résolution de l'équation de Vlasov est maintenant limité au domaine plus restreint des particules (quasi) résonantes ce qui d'un point de vue numérique est un avantage indéniable ;

- plus généralement, cette description, qu'elle soit discrète ou continue permet de mieux appréhender les notions de phénomènes individuels et collectifs tels que nous les avons présentés dans le premier chapitre. Le lecteur est renvoyé au livre récent de Elskens et Escande pour un approfondissement de ce sujet [4].

6.4 Remarque : l'approximation de quasi-neutralité.

A l'équilibre stationnaire ($\partial/\partial t = 0$) toute fonction de l'énergie est solution de l'équation de Vlasov

$$f_e = f\left(\tfrac{1}{2}m_e v^2 - e\phi(x)\right) \quad (6.32)$$

En supposant que f_e est une maxwellienne:

$$f_e = f_M = n_e^0\left(\frac{m_e}{2\pi k_B T}\right)^{1/2} e^{-m_e v^2/2k_B T} \quad (6.33)$$

on aura pour tout x : $n_e(x) = n_e^0 e^{\frac{e\phi}{k_B T}}$.

En remplaçant tous ces grandeurs dans le système Vlasov-Poisson, on en tire l'équation de Poisson déterminant le potentiel ϕ; cette équation est notée :

$$\frac{d^2\phi}{dx^2} = \frac{e}{\varepsilon_0}\left[n_e^0 e^{e\phi/k_B T} - n_i(x)\right] \quad (6.34)$$

(même si on néglige le mouvement ionique : on prend un profil stationnaire pour $n_i(x)$.)

En effectuant un changement de variable tel que : $y = \frac{e\phi}{k_B T}$, cette dernière équation devient :

$$\frac{d^2 y}{dx^2} = \frac{e^y - n_i(x)/n_e^0}{\lambda_D^2} \quad (6.35)$$

où λ_D est la longueur de debye définie en 1.1

Si, sur une distance de l'ordre de λ_D, la variation relative de $n_i(x)/n_e^0$ est faible, on a une solution approchée de l'équation de Poisson donnée par :

$$y \approx \ln\left(\frac{n_i(x)}{n_e^0}\right) = \frac{e\phi}{k_B T} \quad (6.36)$$

cette solution correspond à ce que l'on appelle l'approximation de *quasi-neutralité électrique* du plasma. Elle est souvent utilisée dans des codes numériques pour simplifier la résolution de Vlasov-Poisson.

[4] Y. Elskens and D. Escande, *Microscopic dynamics of plasmas and chaos*, IoP Publishing, Bristol (2002).

7.Codes numériques pour l'équation de Vlasov

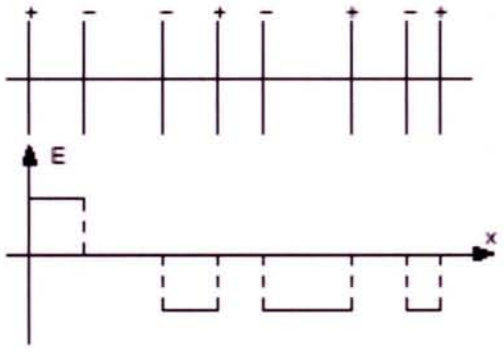
7.1.Introduction

Les codes de plasma ont maintenant une longue histoire puisque les premiers calculs ont été exécutés vers la fin des années 80 employant quelques centaines de macro particules. Très rapidement il s'est avéré que deux genres de problèmes ont dû être considérés :

- physique statistique des particules chargées, y compris la diffusion sur des particules tests.
- Mouvement collectif : c.-à-d. la limite de Vlasov décrite par des systèmes de Vlasov-Poisson ou de Vlasov –Maxwell.

Pour la première classe des problèmes, les ordinateurs disponibles dans les années 60 pouvaient manipuler seulement le plasma de 1D (gaines chargées). Néanmoins les simulations sur ordinateur dans une dimension ont été idéalement conçues afin de vérifier des résultats théoriques. Dans le 1D, le code apporte une solution exacte du problème de N corps excepté les erreurs d'approximation. Entre les gaines, le champ électrique est constant et les gaines ont un mouvement uniformément accéléré. Pour accelerer le code que nous avons calculé seulement les temps croissants, trouver le plus petit et réarranger les gaines **(voir la fig.10)**. Il est très facile manipuler ce code en 1D parce que le calcul de champ est simple, mais il est loin d'être de même en 2D et 3D. Pour un système de 10^6 particules et une machine gigaflops au moins 10^3 secondes sont nécessaires pour un simple pas de temps !

Le schéma 10 : Le code de N-corps de 1D : gaines chargées (ci-dessus) et champ électrique correspondant (ci-dessous).



Mais dans la mesure où les phénomènes collectifs sont notre souci, il vaut la peine de se rappeler que dans un plasma de Vlasov, l'interaction de coulomb entre les particules chargées est remplacée actuellement par un champ moyen calculé à partir des équations de Poisson ou de maxwell en utilisant la densité de charge ou la densité de courant où les fluctuations microscopiques (étant donné qu'un plasma n'est pas un continuum) sont ramenées à une moyenne au-dessus de la longueur de Debye. Ce concept de champ moyen est l'idée fondamentale qui se rapproche du concept de particle in cell ou codes PIC : pendant un pas de temps, la trajectoire de particules est calculée d'un champ prescrit sur une grille fixe avec une maille d'ordre de la longueur de Debye. À la fin du pas de temps, la charge de chaque particule est redistribuée parmi les points voisins de maille laissant résoudre l'équation de Poisson (ou Maxwell) en tout point de la grille. Cette méthode donne des résultats satisfaisants avec un nombre relativement petit de particules. Mais le bruit numérique dû aux différents effets peut être trop important pour obtenir une description précise de la fonction de distribution.

En fait, on a rapidement réalisé qu'une solution numérique de l'équation de Vlasov elle-même était plus appropriée. G. Knorr a donné la première solution par une méthode spectrale (une double transformée de Fourier dans x et v) mais a considéré seulement des problèmes non linéaires près de la théorie linéaire. Le choix de la transformée de Fourier était plutôt évident pour l'espace de configuration : il y a l'avantage que dans une théorie linéaire tous les composants de k sont indépendants. Des problèmes en conséquence légèrement non linéaires ont pu être prévus pour être traités par un nombre restreint de modes de Fourier. Le même choix pour le v-espace était moins évident et un autre traitement a été présenté par Feix et Grant et Armstrong. Les transformé de Fourier ont été gardées sur x, mais des fonctions de Hermite ont été présentées pour le v-espace.

$$f(x,v,t)=\frac{1}{\sqrt{2\pi}}\sum_{n\in\mathbb{Z}}\sum_{s\geq 0}a_{ns}(t)e^{in k_0 x}\bar{H}_s(v)e^{-\frac{v^2}{2}} \quad (7.1)$$

ou

$$\bar{H}_s(v)=(-1)^s e^{v^2/2}\left(\frac{d}{dv}\right)^s e^{-v^2/2}$$
$$\int_{-\infty}^{+\infty}\bar{H}_m(v)\bar{H}_n(v)e^{-v^2/2}dv=\sqrt{2\pi}n!\delta_{mn} \quad (7.2)$$

Néanmoins les développements récents employant ce schéma « time splitting » ont essayé de palier à ce problème de convergence.

Enfin un traitement précis de l'équation de Vlasov-Poisson a été donné par Cheng et Knorr. La méthode sert de base aux codes modernes de Vlasov. Elle présente deux idées :

Time splitting

1. calculer $f^*(x,v) = f^n(x - v\Delta t/2, v)$
2. calculer le champ électrique E^* par substitution de f* dans l'équation de Poisson;
3. calculer $f^{**}(x,v) = f^*(x, v - \frac{eE^*}{2m}\Delta t)$

répéter encore l'étape 1: $f^{n+1}(x,v) = f^{**}(x - v\Delta t/2, v)$

- La reconstruction de **f*, f**** de f et f^{n+1} sur une maille dans l'espace des phases : pour faire cela, des interpolations splines cubiques ont été intensivement employées. Les interpolations de Fourier ont également été développées [39] comme des méthodes de volumes finis (pour une large vue d'ensemble récente sur ces méthodes). Pour le système de Vlasov-Poisson, ce schéma est équivalent à une intégration de l'équation de Vlasov le long des caractéristiques et est correct jusqu'au deuxième ordre.

Revenons aux codes plasmas. Pendant les années 80, les codes PIC effectuaient les simulations 2D pour obtenir l'information utile directement. Mais les solutionneurs directs de Vlasov ont perdu leur intérêt pour la plupart des personnes dus à leur coût numérique élevé. Maintenant (aussi bien dans l'exécution d'unité centrale de traitement comme dans la capacité de la mémoire) les ordinateurs très puissants de vecteurs parallèles sont disponibles et les codes de haute résolution de Vlasov deviennent de plus en plus populaires. Nous voudrions simplement esquisser quelques dispositifs importants qui devraient être pris en considération en décidant d'écrire un code pour un problème spécifique. Le premier concerne le choix entre codes PIC et codes de Vlasov (voir la section 2). Les caractéristiques principales des codes de Vlasov seront esquissées dans la section 3.

7.2. Codes pic vs codes vlasov

Avant de décider quel code devrait être le meilleur pour un problème donné, on doit préciser que la solution des équations de Poisson (ou équations de Maxwell) est la même tâche pour les deux codes (PIC et Vlasov) ; on a besoin de la même grille spatiale, avec une maille Δx de l'ordre de la longueur λD de Debye. Le nombre de points de maille de la grille spatiale pour les deux codes peut être estimé comme $N_x^{dx} = (L / \Delta x)^{dx}$ (pour un plasma avec une longueur typique L, dx étant la dimension spatiale du problème (d_x = 1, 2 ou 3)). D'autre part, les étapes 1 et 3 du schéma de Cheng-Knorr pour le code de Vlasov décrit ci-dessus, ne sont rien d'autre que les équations caractéristiques pour une particule. Les équations de mouvement pour une particule dans un code PIC se résolvent de la même façon. Par conséquent, il est clair qu'en poussant une particule dans un code PIC, vous faites le même effort numérique qu'en reconstruisant un point de la maille de l'espace des phases dans un code eulerien de Vlasov. Par conséquence, le rapport entre l'effort numérique pour un code PIC et un code Vlasov (temps- CPU aussi bien que la condition de mémoire) se mesurera comme N_{vlas} /N_{part} où N_{vlas} est le nombre de tous les points de la maille dans l'espace des phases pour l'équation et N_{part} de Vlasov est le nombre de superparticules dans le code PIC correspondant. Pour un code de Vlasov, N_{vlas} peut être écrit simplement comme le produit de la grille spatiale par la grille de vitesse

$$N_{vlas} = \left(\frac{L}{\Delta x}\right)^{d_x} N_v^{d_v} \quad (7.3)$$

où d_v est la dimension de l'espace de vitesse (d_v = 1, 2 ou 3).

Habituellement $d_x = d_v$, mais dans certains cas, particulièrement quand les caractéristiques de l'équation de Vlasov possèdent un invariable exact, les deux nombres peuvent être différents. Pour un code de PIC avec une densité de superparticles N_0, N_{part} peut être écrit :

$$N_{part} = N_0 L^{d_x} = N_0 (\Delta x)^{d_x} \left(\tfrac{L}{\Delta x}\right)^{d_x} \quad (7.4)$$

N_x^{dx} peut être facilement identifié et nous voyons également l'aspect de $n_0 \Delta x dx$ qui n'est rien d'autre que l'inverse du paramètre de granulation dû aux particules discrétisées du code PIC.

$$g_{pic} = \frac{1}{n_0 \lambda_D^2} \quad (7.5)$$

Cette relation mérite quelques commentaires. Un code de PIC montre une situation paradoxale où l'introduction d'une grille spatiale $\Delta x \sim \lambda D$ nous permet de décrire les effets collectifs par l'approximation moyenne de champ (voyez au-dessus de la section 1) mais au prix de la réintroduction de différents effets dus au nombre fini de superparticles. Ces différents effets sont souvent considérés car un bruit numérique dont la mesure est clairement gpic comme indiquée au (7.5). De ce fait, garder la concordance du modèle de superparticule entraîne que gpic doit être aussi petit que possible. N_{part} peut être maintenant écrit.

$$N_{part} = g_{pic}^{-1}\left(\frac{L}{\Delta x}\right)^{d_x} \quad (7.6)$$

7.3. Time-splitting et schéma lagrangien

Maintenant, nous voudrions esquisser les caractéristiques principales de la situation actuelle. Commençons par le schéma bien connu de Cheng-Knorr décrit ci-dessus dans l'introduction de ce chapitre.

Dans le cas simple considéré ci-dessus, l'advection sur v (respectivement E(x,t)) ne dépend pas de la variable le long de laquelle les décalages sont exécutés sur x (respectivement sur v), et le calcul de $f^*(x,v) = f^n(x - v\Delta t/2, v)$ *ou* $f^{**}(x,v) = f^*(x, v - \frac{qE^*}{m}\Delta t)$ est simple. Cette méthode s'est avérée fonctionner très bien dans ce cas électrostatique. L'avantage est que ce schéma peut s'employer sur de plus grands pas de temps que le schéma eulerien et que le bruit est moindre par rapport aux codes de Particle-In-Cell, mais le prix à payer est de reconstruire une grille régulière en utilisant des interpolations. Cependant ce schéma ne pourrait pas être appliqué en tant que tel à un problème plus compliqué de Vlasov, par exemple à l'équation gyro-cinétique de Vlasov ou à l'équation relativiste de Vlasov dans laquelle le facteur de Lorentz est couplé avec les différents composants de p. Dans ce cas, nous devons considérer un problème comme $\frac{\partial f}{\partial t} + v(x,t)\frac{\partial f}{\partial x} = 0$ (7.7)

où la limite d'advection dépend de la variable pour être advecté (considérez un pas de temps Δt). Il a été montré comment résoudre ce problème en présentant le schéma des caractéristiques arrières, où $f^*(x,v,t_n + \Delta t) = f(x - v\Delta t, v, t_n)$ est remplacé par :

$f^*(x,v,t_n + \Delta t) = f(x - \alpha, v, t_n)$ (7.8)

x -α est le point de départ de la fin caractéristique à x et peut être obtenu à l'exactitude du second degré en résolvant l'équation implicite : $\alpha = \Delta t.v\left(x - \frac{\alpha}{2}, t_{n+\frac{1}{2}}\right)$(7.9)

Dans le cas simple où le champ d'advection v est une constante en ce qui concerne la variable à être advecté, comme dans le cas de Vlasov-Poisson de 1D, la solution de (7.9) est simple et nous récupérons $f^*(x,v,t_n + \Delta t) = f(x - v\Delta t, v, t_n)$.

Cette méthode semi-lagrangienne ou méthode caractéristique arrière est du deuxième ordre précis dans l'étape de temps. Maintenant revenons au time splitting et considérons l'équation de Vlasov sous la forme suivante :

$$\frac{\partial f}{\partial t} + U(X,t).\nabla_x f = 0 \quad (7.10)$$

où les valeurs de X, qui sont les coordonnées pour l'espace de phase, et U, qui est une divergence libre, dirigent le champ ayant jusqu'à six composants dans le volume tridimensionnel. Par exemple, dans le cas de l'équation électrostatique de 3D Vlasov, nous avons X = (x, y, z, v_x, v_y, v_z) et U = (v_x, v_y, v_z, E_x, E_y, E_z), tous les composants du champ électrique dépendent de x, y, z et t.

Pour un champ U d'advection qui est divergent, l'équation de Vlasov, Eq. (7.10) peut également être écrit sous la forme conservatrice

$$\frac{\partial f}{\partial t} + \nabla_x \left[U(X,t) f\right] = 0 \quad (7.11)$$

Si on sépare les composants de X en deux X1 et X2, Eq. (7.10) peut alors être écrit sous la forme :

$$\frac{\partial f}{\partial t} + \nabla_{x1} \left[U_1(X_1,X_2,t) f\right] + \nabla_{x2} \left[U_1(X_1,X_2,t) f\right] = 0 \quad (7.12)$$

D'ailleurs, il est bien connu que cela se résout séparément :

$$\frac{\partial f}{\partial t} + \nabla_{x1} \left[U_1(X_1,X_2,t) f\right] = 0 \quad (7.13)$$

$$\frac{\partial f}{\partial t} + \nabla_{x2} \left[U_1(X_1,X_2,t) f\right] = 0 \quad (7.14)$$

Cela maintient l'exactitude du second degré pour l'équation entière (7.12) qui se résout de manière alternative. Il est maintenant important de préciser que le schéma semi-lagrangien ne résout pas l'équation de Vlasov sous la forme conservatrice, mais sous la forme advective pour faire la pleine utilisation de la méthode caractéristique en arrière. Par conséquent si (et seulement si) les deux conditions se tiennent

$$\nabla_{x_1}.U_1(X_1,X_2,t)=0 \quad (7.15)$$

$$\nabla_{x_2}.U_2(X_1,X_2,t)=0 \quad (7.16)$$

Puis les équations (7.13) et (7.14) peuvent être mis sous la forme advective

$$\frac{\partial f}{\partial t}+U_1.\nabla_{x_1} f=0 \quad (7.17)$$

$$\frac{\partial f}{\partial t}+U_2.\nabla_{x_2} f=0 \quad (7.18)$$

Pour garder l'exactitude du second degré pour l'équation entière (7.10), on résout ces 2 équations en alternance. C'est la base du schéma time splitting accomplissant les conditions (7.15) et (7.16). Les deux méthodes sont actuellement employées.

Comme déjà mentionné, une explication au fait que peu d'effort a été dépensé pour les codes Vlasov pendant les années 80 est liée à la taille limitée des ordinateurs disponibles et jusqu'à très récemment à ces codes qui pouvaient seulement traiter l'espace des phases en 2D.

Ceci achève le traitement de l'équation de Vlasov d'un point de vue purement théorique.
Nous avons vu que la résolution de l'équation de Vlasov nécessite d'importants calculs. C'est pourquoi nous allons étudier, dans la seconde partie, les moyens informatiques que nous avons à disposition pour effectuer les calculs dans les meilleures conditions possibles…

PARTIE II :

INFORMATIQUE

1. Conception de programmes parallèles

Les langages de programmation ont permis aux programmeurs de développer des calculs de manière plus simple que les langages trop proches du modèle d'exécution ; ainsi, les langages de programmation offrent des niveaux d'abstraction de plus en plus élevés, contribuant à une tâche de programmation plus facile et plus fiable. On distingue donc bien les modèles d'exécution (modèle Von Neuman, réseaux d'ordinateurs, machines parallèles à mémoire partagée, ...) et les modèles de programmation comme la programmation impérative, la programmation fonctionnelle, la programmation logique, la programmation objet, la programmation à base de composants, la programmation répartie, la programmation parallèle, la programmation par coordination, etc. Ces philosophies de programmation reposent sur des concepts très simples et sur des techniques logico-mathématiques qui commencent à être plus connues ; ces techniques permettent aussi d'instiller des aspects liés à la sûreté du code développé ou encore à sa qualité. La démarche repose sur une approche parfois appelée la démarche par contrat par B. Meyer [eiffel] ou encore une démarche s'appuyant sur une analyse précise des besoins exprimés sous la forme de pré et de post conditions : l'objectif est de concilier les exigences initiales et le produit final obtenu: on parle donc de contrat. Au-delà de ce contrat, la question est aussi d'expérimenter sur des structures de machines les programmes développés et de les mettre au point ; il ne s'agit pas seulement de produire des programmes corrects mais aussi de produire des programmes efficaces. Le parallélisme est une solution possible pour obtenir des codes efficaces et une démarche progressive basée sur l'établissement clair de contraintes ou d'exigences contractuelles permet

de s'assurer la qualité fonctionnelle du code par rapport au problème à résoudre. Dans ce travail, nous visons plusieurs objectifs liés à la programmation et aux calculs associés à ces équations (équations cinétiques, équation de Vlasov...):

- Une analyse du problème posé et une résolution avec des outils informatiques mettant en œuvre le parallélisme.
- La mise en évidence de modules ou de briques de base en vue de réutiliser celles-ci dans des développements futurs et de favoriser une approche par composants.

Nous allons tout d'abord commenter quelques approches utilisées pour le développement de programmes parallèles. Notre commentaire ne sera pas exhaustif mais mettra en avant les points positifs que l'on peut souhaiter retenir. Notre souci est de programmer un calcul et de le réaliser dans des conditions les plus acceptables pour un physicien ; le modèle d'exécution sous-jacent restera accessoire dans la mesure où un critère d'efficacité est visé mais sans donner d'architecture explicitement.

1.1 Langages de programmation parallèle

Les langages de programmation parallèle sont apparus à la suite d'une évolution des besoins liés aux problèmes à résoudre ; la multiprogrammation a introduit la notion de tâches qui partagent un même processeur. Très naturellement, l'apparition de machines multiprocesseurs a conduit à se poser des questions sur les langages de programmation ad hoc. Ainsi, les langages comme ADA, CSP, OCCAM etc ont permis aux utilisateurs de se poser des questions nouvelles sur les possibilités de programmer dans un nouveau style ; il est à noter que ces langages ont permis de mettre en œuvre des techniques de programmation intégrant la communication entre processus soit via des variables partagées soit via des canaux de communication. L'exemple d'OCCAM est très intéressant, puisque ce langage est devenu un assembleur pour ce type de programmation avec les processeurs appelés transputers. Cette approche orientée vers les langages a conduit à des langages qui étaient conçus pour prendre en compte le parallélisme ou bien des langages qui ont été étendus par des primitives de communication. L'idée principale d'un langage de programmation est de fournir des abstractions aussi puissantes que possible pour prendre en compte les capacités d'une machine donnée. Les techniques de parallélisation fondées sur une

analyse des dépendances permettent d'affranchir l'utilisateur des aspects trop techniques de telle ou telle autre machine ou architecture. Les compilateurs FORTRAN offrent des possibilités de parallélisation automatique et permettent d'optimiser quelque peu les codes développés qui doivent être effectivement utilisés.

1.2 Architectures parallèles

Les architectures parallèles reposaient initialement sur une classification proposée par Flynn (1972) sous les quatre dénominations acronymiques suivantes SISD ((Single Instruction Stream, Single Data Stream) ordinateur sequentiel standard), SIMD ((Single Instruction Stream, Multiple Data Stream) Les N processeurs identiques opèrent sous contrôle d'un flot d'instructions unique généré par une unité centrale unique (pour faciliter la compréhension, on peut supposer que tous les processeurs exécutent le même programme).), MISD ((Multiple Instruction Stream, Single Data Stream) N processeurs, chacun avec sa propre unité de contrôle, se partagent une mémoire commune.), MIMD (Multiple Instruction Stream, Multiple Data Stream), C'est la classe la plus générale et la plus puissante de toute cette classification. Il y a N processeurs, N flots d'instructions et N flots de données. Et on peut aussi avoir fonctionnellement les deux schemas suivants (mémoire partagée ou distribuée)). Les architectures de type SIMD se déclinent en plusieurs types comme SIMD vectoriel (multiprocesseurs vectoriels) ou SIMD parallèle (CM-1) ; ce type de machine induit un style de programmation ou encore un modèle de programmation. Ce type d'architecture est en voie de disparition.

Un second type d'architecture de machines appelées MIMD organise des processeurs et des flux de calculs via un réseau d'interconnexion, mais est difficiles à programmer, puisqu'il faut organiser les calculs effectués par des processeurs distincts mais ne réalisant pas la même tâche. Une première classe MIMD comprend les machines à mémoire distribuée ou à passage de mémoire comme un ensemble de machines en réseau ou encore à mémoire partagée comme SGI Origine 2000.

La question importante reste celle de la programmation d'applications dans ces architectures et les langages de programmation constituent une réponse partielle car il est important de

souligner l'importance du modèle choisi. En effet, un modèle de programmation apporte un niveau d'abstraction favorisant un code plus clair, plus correct et en adéquation avec les besoins. Il est assez clair que le programmeur doit être libéré des détails trop techniques du fonctionnement de la machine parallèle et le modèle de programmation fournit cette possibilité.

1.3 Modèles de programmation parallèle

Un modèle de programmation est lié à l'expression de l'algorithme et permet de donner explicitement les entités communiquant entre elles ; il a un lien avec le modèle d'exécution qui est lié à l'architecture parallèle ou distribuée. Les langages de programmation mettent en œuvre des modèles de programmation, mais il y a aussi des modèles de programmation qui permettent d'étendre un langage de programmation donné qui a été conçu comme un langage de programmation séquentielle ; ainsi, le modèle de programmation basé sur la coordination [linda] LINDA a proposé un cadre d'extension de tout langage de programmation en un cadre de programmation parallèle ou répartie. LINDA n'a pas connu de succès en programmation pratique, mais a permis de dresser les jalons d'un thème très riche en génie logiciel notamment la coordination, la coopération. Rappelons brièvement l'idée centrale de la coordination. Un problème est résolu par la mise en évidence de tâches qui communiquent via un espace de données appelées tuples; les tâches sont programmées dans un langage de programmation séquentielle et l'on ajoute des primitives de communication pour l'espace des données. Galibert [galibert] a implanté sur les machines du CCH une version LINDA pour C++ en fondant son travail sur les objets et en répartissant l'espace des tuples. Plus tard, les approches comme PVM, MPI se sont imposées et Foster a mis en œuvre une méthodologie de conception fondée sur une décomposition en tâches qui sont ensuite organisées. La question difficile reste cette décomposition. En fait, Ian Foster dans "Designing and Building Parallel Programs", traite des extensions parallèles des langages de programmation, parallélisme de contrôle et parallélisme de données, des principes de la programmation data-parallèle : virtualisation et alignement des données, opérations globales, diffusions et réductions, communications régulières.

1.4 Méthodologie de programmation parallèle

Le développement de programmes ou de systèmes parallèles est une activité qui pose des questions sur la qualité des codes produits et sur la portabilité de ces mêmes codes. La qualité des codes produits repose sur l'utilisation de techniques de preuve ou des techniques de validation des programmes en fonction de leurs spécifications. Les recherches menées pour produire ou construire des programmes parallèles corrects de manière aussi indépendante de l'architecture visée sont très nombreuses et on pourra citer les travaux de Chandy et Misra sur UNITY [chandy88] mais aussi les travaux mettant en avant des modèles de programmation comme le modèle de coordination introduit par Gelertner [gelertner]. Les modèles comme PVM ou MPI permettent de combiner de manière transparente des entités de calcul et qui permettent de réaliser des calculs sur des infrastructures hétérogènes, mais il n'y a pas de méthodologie réelle. Tous ces travaux ont concerné en fait les modèles de programmation parallèle qui ont ensuite été mis en œuvre sur des modèles d'exécution mais une question fondamentale est la production de codes réutilisables dans des cas architecturaux divers et leur maintenance. Ainsi, comme les design patterns constituent un moyen d'architecturer les systèmes informatiques, le développement de patrons pour la programmation parallèle dédiée à certains types de problèmes semble être une voie intéressant les utilisateurs de ces domaines dédiés. Beverly Sanders et son équipe a développé une étude de tels patterns et a coécrit un livre [0] sur ce point. Ce livre aide au développement de logiciels parallèles ; ainsi, tandis que la demande du logiciel parallèle est grande et croissante. Le principe est de développer un langage commun qui exécuterait des applications parallèles. Il convient aussi de citer le modèle CSP qui a permis aussi de développer une méthodologie mettant en avant le concept d'échange de messages. Enfin, les techniques de raffinement de systèmes d'actions introduits par Back constituent une perspective très intéressante pour développer de manière progressive des systèmes d'action qui sont ensuite compilées sur des architectures données.

1.5 Synthèse

Les sections précédentes ont brossé un panorama rapide des langages, des architectures et des modèles pour la programmation parallèle. Le travail de nature informatique visait à identifier des codes reproductibles dans les différents problèmes posés dans le contexte de cette physique. L'idée est de chercher à documenter les codes développés et à permettre une réutilisation de ces codes aussi simplement que possible. Le problème physique étudié conduit a développer des programmes informatiques pour l'équation de Vlassov. La démarche suivie consiste à analyser les codes et à rechercher des éléments ou des briques qui peuvent êtres validés en tant que tel.

2. Etude de l'équation de Vlasov

Dans ce chapitre, nous allons étudier l'équation de Vlasov d'un point de vue informatique. Le but étant de calculer cette équation le plus rapidement possible, on introduira la programmation parallèle. La comparaison entre deux méthodes de parallélisation (MPI et OpenMP) sera étudiée de façon à établir une performance optimale. Avant cela, nous tracerons les lignes générales de chaque langage de parallélisme et nous donnerons une idée de leurs performances. Ces langages sont MPI (Message Passing Interface) et OpenMP.

2.1 Spécification du code et décomposition modulaire

Nous étudions dans un premier temps la méthode numérique, appelée « time splitting », qui est particulièrement adaptée à la résolution de cette équation : ainsi nous travaillons d'une manière itérative, à un instant t donné, dans l'espace des phases à deux dimensions (x, v_x). Le grand nombre de calculs nécessite l'introduction des méthodes de programmation en parallèle qui permettent d'obtenir un gain important sur le temps de calcul.

Dans un deuxième temps, nous étudions deux méthodes de parallelisation, MPI et OpenMP, en soulignant les avantages et les désavantages de chacune, ainsi que la façon de les

appliquer à notre problème. En effet, le programme de calcul proposé utilise plusieurs sous routines en FORTRAN 90 et C++ ; nous attachons une attention particulière à l'implémentation du système parallèle (MPI ou OpenMP) sur ces deux langages informatiques, en assurant en même temps leurs interconnexions.

À présent, on doit résoudre un système d'équations qui demande beaucoup de temps de calculs. Nous rappelons que ce système est de la forme

$$\begin{cases} \left[\partial_t + p\cdot\partial_x + \left(X'\sin(x)+Y'\cos(x)\right)\varepsilon'\beta\cdot\partial_p\right]f = 0 \\ X' = Y' + \varepsilon'\beta\cdot\iint f(x,p)siin(x)\cdot dxdp \\ Y' = X' + \varepsilon'\beta\cdot\iint f(x,p)\cos(x)\cdot dxdp \end{cases}$$

Pour intégrer ce système, on doit faire appel à l'algorithme numérique de Shanon, appelé time splitting. On discrétise les valeurs de X' et Y' pour chaque pas de temps Δt tels que

$$X^* = \Delta t\cdot X'\cdot\sum_n \delta\left(t - t_{n+\frac{1}{2}}\right)$$

$$Y^* = \Delta t\cdot Y'\cdot\sum_n \delta\left(t - t_{n+\frac{1}{2}}\right)$$

On peut voir que la première équation se décompose en deux équations plus simples suivant la période.

$$\frac{\partial f}{\partial t} + p\cdot\frac{\partial f}{\partial x} = 0$$

$$\frac{\partial f}{\partial t} + \left(X^*\sin(x)+Y^*\cos(x)\right)\cdot\varepsilon'\beta\cdot\frac{\partial f}{\partial p} = 0$$

Donc le programme effectue sur un intervalle Δt :

Calcul de $\varepsilon'\beta\cdot\iint f(x,p,t_n)\cdot e^{ikx}dxdp$ (programme M_density)

Effectue $\frac{\partial f}{\partial t} + p\cdot\frac{\partial f}{\partial x} = 0$ pendant $\frac{\Delta t}{2}$

Implémentation du nouveau repère (X', Y')

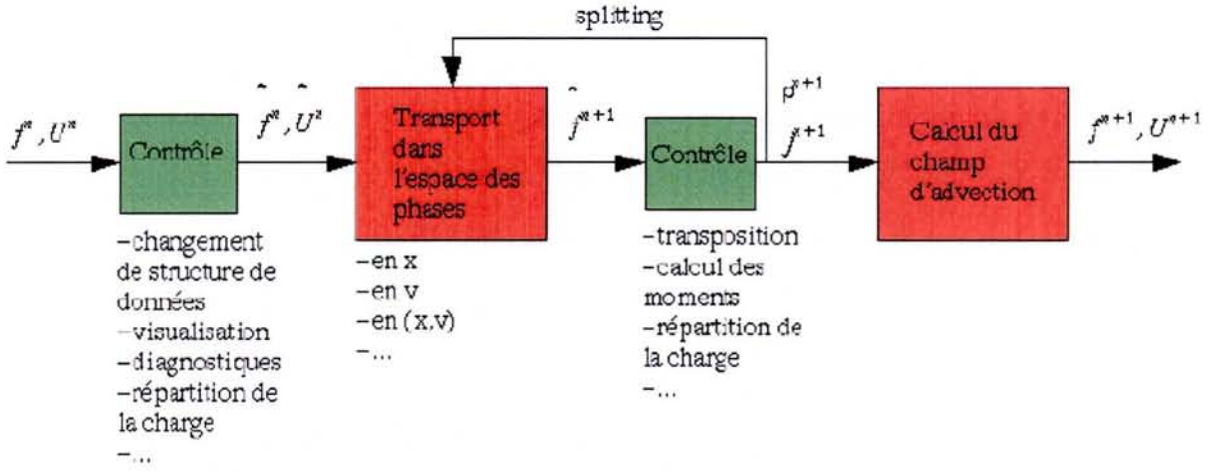
Avec ces nouvelles coordonnées, on réapplique la seconde équation sur le résultat précédent

$$f(x,p,t_+) = f(x, p-\varepsilon'\beta\cdot\Delta t\cdot(X'\sin(x)+Y'\cos(x)),t_-)$$

Et de nouveau, on effectue l'équation $\frac{\partial f}{\partial t} + p\cdot\frac{\partial f}{\partial x} = 0$ pendant $\frac{\Delta t}{2}$.

Cette méthode permet d'intégrer f le long des séparatrices, elle est appelée méthode semi-lagrangienne.

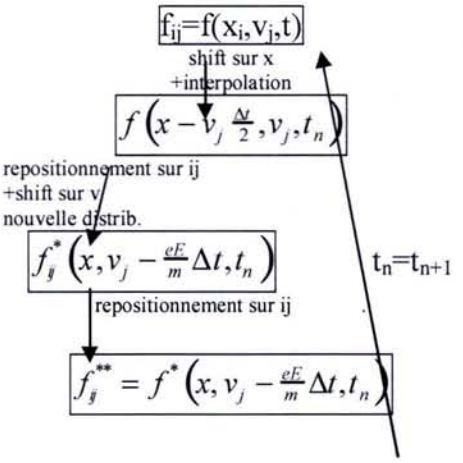
Schéma récapitulatif :



Ce schéma illustre ce que le programme va effectuer. Ce code contient un programme principal et des sous-broutines effectuant essentiellement des transpositions de matrices et des changements de repères.

Schéma de résolution :

Programme vlasov(pour les calculs, on a pris m=1 donc p=mv=v)



Ce code se décompose en modules inter dépendants. Chaque module effectue une tâche simple dans un pas de temps. Voici le tableau des opérations des modules.

M_DENSITY	Calcule la densité puis le champ électrique
M_SHIFT	Décalage de vdt/2 sur x
M_ACC	Calcule de fftrans pour V+DV
M_INICON	Initialisation des variables pour t=0
M_PREP	Calcule des champs
M_transF	Transposition de la matrice F => FFTRANS

Ce code est modulable selon la fonction de distribution, la période…

Pour réaliser ce projet, il suffit de changer les données de M_PREP et M_INICON…

2.2 Analyse du code

Nous avons vu dans les sections précédentes que ce code repose sur une suite d'advections toujours identiques. Il est donc clair que ce code est fondé sur des calculs itératifs, nous allons utiliser des techniques de parallélisation, afin de prendre en compte cet aspect-là.

3. Parallélisation

Avant de paralléliser les codes, il faut les produire et surtout définir clairement leurs spécifications. L'idée est de dégager ce qui peut être réutilisé ou non ultérieurement. Ian Foster, dans son livre *Designing and Building Parallel Programs* considère la programmation parallèle comme une activité de technologie, dans laquelle des programmes sont développés d'une manière méthodique. La performance et l'exécution sont considérées à chaque étape comme étant une conception à part entière. Nous allons nous reposer sur cette idée pour les programmes liés a Vlasov.

3.1 MPI

Ce programme a été implémenté pour MPI. On va s'intéresser à cette méthode de parallélisation. Ce langage de parallélisation a été étudié grâce au livre Using MPI, 2nd Edition. Tout d'abord, MPI est une bibliothèque utilisable par FORTRAN ou C++, qui est basée sur l'envoi et la réception de messages (MPI_send et MPI_receive). De la mémoire tampon est alors répartie entre les processeurs grâce à un processeur maître qui régit aussi la réception de leurs résultats. Cette notion d'échange de données entre les processeurs est appelée communication point à point.

Chaque processeur peut exécuter une partie du programme. Un message entre les processeurs maître esclaves permet de transférer des données. MPI est autonome, c'est-à-dire que les processus exécutent un bout du programme qui est ensuite transmis, à l'aide de sous-programme, au processeur maître. Les données échangées sont, en général, des nombres réels qui sont fixés par une étiquette (tag). On peut faire une série de communications point à point en une seule instruction : c'est la communication collective. On peut l'utiliser de trois manières différentes :

- Synchronisation globale (MPI_barrier)
- Transfert de données (MPI_bcast, MPI_scatter, MPI_gather, MPI_allgather, MPI_alltoall)
- Gestion des communications (MPI_reduce)

L'intérêt à mettre MPI_barrier réside parfois dans la nécessité d'avoir tous les résultats pour pouvoir continuer le programme. Le but de MPI est l'optimisation (éviter de recopier les

buffers et limiter les appels aux sous-programmes). En général, on perd 75 % du temps total pour les communications.

L'envoi des données nécessite une mémoire suffisamment importante pour être capable de restituer les informations. Pour être sûr qu'un envoi a bien été reçu, on peut mettre la fonction *synchroneous* qui est couplée avec la réception. Parfois, on utilise un envoi synchrone bloquant pour la copie directe de mémoire à mémoire. Le principal problème de tous les langages parallèles est de synchroniser les processeurs dans un ordre naturel de calculs,(le premier processeur à calculer sera le premier à envoyer son résultat). Dans ce dessein, il existe des commandes permettant de vérifier l'avancée des calculs de chacun : MPI_TEST() contrôlant si la progression est achevée. Les données échangées sont cataloguées par types par MPI : il faut définir le type de toutes les variables avant de les utiliser. Par défaut, MPI les met en réelles, mais toutes les formes sont possibles (entier, complexe, vecteur). MPI permet aussi de créer des sous-ensembles à l'intérieur desquels on peut faire une communication point à point : cela s'appelle les communicateurs (MPI_COMM_WORLD). Cette entité est, par définition, entière.

Un message est divisé en deux :

- Des données (adresse, éléments, types)
- Des enveloppes (processus, étiquettes, communicateurs)

3.2 OpenMP

À l'inverse, OpenMP est un langage de parallèlisation implicite. Il a pour but aussi de partager le travail entre les processeurs. En général, son mode de programmation s'approche d'un code séquentiel, il se décompose en trois directives principales :

- Partage du travail (work-sharing)
- Partage des données
- Synchronisation

Ce code est portable, c'est-à-dire que, si le compilateur n'a pas une bibliothèque openMP, il fera comme si les directives n'existaient pas, et exécutera le programme normalement. La programmation est basée sur le modèle *fork and join* ; cela signifie que

1. Les programmes sont séquentiels

2. Seul le processeur maître exécute la partie séquentielle
3. Des tâches exécutent la partie parallèle et transmettent leurs résultats au processeur maître.

À la fin d'une région parallèle, toutes les tâches se synchronisent avant que le processeur maître continue, tout seul, la suite du programme. On ne peut garantir que le nombre de processeurs soit identique à celui des tâches. Toutes les tâches sont numérotées, et l'on peut savoir quelles sont les tâches qui travaillent avec l'instruction OMP_GET_THREAD_NUM(). Une région parallèle doit commencer par **!SOMP PARALLEL** et finir par **!SOMP END PARALLEL**.

La mémoire d'openMP est partagée en cinq :

- La zone U sert à faire des appels systèmes
- La zone TXT elle garde le code exécutable en mémoire
- La zone DATA stocke les valeurs numériques
- La zone BSS contient les valeurs numériques au fur et à mesure que le programme calcule.
- La pile et le tas sont des mémoires dynamiques.

Les variables peuvent êtres privées ou partagées. On peut forcer le compilateur à travailler avec un seul processeur (processeur maître) pour exécuter une instruction du programme : cette fonction est appliquée grâce à la directive MASTER. Pour paralléliser une instruction, il faut savoir si les variables sont privées ou partagées. Si la variable est privée, tous les processeurs l'ont en mémoire et peuvent la manipuler comme ils le souhaitent. Si, par contre, elle est partagée, les processeurs peuvent seulement la lire, sans faire de changement sur elle. Par défaut, toutes les variables sont partagées. Il existe un dernier type de variables appelé réduction, il est utilisé afin de faire des opérations simples sur cette variable. Chaque processeur effectue une opération différente dans la section parallèle (boucle).

3.3 PVM

À l'aide du manuel : PVM: Parallel Virtual Machine : A Users' Guide and Tutorial for Networked Parallel Computing, on va donner une vision non exhaustive de PVM…

Ce système est conçu d'une manière inter opératrice entre plusieurs architectures différentes. Cette hétérogénéité des environnements de travail introduit le concept d'une super machine virtuelle. À l'intérieur de cette machine virtuelle, le travail est partagé entre plusieurs processeurs grâce à des programmes de connexions, appelés deamons, qui sont l'intermédiaire entre la mémoire virtuelle et les processeurs ; ces programmes deamons sont rassemblés dans une bibliothèque appelée pvmd3 ou tout simplement pvmd. Pour assurer la coopération entre les différentes tâches et applications exécutables, le système PVM comporte aussi une bibliothèque de sous-routines citables par des programmes principaux en FORTRAN (PVMF...) ou C (PVM_...). Les possibilités d'interconnexion entre architectures différentes diminuent cependant les performances de communication au niveau des multiprocesseurs, par rapport à l'autre système existant MPI.

Pour exécuter n'importe quel programme Fortran ou C en PVM, on lance d'abord les programmes deamons en appelant leur bibliothèque pvmd3 à l'aide de l'instruction homonymique: pvmd3. Il est intéressant de préciser que l'on doit toujours faire appel à la commande kill pour détruire ces programmes deamons, car si on lance seulement la commande quit ces programmes tournent toujours. D'une certaine manière, les deamons jouent le rôle de transporteurs d'information ; ils envoient des tâches à effectuer aux différents processeurs, qui orientent leurs résultats obtenus vers un processeur maître, d'où la notion de maitre-esclaves.

Récapitulatif des instructions à effectuer pour un programme avec PVM :

1) pvmd3 [=>lancement des damons]
2) nom du programme
3)Kill [=>destruction du deamon]

4. Perspectives d'une bibliothèque pour la réutilisation du code

Dans beaucoup de laboratoires tournés vers les applications et non pas vers l'informatique, le pourcentage du temps des chercheurs passé à développer et à mettre au point des programmes est considérable. Il est souvent proche de 75% au détriment du temps d'utilisation effectif. De plus, le taux de réutilisation des logiciels est très faible.

Pour ce code de Vlasov, l'idée serait de séparer les modules en les rendant exécutables pour d'autres fonctions et d'autres caractéristiques (prise en compte des champs magnétiques, modification du champ électrostatique...).

La simulation numérique joue un rôle fondamental dans le développement et la fabrication de composants industriels. Mais souvent la complexité de ces composants nécessite des codes à facettes multiples (avec beaucoup de boucles) et tirant pleinement avantage des moyens de calcul disponibles.

La mise en œuvre pratique et la maintenance de codes de calcul en physique des plasmas est un travail difficile qui occupe des équipes entières en physique numérique. Pour développer des codes de calcul de recherche 2D et 3D performants, l'accent doit être mis sur les points suivants : algorithmique parallèle, programmation orientée objet et programmation modulaire.

La taille des problèmes considérés dans la simulation numérique de plasmas, doit nous encourager à tirer profit au maximum des nouvelles architectures: l'émergence de mémoires hiérarchiques va inciter à développer des algorithmes à plusieurs niveaux de parallélisme. De plus, l'utilisation de routines parallèles utilisables en ``boîte noire" (telle que FFT, raffinement de maillage,...) peut s'avérer très utile. Plus généralement, pour pouvoir augmenter le niveau de complexité des problèmes traités, il est indispensable de réutiliser le plus possible des bibliothèques déjà écrites, que ce soit en Fortran77, Fortran90, C++, et de développer des bibliothèques réutilisables. L'encapsulation de morceaux de codes existant dans des classes abstraites de haut niveau doit être une priorité dans cette optique. De même, l'indépendance des codes de la machine utilisée est fondamentale pour des logiciels de recherche tels que ceux dont nous poursuivons le développement.

Pour pouvoir simuler de manière exhaustive un composant industriel, il est en général indispensable de faire appel à des modèles mathématiques différents, pour différentes parties

du composant ou pour décrire une évolution en temps. Ceci nécessite donc le couplage de codes différents. La tendance actuelle pour réaliser ce couplage de manière efficace est d'utiliser des langages interprétés tels que Perl, Python ou Tcl pour lier les routines de calcul intensif qui sont écrites dans un ou plusieurs langages compilés plus performants au niveau du coût de calcul, ou/et d'utiliser CORBA pour gérer la communication entre les composants logiciels. L'utilisation d'un langage interprété permet également de développer rapidement des diagnostics sur mesure pour le problème simulé ou une intervention en cours de calcul et dispose en général de bibliothèques graphiques qui peuvent être utilisées pour la visualisation interactive ou en post-traitement. Il est également possible, au même titre que les modules de calcul, de développer des bibliothèques graphiques sur mesure qui pourront être couplées facilement aux modules de calcul à l'intérieur de l'interpréteur.

Ce schéma numérique en modules pourrait sans nul doute se plier à d'autres spécificités physiques. Il est donc possible d'imaginer et de réaliser une bibliothèque dans laquelle se trouveraient des modules déjà opérationnels (par exemple un module pour transposer une matrice F, ou un module calculant le champ E(x,t)). L'utilisateur n'aurait plus qu'à assembler ces modules et rentrer ses caractéristiques liées à son équation…

Concernant les résultats avec MPI et OpenMP (annexes A et B), nous pouvons dire que la tache la plus ardue est la synchronisation des processus qui envoient leurs résultats dans un ordre aléatoire. Le processeur maître doit alors allouer de la mémoire pour retenir ces informations afin de les placer dans un ordre correct. Quant à la vitesse d'exécution de ces 2 programmes (en MPI et OpenMP), elle est subjective car ce code contient essentiellement des boucles de calcul…

CONCLUSION

Le but de ce travail était d'étudier et tenter de modéliser les phénomènes collectifs dans un plasma chaud avec comme point de mire la résolution de l'équation de Vlasov à une dimension en l'absence de champ magnétique.

Nous avons appliqué, à ces codes, des techniques parallèles afin d'accroître la vitesse d'exécution en nous servant, notamment, des nombreuses itérations jalonnant le programme. Il est clair que la synchronisation des processus etait un de nos soucis le plus important et délicat : la mise en place de « barrières » ne donnant pas entières satisfactions.

Comme nous l'avons suggéré dans la partie informatique, il serait envisageable de découper ce programme en une suite de petits modules indépendants (densité, transposée, transformée de Fourrier...) : l'utilisateur n'aurait plus qu'à assembler les modules afin de créer un programme résolvant l'équation de Vlasov avec les conditions souhaitées...

REFERENCES

[1] P. Bertrand, A. Ghizzo, S.J. Karttunen, T.J.H. Pattikangas, R.R.E. Salomaa, M. Shoucri, "Two-stage electron acceleration by simultaneous stimuled Raman backward and forward scattering", Phys. Plasmas 2 (8), p.3115-3129 (1995)

[2] A. Ghizzo, P. Bertrand, M.L. Begue, T.W. Johnston, M. Shoucri, "A Hilbert-Vlasov code for the study of high-frequency plasma beatwave accelerator", IEEE Transactions on Plasma Science, 24 (2), p. 370-378 (1996)

[3] E.Sonnendrucker, J.Roche, P.Bertrand, A.Ghizzo, "The semi-lagrangian method for the numerical resolution of Vlasov equations", journal of Computational Phisics 149, p.201 (1999)

[4] E. Esarey, P. Sprangle, J. Krall, A. Ting, "Overview of plasma-based accelerator concepts", IEEE Transactions on Plasma Science, 24 (2), p. 252-288 (1996)

[5] M. Tabak, J. Hammer, M. E. Glinsky, W.L. Kruer, S.C. Wilks, J. Woodworth, E. M. Campbell, M.D. Perry, R. J. Mason, "Ignition and high gain with ultrapowerful lasers", Phys. Plasmas 1 (5), p. 1626-1634 (1994)

[6] C.Z. Cheng, G. Knorr, "The integration of the Vlasov equation in configuration space", Journal of Computational Physics 22, p.330-351 (1976)

[7] C.K. Birsdall, A.B. Langdon, "Plasma physics via computers simulations", Plasma Physics Series. Editor professor E.W. Laing, Institute of Physics, publishing Bristol, Philadelphia (1995)

[8] E. Sonnendrucker, J. Roche, P. Bertrand, A. Ghizzo, "The semi-lagrangian method for the numerical resolution of Vlasov equations", Journal of Computational Physics 149, p.201 (1999)

[9] N. Rosteker, M.N. Rosenbluth, "Test particles in a completely ionised plasma", Phys. Fluids 3 (1), p.1-14(1960)

[10] F. Huot, A. Ghizzo, P. Bertrand, E. Sonnendrucker, O. Coulaud, "Study of propagation of ultraintense electromagnetic wave through plasma using semi-lagrangian Vlasov codes", IEEE transactions on IFP, 1996.

[11] G. Manfredi, M. Shoucri, M.R. Feix, P. Bertrand, E. Fijalkow, A. Ghizzo, "The numerical integration of the Vlasov equation possessing an invariant", Journal of Computational Physics 121, p. 298-313 (1995)

[12] G. Depret, X. Garbet, P. Bertrand, A. Ghizzo, "Trapped-ion driven turbulence in tokamak plasmas", Plasma Phys. Control. Fusion 42, p.949-971 (2000)

[13] A.Staniforth, J. Cote, "Semi-lagrangian integration schemes for atmospheric models - A review", Monthly Weather Rev., 119 (1991)

[14] Timothy G. Mattson, Beverly A. Sanders, and Berna L. Massingill, Patterns for Parallel Programming Software Patterns Series, Addison Wesley 2004.

[15] G. Allaire, Z. Belhachmi, and F. Jouve. The homogenization method for topology and shape optimization. simple and multiple loads case. Revue europÈenne des Eléments finis, 5 :649_672, 1996.

[16] A.Choquet-Geniet, D.Geniet-R Schott, "A Methodology for Transforming Sequential Flowcharts into Parallel Ones Using Fifo Nets", vo11 pp :529-545, 1995.

[17] L. Alonso and R. Schott. A parallel algorithm for the generation of a permutation and applications. Theoretical Computer Science, 159(1):15 - 28, 1996

[18] Y. Atassi.Visibility graph of a set of line segments: A dynamic sequential3' algorithm and its parallel version. In Proceedings of CCCG'96 (Eight Canadian Conference on Computational Geometry), pages 246 - 251, Ottawa, Canada, August 1996..

[19] A.Darte, F.Desprez, J.C.Mignot et Y.Robert. TransTool : A restructuring tool for the parallelization of applications using high performance fortran. Journal of the brazilian computer society, 3(2) :5-15, nov1996

[20] G Blanc and I.-A. Ene. Indice de productivité en régime transitoire pour la simulation numérique des essais de puits. rapport

[21] E.Dilon. Propositions pour la maîtrise de la programmation par échange de messages. Thèse uhp, 1997.

[22] R. Bunoiu, S. Richard, and J. Saint Jean Paulin. Calcul numérique de la charge hydraulique dans un module de stockage. Poster au congrès national d'analyse numérique, 1996.

[23] R. Bunoiu and J. Saint Jean Paulin. Nonlinear viscous row through a thin slab in the lubrication case. (Sur invitation) International Workshop on Optimization of non linear systems and of free boundaries, Constanμa, Roumanie, 1996.

[24] R. Bunoiu. Homogénéisation des caractéristiques hydrauliques d'un
Module de stockage, Etude théorique et calculs numériques. rapport de contrat CEA, janvier 1996.

[25] R. Bunoiu. Sur quelques problèmes mathématiques en mécanique des fluides. Forum des jeunes mathématiciens, 1997.

[26] R. Baeza-Yates and R. Schott. Parallel searching in the plane. Computational Geometry: Theory and Applications, 5:143 _ 154, 1996.

[27] O. Coulaud and Dillon E. Early implementation of para++ with mpi-2. In MPI developers conference'96, Notre Dame, Indiana, july 1996.

[28] Using MPI, 2nd Edition, by William Gropp, Ewing Lusk, and Anthony Skjellum, published by MIT Press ISBN 0-262-57132-3.

[29] PVM: Parallel Virtual Machine: A Users' Guide and Tutorial for Network Parallel Computing (Scientific and Engineering Computation) by Al Geist, Adam Beguelin, Jack Dongarra, Weicheng Jiang, Robert Manchek, Vaidyalingam S. Sunderam

[30] Automatic perfomance analysis of hybrid MPI/OpenMP applications, Felix Wolf, Bernd Mohr, Journal of system architecture 49 (2003) 421-439.

[31] Some Simple OpenMP Optimization Techniques, M.Muller, WOMPAT 2001, LNCS pp.31-39 2001.

[32] R.Couturier, Utilisation des methodes formelles pour le developpement de programmes paralleles. These uhp, janvier 2000.

[33] F.Califano, A.Mangeney, C.Cavazzoni, P.Travnicek, A numerical scheme for the integration of the Vlasov-Maxwell system of equations, Science and supercomputing of CINECA – rapport 2001 pp.456-471

[34] Trong T.Bui, A parallel, finite-volume algorithm for large-eddy simulation of turbulent flows. NASA dryden flight research center edwards CA, Computers & Fluids 29 (2000) pp.877-915.

[35] F.Filbet, E.Sonnendrucker and P.Bertrand, Conservative numerical schemes for the Vlasov equation. Journal of computational physics 172, pp.166-187, 2001

[36] B. Meyer , « Eiffel : the language » livre publie chez Prentice-Hall, Inc. Upper Saddle River, NJ, USA 1992

RESULTATS NUMERIQUES

Pour la résolution du problème d'interaction onde particules nous avons associé au système Vlasov-onde des conditions initiales pour la fonction de distribution et le champ électrique. En ce qui concerne la fonction de distribution toute fonction continue convient théoriquement, mais nous nous plaçons dans le cas particulier d'un taux d'amortissement Landau de la théorie linéaire où l'on veut une condition initiale sur la fonction de distribution telle que le système reste stationnaire si on prend le champ électrique initial nul. En ce cas la fonction de distribution a la forme analytique suivante:

$$f(v)=\begin{cases} 0 \quad si \quad v_1 > v \; ou \; v > v_2 \\ C_1 + C_2\left(1 - \dfrac{v_1}{v_2 + v_1 - v}\right) \quad si \quad v_1 \le v \le v_2 \end{cases}$$

Avec les constantes positives

$$C_1 = \frac{1 + C_2\left[v_1 \ln\left(\frac{v_2}{v_1}\right) - v_2 + v_1 \right]}{v_2 - v_1} \text{ et } C_2 = -\frac{2\gamma_L}{\pi\eta v_1}$$

déterminées par les conditions sur la normalisation de la fonction de distribution et sur l'amortissement de Landau. Les simulations numériques sont faites avec un taux d'amortissement $\gamma_L = \frac{1}{200}$ et le coefficient de dispersion $\beta = \frac{1}{\sqrt{2}}$.

En ce qui concerne le champ électrique la valeur initiale est

$$E = \varepsilon\beta\sqrt{2I}$$

Il faut remarquer que le champ électrique non-stationnaire est uniforme dans l'espace des phases.

ANNEXE A :

RESULTATS OPENMP

VLASOV AVEC OPENMP :

INTEGRATION OF THE VLASOV EQ. IN V-X SPACE

XL	VMAX	DT	TMAX	EPSB
6.28319	1.45000	0.05000	50.00000	0.02241

T	Part.reelle	Part.imag.	Champ	mode k	dens moy
S= 5.3791391876958068					
0.	0.133E-04	0.000E+00	-0.500E+01	0.512E-02	0.000E+00
0.	0.133E-04	0.000E+00	-0.500E+01	0.512E-02	0.000E+00
0.	0.133E-04	0.000E+00	-0.500E+01	0.512E-02	0.000E+00
0.	0.156E-04	0.592E-05	-0.477E+01	0.512E-02	0.000E+00
0.	0.156E-04	0.592E-05	-0.477E+01	0.512E-02	0.000E+00
0.	0.156E-04	0.592E-05	-0.477E+01	0.512E-02	0.000E+00
0.	0.156E-04	0.592E-05	-0.477E+01	0.512E-02	0.000E+00
S= 813.67973408649937					
0.	0.156E-04	0.592E-05	-0.477E+01	0.512E-02	0.795E+00
0.	0.156E-04	0.592E-05	-0.477E+01	0.107E+00	0.789E+00
0.	0.156E-04	0.592E-05	-0.477E+01	0.107E+00	0.789E+00
0.	0.156E-04	0.592E-05	-0.477E+01	0.107E+00	0.789E+00
0.	-0.311E-04	-0.103E-03	-0.291E+01	0.107E+00	0.789E+00
0.	-0.311E-04	-0.103E-03	-0.291E+01	0.107E+00	0.789E+00
0.	-0.311E-04	-0.103E-03	-0.291E+01	0.107E+00	0.789E+00
0.	-0.311E-04	-0.103E-03	-0.291E+01	0.107E+00	0.789E+00
S= 813.67651687640262					
0.	-0.311E-04	-0.103E-03	-0.291E+01	0.107E+00	0.795E+00
DENSK = 0.1065143298242703 DENSMOY = 0.78935290789912771					
0.	-0.311E-04	-0.103E-03	-0.291E+01	0.107E+00	0.789E+00
0.	-0.311E-04	-0.103E-03	-0.291E+01	0.107E+00	0.789E+00
0.	-0.311E-04	-0.103E-03	-0.291E+01	0.107E+00	0.789E+00
0.	-0.832E-04	-0.214E-03	-0.215E+01	0.107E+00	0.789E+00
0.	-0.832E-04	-0.214E-03	-0.215E+01	0.107E+00	0.789E+00
0.	-0.832E-04	-0.214E-03	-0.215E+01	0.107E+00	0.789E+00
0.	-0.832E-04	-0.214E-03	-0.215E+01	0.107E+00	0.789E+00
S= 813.68053851789193					
0.	-0.832E-04	-0.214E-03	-0.215E+01	0.107E+00	0.795E+00
DENSK = 0.1065185443282461 DENSMOY = 0.78935683528339473					
0.	-0.832E-04	-0.214E-03	-0.215E+01	0.107E+00	0.789E+00
0.	-0.832E-04	-0.214E-03	-0.215E+01	0.107E+00	0.789E+00
0.	-0.832E-04	-0.214E-03	-0.215E+01	0.107E+00	0.789E+00
0.	-0.141E-03	-0.328E-03	-0.171E+01	0.107E+00	0.789E+00
0.	-0.141E-03	-0.328E-03	-0.171E+01	0.107E+00	0.789E+00
0.	-0.141E-03	-0.328E-03	-0.171E+01	0.107E+00	0.789E+00

```
0.     -0.141E-03     -0.328E-03     -0.171E+01      0.107E+00      0.789E+00
 S= 813.68135161198234
0.     -0.141E-03     -0.328E-03     -0.171E+01      0.107E+00      0.795E+00
DENSK = 0.10652235895841293 DENSMOY = 0.78935762932059228
0.     -0.141E-03     -0.328E-03     -0.171E+01      0.107E+00      0.789E+00
0.     -0.141E-03     -0.328E-03     -0.171E+01      0.107E+00      0.789E+00
0.     -0.141E-03     -0.328E-03     -0.171E+01      0.107E+00      0.789E+00
0.     -0.204E-03     -0.444E-03     -0.140E+01      0.107E+00      0.789E+00
0.     -0.204E-03     -0.444E-03     -0.140E+01      0.107E+00      0.789E+00
0.     -0.204E-03     -0.444E-03     -0.140E+01      0.107E+00      0.789E+00
0.     -0.204E-03     -0.444E-03     -0.140E+01      0.107E+00      0.789E+00
 S= 813.65660104084509
0.     -0.204E-03     -0.444E-03     -0.140E+01      0.107E+00      0.795E+00
DENSK = 0.10654656106427862 DENSMOY = 0.78933345884096606
0.     -0.204E-03     -0.444E-03     -0.140E+01      0.107E+00      0.789E+00
0.     -0.204E-03     -0.444E-03     -0.140E+01      0.107E+00      0.789E+00
0.     -0.204E-03     -0.444E-03     -0.140E+01      0.107E+00      0.789E+00
0.     -0.273E-03     -0.563E-03     -0.115E+01      0.107E+00      0.789E+00
0.     -0.273E-03     -0.563E-03     -0.115E+01      0.107E+00      0.789E+00
0.     -0.273E-03     -0.563E-03     -0.115E+01      0.107E+00      0.789E+00
0.     -0.273E-03     -0.563E-03     -0.115E+01      0.107E+00      0.789E+00
 S= 813.6645122775094
0.     -0.273E-03     -0.563E-03     -0.115E+01      0.107E+00      0.795E+00
DENSK = 0.10654539417432277 DENSMOY = 0.78934118465802106
0.     -0.273E-03     -0.563E-03     -0.115E+01      0.107E+00      0.789E+00
0.     -0.273E-03     -0.563E-03     -0.115E+01      0.107E+00      0.789E+00
0.     -0.273E-03     -0.563E-03     -0.115E+01      0.107E+00      0.789E+00
0.     -0.348E-03     -0.686E-03     -0.944E+00      0.107E+00      0.789E+00
0.     -0.348E-03     -0.686E-03     -0.944E+00      0.107E+00      0.789E+00
0.     -0.348E-03     -0.686E-03     -0.944E+00      0.107E+00      0.789E+00
0.     -0.348E-03     -0.686E-03     -0.944E+00      0.107E+00      0.789E+00
 S= 813.67823198696408
0.     -0.348E-03     -0.686E-03     -0.944E+00      0.107E+00      0.795E+00
DENSK = 0.10654551380181022 DENSMOY = 0.78935458281178539
0.     -0.348E-03     -0.686E-03     -0.944E+00      0.107E+00      0.789E+00
0.     -0.348E-03     -0.686E-03     -0.944E+00      0.107E+00      0.789E+00
0.     -0.348E-03     -0.686E-03     -0.944E+00      0.107E+00      0.789E+00
0.     -0.429E-03     -0.812E-03     -0.766E+00      0.107E+00      0.789E+00
0.     -0.429E-03     -0.812E-03     -0.766E+00      0.107E+00      0.789E+00
0.     -0.429E-03     -0.812E-03     -0.766E+00      0.107E+00      0.789E+00
0.     -0.429E-03     -0.812E-03     -0.766E+00      0.107E+00      0.789E+00
 S= 813.70080333290377
0.     -0.429E-03     -0.812E-03     -0.766E+00      0.107E+00      0.795E+00
DENSK = 0.10655462876162786 DENSMOY = 0.78937662514180462
0.     -0.429E-03     -0.812E-03     -0.766E+00      0.107E+00      0.789E+00
0.     -0.429E-03     -0.812E-03     -0.766E+00      0.107E+00      0.789E+00
0.     -0.429E-03     -0.812E-03     -0.766E+00      0.107E+00      0.789E+00
0.     -0.516E-03     -0.942E-03     -0.610E+00      0.107E+00      0.789E+00
0.     -0.516E-03     -0.942E-03     -0.610E+00      0.107E+00      0.789E+00
0.     -0.516E-03     -0.942E-03     -0.610E+00      0.107E+00      0.789E+00
0.     -0.516E-03     -0.942E-03     -0.610E+00      0.107E+00      0.789E+00
 S= 813.63908309887097
...
...
...
 DENSK = 6.24634477658416315E-2 DENSMOY = 7.05790144250030976E-2
9.     -0.115E+02     -0.115E+02      0.902E+01      0.625E-01      0.706E-01
9.     -0.115E+02     -0.115E+02      0.902E+01      0.625E-01      0.706E-01
9.     -0.115E+02     -0.115E+02      0.902E+01      0.625E-01      0.706E-01
9.     -0.121E+02     -0.121E+02      0.907E+01      0.625E-01      0.706E-01
9.     -0.121E+02     -0.121E+02      0.907E+01      0.625E-01      0.706E-01
```

```
9.    -0.121E+02     -0.121E+02      0.907E+01      0.625E-01      0.706E-01
9.    -0.121E+02     -0.121E+02      0.907E+01      0.625E-01      0.706E-01
 S= 82.938203350335044
9.    -0.121E+02     -0.121E+02      0.907E+01      0.625E-01      0.810E-01
DENSK = 6.7726773589391609E-2 DENSMOY = 7.5741273596327388l-2
9.    -0.121E+02     -0.121E+02      0.907E+01      0.677E-01      0.757E-01
9.    -0.121E+02     -0.121E+02      0.907E+01      0.677E-01      0.757E-01
9.    -0.121E+02     -0.121E+02      0.907E+01      0.677E-01      0.757E-01
9.    -0.127E+02     -0.127E+02      0.911E+01      0.677E-01      0.757E-01
9.    -0.127E+02     -0.127E+02      0.911E+01      0.677E-01      0.757E-01
9.    -0.127E+02     -0.127E+02      0.911E+01      0.677E-01      0.757E-01
9.    -0.127E+02     -0.127E+02      0.911E+01      0.677E-01      0.757E-01
 S= 66.875044443135025
9.    -0.127E+02     -0.127E+02      0.911E+01      0.677E-01      0.653E-01
DENSK = 5.25140381704877549E-2 DENSMOY = 6.0054594976014862E-2
9.    -0.127E+02     -0.127E+02      0.911E+01      0.525E-01      0.601E-01
10.    -0.127E+02     -0.127E+02      0.911E+01      0.525E-01      0.601E-01
10.    -0.127E+02     -0.127E+02      0.911E+01      0.525E-01      0.601E-01
10.    -0.133E+02     -0.133E+02      0.916E+01      0.525E-01      0.601E-01
10.    -0.133E+02     -0.133E+02      0.916E+01      0.525E-01      0.601E-01
10.    -0.133E+02     -0.133E+02      0.916E+01      0.525E-01      0.601E-01
10.    -0.133E+02     -0.133E+02      0.916E+01      0.525E-01      0.601E-01
 S= 57.385260036292742
10.    -0.133E+02     -0.133E+02      0.916E+01      0.525E-01      0.560E-01
DENSK = 4.4134954337848607 3E-2 DENSMOY = 5.0787227391207945 3E-2
10.    -0.133E+02     -0.133E+02      0.916E+01      0.441E-01      0.508E-01
…
...
```

ANNEXE B :

RESULTATS MPI

VLASOV AVEC MPI :

INTEGRATION OF THE VLASOV EQ. IN V-X SPACE

XL	VMAX	DT	TMAX	EPSB
6.28319	1.45000	0.10000	10.00000	0.02241

T	Part.reelle	Part.imag.	Champ	mode k	dens moy
S= 1087.2026350043973					
0.000	0.133E-04	0.000E+00	-0.500E+01	0.148E+00	0.000E+00
0.100	0.133E-04	0.000E+00	-0.500E+01	0.148E+00	0.000E+00
0.100	0.133E-04	0.000E+00	-0.500E+01	0.148E+00	0.000E+00
0.100	-0.128E-03	-0.301E-03	-0.180E+01	0.148E+00	0.000E+00
0.100	-0.128E-03	-0.301E-03	-0.180E+01	0.148E+00	0.000E+00
0.100	-0.128E-03	-0.301E-03	-0.180E+01	0.148E+00	0.000E+00
0.100	-0.128E-03	-0.301E-03	-0.180E+01	0.148E+00	0.000E+00
S= 1202.7998343134784					
0.100	-0.128E-03	-0.301E-03	-0.180E+01	0.148E+00	0.117E+01
0.200	-0.128E-03	-0.301E-03	-0.180E+01	0.148E+00	0.117E+01
0.200	-0.128E-03	-0.301E-03	-0.180E+01	0.148E+00	0.117E+01
0.200	-0.193E-03	-0.363E-03	-0.157E+01	0.148E+00	0.117E+01
0.200	-0.193E-03	-0.363E-03	-0.157E+01	0.148E+00	0.117E+01
0.200	-0.193E-03	-0.363E-03	-0.157E+01	0.148E+00	0.117E+01
0.200	-0.193E-03	-0.363E-03	-0.157E+01	0.148E+00	0.117E+01
S= 1202.7988987042247					
0.200	-0.193E-03	-0.363E-03	-0.157E+01	0.148E+00	0.117E+01
0.300	-0.193E-03	-0.363E-03	-0.157E+01	0.148E+00	0.117E+01
0.300	-0.193E-03	-0.363E-03	-0.157E+01	0.148E+00	0.117E+01
0.300	-0.264E-03	-0.419E-03	-0.138E+01	0.148E+00	0.117E+01
0.300	-0.264E-03	-0.419E-03	-0.138E+01	0.148E+00	0.117E+01
0.300	-0.264E-03	-0.419E-03	-0.138E+01	0.148E+00	0.117E+01
0.300	-0.264E-03	-0.419E-03	-0.138E+01	0.148E+00	0.117E+01
S= 1202.79785607246					
0.300	-0.264E-03	-0.419E-03	-0.138E+01	0.148E+00	0.117E+01
0.400	-0.264E-03	-0.419E-03	-0.138E+01	0.148E+00	0.117E+01
0.400	-0.264E-03	-0.419E-03	-0.138E+01	0.148E+00	0.117E+01
0.400	-0.341E-03	-0.467E-03	-0.123E+01	0.148E+00	0.117E+01
0.400	-0.341E-03	-0.467E-03	-0.123E+01	0.148E+00	0.117E+01
0.400	-0.341E-03	-0.467E-03	-0.123E+01	0.148E+00	0.117E+01
0.400	-0.341E-03	-0.467E-03	-0.123E+01	0.148E+00	0.117E+01
S= 1202.7967496452002					
0.400	-0.341E-03	-0.467E-03	-0.123E+01	0.148E+00	0.117E+01
0.500	-0.341E-03	-0.467E-03	-0.123E+01	0.148E+00	0.117E+01
0.500	-0.341E-03	-0.467E-03	-0.123E+01	0.148E+00	0.117E+01
0.500	-0.422E-03	-0.507E-03	-0.110E+01	0.148E+00	0.117E+01
0.500	-0.422E-03	-0.507E-03	-0.110E+01	0.148E+00	0.117E+01
0.500	-0.422E-03	-0.507E-03	-0.110E+01	0.148E+00	0.117E+01

```
  0.500    -0.422E-03    -0.507E-03    -0.110E+01    0.148E+00    0.117E+01
 S= 1202.7956263093238
  0.500    -0.422E-03    -0.507E-03    -0.110E+01    0.148E+00    0.117E+01
  0.600    -0.422E-03    -0.507E-03    -0.110E+01    0.148E+00    0.117E+01
  0.600    -0.422E-03    -0.507E-03    -0.110E+01    0.148E+00    0.117E+01
  0.600    -0.506E-03    -0.540E-03    -0.983E+00    0.148E+00    0.117E+01
  0.600    -0.506E-03    -0.540E-03    -0.983E+00    0.148E+00    0.117E+01
  0.600    -0.506E-03    -0.540E-03    -0.983E+00    0.148E+00    0.117E+01
  0.600    -0.506E-03    -0.540E-03    -0.983E+00    0.148E+00    0.117E+01
 S= 1202.7945357607541
  0.600    -0.506E-03    -0.540E-03    -0.983E+00    0.148E+00    0.117E+01
  0.700    -0.506E-03    -0.540E-03    -0.983E+00    0.148E+00    0.117E+01
  0.700    -0.506E-03    -0.540E-03    -0.983E+00    0.148E+00    0.117E+01
  0.700    -0.594E-03    -0.563E-03    -0.882E+00    0.148E+00    0.117E+01
  0.700    -0.594E-03    -0.563E-03    -0.882E+00    0.148E+00    0.117E+01
  0.700    -0.594E-03    -0.563E-03    -0.882E+00    0.148E+00    0.117E+01
  0.700    -0.594E-03    -0.563E-03    -0.882E+00    0.148E+00    0.117E+01
 S= 1202.7935295940492
  0.700    -0.594E-03    -0.563E-03    -0.882E+00    0.148E+00    0.117E+01
  0.800    -0.594E-03    -0.563E-03    -0.882E+00    0.148E+00    0.117E+01
  0.800    -0.594E-03    -0.563E-03    -0.882E+00    0.148E+00    0.117E+01
  0.800    -0.683E-03    -0.578E-03    -0.793E+00    0.148E+00    0.117E+01
  0.800    -0.683E-03    -0.578E-03    -0.793E+00    0.148E+00    0.117E+01
  0.800    -0.683E-03    -0.578E-03    -0.793E+00    0.148E+00    0.117E+01
  0.800    -0.683E-03    -0.578E-03    -0.793E+00    0.148E+00    0.117E+01
  0.800    -0.683E-03    -0.578E-03    -0.793E+00    0.148E+00    0.117E+01
...
...
...
  9.800    -0.136E-02     0.106E-02    -0.138E+00    0.148E+00    0.117E+01
  9.800    -0.136E-02     0.106E-02    -0.138E+00    0.148E+00    0.117E+01
  9.800    -0.128E-02     0.111E-02    -0.153E+00    0.148E+00    0.117E+01
  9.800    -0.128E-02     0.111E-02    -0.153E+00    0.148E+00    0.117E+01
  9.800    -0.128E-02     0.111E-02    -0.153E+00    0.148E+00    0.117E+01
  9.800    -0.128E-02     0.111E-02    -0.153E+00    0.148E+00    0.117E+01
 S= 1202.8760871162312
  9.800    -0.128E-02     0.111E-02    -0.153E+00    0.148E+00    0.117E+01
  9.900    -0.128E-02     0.111E-02    -0.153E+00    0.148E+00    0.117E+01
  9.900    -0.128E-02     0.111E-02    -0.153E+00    0.148E+00    0.117E+01
  9.900    -0.120E-02     0.116E-02    -0.169E+00    0.148E+00    0.117E+01
  9.900    -0.120E-02     0.116E-02    -0.169E+00    0.148E+00    0.117E+01
  9.900    -0.120E-02     0.116E-02    -0.169E+00    0.148E+00    0.117E+01
  9.900    -0.120E-02     0.116E-02    -0.169E+00    0.148E+00    0.117E+01
 S= 1202.8875197349282
  9.900    -0.120E-02     0.116E-02    -0.169E+00    0.148E+00    0.117E+01
 10.000    -0.120E-02     0.116E-02    -0.169E+00    0.148E+00    0.117E+01
 10.000    -0.120E-02     0.116E-02    -0.169E+00    0.148E+00    0.117E+01
 10.000    -0.112E-02     0.120E-02    -0.189E+00    0.148E+00    0.117E+01
 10.000    -0.112E-02     0.120E-02    -0.189E+00    0.148E+00    0.117E+01
 10.000    -0.112E-02     0.120E-02    -0.189E+00    0.148E+00    0.117E+01
 10.000    -0.112E-02     0.120E-02    -0.189E+00    0.148E+00    0.117E+01
 S= 1202.8985737084017
 10.000    -0.112E-02     0.120E-02    -0.189E+00    0.148E+00    0.117E+01
...
```

Université HENRI POINCARÉ
NANCY 1

Monsieur **PEYROUX Julien**

DOCTORAT DE L'UNIVERSITE HENRI POINCARE, NANCY 1

en PHYSIQUE DES PLASMAS

VU, APPROUVÉ ET PERMIS D'IMPRIMER N° 1126

Nancy, le 14 novembre 2005.

Le Président de l'Université

UNIVERSITÉ HENRI POINCARÉ NANCY 1
LE PRÉSIDENT
P. FINANCE

Université Henri Poincaré, Nancy I
24-30 rue Lionnois - B.P. 3069 - 54013 NANCY Cédex
Tél. : 03 83 68 20 00 - Fax : 03 83 68 21 00