



REPUBLIQUE ALGERIENNE DEMOCRATIQUE ET POPULAIRE
MINISTRE DE L'ENSEIGNEMENT SUPERIEUR ET DE LA RECHERCHE SCIENTIFIQUE

UNIVERSITE ABOU-BEKR BELKAID - TLEMCCEN

MEMOIRE

Présenté à :

FACULTE DES SCIENCES – DEPARTEMENT DE PHYSIQUE

MASTER EN PHYSIQUE

Spécialité : Physique des Plasmas

Par :

Mlle LAISSOUF SARRA KHADIDJA

Sur le thème

Simulation Numérique de la dynamique des particules dans un plasma électrostatique par la méthode PIC

Soutenu publiquement le 10 Juin 2023 à Tlemcen devant le jury composé de :

BEDRANE Zeyneb	Professeur	Université de Tlemcen	Présidente
SENOUDI Assia Rachida	Maître de Conférences (A)	Université de Tlemcen	Encadrant
BABA AHMED Nassim	Maître de Conférences (A)	Université de Tlemcen	Examineur

Année Universitaire : 2022 ~ 2023

Simulation Numérique de la dynamique
des particules dans un plasma électrostatique par la méthode
PIC.

Laissouf / 2023

Remerciement

Tout d'abord je remercie ALLAH tout puissant de m'avoir donné la force et le courage pour pouvoir dépasser toutes les difficultés et d'accomplir ce travail.

Ce mémoire a été réalisé au Laboratoire de Physique Théorique, du Département de Physique, Faculté des Sciences de l'Université Abou Bekr Belkaid-Tlemcen.

J'adresse de chaleureux remerciements à Madame SENOUDI Assia Rachida Maître de Conférences Classe A à l'Université Abou Bekr Belkaid-Tlemcen pour m'avoir fait le grand honneur d'encadrer ce mémoire, et partagé son expérience dans un domaine très compliqué, et pour le temps qu'elle m'a accordé et je veux lui exprime ma gratitude pour ses conseils et sa patience.

Je tiens aussi à remercier vivement les membres du jury :

Madame BEDRANE Zeyneb Professeur à l'Université Abou Bekr Belkaid, qui m'a fait le grand honneur de présider le jury.

Monsieur BABA AHMED Nassim Maître de Conférences Classe A à l'Université Abou Bekr Belkaid pour avoir bien voulu accepter d'évaluer et d'examiner mon travail.

J'exprime ma profonde reconnaissance à celui qui a sacrifié sa vie pour mon éducation et qui voulait voir ma réussite ; mon défunt père repose en paix.

Je tiens à exprimer mes sincères remerciements à ma chère mère pour son soutien indéfectible devant les obstacles.

Je désire aussi remercier ma sœur et mon frère pour leur soutien et encouragement tout au long de mes études.

Je tiens sincèrement à remercier tous les professeurs qui m'ont enseigné, sans eux je n'aurais pu réussir ce travail de mémoire.

Enfin, un grand merci à ma famille, mes amies et tous mes proches pour m'avoir encouragée et m'avoir poussée vers la réussite.

Mlle S.Kh.LAISSOUF

Table des matières :

Introduction générale	1
Chapitre 1 : Physique des Plasmas	3
I. Introduction	4
II. Généralités	4
II.1. La matière	5
II.2. Les états de la matière	6
III. Physiques des plasmas	6
III.1. C'est quoi un plasma	7
III.2. Création de plasma	7
III.3. Classification des plasmas	9
III.4. Grandeurs caractéristiques de plasma	10
1. Le degré d'ionisation	10
2. Le libre parcours moyen	10
3. La longueur de Debye	10
4. La longueur de Landau	11
5. La fréquence plasma	11
III.5 Conditions d'existence d'un plasma	12
III.6. Interactions dans un plasma	13
IV. Description d'un plasma	14
V. Modélisation Mathématique de la dynamique d'un plasma	15
V.1. Fonction de distribution	15
V.2. Densité de charge	17
V.3. Force de Lorentz	17
V.4. Champ électrique	18
V.5. Equations de Maxwell	18
VI. Conclusion	18
Chapitre 2 : Etapes de Simulation par la Méthode PIC	19
Introduction	20
I. Méthodologie de travail	20
II. Subdivision du domaine d'étude	21
IV. Modèle électrostatique unidimensionnel	23
IV.1. Equations	23
IV.2. Subdivision du domaine d'étude par la méthode PIC 1D	25
V. Schémas numériques de résolution	26
V.1 Calcul du potentiel électrique $\phi(x)$	26
V.2 Calcul de la densité es charges	27
V.3 Calcul du champ électrique	29
V.4 Calcul de l'accélération	30
V.5 Pushing particle : Calcul des nouvelles positions et vitesses des particules	31
1. Schéma d'Euler	32
2. Schéma Leap-Frog	32
3. Schéma de Runge-Kutta	33
VI. Choix des paramètres initiaux	35
VI.1 Positions initiales	35
VI.2 Vitesses initiales	36
VII. Paramètres de stabilité et de précision	37
VIII. Conditions Périodiques	37
IX. Algorithme PIC	38

X.	Conclusion	39
Chapitre 3 : Résultats de la Simulation PIC 1D		40
I.	Introduction	40
II.	Distribution initiale des positions des particules	41
III.	Distribution initiale des vitesses des particules	42
IV.	Simulation d'un seul faisceau d'électrons	43
IV.1	Oscillation de plasma froid	43
IV.2	Dynamique des électrons dans un plasma chaud	44
V.	Two-Stream-Instability	45
VI.	VI.Conclusion	48
Conclusion générale		49
Références		
Annexes		

Introduction générale

Ces dernières années, la physique du plasma est devenue une des sciences les plus fondamentales, en raison du vaste domaine de ses applications, dans l'industrie, dans l'environnement, dans le domaine médical, dans le domaine d'éclairage, et aussi dans le domaine énergétique comme la fusion nucléaire et les réacteurs Tokamaks.

Pour modéliser un plasma qui est assez complexe, il existe l'approche cinétique qui décrit le plasma en suivant les particules dans l'espace des phases et en résolvant les équations de Vlasov, alors que l'approche fluide est basée sur la détermination des grandeurs macroscopiques en résolvant les équations de la magnétohydrodynamique. Ces deux approches peuvent être rassemblées dans une seule approche appelée hybride, elle se rapproche des méthodes dites de décomposition de domaines, pour lesquelles le modèle fluide est utilisé dans les zones où le système est à l'équilibre alors que le modèle cinétique est utilisé uniquement dans les zones où le système est hors équilibre.

Les méthodes Particles in Cell (PIC) sont des outils numériques fondamentaux pour simuler la dynamique des particules (électrons, ions ou trous et mêmes molécules) dans un plasma. Ces méthodes ont été utilisées pour la première fois en 1955, avant même que les premiers compilateurs Fortran ne soient disponibles. La méthode a gagné en popularité à partir de 1960 lorsque les chercheurs Buneman, Dawson, Hockney, Birdsall, Morse et autres l'ont exploité dans la modélisation plasma [1].

Cette méthode est couramment utilisée en physique des plasmas de fusion, dans l'étude des différents phénomènes intervenant dans les propulseurs plasmas pour la simulation de la source d'ion dans le cadre du projet ITER¹, dans l'étude des phénomènes de l'environnement spatial tels que les vents solaires, les environnements atmosphériques, les interactions laser-plasmas. . . [2]. En outre, elle est devenue un outil efficace pour fournir des prévisions précises des performances dans les applications de la physique des plasmas.

Le principe de la méthode PIC est de suivre les trajectoires de milliards de particules chargées dans un champ électromagnétique ; Elle détermine la vitesse et la position de chaque particule, tandis que en même temps, elle calcule les densités et les courants sur les points de la grille de subdivision du domaine, et elle résout de façon simultanée les équations du champ.

L'objectif de ce travail est de présenter le principe de la méthode PIC et de concevoir un code Python pour la simulation de la dynamique des particules chargées dans un plasma soumis initialement à une perturbation extérieure provoquant le mouvement des électrons. Nous avons supposé que 1/ le champ magnétique est négligeable devant le champ électrostatique, 2/ le champ électrostatique à l'intérieur du plasma est unidimensionnel 1D c'est-à-dire qu'il ne varie que selon l'axe (ox), 3/ les effets collisionnels entre particules ne sont pas pris en considération, 4/ on néglige le mouvement des ions devant celui des électrons à cause de leur grande masse, 5/ Le plasma est soumis initialement à un champ extérieur électrostatique qui perturbe les positions et les vitesses initiales des électrons, provoquant une dynamique dans la direction d'induction du champ électrique (on prend l'axe (ox) comme direction de déplacement des particules).

¹ ITER : international thermonuclear experimental reactor

Les électrons seront traqués à partir de la méthode PIC 1D électrostatique non collisionnelle.

Le mémoire est décomposé en 03 chapitres comme suit :

Le premier chapitre où sont données quelques propriétés générales du plasma, ses paramètres caractéristiques, ainsi que le modèle mathématique qui permet de décrire le mouvement des électrons soumis à un champ électromagnétique. Les équations à résoudre sont les équations de Poisson et de Newton, couplées à la force de Lorentz (dans ce travail, nous avons négligé le champ magnétique).

Le chapitre 2 qui est le plus important est composé de plusieurs parties explicatives de la méthode PIC 1D, accompagnées de figures illustratives. Nous avons montré qu'il est préférable du point de vue numérique, de normaliser les grandeurs à calculer. Ensuite nous avons procédé de façon méthodique à développer clairement toutes les phases de la simulation.

- Le domaine du plasma est subdivisé en un nombre de cellules de géométrie simple. La subdivision peut être simplement une grille à une dimension si la variation des grandeurs ne se fait qu'en fonction d'une seule variable spatiale (exemple x).

- Une distribution initiale des particules va les placer dans ces cellules. Cette distribution spatiale initiale peut être soit homogène ou bien aléatoire. Une perturbation des positions et des vitesses entrainera donc une dynamique à suivre dans le temps, pour chaque particule et en fonction de plusieurs paramètres tels que la vitesse thermique et la vitesse de dérive.

Pour cela deux algorithmes nécessaires ont été établis : 1/ un algorithme pour la phase initialisation 2/ un algorithme pour suivre le mouvement des électrons. Cet algorithme a pour fonction de calculer sur chaque point de la grille spatiale, la densité des particules $n(x)$, le potentiel électrostatique $\phi(x)$, le champ électrostatique $E(x)$, les positions (x_p) et les vitesses des particules (v_p) dans un cycle de simulation temporel.

Les outils numériques utilisés dans ce travail sont les schémas des différences finies, l'interpolation polynomiale de Lagrange, les schémas d'Euler, de Leap Frog et de Runge Kutta d'ordre 4. Tous ces algorithmes ont été implémentés en Langage Python avec pour seuls modules Numpy et Matplotlib.

Dans le troisième chapitre, différentes simulations ont été faites à partir du code Python que nous avons réalisé pour donner différentes configurations de plasmas : plasma oscillatoire, flux d'électrons, instabilité de deux flux d'électrons. Nos résultats sont donnés sous forme de représentation graphique : 1/ diagramme de phase qui est la représentation des vitesses en fonction des positions des électrons, 2/ évolution spatiale du potentiel et du champ électrostatique en fonction du temps, 3/ évolution de l'énergie totale des électrons en fonction du temps. 4/ fonction de distribution des vitesses des électrons.

Des annexes sont données comme support supplémentaire pour notre développement mathématique.

Finalement une conclusion est donnée avec quelques perspectives.

Chapitre 1 :

Physique des Plasmas

I. Introduction :

La physique des plasmas est actuellement une discipline importante puisque 99% de la matière dans l'univers est constituée de plasma. Une grande partie de la connaissance des plasmas provient de l'étude de la fusion nucléaire contrôlée, pour laquelle la physique des plasmas fournit la base scientifique.

La physique des plasmas consiste à résoudre des problèmes liés à l'interaction des champs électromagnétiques et des particules chargées (électrons, ions et trous). La dynamique des particules est gouvernée par l'interaction de ces particules avec les champs externes et les champs créés par ces mêmes particules.

Ce chapitre présente les éléments de base de la physique des plasmas, et aussi les équations de l'électromagnétisme qui serviront à poser le problème étudié.

II. Généralités

II.1 La matière :

Définition de la matière : La définition commune de la matière est toute substance qui a une masse et qui occupe de l'espace (volume). Elle est constituée d'assemblage d'atomes ou de molécules.

Description classique de l'atome : l'atome est la plus petite unité de la matière, en son centre se trouve un noyau composé de protons et de neutrons autour duquel se distribuent des électrons (Figure I.1) qui occupent des orbitales atomiques en interaction avec le noyau via la force électromagnétique. Le nuage électronique est stratifié en niveaux d'énergies quantifiés autour du noyau, niveaux qui définissent des couches et des sous-couches électroniques [3].

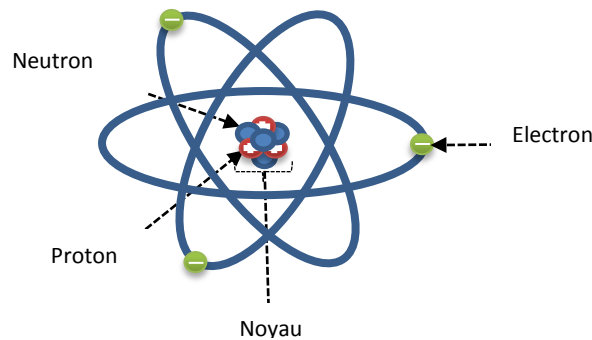


Figure I.1 : Modèle de l'atome de Lithium ${}^7_3\text{Li}$ pour illustration.

Les protons ont une charge positive, les électrons ont une charge négative et les neutrons ne possèdent pas de charge. Si le nombre des électrons et des neutrons n'est pas le même, l'atome aura une charge électrique et devient un *ion*. Si deux atomes ayant le même nombre atomique mais un nombre de neutrons différents, ils sont appelés *isotopes*.

Malgré la répulsion électrostatique entre les protons positifs, le noyau tient ensemble grâce à la force nucléaire forte qui permet la cohésion du noyau de l'atome (Figure I.2). Cette force d'attraction agit à courte portée au sein du proton et du neutron mais elle est plus forte que la force électromagnétique. Elle confine les quarks, particules élémentaires qui composent les protons et neutrons, en couples "quark-antiquark" (mésons), ou dans des triplets de quarks « un ou deux autres "quarks-antiquarks" » (baryons). Cette interaction se fait par l'échange de bosons appelés "gluons" [4].

Remarque : en 1925 Erwin Schrodinger révolutionne l'idée d'orbite électronique, en proposant un modèle ondulatoire dans lequel l'électron n'est plus décrit comme une particule à laquelle on attribue une orbite précise, mais comme une onde à laquelle on associe une

orbitale. Ce modèle donne une représentation de la région de l'espace autour du noyau où l'électron a la plus grande probabilité de se trouver.

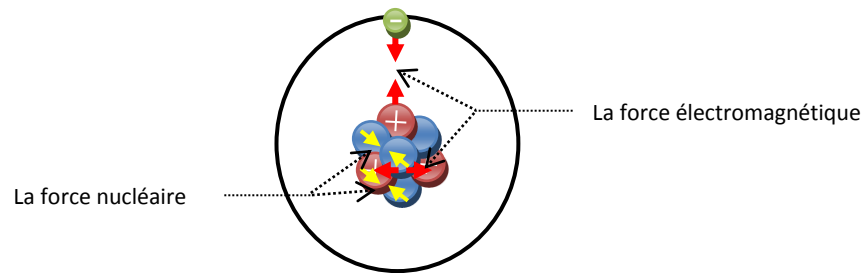


Figure I.2 : Les forces dans un atome

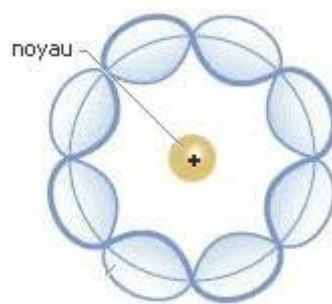


Figure I.3 : Modèle atomique de Schrodinger [5]

II.2 Les états de la matière :

La matière peut être dans différents états : liquide, solide, gazeux ou plasma (figure I.4) :

Solide : la matière possède un volume et une forme, les particules sont liées les unes aux autres.

Liquide : la matière possède un volume mais pas de forme propre (il prend la forme du récipient qui le contient), les particules sont faiblement liées.

Gaz : la matière ne possède ni forme propre ni volume propre, tend à occuper tout le volume disponible, les particules sont très faiblement liées, quasiment indépendantes.

Plasma : la matière ne possède ni forme propre ni volume propre, c'est un gaz hautement ionisé, les particules sont détachées de leurs électrons [6].

Remarque : La matière peut aussi exister dans l'univers avec un volume infiniment petit et une masse extrêmement importante comme le cas du trou noir¹ qui est créé après la mort d'une étoile très massive, le reste de cette étoile se concentre en un petit point

¹ Trou noir : est une région de l'espace dont le champ gravitationnel est si intense qu'il empêche toute forme de matière ou de rayonnement de s'en échapper [7].

appelé singularité ; La singularité est un point infiniment petit au centre d'un trou noir. Le pourtour d'un trou noir (l'horizon des événements) est de taille variable, mais la taille de trou noir est très petite [7].

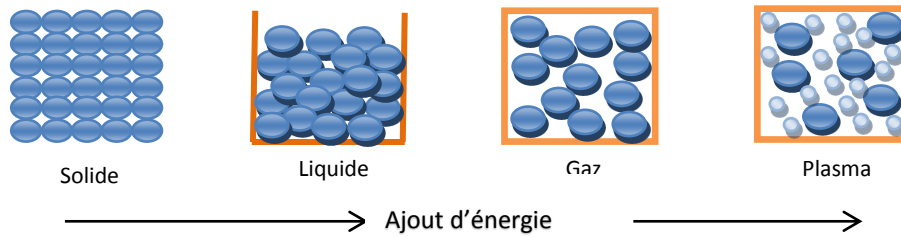


Figure I.4 : Les états de la matière qui dépendent de la température et de la pression et caractérisent un niveau d'organisation de la matière.

Le passage d'un état à un autre se réalise par l'ajout d'une énergie externe, soit par augmentation de la température ou de la pression. Le changement de l'état de la matière affecte les propriétés de la matière (densité, viscosité, et conductivité), mais l'élément reste le même.

Dans ce qui suit, on s'intéresse à introduire des notions de base de la physique des plasmas.

III. Physique des plasmas :

III.1 C'est quoi un plasma ?

Le plasma fut identifié pour la première fois en 1928 par le physicien américain Langmuir lorsqu'il observait le comportement d'un gaz ionisé dans des tubes à décharges [8-10].

Le terme plasma désigne un état dilué de la matière, analogue à un gaz, constitué de particules chargées (électrons et ions), de même proportion tel que le milieu soit de charge globalement neutre. L'hydrogène peut servir à constituer un plasma de façon simple étant donné qu'il est constitué d'un seul électron.



Figure I.5: Plasma naturel (Aurores boréales) à droite, plasma de laboratoire (lampe à plasma) à gauche [11,12]

La grande quantité de particules chargées libres dans un plasma fournit des conductivités électriques élevées et une possibilité d'établir des courants électriques, qui interagissent avec les champs électromagnétiques externes et les propres champs électromagnétiques générés par ces mêmes courants.

Grace à sa constitution, le plasma se définit comme un ensemble statistique de particules chargées, suffisamment dense pour avoir un comportement **collectif**.

III.2 Création du plasma :

En général, les atomes et les molécules d'un gaz possèdent une charge globale nulle dans les conditions normales de température et de pression². Pour créer un plasma à partir d'un gaz neutre, il faut fournir de l'énergie pour arracher un ou plusieurs électrons de chaque atome. Cette énergie doit être suffisante aux atomes pour qu'ils soient partiellement ou même totalement ionisés.

L'énergie peut être fournie sous plusieurs formes :

- **Les décharges électriques** : Appelées aussi décharges d'arc, comme l'a montré Langmuir initialement, en provoquant une décharge électrique dans un gaz, on peut l'ioniser partiellement (ou totalement), ainsi se crée un plasma. Ce type de plasmas est présent dans notre entourage familial tel que les enseignes au néon et les chalumeaux.

- **Le chauffage** : Dans la nature, l'effet qui permet l'ionisation est très souvent la température tout simplement.

Typiquement l'énergie d'ionisation d'un atome ou d'une molécule est de quelques électronvolts (eV). La température nécessaire pour former un plasma en équilibre thermodynamique local est donc celle à partir de laquelle l'énergie thermique, qui peut être estimée par le produit $k_B T^3$, atteint cet ordre de grandeur, c'est-à-dire lorsque $k_B T \approx 1$ eV, soit une température d'environ 11 000 K.

- **L'absorption d'une radiation lumineuse** : L'absorption d'un photon par un atome peut produire un ion et un électron. Si la recombinaison est suffisamment lente, un plasma se forme. Un exemple de plasma créé de cette façon à basse température est celui de l'ionosphère de la Terre ou les planètes qui possèdent une atmosphère. La source de radiation lumineuse peut être un laser à impulsions [13].

² Les conditions normales de température et de pression (parfois abrégé CNTP) sont des conditions pratiques pour l'expérimentation et la mesure en laboratoire en physique et en chimie. Les conditions les plus usuelles fixent la température normale à 0 °C (273,15 K) et la pression normale à 1 atm (1,013 bar = 101 325 Pa).

³ $k_B T$ est le produit de la constante de Boltzmann k_B et de la température absolue T (en kelvins). Du fait de l'unité de k_B (des joules par kelvin), le produit de ces deux éléments est une énergie, qui dépend donc de la température ; Elle vaut $4,11 \times 10^{-21}$ J à 25°C.

Il peut exister dans un plasma, les processus d'excitation, d'ionisation, d'attachement, de recombinaison et transfert de charges [10].

III.3 Classification des plasmas :

Les plasmas peuvent être classés selon deux catégories :

Plasmas naturels : ils sont répandus dans l'ionosphère, magnétosphère, vent et courant solaire. Les nébuleuses sont également constituées de gaz ionisés. La température des ions et des électrons est supérieure à 10 millions °C. Le gaz est complètement ionisé et le milieu est à l'équilibre thermodynamique complet [6,14].

Plasmas de laboratoires : les plasmas peuvent aussi être produits dans les laboratoires [9] (grâce à des décharges électriques, réacteur à plasma), et il existe deux types :

- Les plasmas chauds ou plasmas thermiques : Ce sont des plasmas fortement ionisés, caractérisés par une densité d'énergie élevée et une température égale pour toutes les espèces, où la température peut dépasser la valeur de 10^6 K pour les plasmas de fusion thermonucléaire.

- Les plasmas froids ou plasmas hors équilibre thermodynamique : Ce sont des plasmas faiblement ionisés, de température modérée entre 10^4 et 10^5 K. Les ions possèdent une température nettement inférieure à celle des électrons. Comme application de plasmas froids, on trouve les lampes à plasmas.

III.4 Grandeurs caractéristiques d'un plasma :

1. Le degré d'ionisation : le degré d'ionisation noté α (terme sans dimensions) est la proportion de particules neutres qui sont ionisées [10] :

$$\alpha = \frac{n_e}{n_n + n_e} \quad \text{I.1}$$

avec n_e , la densité des électrons et n_n , la densité des particules neutres.

- Si $\alpha = 0$: le plasma est neutre.
- Si $\alpha \ll 1$: le plasma est faiblement ionisé, la valeur de α peut varier entre 10^{-7} - 10^{-4} .
- Si $\alpha \approx 1$: le plasma est fortement ionisé, la valeur de α peut varier entre 0.1 et 1.

2. Le libre parcours moyen : il correspond à la distance moyenne parcourue par une particule chargée entre deux collisions successives, il est défini par la relation suivante [10] :

$$\lambda = \frac{1}{n_n \sigma} \quad \text{I.2}$$

Où σ représente la section efficace de collision en cm^2 , donnée par la relation : $\sigma = \pi r^2$ et r est le rayon de la particule cible (particule en collision).

3. La longueur de Debye : c'est la plus petite distance au-delà de laquelle le champ électrique est écranté [10] :

$$\lambda_D = \sqrt{\frac{\varepsilon_0 k_B T_e}{n_e q_e}} \quad \text{I.3}$$

Où ε_0 est la permittivité du vide⁴ ; k_B est la constante de Boltzmann ; q_e est la charge de l'électron⁵.

L'écrantage du champ électrique consiste en l'atténuation du champ électrique en raison de la présence de porteurs de charge électrique mobiles dans le matériau. La longueur de Debye peut être aussi définie comme le rayon de la sphère de Debye qui divise le milieu plasma en deux régions : l'une avec l'interaction à courte portée à l'intérieur de la sphère et l'autre région avec l'interaction à longue portée à l'extérieur.

4. La longueur de Landau : c'est la distance pour laquelle l'énergie potentielle d'interaction entre deux particules de même charge est égale à leurs énergies cinétiques d'agitation thermique [10] :

$$\lambda_L = \frac{q_e^2}{4\pi \varepsilon_0 k_B T_e} \quad \text{I.4}$$

5. La fréquence plasma : dite aussi fréquence de Langmuir, ou encore pulsation plasma, est la fréquence caractéristique des oscillations des charges électriques présentes dans les plasmas [10].

Si, dans un plasma électriquement neutre, on produit une perturbation locale sous la forme d'un excès de charge électrique positive ou négative, celui-ci va tendre à revenir vers l'état de l'équilibre de neutralité. Cette perturbation engendre des oscillations des particules chargées autour de son état d'équilibre. Les ions sont supposés infiniment lourds, donc au repos, les électrons sont mobiles. La perturbation spatiale (notée $\delta\vec{x}$) induit alors une séparation de charge δn_e qui se traduit par l'apparition d'un champ électrique \vec{E} . La relation fondamentale de la dynamique s'écrit comme :

⁴ $\varepsilon_0 = 1/\mu_0 c^2$; Elle vaut : $8.854\,187\,82 \times 10^{-12}$ (F m⁻¹).

⁵ q_e vaut : $-1.602\,176\,634 \times 10^{-19}$ (C).

$$m \frac{d^2 \delta \vec{x}}{dt^2} = -e \vec{E} = e \vec{\nabla} \phi \quad \text{I.5}$$

Où m est la masse de l'électron ; ϕ est le potentiel électrique ; e est la charge élémentaire.

En appliquant l'opérateur gradient ($\vec{\nabla}$) sur les deux membres de l'équation I.5, cette dernière s'écrit de nouveau comme :

$$m \frac{d^2 \vec{\nabla} \cdot \delta \vec{x}}{dt^2} = e \Delta \phi \quad \text{I.6}$$

L'équation de Poisson relie le potentiel ϕ à la densité de charge et peut s'écrire comme suit :

$$\Delta \phi = \frac{e}{\epsilon_0} \delta n_e \quad \text{I.7}$$

La valeur ($e \delta n_e$) représente la densité de charges.

Remplaçant le second membre de l'équation I.7 dans l'équation I.6, on trouve :

$$m \frac{d^2 \vec{\nabla} \cdot \delta \vec{x}}{dt^2} = \frac{e^2}{\epsilon_0} \delta n_e \quad \text{I.8}$$

La conservation de la matière au cours de cette perturbation s'exprime selon la loi de conservation [8] selon la relation suivante :

$$\frac{\partial \delta n_e}{\partial t} + \vec{\nabla} \cdot [(n_e + \delta n_e) \vec{v}] = 0 \quad \text{I.9}$$

En supposant que la perturbation est petite (approximation linéaire) l'équation I.9 devient :

$$\frac{\partial \delta n_e}{\partial t} + n_e \vec{\nabla} \cdot \frac{\partial \delta \vec{x}}{\partial t} = 0 \quad \text{I.10}$$

En intégrant l'équation I.10, nous trouvons que :

$$\delta n_e = -n_e \vec{\nabla} \cdot \delta \vec{x} \quad \text{I.11}$$

Remplaçant l'expression de $\vec{\nabla} \cdot \delta \vec{x}$ trouvée dans I.11 dans I.8, on obtient :

$$\frac{1}{n_e} \frac{d^2 \delta n_e}{dt^2} = -\frac{e^2}{m_e \epsilon_0} \delta n_e \quad \text{I.12}$$

En modifiant l'équation I.12, on obtient finalement la relation d'oscillation des électrons :

$$\frac{d^2 \delta n_e}{dt^2} + \frac{n_e e^2}{m_e \epsilon_0} \delta n_e = 0 \quad \text{I.13}$$

Cette dernière équation est l'équation d'un oscillateur harmonique oscillant à une pulsation appelée fréquence plasma donnée par la formule suivante :

$$\omega_p = \sqrt{\frac{n_e e^2}{m_e \epsilon_0}} \quad \text{I. 14}$$

Tableau I.2 : Propriétés physique de différents plasmas [15].

Plasma	Densité $n_e (m^{-3})$	Température $T_e (keV)$	Champ magnétique induit $B (T)$	Longueur de Debye $\lambda_D (m)$	Fréquence Plasma $\omega_p (rd/s)$
Interstellaire	10^6	10^{-5}	10^{-9}	0.7	$6 \cdot 10^4$
Vent solaire	10^7	10^{-2}	10^{-8}	7	$2 \cdot 10^5$
Ionosphère	10^{12}	10^{-4}	10^{-5}	$2 \cdot 10^3$	$6 \cdot 10^7$
Couronne solaire	10^{12}	0.1	10^{-3}	0.037	$6 \cdot 10^7$
Propulseur ionique	10^{15}	10^{-3}	-	$4 \cdot 10^{-4}$	$2 \cdot 10^{10}$
Décharge d'arc	10^{20}	10^{-3}	0.1	$7 \cdot 10^{-7}$	$6 \cdot 10^{11}$
Tokamak	10^{20}	1	10	$7 \cdot 10^{-5}$	$6 \cdot 10^{11}$
Fusion par confinement inertiel	10^{28}	10	-	$7 \cdot 10^{-9}$	$6 \cdot 10^{15}$

III.5 Conditions d'existence d'un plasma :

Les trois conditions que doit satisfaire un gaz ionisé pour être un plasma sont les suivantes [10] :

1. La longueur de Debye doit être très inférieure à l'ordre de grandeur des dimensions du système (notée L), pour que les effets d'écran (effets collectifs) puissent être observés :

$$\lambda_D \ll L$$

2. La fréquence plasma doit être plus grande que la fréquence qui résulte des collisions entre les électrons et les particules neutres (elle est notée ν) et ceci pour que les électrons puissent se déplacer pour former un plasma collectif :

$$\omega_p \gg \nu$$

3. Le plasma doit être globalement électriquement neutre, c'est-à-dire le nombre des charges positives doit être égal au nombre des charges négatives, c'est le critère de quasi-neutralité :

$$n_e = n_i$$

III.6 Interactions dans un plasma :

Comment traiter les interactions entre particules dans un plasma, faut-il considérer les interactions de façon binaires ou collectives ?

Le modèle le plus simple est celui de l'interaction *binaire* où seules deux particules entrent en jeu dans la collision, où l'effet sur la trajectoire ne se fait sentir que pendant de très courts instants, en dehors desquels chaque particule retrouve son mouvement libre [8,10].

L'ensemble des particules intervient par contre de manière collective sur le mouvement global. Ce caractère *collectif* est une propriété essentielle des plasmas.

Les constituants d'un plasma sont en interaction incessante les uns avec les autres. Ces interactions peuvent se dérouler entre particules de même espèce (ex : e-e et i-i) ou d'espèces différentes (ex : e-i).

1. Collisions :

Deux ou plusieurs particules peuvent se rapprocher à des distances assez faibles en produisant des collisions avec des trajectoires courbées, ceci se produit avec un échange de quantité de mouvement et d'énergie cinétique. Il existe deux types de collisions :

- *Collisions élastiques* : elles se caractérisent par leur absence de perte d'énergie et de déformation.
- *Collisions inélastiques* : les collisions inélastiques sont les plus fréquentes et surviennent lorsqu'il y'a une perte d'énergie au cours du choc qui se traduit sous la forme d'une dissipation de chaleur, ou une déformation, ou bien une création de nouvelles particules.

2. Réactions nucléaires :

Il existe trois types de réactions nucléaires dans le plasma :

- *La fusion nucléaire* appelée aussi thermonucléaire : Deux ou plusieurs petits noyaux entrent en collision pour former de nouveaux noyaux plus gros (Figure 1.6).
- *La fission nucléaire* : Un gros noyau se divise en noyaux plus petits.
- *La désintégration radioactive* : Un noyau se transforme spontanément en un autre type de noyau.

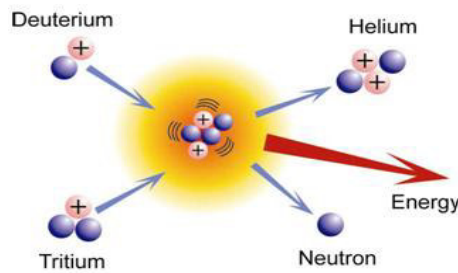


Figure I.6 : schéma représentant la réaction de fusion des atomes de Deutérium et de Tritium [1].

IV. Description d'un plasma :

Pour décrire un système de particules chargées, il existe plusieurs méthodes :

- **La description dite fluide**, qui modélise le plasma à partir des équations de la mécanique des fluides (comme les équations d'Euler ou de Navier-Stokes). Ce type de modélisation [16] est bien adapté pour des pressions relativement grandes, c'est à dire pour des conditions où le libre parcours moyen noté λ (distance entre deux collisions) est petit devant la taille du système (L) c'est-à-dire $\lambda/L \ll 1$. Les inconnues dans ces équations sont des quantités dites macroscopiques (mesurables expérimentalement) comme la densité, la vitesse moyenne ou la pression cinétique du plasma. Cette description suppose que le système étudié est à l'équilibre thermodynamique⁶, caractérisé par la répartition des vitesses des particules de type Maxwellienne.

La description fluide est effectuée à l'aide de grandeurs moyennes, pour chaque espèce de particules en présence : neutres, électrons et ions. On définira ainsi : la densité n_s , la vitesse moyenne v_s et la pression cinétique P_s (s est un indice qui indique : $s = i$ pour l'ion, $s = e$ pour l'électron et $s = n$ pour le neutre).

Les grandeurs physiques : densité, vitesse et pression des constituants du plasma, mesurent pour chaque espèce, l'agitation désordonnée autour de la vitesse moyenne et pourront servir à définir à travers une loi du type gaz parfait, les températures cinétiques T_s . Cette notion de température n'étant en fait bien définie que dans le cas de l'équilibre thermodynamique.

⁶ Equilibre thermodynamique : C'est un état physique idéal vers lequel tend tout plasma dense parfaitement isolé et conservant toutes ses formes d'énergie en absence des gradients de température et de pression. C'est-à-dire que tous les processus microscopiques sont parfaitement contrebalancés par leurs processus inverses ; c'est le principe de micro-réversibilité. De plus, la température des électrons, les températures cinétiques des neutres, et des ions, les températures d'excitation et d'ionisation des espèces et la température du rayonnement de Planck sont égales : $T_e = T_n = T_i = T_{ex} = T_{ion} = T_r = T_{eq}$ [14].

- **La description cinétique** : si la pression est très basse et que le libre parcours moyen devient de l'ordre ou supérieur à la taille du système ($\lambda/L \geq 1$) et lorsque le système est parcouru par une onde de choc ou lorsqu'une population de particules chaudes est présente dans le système, des phénomènes hors équilibre thermodynamique sont à prendre en considération exigeant une description plus précise ; on utilise alors la description cinétique. Une fonction de distribution (définie plus loin dans ce chapitre) dépendant du temps, de l'espace mais aussi de la vitesse des particules, permet de prendre en compte les aspects hors équilibre [16].

Remarque : La complexité de description apportée par le modèle cinétique se traduit numériquement par un coût en temps de calcul et en utilisation de la mémoire d'un ordinateur.

- Entre les deux approches, **une approche hybride** est posée [16]. Cette approche assemble les avantages des descriptions fluide et cinétique. Elle se rapproche des méthodes dites de *décomposition de domaines* pour lesquelles le modèle fluide est utilisé dans les zones où le système est à l'équilibre alors que le modèle cinétique est utilisé uniquement dans les zones où le système est hors équilibre.

Quelle que soit l'approche utilisée, le transport des particules chargées est ensuite couplé aux équations de Maxwell afin de déterminer les profils des champs électromagnétiques.

V. Modélisation Mathématique de la dynamique d'un plasma :

La dynamique d'un plasma est d'une grande complexité, elle peut être décrite en utilisant différentes approximations en fonction de l'aspect physique à analyser [17].

Dans la section suivante, nous allons introduire les grandeurs nécessaires à connaître dans la modélisation par la méthode PIC.

V.1 **Fonction de distribution** :

La description de la dynamique d'un plasma la plus complète est fournie par la théorie cinétique qui permet de déterminer les vitesses microscopiques des particules à partir de la fonction de distribution de chaque espèce. La fonction de distribution f_s [6,10] indique le nombre de particules de l'espèce s par unité de volume dans l'espace des phases (\vec{x}, \vec{v}) .

Un espace de phases est un espace mathématique dans lequel tous les états possibles d'un système sont représentés ; chaque état possible correspond à un point unique de cet espace. Chaque particule est décrite par six variables, trois pour la position et trois pour la vitesse. Si le système est composé de N particules, il est donc décrit par un point dans l'espace des phases à $6N$ dimensions.

Le nombre de particules à l'instant t , contenu dans un petit élément d'hyper-volume ($d\vec{x}, d\vec{v}$) de position (\vec{x}, \vec{v}) est donné par :

$$f_s(\vec{x}, \vec{v}, t) d\vec{x} d\vec{v} \quad \text{I.15}$$

Si le nombre de particules du milieu est très important (s'il y a un grand nombre de particules par élément d'hyper-volume de l'espace des phases), ou si le pas de temps est trop grand pour décrire parfaitement les collisions, l'évolution temporelle de la fonction de distribution est donnée, en fonction de l'importance de l'influence des collisions, en utilisant l'un ou l'autre équation :

- *Equation de Boltzmann,*
- *Equation de Vlasov.*

a. Equation de Boltzmann :

Pour un mélange d'espèces, la fonction de distribution évolue au cours du temps en fonction des forces exercées sur les particules par une influence externe (le modèle peut s'étendre aux forces s'exerçant entre les particules elles-mêmes), de la diffusion des particules liée à leur mouvement dans l'espace, et de leurs éventuelles collisions directes. L'équation générale peut alors s'écrire [10] :

$$\frac{\partial f_s}{\partial t} + \vec{v} \frac{\partial f_s}{\partial \vec{x}} + \frac{\vec{F}_{total}}{m_s} \frac{\partial f_s}{\partial \vec{v}} = C_s \quad \text{I.16}$$

Où C_s représente les collisions des particules de l'espèce s avec toutes les espèces restantes ; m_s est la masse de particule ; t est le temps ; \vec{v} vecteur vitesse de la particule ; \vec{x} vecteur position de la particule ; \vec{F}_{total} est la force exercée sur les particules du plasma, elle est composée de :

- Forces extérieures \vec{F}_{ext} fournies par les champs externes.
- Forces intérieures \vec{F}_{int} fournies par les champs générés par les particules éloignées de la sphère de Debye.

Lorsque les forces entre particules sont répulsives à longue portée (telle que l'interaction coulombienne) les collisions directes sont impossibles; on obtient l'équation de *Boltzmann sans collision*, généralement appelée *équation de Vlasov*.

b. Equation de Vlasov :

L'équation de Vlasov décrit l'évolution temporelle de la fonction de distribution f_s des particules de masse m_s et de charge q_s dans un plasma en négligeant l'effet des collisions, donc C_s est considérée comme égale à 0. On dit que les particules ne sont pas corrélées

lorsqu'il n'y a pas de collisions entre elles. Cette équation a été obtenue par Anatoli Vlasov et s'écrit selon [10] :

$$\frac{df_s}{dt} = \frac{\partial f_s}{\partial t} + \vec{v} \frac{\partial f_s}{\partial \vec{x}} + \left(\frac{\vec{F}_{ext}}{m_s} + \int \frac{\vec{F}_{int}}{m_s} f_s(\vec{x}, \vec{v}, t) d\vec{x} d\vec{v} \right) \cdot \frac{\partial f_s}{\partial \vec{v}} = 0 \quad \text{I.17}$$

La détermination de la fonction de distribution à partir l'équation I.16 ou I.17 permet de calculer la densité de charge ρ et la densité de courant \vec{j} :

$$\rho = \sum_s q_s \int_{-\infty}^{+\infty} f_s d^3v \quad \text{I.18}$$

$$\vec{j} = \sum_s q_s \int_{-\infty}^{+\infty} \vec{v} f_s d^3v \quad \text{I.19}$$

V.2 Densité de Charge ρ :

La densité de charge ρ se définit comme la quantité de charge présente par unité de volume. Son unité de mesure standard est le coulomb par mètre cube (C/m^3) :

$$\rho = \frac{q}{V} \quad \text{I.20}$$

Où V est le volume du milieu ; q est la quantité de charge présente dans ce volume. Par contre la densité des particules n se définit comme le nombre de charges par unité de volume.

Les deux grandeurs (ρ et n) sont reliées par la relation suivante :

$$\rho = e(n_i - n_e) \quad \text{I.21}$$

Où n_i est la densité des ions ; n_e est la densité des électrons.

Si le nombre des électrons subit une légère variation au voisinage de l'ion considéré, alors la densité des électrons satisfera à la relation de Boltzmann suivante [8] :

$$n_e = n_0 \exp - \frac{q_e(\phi - \phi_0)}{k_B T} \quad \text{I.22}$$

Où n_0 est la densité des électrons initiale à $t=0$; ϕ est le potentiel électrostatique à l'instant t .

V.3 Force de Lorentz :

La force externe est modélisée par la force de Lorentz, qui calcule le mouvement d'une particule de l'espèce s de masse m_s , de vitesse \vec{v} et de charge q_s se déplaçant dans un champ électrique \vec{E} et un champ magnétique \vec{B} . Elle est donnée par la relation suivante :

$$\vec{F}_{ext} = q_s[\vec{E} + \vec{v} \wedge \vec{B}] \quad \text{I.23}$$

L'équation du mouvement de la particule (espèce s) relie sa vitesse à la force à travers l'équation I.24 :

$$\vec{F}_{ext} = m_s \frac{d\vec{v}}{dt} \quad \text{I.24}$$

V.4 Champ électrique :

Le champ électrique est le paramètre le plus important, parce qu'il est responsable de la production des décharges électriques (ionisation). La connaissance des valeurs du champ électrique dans chaque point du domaine d'étude nous permet d'avoir des informations sur les phénomènes qui se manifestent dans les décharges électriques. Le champ électrique dérive d'un potentiel scalaire, tel que :

$$\vec{E} = -\nabla\phi \quad \text{I.25}$$

Où ϕ est le potentiel électrostatique gouverné par l'équation de Poisson :

$$\nabla^2\phi = -\frac{\rho}{\epsilon_0} \quad \text{I.26}$$

V.5 Equations de Maxwell :

Les champs électromagnétiques dans un plasma sont modélisés [18] par les équations suivantes :

$$\nabla \cdot \vec{E} = \frac{\rho}{\epsilon_0} \quad (\text{Maxwell-Gauss}) \quad \text{I-27}$$

$$\nabla \cdot \vec{B} = 0 \quad (\text{Maxwell}) \quad \text{I.28}$$

$$\nabla \wedge \vec{E} = -\frac{\partial \vec{B}}{\partial t} \quad (\text{Maxwell-Faraday}) \quad \text{I.29}$$

$$\nabla \wedge \vec{B} = \mu_0 \vec{j} + \frac{1}{c^2} \frac{\partial \vec{E}}{\partial t} \quad (\text{Maxwell-Ampère}) \quad \text{I.30}$$

Où c est la vitesse de la lumière dans le vide, $\mu_0 = 4\pi \times 10^{-7} \text{ kgmA}^{-2}\text{s}^{-2}$ est la perméabilité magnétique du vide, $\vec{j} = \sigma \vec{E}$ est la densité de courant de conduction électrique et σ la conductivité électrique.

VI. Conclusion :

Dans ce chapitre, nous avons présenté des généralités sur la physique des plasmas ainsi que leurs paramètres caractéristiques. Nous avons montré que l'action d'un champ électrique entraînant une perturbation spatiale des électrons est caractérisée par une fréquence plasma. Nous avons aussi introduit les différentes équations que nous allons utiliser dans le chapitre 2, pour déterminer la dynamique des particules.

Chapitre 2 :

Étapes de Simulation par la Méthode PIC

I. Introduction :

La méthode PIC est un outil numérique très puissant pour étudier les plasmas et simuler les trajectoires de particules chargées [19-27]. Macroscopiquement, la dynamique des particules chargées est décrite par les équations de Maxwell, alors que microscopiquement elle est modélisée par un ensemble de particules numériques. Le plasma considéré sera caractérisé par des quantités fluides (telle que la densité) et des quantités cinétiques (qui sont les particules).

Dans ce chapitre nous allons exposer dans les détails, le principe de résolution par la méthode PIC dans le cas électrostatique où on négligera le champ magnétique. Le mouvement des ions sera ignoré (les ions vont constituer une charge statique neutralisant le fond autour du mouvement des électrons) et la méthode consistera donc à suivre les trajectoires d'un nombre représentatif d'électrons dans l'espace des phases, et de décrire l'interaction collective de ces particules en résolvant l'équation de Poisson.

II. Méthodologie de travail :

La méthode PIC consiste en plusieurs étapes, elle se résume comme l'ont indiqué les auteurs C. K. Birdsall et A. B. Langdon dans la réf. [19] :

- Subdivision du domaine d'étude en cellules (constituant ainsi une grille ou maille) ;
- Phase initialisation des vitesses et positions des particules (électrons ou ions) ;
- Calcul densité, potentiel, champ électrostatique sur les points de la grille (nœuds ou sommets) ;
- Interpolation de ces grandeurs pour le calcul de la force et de l'accélération que subit chaque particule ;
- Mise à jour des vitesses et positions de la particule (équation du mouvement) ;
- Toutes ces étapes se répètent dans une boucle de temps pour simuler la dynamique de l'espèce choisie (le temps varie de $t=0$ jusqu'au temps final de la simulation noté t_{fin}).

Nous avons réalisé un code Python pour chaque étape de la résolution ; les parties du code seront données au fur et à mesure. Dans le code Python tout tableau doit commencer par un compteur nul. Nous avons utilisé comme modules python : numpy¹ et matplotlib².

¹ NumPy abrégé de numerical Python est une bibliothèque Python très populaire qui est principalement utilisée pour effectuer des calculs mathématiques et scientifiques. Elle offre de nombreuses fonctionnalités et outils : tableaux, fonctions mathématiques, fonctions de l'analyse matricielle ect [28].

² Matplotlib est une bibliothèque Python open source permettant de créer des visualisations graphiques des données [28].

Nous avons dû reposer certaines équations déjà posées dans le chapitre 1 et ceci pour les besoins explicatifs. Des figures illustratives seront aussi tracées pour chaque phase de la simulation.

III. Subdivision du domaine d'étude :

Pour illustrer la première des étapes qui consiste en la subdivision du domaine, considérons à titre provisoire la figure II.1 qui montre la distribution des particules d'un plasma en 2D.

Le plasma est composé d'un grand nombre de particules donc un grand nombre d'équations aux dérivées partielles à résoudre. La géométrie étudiée devient plus simple lorsque la structure est découpée en un nombre fini de petites cellules formant une grille (voir La figure II.2). L'ensemble de ces éléments constituent le maillage du domaine.

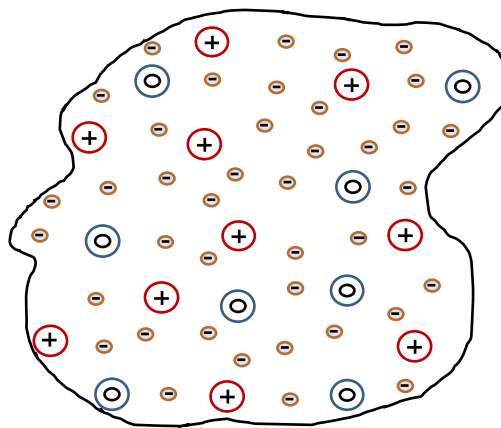


Figure II.1 : Représentation d'un milieu plasma en 2D.

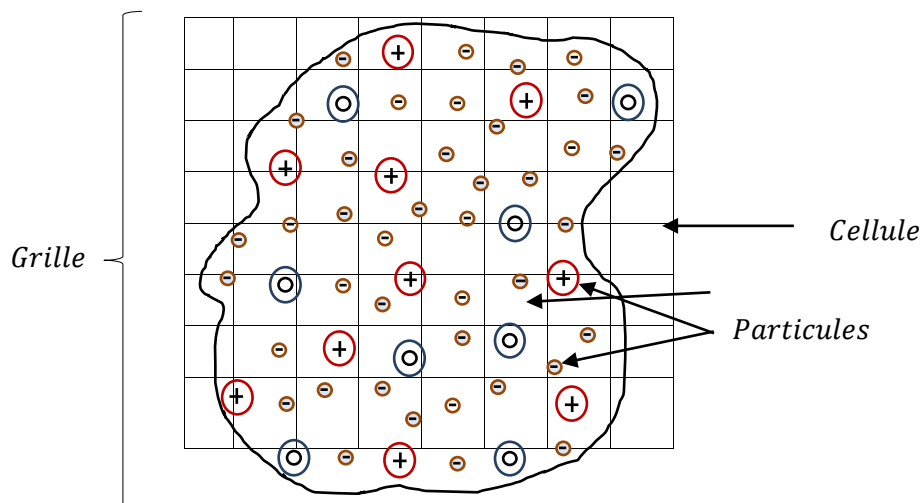


Figure II.2 : Subdivision du milieu 2D en grille de petites cellules.

La figure II.3 montre une cellule bidimensionnelle de la grille avec quatre nœuds (sommets géométriques). L'étape de subdivision (qui se fait numériquement) permet de construire ces cellules en déterminant parfaitement les coordonnées de chaque sommet.

Chaque cellule de la grille va contenir une particule ou bien une macro particule localisée au point de coordonnées (x_p, y_p) .

Le calcul de la densité, du potentiel et du champ électrostatique se fera sur les sommets de la cellule, donc se seront des valeurs discrètes.

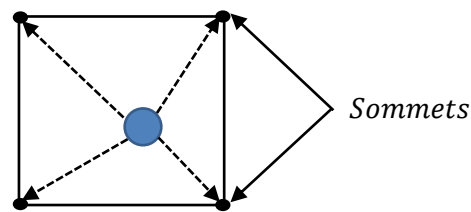


Figure II.3 : Une cellule de la grille en deux dimensions (2d) ; la particule (en couleur bleue) est située à l'intérieur de la cellule.

La force exercée sur la particule située à la position (x_p, y_p) , le champ et du potentiel électrostatiques seront tous calculés à cette position. Cette étape se fera grâce à l'interpolation. Ces petites surfaces en couleur, illustrées dans la figure II.4 serviront comme polynômes d'interpolation en 2D (voir annexe A.1 : [fonctions de Pondération](#)).

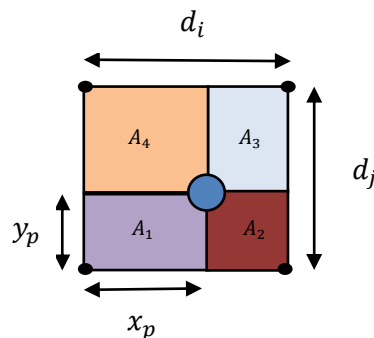


Figure II.4 : Schéma représentant les petites surfaces pour servir à l'interpolation en 2d.

Dans la section suivante, nous allons donner le développement de la méthode PIC pour le cas unidimensionnel (PIC 1D) qui signifie que la densité de charges subit des variations selon une direction unique que nous avons choisi comme étant l'axe (ox).

IV. Modèle électrostatique unidimensionnel :

Le mouvement de chaque électron de charge $q = -e$ et de masse m , le long de la direction (ox) est gouverné par les lois de Newton couplées à l'interaction du champ électrostatique $E(x)$. Ces équations sont établies pour chaque particule en mouvement :

$$a = \frac{dv}{dt} = \frac{q}{m} E(x) \quad \text{II.1}$$

$$v = \frac{dx}{dt} \quad \text{II.2}$$

Où, a est l'accélération des charges ; x et v sont les positions et vitesses des particules, respectivement.

Le champ électrique dans ce modèle électrostatique dérive du potentiel électrique $\phi(x)$ comme :

$$E(x) = - \frac{d\phi(x)}{dx} \quad \text{II.3}$$

Le potentiel électrique est relié de façon générale à la charge à travers l'équation de Poisson-Maxwell :

$$\frac{d^2\phi(x)}{dx^2} = - \frac{\rho}{\epsilon_0} \quad \text{II.4}$$

Sachant que $\rho = -e(n_e - n_i)$, en posant $n(x) = n_e$ et $n_0 = n_i$ alors l'équation II.4 devient comme suit :

$$\frac{d^2\phi(x)}{dx^2} = \frac{e}{\epsilon_0}(n(x) - n_0) \quad \text{II.5}$$

Avec $n(x)$ est la densité des électrons ; n_0 est la densité des ions ; $n(x)dx$ est le nombre de charges dans l'intervalle $(x; x + dx)$; la valeur moyenne de $n(x)$ est égale à n_0 .

L'énergie totale pour chaque particule est la somme des deux énergies cinétique et potentiel :

$$U = \frac{1}{2}mv^2 - e\phi \quad \text{II.6}$$

Les isovaleurs des énergies totales fournissent les orbites des électrons dans l'espace des phases.

VI.1 Normalisation des équations :

Il est nécessaire de normaliser les équations précédentes. Cette étape est très importante pour simplifier les calculs et aussi ne pas encombrer la taille mémoire du PC.

La normalisation [23,24] se fait par rapport aux valeurs caractéristiques suivantes :

- Pour le temps, l'échelle caractéristique est la fréquence plasma $\omega_p = \sqrt{\frac{ne^2}{m_e \epsilon_0}}$
- Pour les vitesses, l'échelle caractéristique est la vitesse thermique $v_{th} = \sqrt{\frac{k_B T_e}{m_e}}$
- Pour les longueurs, l'échelle caractéristique est la longueur de Debye $\lambda_D = \frac{v_{th}}{\omega_p}$

Pour que les effets collectifs (ondes et oscillations) puissent apparaître dans un plasma, il faut que le vecteur d'onde du champ électrique k soit inférieur à $1/\lambda_D$.

Grâce à ces échelles caractéristiques, normaliser les équations II.1 à II.5 qui gouvernent la dynamique des particules chargées dans un plasma se fait comme suit :

Posant les relations suivantes :

$$\bar{v} = \frac{v}{v_c}; \quad \bar{x} = \frac{x}{x_c}; \quad \bar{t} = \frac{t}{t_c}; \quad \bar{\phi} = \frac{\phi}{\phi_c}; \quad \bar{\rho} = \frac{\rho}{\rho_c}; \quad \bar{E} = \frac{E}{E_c}; \quad \bar{U} = \frac{U}{U_c}; \quad \bar{n} = \frac{n}{n_0} \quad \text{II.7}$$

où les grandeurs caractéristiques (identifiées par l'indice c) choisies pour la normalisation sont :

$$\begin{aligned} v_c = v_{th} &= \sqrt{\frac{k_B T_e}{m_e}} & x_c &= \lambda_D & t_c &= \omega_p^{-1} \\ \phi_c &= \frac{k_B T_e}{e} & \rho_c &= n_0 e & E_c &= \sqrt{\frac{n_0 k_B T_e}{\epsilon_0}} & U_c &= k_B T_e \end{aligned} \quad \text{II.8}$$

Et remplaçant les formules II.7 et II.8 dans les équations II.1 à II.5 (voir développement dans l'annexe A.2 : Normalisation des équations), nous obtenons finalement les équations à résoudre par la méthode PIC 1D :

$$\bar{v} = \frac{d\bar{x}}{d\bar{t}} \quad \text{(II.9 a)}$$

$$\frac{d\bar{v}}{d\bar{t}} = -\bar{E}(x) \quad \text{(II.9 b)}$$

$$\bar{E}(x) = -\frac{d\bar{\phi}}{d\bar{x}} \quad \text{(II.9 c)}$$

$$\frac{d^2\bar{\phi}}{d\bar{x}^2} = \bar{n} - 1 \quad \text{(II.9 d)}$$

$$\bar{U} = \frac{1}{2} \bar{v}^2 - \bar{\phi} \quad (\text{II.9 e})$$

Remarque importante: La notation barre sur les grandeurs à calculer par la méthode PIC sera ignorée dans la section suivante pour ne pas alourdir le développement mathématique.

VI.2. Subdivision du domaine d'étude par la méthode PIC 1D :

Soit L la longueur du domaine de computation définissant le plasma, N le nombre total de charges (électrons). L'axe (ox) est subdivisé en N_c cellules avec un pas $\Delta x = L/N_c$ (voir la Figure II.5). Le nombre de points de la grille est noté $N_g = N_c + 1$.

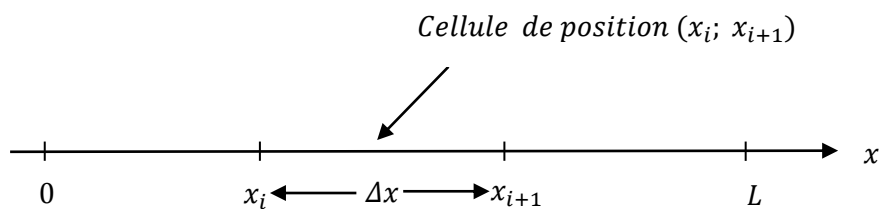


Figure II.5 : Subdivision de la grille spatiale.

La particule à simuler (qui est en mouvement) est localisée dans la cellule (x_i, x_{i+1}) avec une position $x = x_p$ qui est la distance d_1 par rapport à x_i et d_2 par rapport à x_{i+1} (voir la Figure II.6).

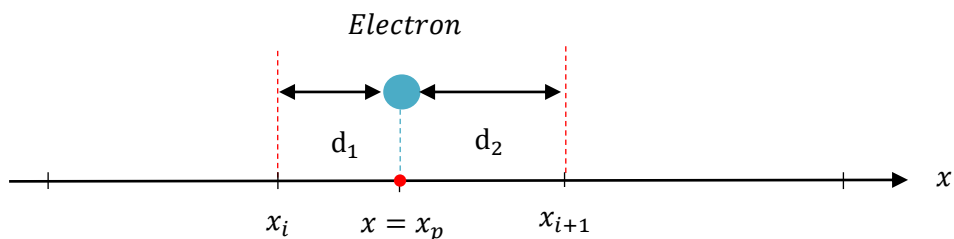


Figure II.6 : la position de la particule (en bleu) dans une cellule de la grille.

$$d_1 = x_p - x_i \text{ et } d_2 = x_{i+1} - x_p$$

La particule avec masse et charge peut voyager à travers les lignes de la grille, cependant le potentiel, le champ électrique et la densité se calculent uniquement aux points de la grille.

```
# Part 1 : Implémentation python :
# calcul de la longueur de la cellule de la grille : dx
dx = L/(Ng-1)
# j nombre entier qui donne le numéro de la cellule où se trouve la particule
j= np.floor(xp/dx).astype(int)
```

V. Schémas numériques de résolution :

V.1. Calcul du potentiel électrostatique $\phi(x)$:

Pour déterminer la valeur du potentiel électrique dans la cellule, il faut résoudre l'équation de Poisson II.9 d (équation normalisée) :

$$\frac{d^2\phi(x)}{dx^2} = n(x) - 1 \quad \text{II.10}$$

L'équation II.10 est composée de deux termes, le potentiel $\phi(x)$ et la densité $n(x)$. Nous allons commencer par le premier membre de l'équation et discrétiser l'opérateur $\frac{d^2\phi(x)}{dx^2}$.

Le potentiel se calcule uniquement sur les points de la grille de la subdivision (voir la Figure II.7) :

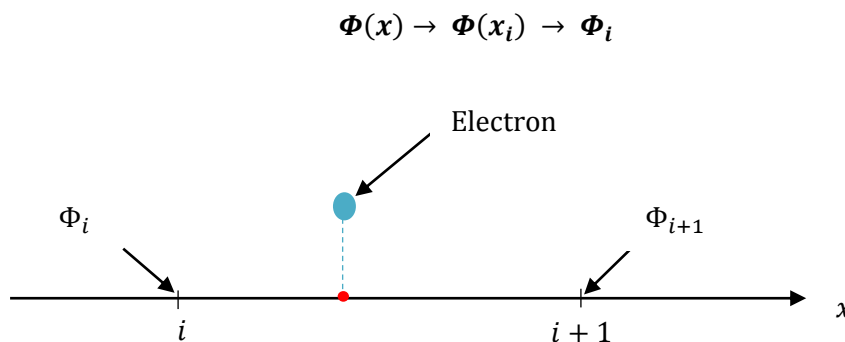


Figure II.7 : Calcul du potentiel sur les points de la grille.

Nous allons montrer comment calculer numériquement le potentiel ϕ_i sur chaque point de la grille, d'indice i . L'opérateur dérivée seconde (appelé « Laplacien ») dans l'équation de Poisson peut être approximé sur les nœuds du maillage en utilisant la méthode des différences finies (travail effectué dans l'annexe A.3 : schémas aux différences finies) :

$$\frac{d^2\phi_i}{dx^2} = \frac{\phi_{i-1} - 2\phi_i + \phi_{i+1}}{\Delta x^2} \quad i = 0, Ng - 1 \quad \text{II.11}$$

Comme illustration, l'équation II.11 donne pour i variant de 0 à 2, les équations suivantes :

$$\begin{cases} \text{pour } i = 0 & (\phi_{-1} - 2\phi_0 + \phi_1)/\Delta x^2 \\ \text{pour } i = 1 & (\phi_0 - 2\phi_1 + \phi_2)/\Delta x^2 \\ \text{pour } i = 2 & (\phi_1 - 2\phi_2 + \phi_3)/\Delta x^2 \end{cases}$$

Le développement de l'équation II.11 sur toutes les valeurs de compteur i aboutit au système suivant :

$$\frac{1}{\Delta x^2} \begin{pmatrix} -2 & 1 & 0 & 0 & 0 & \dots \\ 1 & -2 & 1 & 0 & 0 & \dots \\ 0 & 1 & -2 & 1 & 0 & \dots \\ & & \vdots & & & \\ \dots & 0 & 0 & 1 & -2 & \dots \end{pmatrix} \begin{bmatrix} \phi_0 \\ \phi_1 \\ \phi_2 \\ \vdots \\ \phi_{N_g-1} \end{bmatrix}$$

avec ϕ_{-1} et ϕ_{N_g} hors compteur i , elles sont donc considérées comme nulles.

```
# Part 2 : Implémentation python : introduction de la matrice Mat_Phi
def matrice_Pot(L, Ng, Mat_Phi):
    dx = L/(Ng-1)
    for i in range(Ng):
        for j in range(Ng):
            if i==j:
                Mat_Phi[i, j]=-2
            elif abs(i-j)==1:
                Mat_Phi[i, j]=1
    Mat_Phi /= (dx**2)
    return
```

V.2. Calcul de la densité des charges :

Nous allons montrer maintenant les étapes de calcul du second membre de l'équation II.10 à savoir la densité $n(x)$ (qui se calcule uniquement sur les points de la grille).

La densité créée par la présence d'une particule ou une macro particule dans chaque cellule est donnée sur les deux points de la cellule est trouvée à partir d'une pondération (voir annexe A.1 : [Fonction de Pondération](#)) comme suit [26] :

$$\begin{cases} n_i = n_i + \frac{x_{i+1} - x_p}{\Delta x} \\ n_{i+1} = n_{i+1} + \frac{x_p - x_i}{\Delta x} \end{cases} \quad i = 0, N_g - 1 \quad \text{II.12}$$

$w_1 = \frac{x_{i+1} - x_p}{\Delta x}$ et $w_2 = \frac{x_p - x_i}{\Delta x}$ sont appelés coefficients de pondération (weighting factors) ;
 $(n_0, n_1, n_2, \dots, n_{N_g-1})$ sont les valeurs de la densité de charge aux points de la grille.

Part 3: Implémentation python: calcul du vecteur densité n sur les points de la grille

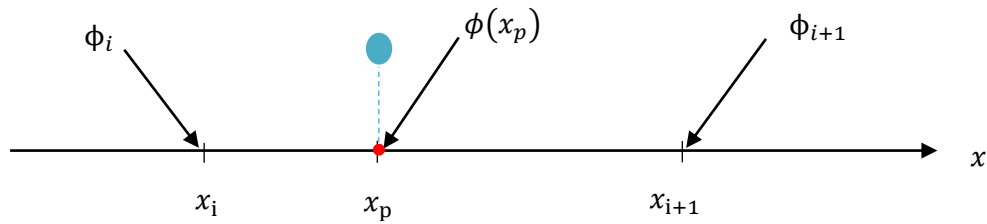
```
def vect_density(xp, N, Ng, L):
    n=np.zeros(Ng)
    dx = L/(Ng-1)
    j= np.floor(xp/dx).astype(int)
    jp=j+1
    w1=(jp*dx-xp)/dx
    w2=(xp-j*dx)/dx
    jp= np.mod(jp, Ng)
    n= np.bincount(j[:,], weights=w1[:,], minlength=Ng)
    n += np.bincount(jp[:,], weights=w2[:,], minlength=Ng)
    n *= L / N / dx
    return (n)
```

Les valeurs de la densité n_i sont arrangées sous forme d'un vecteur colonne de même dimension que le vecteur ϕ_i , donc l'équation II.10 se transforme en une forme matricielle linéaire de type $A\phi = b$

$$\frac{1}{\Delta x^2} \begin{pmatrix} -2 & 1 & 0 & 0 & 0 \dots \\ 1 & -2 & 1 & 0 & 0 \dots \\ 0 & 1 & -2 & 1 & 0 \dots \\ & & \vdots & & \\ \dots & 0 & 0 & 0 & 1 & -2 \end{pmatrix} \begin{bmatrix} \phi_0 \\ \phi_1 \\ \phi_2 \\ \vdots \\ \phi_{N_g-1} \end{bmatrix} = \begin{bmatrix} n_0 \\ n_1 \\ n_2 \\ \vdots \\ n_{N_g-1} \end{bmatrix} - 1 \quad \text{II.13}$$

La résolution du système matriciel donnée dans l'équation II.13 a été faite par inversion de matrice telle que $\phi = A^{-1}b$. Le résultat sont les valeurs du potentiel électrostatique en tout point de la grille $(\phi_0, \phi_1, \phi_2, \dots, \phi_{N_g-1})$. Maintenant, à partir de ces valeurs, on peut déduire la valeur du potentiel ϕ à la position où est située la particule comme indiqué sur la Figure II.8.

Il est facile de montrer qu'à partir de l'interpolation de Lagrange (voir annexe A.4 : Interpolation de Lagrange) pour deux points $(x_i$ et $x_{i+1})$, le potentiel électrostatique à la position x_p où se situe la particule, s'exprime comme donné dans l'équation II.14 :

Figure II.8 : Calcul du potentiel électrique à la position x_p .

$$\phi(x = x_p) = \frac{x_{i+1} - x_p}{\Delta x} \phi_i + \frac{x_p - x_i}{\Delta x} \phi_{i+1}$$

II.14

Avec : $\Delta x = x_{i+1} - x_i$ est la longueur de la cellule.

```
# Part 5: Implémentation python : calcul du potentiel électrique au point xp
def vecteur_pot_elec(xp, Ng, L, phi_grid):
    dx = L/(Ng-1)
    j= np.floor(xp/dx).astype(int)
    jp=j+1
    w1=(jp*dx-xp)/dx
    w2=(xp-j*dx)/dx
    jp= np.mod(jp, Ng) #condition BC periodic
    Phi=w1* phi_grid[j] + w2 * phi_grid[jp]
    return (Phi)
```

V.3. Calcul du champ électrique :

Tout d'abord il faut calculer la valeur de champ électrique E_i sur les points de la grille, en utilisant l'équation II.9.c :

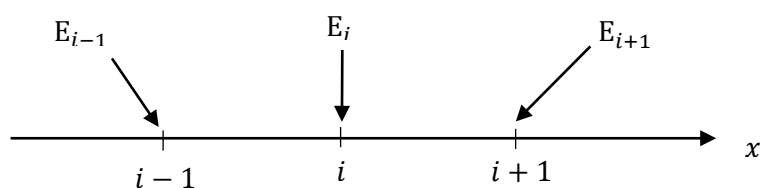


Figure II.9 : Calcul du champ électrique sur les points de la grille.

L'équation II.9.c peut-être approximée sur le maillage par une formule des différences finies :

$$E_i = -\frac{\phi_{i+1} - \phi_{i-1}}{2\Delta x} \quad i = 0, N_{g-1} \quad \text{II.15}$$

Comme illustration, on varie l'indice i de 0 à 2 dans l'équation II.15, ce qui donne :

$$\begin{cases} \text{pour } i = 0 & E_0 = -(\phi_1 - \phi_{-1})/2\Delta x \\ \text{pour } i = 1 & E_1 = -(\phi_2 - \phi_0)/2\Delta x \\ \text{pour } i = 2 & E_2 = -(\phi_3 - \phi_1)/2\Delta x \end{cases}$$

Le développement de l'équation II.15 pour toutes les valeurs du compteur i aboutit au système matriciel sous la forme $E = B\phi$ trouvé comme suit :

$$\begin{pmatrix} E_0 \\ E_1 \\ E_2 \\ \vdots \\ E_{N_{g-1}} \end{pmatrix} = -\frac{1}{2\Delta x} \begin{pmatrix} 0 & 1 & 0 & 0 & 0 & \dots \\ -1 & 0 & 1 & 0 & 0 & \\ 0 & -1 & 0 & 1 & 0 & \\ & & \vdots & & & \\ 0 & 0 & 0 & -1 & 0 & \end{pmatrix} \begin{pmatrix} \phi_0 \\ \phi_1 \\ \phi_2 \\ \vdots \\ \phi_{N_{g-1}} \end{pmatrix} \quad \text{II.16}$$

```
# Part 6 : Implémentation python : Calcul de la Matrice B
```

```
def matrice_Electric(L, Ng, Mat_champ):
```

```
    dx = L/(Ng-1)
```

```
    for i in range(Ng):
```

```
        for j in range(Ng):
```

```
            if (i==j):
```

```
                Mat_champ[i, j]=0
```

```
            elif (i-j)==1:
```

```
                Mat_champ[i, j]=-1
```

```
            elif (j-i)==1:
```

```
                Mat_champ[i, j]=1
```

```
    Mat_champ /= (2*dx)
```

```
    return
```

```
# Part 7 : Implémentation python : calcul du champ E_grid aux points de la grille
```

```
E_grid = - np.matmul(Mat_champ, phi_grid)
```

V.4. Calcul de l'accélération :

Pour calculer la force exercée sur chaque particule située au point $x = x_p$ et son accélération, il faut connaître l'expression du champ électrostatique à la même position x_p .

Nous avons déjà montré comment déterminer les valeurs du champ sur la grille :

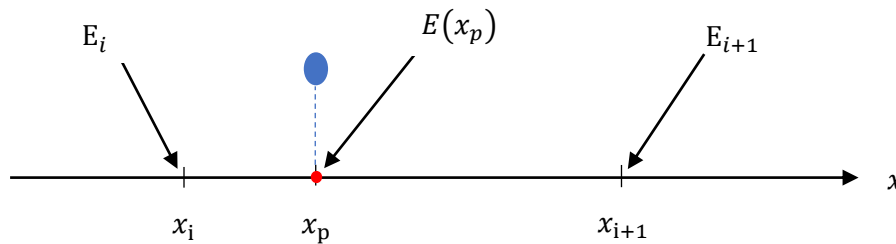


Figure II.10 : Calcul du champ électrique à la position x_p .

En utilisant toujours l'interpolation de Lagrange (voir annexe A.4) pour deux points $(x_i; x_{i+1})$, le champ électrostatique à la position où se trouve la particule (x_p) est trouvé :

$$E(x = x_p) = \frac{x_{i+1} - x_p}{\Delta x} E_i + \frac{x_p - x_i}{\Delta x} E_{i+1} \quad \text{II. 17}$$

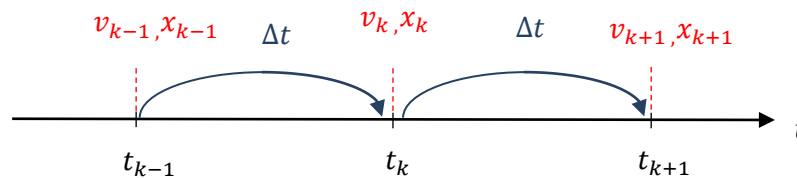
```
# Part 8 : Implémentation python : calcul du champ et de l' accélération Acc
def vecteur_champ_elec(xp, Ng, L, E_grid):
# interpolation champ Electrique
    n=np.zeros(Ng)
    dx = L/(Ng-1)
    j= np.floor(xp/dx).astype(int)
    jp=j+1
# polynome interpolation
    w1=(jp*dx-xp)/dx
    w2=(xp-j*dx)/dx
    jp= np.mod(jp, Ng) #condition BC periodic
    Ex=w1* E_grid[j] + w2 * E_grid[jp] #interpolation
    Acc=-Ex #acceleration électron
    return (Acc)
```

V.5. Pushing particle : Calcul des nouvelles positions et vitesses des particules :

Connaissant l'accélération des particules notée Acc (qui est proportionnelle au champ électrostatique), il faut mettre à jour les nouvelles positions et vitesses pour déplacer les charges de leur état initial (cette phase est appelée pushing particle). Pour cela, il faut résoudre numériquement les équations du mouvement (équations II.9 a et II.9 b) qui se présentent sous la forme d'équations différentielles ordinaires et il existe plusieurs méthodes qu'on peut utiliser [29], parmi elles on cite :

1. Schéma d'Euler :

La méthode d'Euler [29] est une méthode très simple. Elle calcule les nouvelles positions et vitesses au même pas temporel $t + \Delta t$ en fonction des valeurs au temps t . Elle est facile à



implémenter mais elle manque de précision.

Figure II.11 : Schéma d'Euler explicite.

Les positions et vitesses des particules au temps futur $t_{k+1} = t + \Delta t$ se calculent en fonction des valeurs au temps $t_k = t$:

```
# Part 9 : Implémentation python : Equations du mouvement
```

```
v=v+Acc*dt
```

```
xp =xp+ v * dt
```

```
xp = np.mod(xp, L) # condition pour garder la particule entre 0,L
```

$$v^{k+1} = v^k + Acc^k \Delta t$$

$$x^{k+1} = x^k + v^k \Delta t$$

$$k = 0, t_{fin}/\Delta t$$

II.18

2. Schéma Leap-Frog :

Le schéma de Leap-Frog appelé aussi Saute-Mouton [29] est souvent utilisé dans la simulation PIC ; il est plus précis que le schéma d'Euler. Cette méthode permet le calcul des vitesses et des positions sur deux échelles de temps différentes : $t + \Delta t/2$ et $t + \Delta t$.

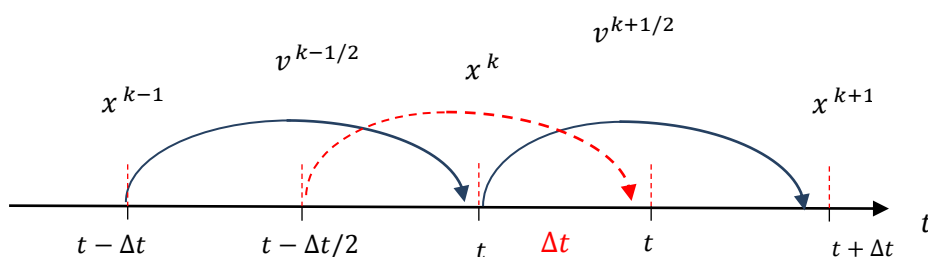


Figure II.12 : schéma Leap-Frog ou Saute-Mouton.

Figure II.12 : Schéma de Leap-Frog ou Saute-Mouton.

Les nouvelles vitesses et positions selon le schéma Leap-Frog se calculent comme suit :

$$\begin{aligned} v^{k+1/2} &= v^{k-1/2} + Acc^k \Delta t/2 \\ x^{k+1} &= x^k + v^{k+1/2} \Delta t \end{aligned} \quad \text{II.19}$$

```
# Part 9 : Implémentation python : Equations du mouvement
v=v+Acc*dt/2.
xp =xp+ v * dt
xp = np.mod(xp, L)      # condition pour garder la particule entre 0,L
# update densité, potentiel et champ
n=vect_density(xp, N, Ng, L)
Inv=np.linalg.inv(Mat_Phi)
phi_grid =np.dot(Inv, n-1)
matrice_Electric(L, Ng, Mat_champ)
E_grid = -np.matmul(Mat_champ, phi_grid)
Acc= vecteur_champ_elec(xp, Ng, L, E_grid)
Phix=vecteur_Potent_elec(xp, Ng, L, phi_grid)
v+= Acc * dt/2.
# update time
t += dt
```

3. Schéma de Runge-Kutta :

Les méthodes Runge-Kutta (RK) sont des méthodes très précises pour la résolution des équations différentielles ordinaires d'ordre 1 [29]. Elles calculent aussi les nouvelles positions et vitesses au même pas de temps Δt , mais avec plusieurs points intermédiaires. Elles sont stables et précises.

Dans notre travail, nous avons utilisé la méthode de Runge-Kutta d'ordre 4 avec quatre points intermédiaires. Connaissant les valeurs initiales des vitesses alors les nouvelles positions (pour chaque particule) s'écrivent comme suit :

$$x^{k+1} = x^k + \frac{\Delta t}{6} (L_1 + 2L_2 + 2L_3 + L_4) \quad k = 0, t_{fin}/\Delta t \quad \text{II. 20. a}$$

avec :

$$\begin{cases} L_1 = v^k \\ L_2 = v^k + \frac{\Delta t}{2} L_1 \\ L_3 = v^k + \frac{\Delta t}{2} L_2 \\ L_4 = v^k + \Delta t L_3 \end{cases}$$

Les nouvelles vitesses sont données comme suit :

$$v_{k+1} = v_k + \frac{\Delta t}{6} (K_1 + 2K_2 + 2K_3 + K_4) \quad k = 0, t_{fin}/\Delta t \quad \text{II. 20. b}$$

avec :

$$\begin{cases} K_1 = Acc^k \\ K_2 = Acc^k + \frac{\Delta t}{2} K_1 \\ K_3 = Acc^k + \frac{\Delta t}{2} K_2 \\ K_4 = Acc^k + \Delta t K_3 \end{cases}$$

Remarque : l'indice k représente le compteur pour le temps t et l'indice $k+1$ représente le

```
# Part 9 : Implémentation python : Equations du mouvement schémas RK4
L1=v
L2=v+(dt/2)*L1
L3=v+(dt/2)*L2
L4=v+dt*L3
xp =xp+1/6*(L1+2*L2+2*L3+L4)*dt
xp = np.mod(xp, L) # condition pour garder la particule entre 0,L
# Calcul densité, potentiel et champ
n=vect_density(xp, N, Ng, L)
Inv=np.linalg.inv(Mat_Phi)
phi_grid =np.dot(Inv, n-1)
matrice_Electric(L, Ng, Mat_champ)
E_grid = -np.matmul(Mat_champ, phi_grid)
Acc= vecteur_champ_elec(xp, Ng, L, E_grid)
Phix=vecteur_Potent_elec(xp, Ng, L, phi_grid)
# nouvelles vitesses
K1=Acc
K2=Acc+(dt/2)*K1
K3=Acc+(dt/2)*K2
K4=Acc+dt*K3
v=v+1/6*(K1+2*K2+2*K3+K4)*dt
```

compteur pour le temps $t+\Delta t$. Les grandeurs Acc^k, v^k, x_p^k sont l'accélération, vitesse et position des particules au temps t , respectivement.

VI. Choix des paramètres initiaux :

Pour simuler la dynamique des particules (électron, trou ou ion), il faut introduire pour chaque particule (ou macro-particule) les informations suivantes :

- Sa position initiale et par conséquent sur la densité des charges.
- Sa vitesse initiale et par conséquent sur la densité des charges.

VI.1. Positions initiales :

Il est possible de distribuer initialement les particules d'une :

- Manière homogène.
- Manière aléatoire avec la probabilité de trouver une particule dans un petit volume $\Delta x \Delta v$ de l'espace des phases.

Cette position sera notée x_0 . En python :

$x_0 = np.linspace(0, L, N)$, dans le cas où nous choisissons une distribution spatiale initiale uniforme.

$x_0 = np.random.rand(N) * (L)$, dans le cas où nous choisissons une distribution spatiale initiale aléatoire uniforme.

A cette valeur sera ajouté un terme défini comme une perturbation provoquée par l'application initiale d'un champ électrique sur le plasma (source de l'onde dans le plasma).

Le terme électrique dans le plasma est composé de deux champs :

- Champ qui rend compte des interactions à courtes distances entre la particule à simuler et les autres charges (électrons et ions). Cette composante est importante dans le cas collisionnel. Dans ce travail on suppose que le plasma est très dilué, par conséquent ce champ est négligé.
- Champ appelé mésoscopique, qui est produit à la fois par la source de l'onde à l'extérieure du plasma et par les particules en mouvement (électrons et ions) dans le plasma de façon collective.

C'est le champ électrique associé à l'onde qui se propage dans le plasma, il s'écrit :

$$\vec{E} = E_0 \vec{u}_z e^{i(\omega t - kx)}$$

Où k est le vecteur d'onde ; ω est la pulsation (fréquence plasma) ; \vec{u}_z est le vecteur unitaire de l'axe (Oz) et représente la direction de polarisation de l'onde.

Cette onde se manifeste elle-même comme une perturbation dans la densité des charges, elle peut être représentée numériquement par le déplacement de la position initiale des particules, comme l'a montré Birdsall et Langdon [19]. *Le déplacement initial* est donné par :

$$\delta x = A \cos\left(\frac{2\pi}{L} x_0\right) \quad \text{II. 21}$$

Avec : A est l'amplitude de la perturbation initiale ; x_0 est la position des particules avant la perturbation.

Finally, la position initiale de la particule est :

$$x_p = x_0 \pm \delta x \quad \text{II. 22}$$

Remarque : en fait si un champ est appliqué initialement sur un plasma, toutes les grandeurs physiques seront perturbées : densité, charge et potentiel.

VI.2. Vitesses initiales :

En général, il n'est pas facile de connaître la fonction de distribution initiale des particules $f_s(x, v, t = 0)$, cependant les vitesses initiales sont distribuées selon une fonction de distribution la plus proche possible d'une distribution Maxwellienne, où les particules se déplacent librement sans interagir les unes avec les autres.

- Pour un plasma froid, le taux d'ionisation observé étant faible, les collisions électron-électron sont rares et il en résulte généralement que la fonction de distribution des électrons s'écarte significativement d'une distribution d'équilibre Maxwellienne (les ions sont supposés intervenir seulement dans les réactions chimiques).

Un plasma froid spatialement uniforme est le plus simple ; les particules sont espacées uniformément avec des vitesses initiales nulles [19].

Pour créer un faisceau froid (cold stream ou flux), on ajoute une vitesse supplémentaire de dérive notée v_b (beam velocity) à chaque particule [19,24].

- Pour un plasma chaud, pour chaque espèce, les vitesses résultent de la superposition de (a) une dérive simple, (b) quasi-maxwellienne aléatoire et (c) non aléatoire à partir de n'importe quelle distribution.

Rappelons que l'expression d'une fonction de distribution Maxwellienne à $t=0$ [25] est :

$$f_s(x, v, t = 0) = \frac{n_0}{2} \left(\frac{1}{\sqrt{2\pi}v_{th}} e^{-\frac{(v-v_b)^2}{2v_{th}^2}} \right) \quad \text{II. 23}$$

```
# Part 10 : Implémentation python : Initialisation des positions et des vitesses
# positions initiales aléatoires
x0=np.random.rand(N)*(L)
x=A*np.cos((2*np.pi/L)*x0)
xp=x0+x

#vitesses initiales selon la distribution Gaussienne (Maxwellienne)
vb = 3          # vitesse de dérive
vth = 1         # Vitesse thermique
v = vth * np.random.randn(N) + vb
```

Avec, n_0 la densité des électrons initiale ; v est la vitesse de la particule.

VII. Paramètres de stabilité et de précision :

Dans le modèle PIC 1d électrostatique, afin d'assurer la stabilité numérique et la précision, la longueur de la cellule Δx et le pas de temps Δt doivent être déterminés en fonction de la longueur de Debye λ_D et de la fréquence plasma ω_p [24] :

$$\begin{cases} \Delta x \leq \lambda_D = \sqrt{\frac{\epsilon_0 T_e}{e n_e}} \\ \Delta t^{-1} \leq \omega_p = \sqrt{\frac{e^2 n_e}{\epsilon_0 m_e}} \end{cases} \quad \text{II. 24}$$

Remarque : La masse des électrons est beaucoup plus petite que celle des ions et dans le référentiel des ions, le déplacement des électrons semble instantané. C'est pourquoi, pour la résolution du mouvement des particules, les temps caractéristiques de la simulation sont définis à partir de la fréquence électronique ω_p qui représente l'échelle du temps dans le plasma [25].

VIII. Conditions périodiques :

Dans notre code Python, nous avons appliqué des *conditions aux limites périodiques* grâce à l'utilisation de la fonction Python `np.mod(xp, L)`.

Avec une telle hypothèse toute particule qui s'échappe d'une paroi retourne au domaine par la paroi opposée. Ce type de conditions aux limites maintient le nombre des particules dans le domaine et n'introduit aucune perte de quantité de mouvement, ni d'énergie [27].

IX. Algorithme PIC :

Finalement, nous pouvons résumer les étapes de la simulation réalisée selon deux algorithmes :

- Le premier présente l'état d'initialisation des données nécessaires pour la simulation, les positions et vitesses initiales des électrons.
- Le deuxième présente les différentes étapes de résolution numérique des équations qui gouvernent la dynamique de particules chargées dans un plasma ; Ce processus sera répété dans une boucle de temps.

L'algorithme 1 : Phase d'initialisation

1. Initialisation des positions et des vitesses des particules
 2. Introduction des données : $L, N, N_g, n_0, t_{fin}, \Delta t$
 3. Construction de la grille spatiale
 4. $t=0$ (initialisation du temps t)
-

L'algorithme 2 : Schéma PIC 1d Electrostatique

1. Boucle sur le temps : tant que $t < t_{fin}$
 2. Calculer la densité des charges n sur les points de la grille (Part 3)
 3. Calculer le potentiel électrique ϕ sur les points de la grille (Part 2+ Part 4)
 4. Calculer le champ électrique E sur les points de la grille (Part 6+ Part 7)
 5. Interpolation du champ électrique sur la position de la particule chargée (Part 8)
 6. On déduit la force électrique appliquée sur la particule et son accélération
 7. Mettre à jour les nouvelles positions et vitesses des particules (Part 9)
 8. $t = t + \Delta t$ (incrémentaire temporel)
 9. Fin tant que (retour à 2)
-

Ces deux algorithmes peuvent être donnés différemment par l'organigramme sur la figure II.13.

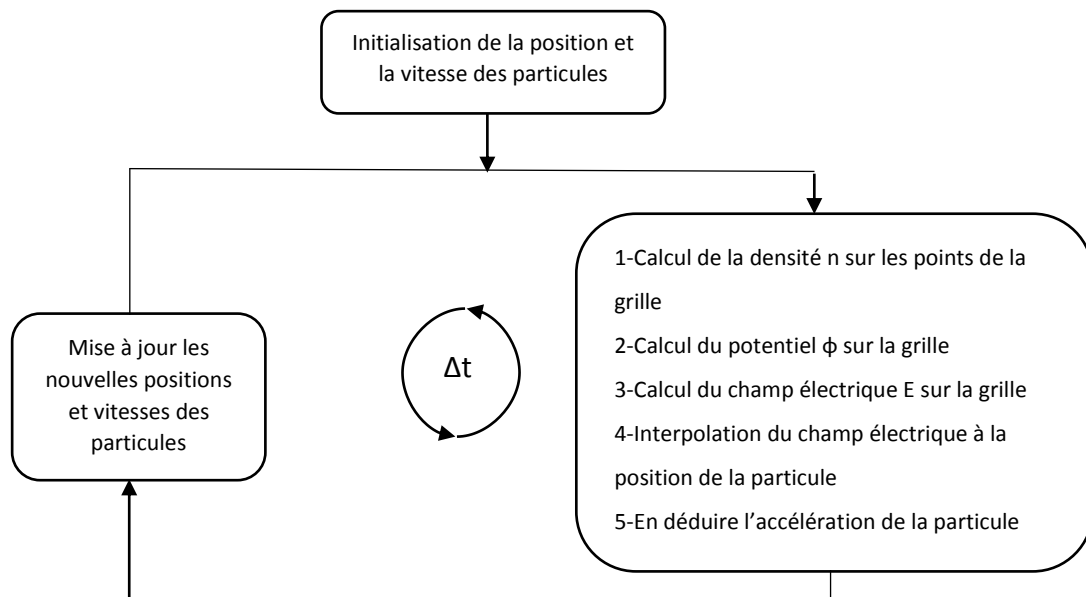


Figure II.13 : Schéma représentant les étapes de simulation PIC.

X. Conclusion :

Dans ce chapitre, nous avons présenté en détails toutes les étapes de la simulation de la dynamique des particules chargées dans un plasma soumis initialement à un champ électrique, par la méthode PIC unidimensionnel en résolvant les équations de Maxwell couplées par les lois de la dynamique de Newton. Nous avons donné les schémas numériques et le code Python de chaque partie réalisée.

Les valeurs calculées avec le code sont toutes normalisées, et pour revenir aux valeurs réelles, il suffit d'utiliser les relations II.7.

Dans le chapitre suivant, nous allons présenter les différents résultats obtenus grâce à la méthode PIC 1D.

Chapitre 3 :

Résultats de la Simulation PIC 1D

I. Introduction :

Dans ce chapitre, nous allons présenter les différents résultats de la simulation de la dynamique des particules chargées (les électrons) pour différents types de plasma, en utilisant le code Python PIC 1D électrostatique que nous avons conçu.

- 1- Distribution des positions et des vitesses initiales des particules et notion de macro particule.
- 2- Diagramme de phase pour plasma froid
- 3- Diagramme de phase pour plasma chaud
- 4- Instabilité de deux flux d'électrons (two stream instability)
- 5- Variation du potentiel électrostatique en fonction de la position x
- 6- Variation du champ électrostatique en fonction de la position x
- 7- Evolution de l'énergie totale en fonction du temps

II. Distribution initiale des positions des particules :

Traitons en premier lieu, la phase initialisation qui consiste en :

- Initialiser les positions x_0 des particules aléatoirement,
- Ajouter une perturbation spatiale $A = 0.001$ qui entraîne un petit déplacement des particules,
- Identifier la cellule où se trouve chaque particule.

Comme illustration, on prend six particules ($N = 6$) de positions initiales aléatoires x_p , et nous allons voir leur distribution dans une grille de longueur $L = 3$, constituée de 03 cellules ($N_c = 3$) et $Ng = 4$ points de la grille.

Une première exécution du programme donne les résultats suivants :

```
xp= [ 0.02170263  2.04768875  2.8675718   0.08764912  0.66110979  1.94889297]
dx= 1.0
j= [0 2 2 0 0 1]
n= [ 2.22953846  0.82156856  2.03363242  0.91526055]
```

Maintenant nous allons visualiser la distribution des particules selon l'axe (ox) :

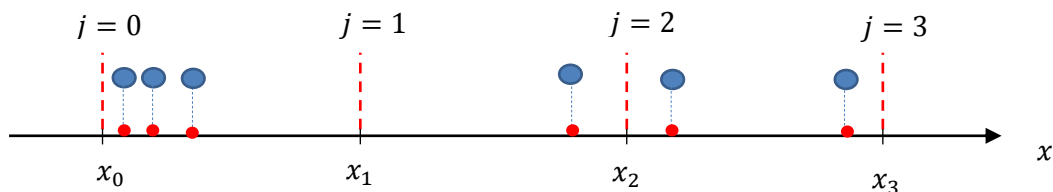


Figure III.1 : Distribution des particules dans la grille spatiale (première Exécution). En Bleu les électrons, en rouge leur position dans la cellule.

Si l'indice $j = 0$ signifie que la particule se situe dans la cellule (x_0, x_1) .

Si l'indice $j = 1$ signifie que la particule se situe dans la cellule (x_1, x_2) .

Si l'indice $j = 2$ signifie que la particule se situe dans la cellule (x_2, x_3) .

Si l'indice $j = 3$ signifie que la particule se retrouvera dans la cellule (x_0, x_1) grâce à la condition périodique $j = np.mod(j, Ng)$.

Si on refait l'exécution du programme avec les mêmes données, on trouve de nouvelles positions donc une distribution différente des charges :

```

xp= [ 1.54204538  2.32425498  1.08677908  1.71352528  0.11265602  0.44638323]
dx= 1.0
j= [1 2 1 1 0 0]
n= [ 1.44096075  2.21668951  2.01809476  0.32425498]
    
```

Les électrons sont distribués de la façon suivante :

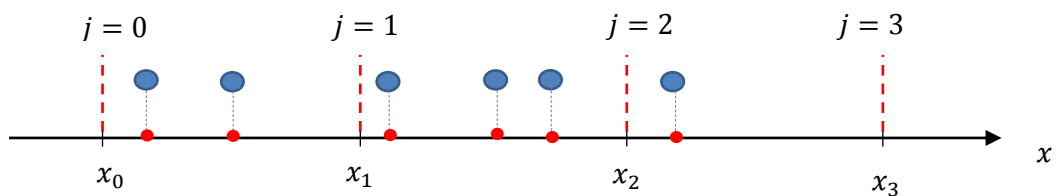


Figure III.2 : distribution des particules dans la grille (deuxième exécution).

Remarque : on remarque souvent qu'il existe plus d'une particule dans une cellule, ce groupe de charges qui ont les coordonnées assez proches est appelé macro-particule ou super particule. Une macro-particule est une particule computationnelle qui représente de nombreuses particules réelles ; il peut s'agir de millions d'électrons ou d'ions dans le cas de la simulation plasma, elle suit la même trajectoire qu'une particule réelle parce que l'accélération de la force de Lorentz ne dépend que du rapport charge sur masse.

La densité de charges dans la cellule sera donc la contribution de la macro particule se trouvant dans la cellule.

III. Distribution initiale des vitesses des particules :

Pour un plasma froid les vitesses des électrons ne suivent pas une distribution Maxwellienne, pour un plasma chaud c'est le cas. Dans la Figure III.3 (à gauche), on a tracé la fonction de distribution des vitesses initiales $f_s(x, v, t = 0)$ dans le cas d'un seul faisceau d'électrons d'un plasma chaud avec un $N = 100$ électrons possédant une vitesse thermique

faible $v_{th} = 1$ et pour $N = 1000$ électrons possédant une vitesse thermique plus grande $v_{th} = 2$.

Dans la Figure III.3 (à droite), on a tracé la distribution des vitesses initiales pour $N = 1000$ avec $v_{th} = 1$ et $N = 10000$ et $v_{th} = 2$ correspondantes à deux faisceaux d'électrons se déplaçant dans deux directions opposées.

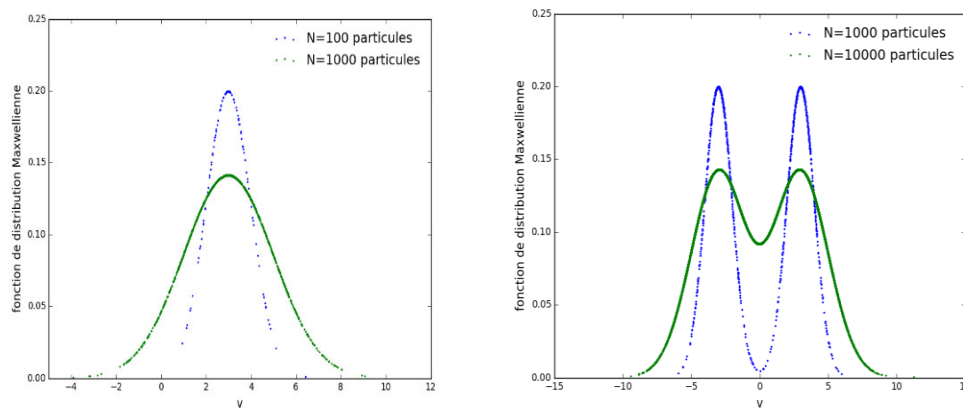


Figure III.3 : Distribution Maxwellienne des vitesses initiales pour un seul faisceau (à gauche).

Distribution Maxwellienne des vitesses initiales pour deux faisceaux en mouvement opposé (à droite).

IV. Simulation d'un seul faisceau d'électrons :

IV.1. Oscillation d'un plasma froid :

Un plasma froid est peu dense et faiblement ionisé, les ions sont supposés immobiles. Quand l'électron subit une légère perturbation spatiale de forme sinusoïdale (équation II.21) due à un champ extérieur, les électrons commencent à se déplacer de leurs positions initiales; un champ électrique de rappel est créé dans la direction opposée pour restaurer la neutralité du plasma en tirant l'électron à sa position d'équilibre. A cause de son inertie, l'électron va osciller autour de sa position d'équilibre à la fréquence de plasma ω_p .

Pour simuler l'oscillation d'un plasma froid, on prend les données suivantes : une vitesse thermique nulle $v_{th} = 0$ et une vitesse de dérive nulle aussi $v_b = 0$ (les vitesses initiales des électrons sont donc nulles), $L = 2\pi$, $N = 1000$, $N_g = 80$, une faible perturbation spatiale $A = 0.1$ est appliquée au système. Le temps de simulation $t_{fin} = 150$ avec un pas $t = 0.2$.

Un des résultats importants de la simulation PIC est le diagramme de phase, où l'axe (ox) représente la position des particules x_p et l'axe (oy) représente leur vitesse. La figure III.4 montre l'oscillation de plasma froid dans l'espace des phases (x, v) pour quatre temps différents. L'ensemble des électrons commencent à osciller de façon harmonique ; Avec le temps, le faisceau des électrons continue à osciller mais les vitesses commencent à se différencier entre elles.

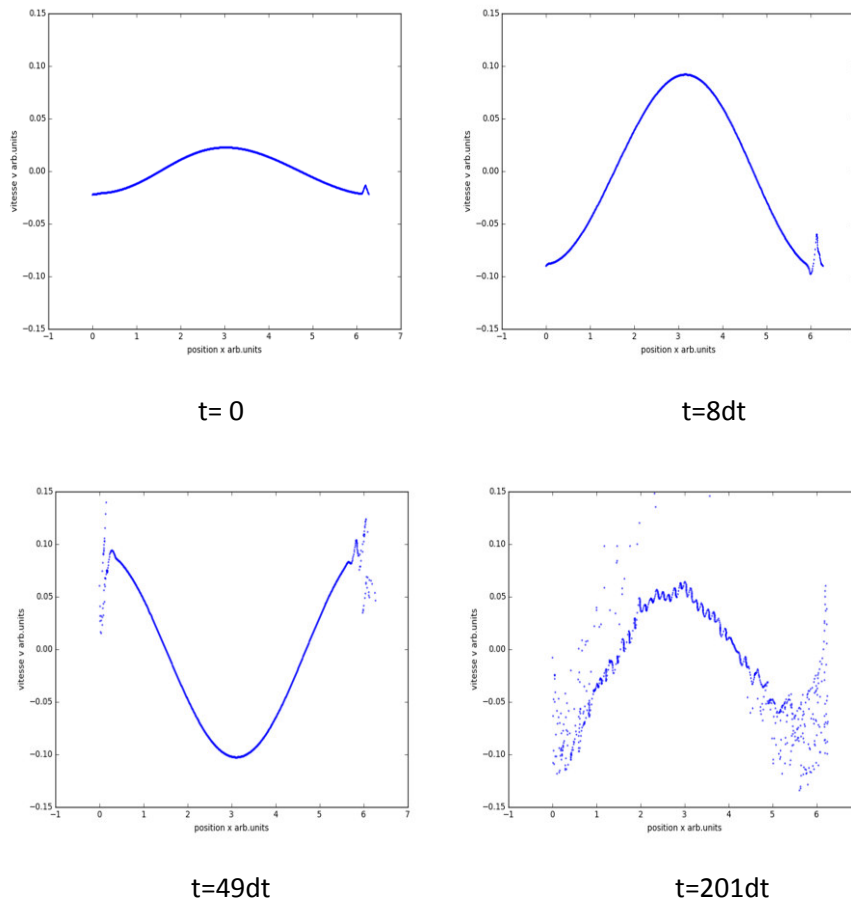


Figure III.4 : Oscillation d'un plasma froid.

IV.2. Dynamique des électrons dans un plasma chaud :

Contrairement au plasma froid, les positions initiales des particules sont aléatoires mais la fonction de distribution des vitesses est proche de la maxwellienne.

Pour simuler la dynamique d'un faisceau d'électrons chauds, on tient compte de la vitesse thermique (température importante des électrons) qui fait disperser les particules, on la pose égale à $v_{th} = 0.5$, $L = 50$, $N = 2000$, $N_g = 300$, une faible perturbation spatiale $A = 0.1$ est appliquée au système. Le temps de simulation est $t_{fin} = 150$ avec un pas $dt = 0.1$. Dans ce cas de figure, on ajoute une vitesse de dérive initiale égale à $v_b = 3$.

La figure III.5 représente le déplacement du faisceau d'électrons chauds dans l'espace des phases (x, v) pris à deux temps différents. On observe un faisceau d'électrons en mouvement translatore oscillatoire à cause de la vitesse de dérive initiale.

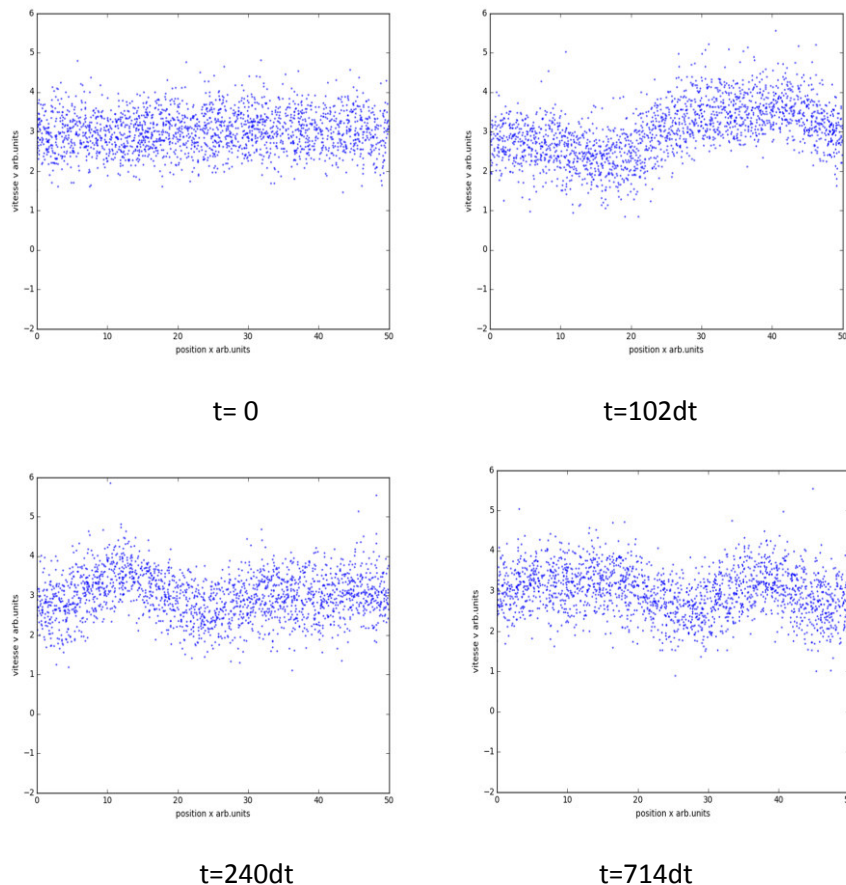


Figure III.5 : La dynamique d'un seul faisceau d'électrons chauds.

V. Two-Stream-Instability :

L'instabilité à deux flux (two stream instability TSI) est une instabilité très courante en physique des plasmas. Elle peut être induite par un flux de particules énergétiques injecté dans le milieu, ou pour établir un courant le long du plasma afin que différentes espèces (ions et électrons) puissent avoir des vitesses de dérive différentes.

Le modèle se compose de deux flux opposés d'électrons comme Indiqué dans la figure III.6. On peut facilement voir qu'un système de flux opposés est instable. Dans notre travail, on considère deux faisceaux d'électrons, initialement les électrons du côté droit se déplacent sous la forme d'un faisceau en couleur bleue, tandis que les électrons du côté gauche se déplacent sous la forme d'un faisceau en couleur verte.

Les données de simulation sont : $v_{th} = 0.5$, $v_b = \pm 3$, $L = 50$, $N = 5000$, $N_g = 300$, $A = 0.1$. Le temps de simulation est $t_{fin} = 150$ avec un pas $dt = 0.1$.

La figure III.7 montre l'évolution de diagramme de phase au cours de temps pour 6 instants.

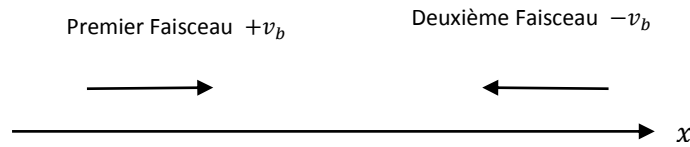


Figure III.6 : Schéma explicatif de la direction des deux faisceaux d'électrons.

On observe que la distribution initiale prend la forme de deux bandes uniformes, cependant plus le temps augmente, plus les bandes vont interagir et développer une configuration augmentant en magnitude jusqu'à atteindre une distribution phase-espace non linéaire de forme tourbillons connectés (Vortex). Cette configuration est appelée instabilité de deux faisceaux d'électrons.

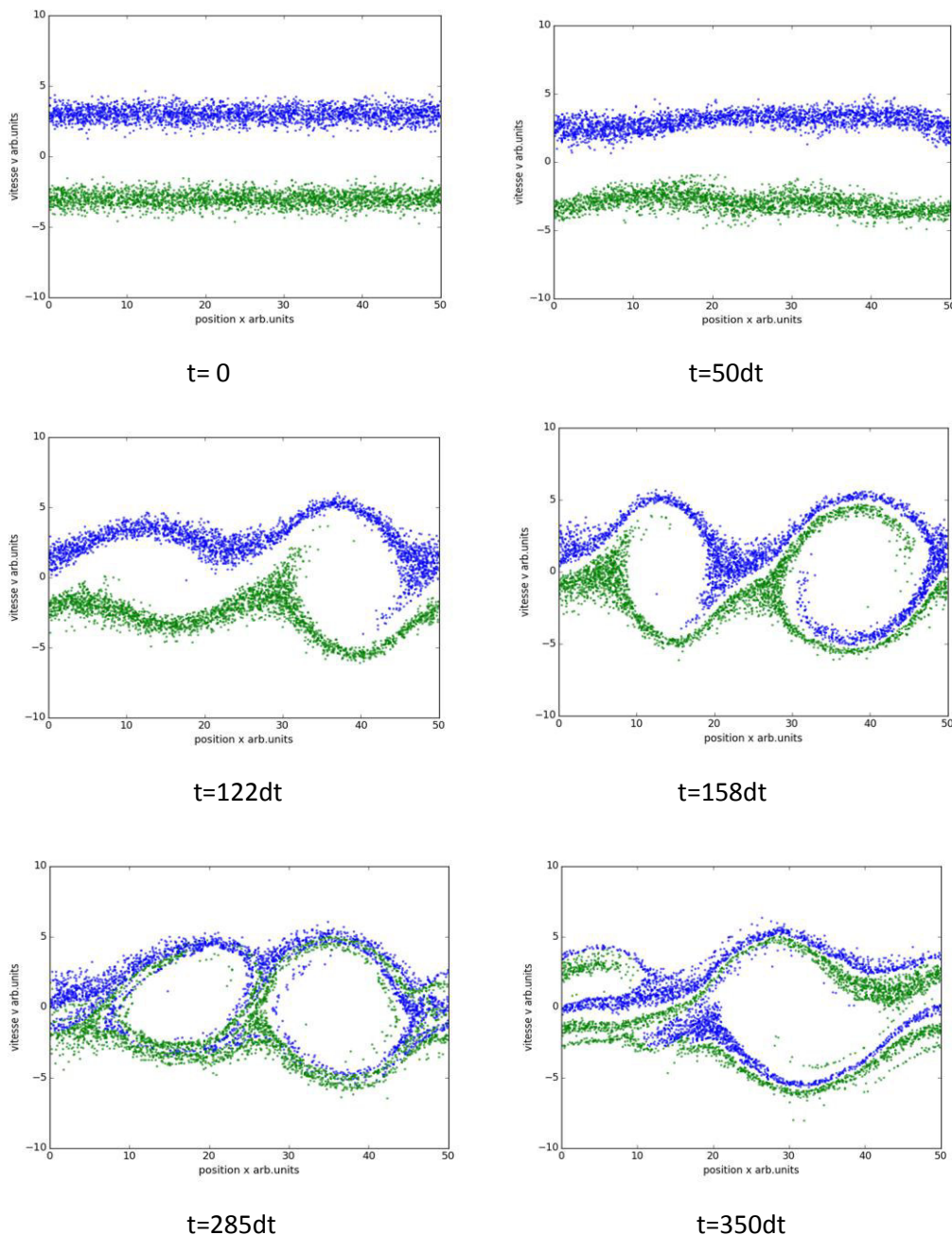
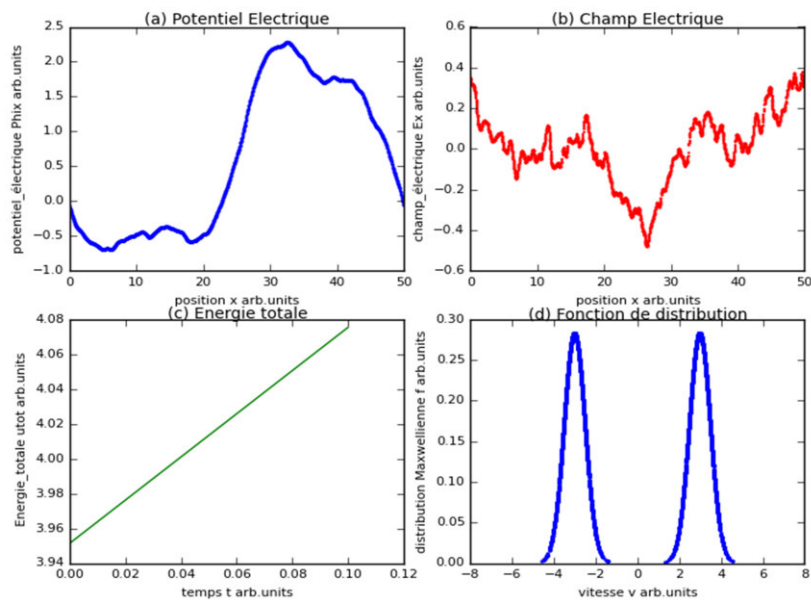
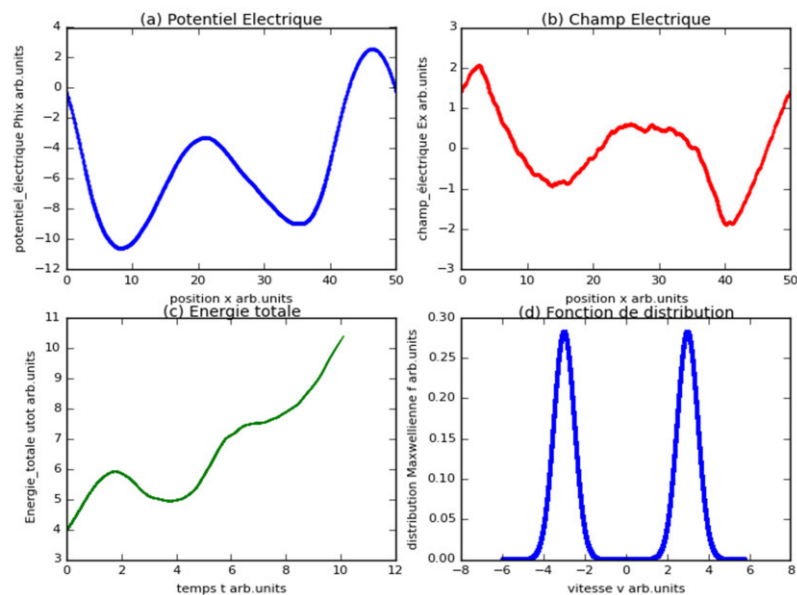


Figure III.7 : Diagramme de phase pour un TSI.

Il est possible de tracer avec le code PIC 1D, l'évolution au cours du temps, du champ E et du potentiel électrique ϕ en fonction de la position des particules, ainsi que : l'évolution de l'énergie cinétique, l'énergie potentielle, l'énergie totale U , et la variation de la fonction de distribution des vitesses. Ceci est illustré dans la figure III.8 pour deux valeurs de temps.



$t=0$



$t=100dt$

Figure III.8 : Variations des grandeurs physiques du TSI prises à deux temps.

VI. Conclusion :

Dans ce chapitre, nous avons utilisé le code PIC 1D électrostatique implémenté en python basé sur deux algorithmes : un premier algorithme d'initialisation des positions et des vitesses des électrons dans un plasma et un second algorithme de suivi des trajectoires de ces électrons. La simulation est réalisée en modifiant les paramètres : taille du plasma, nombre de cellules de la grille, vitesse thermique des électrons, vitesse de dérive du faisceau d'électrons, l'amplitude de la perturbation initiale et le temps de simulation. Nous avons simulé la dynamique des électrons dans un plasma dans différentes situations physiques : oscillations, flux des électrons, instabilité de deux flux d'électrons se déplaçant dans deux directions opposées.

Conclusion générale

Dans ce travail, nous avons réalisé un code Python basé sur la méthode *particle in Cell* qui permet de suivre la dynamique d'un nombre de particules formant une espèce (électrons, trous, ions ou même molécules) dans un plasma. Nous avons considéré le cas où les particules (dans ce travail ce sont des électrons) sont sous l'action d'un champ électrostatique induit où nous avons négligé le champ magnétique.

Il est vrai qu'il existe des codes Fortran, C et Python de la méthode PIC (gratuit et commercial), mais la méthode en elle-même est généralement difficile à comprendre ou bien à maîtriser car elle utilise beaucoup d'outils numériques : méthodes des différences finies, fonctions de pondération, interpolation de Lagrange, schémas de résolution des équations différentielles ordinaires d'ordre 1 (Euler, Leap-Frog et Runge Kutta). L'implémentation numérique nécessite aussi la résolution de système matriciel linéaire ainsi que l'inversion de matrice.

Le travail réalisé dans le mémoire est partagé en deux parties liées : développer la partie théorique de la méthode PIC et faire son implémentation numérique en utilisant le langage scientifique Python. Nous avons reporté quelques développements mathématiques dans des annexes supplémentaires. L'utilisation des fonctions préprogrammées Python a beaucoup facilité le codage numérique.

Nous avons pu simuler et tracer le diagramme de phase pour différentes configurations : 1/dynamique des électrons d'un plasma oscillatoire 2/ flux d'électrons lorsqu'une vitesse de dérive initiale est fournie aux électrons 3/ dynamique de deux faisceaux d'électrons dans deux directions opposées peut aboutir à une instabilité dans l'espace des phases qui se produit quand les vitesses ne sont plus linéaires. Nous avons pu aussi tracer le potentiel et le champ électrostatiques en fonction de la position ainsi que l'évolution de l'énergie totale (cinétique et potentielle) en fonction du temps.

Il existe la méthode PIC électromagnétique et PIC-MCC (PIC associée avec la méthode de Monté Carlo) qui simule le comportement du plasma en présence des collisions

Comme perspectives, il serait intéressant de compléter le code en ajoutant la composante champ magnétique et par conséquent de travailler dans une grille en 2D.

Références

- [1] K. Nishikawa et al., PIC methods in astrophysics: Simulations of relativistic jets and kinetic physics in astrophysical systems. Living Reviews in Computational Astrophysics manuscript, 2021.
- [2] Nuclear Physics Experience and its respective owners -2021, <http://nupeu.eu/index.php?g=textcontent/nuclearenergy/nuclearfusion&lang=en>
- [3] M. Guymont, Structure de la matière Atomes, liaisons chimiques et cristallographie, bellinédition, 2003.
- [4] Découvrir & Comprendre – Les 4 interactions fondamentales. Available at www.cea.fr
- [5] <http://physique.coursgratuits.net/physique-quantique-ondulatoire/modele-de-schrodinger.php>
- [6] Jean-Luc Raimbault, Introduction à la Physique des Plasmas, 2013 available at : <https://www.lpp.polytechnique.fr/IMG/pdf/m2apim2012.pdf>
- [7] Fiche d'information : les trous noirs, Agence spatiale canadienne, Available at : www.asc-csa.gc.ca
- [8] P. M. Bellan, Fundamentals of Plasma Physics, available at <http://www.gammaexplorer.com/wp-content/uploads/2014/03/Fundamentals-Of-Plasma-Physics-Paul-M.-Bellan.pdf>
- [9] A. Piel, Plasma Physics An Introduction to Laboratory, Space, and Fusion Plasmas, Springer-Verlag Berlin Heidelberg 2010.
- [10] P. Savoini, Introduction à la physique des plasmas. 2009, available at: https://www.lpp.polytechnique.fr/IMG/pdf/Cours_Intro_Plasma_M1_WEB.pdf
- [11] https://fr.wikipedia.org/wiki/Aurore_polaire
- [12] https://fr.wikipedia.org/wiki/Lampe_%C3%A0_plasma
- [13] Guillaume Clair, Etudes théorique et expérimentale des plasmas produits par laser en vue de leur application à l'analyse chimique des matériaux en environnement complexe. Thèse doctorat, Université d'Aix-Marseille II, 2011.
- [14] Chebbah A. univ. Contribution à l'étude de la production de l'ozone par décharge à barrière diélectrique surfacique et à électrodes planes. Application au traitement des eaux polluées. PhD these, Sidi Belabbes Djillali Liabes ; 2018.
- [15] P. Gibbon, Introduction to Plasma Physics, Proceedings of the CAS-CERN Accelerator School: Plasma Wake Acceleration, Geneva, 2014.
- [16] J. Massot ; Modèles hybrides fluide/cinétique pour les plasmas chauds ; Thèse de Doctorat ; Rennes université, France 2021.
- [17] C. Besse, P. Degond, M. H. Vignal ; La modélisation mathématique des plasmas au service des technologies spatiales ; thèse de doctorat Université Paul Sabatier, France, 2001.
- [18] J.A. Bittencourt. Fundamentals of Plasma Physics, third edition. Springer Science+Business Media New York, 2004.
- [19] C. K. Birdsall et A. B. Langdon; Plasma Physics via Computer Simulation; The Adam Hilger Series on Plasma Physics; Adam Hilger Bristol, Philadelphia and New York. IOP Publishing, 1991.
- [20] Bruce Langdon; Charles K. Birdsall. Theory of Plasma Simulation Using Finite-Size Particles, Physics of Fluids 13, 2115–2122, 1970.
- [21] A. C. J. Paes, N. M. Abe, V. A. Serrao, and A. Passaro. Simulations of Plasmas with Electrostatic PIC Models Using the Finite Element Method, Brazilian Journal of Physics, (33)2, 2003.
- [22] X. Aguilar, Stefano Markidis; A Deep Learning-Based Particle-in-Cell Method for Plasma Simulations; Electrical Engineering and Computer Science School KTH Royal Institute of Technology Stockholm, Sweden. 2021.

- [23] W.Konior; Particle-in-Cell Electrostatic Numerical Algorithm; Institute of Aviation,transactions of the institute of aviation No. 3(248), pp. 24-45, 2017.
- [24] Daniela F. Lopez-Astorquiza; Jaime H. Hoyos; Tutorial for the solution of a plasma electrostatic one-dimensional model using a PIC scheme; Sciences School; EAFIT; Basic Sciences Faculty;University of Medellin; Medellín; CP 050026. 2021.
- [25] J S Blandón et al 2017 J. Phys; Electrostatic plasma simulation by Particle-In-Cell method using Anaconda package; Journal of Physics: Conference Series: Conf. Ser. 850 012007, 2017.
- [26] V. Saini, S. K. Pandey and al, Coherent phase space structures in a 1D electrostatic plasma using particle-in-cell and Vlasov simulations: A comparative study, Physics of Plasmas 25, 092104, 2018.
- [27] P.Savoini, Etude de la physique des plasmas via la simulation numérique : une introduction ; Université Pierre et Marie Curie ; La science à Paris. 2008.
https://www.lpp.polytechnique.fr/IMG/pdf/Cours_simulationPIC_M2_Web.pdf
- [28] P. Fuchs et P. Poulain, Cours de Python, Université Paris Cité, France, 2022,
<https://python.sdv.univ-paris-diderot.fr/cours-python.pdf>
- [29] K. Atkinson, W. Han, D.Stewart, Numerical solution of ordinary differential equations. John Wiley & Sons, Inc, 2009.

Annexe A-1 :

Fonctions de Pondération

1- Fonction de Pondération : Grille 2D :

Pour calculer les moments de la distribution (n, ρ) au point fixe de la grille, on utilise une interpolation bilinéaire basée sur la méthode de pondération d'ordre un CIC « Cloud In Cell » qui consiste à attribuer une charge partielle, approchée linéairement aux quatre nœuds.

Dans le cas d'une cellule à deux dimensions, le coefficient de pondération correspond au nœud $(i; j)$ s'écrit comme suit :

$$w_{i,j} = \frac{A_1}{A} = \frac{(d_i - x_p) \cdot (d_j - y_p)}{d_i \cdot d_j}$$

Avec : A_1 est la surface entre la particule et le nœud $(i; j)$; A est la surface de la cellule ; x_p et y_p sont les coordonnées de la particule ; d_i , d_j sont la largeur et la hauteur de la cellule.

En faisant la même chose pour les nœuds $(i + 1; j)$, $(i; j + 1)$, $(i + 1; j + 1)$, on obtient :

$$w_{i+1,j} = \frac{A_2}{A} = \frac{x_p \cdot (d_j - y_p)}{d_i \cdot d_j}$$

$$w_{i,j+1} = \frac{A_3}{A} = \frac{(d_i - x_p) \cdot y_p}{d_i \cdot d_j}$$

$$w_{i+1,j+1} = \frac{A_4}{A} = \frac{x_p \cdot y_p}{d_i \cdot d_j}$$

Ces fonctions (appelées ou coefficients de pondération) en fait sont la base du polynôme d'interpolation de Lagrange en deux dimensions et elles représentent les coordonnées barycentriques. La somme des quatre poids doit être égale à un.

2- Fonction de Pondération : Grille 1D

Cependant, pour le cas d'une cellule unidimensionnelle, les coefficients sont donnés par :

$$w_i = \frac{d_1}{\Delta x} = \frac{x_p - x_i}{\Delta x}$$

$$w_{i+1} = \frac{d_2}{\Delta x} = \frac{x_{i+1} - x_p}{\Delta x}$$

Avec : d_1 et d_2 sont les distances entre la charge et les nœuds ; x_p est la position de la particule ; x_i et x_{i+1} sont les nœuds de la cellule.

Annexe A-2 :

Normalisation des équations

Normalisation des équations :

Pour obtenir des équations normalisées, on remplace les équations II.7 et II.8 dans les équations II.1 jusqu'à II.6, comme suit :

- Equation II.1 :

$$\frac{dv}{dt} = \frac{q}{m} E(x) \quad \text{II.1}$$

Remplaçant v, t , et E par :

$$v = \bar{v} v_{th} = \bar{v} \sqrt{\frac{k_B T_e}{m_e}} \quad (1)$$

$$t = \bar{t} \omega_p^{-1} = \bar{t} \sqrt{\frac{m_e \epsilon_0}{n_0 e^2}} \quad (2)$$

$$E = \bar{E} E_c = \bar{E} \sqrt{\frac{n_0 k_B T_e}{\epsilon_0}} \quad (3)$$

On obtient :

$$\frac{v_{th}}{\omega_p^{-1}} \frac{d\bar{v}}{d\bar{t}} = \frac{q_e}{m_e} \bar{E} E_c \Rightarrow \frac{d\bar{v}}{d\bar{t}} = \frac{\omega_p^{-1} q_e E_c}{v_{th} m_e} \bar{E}$$

Calculons le rapport $\frac{\omega_p^{-1} q_e E_c}{v_{th} m_e}$ sachant que $q_e = -e$, on trouve :

$$\frac{\omega_p^{-1} q_e E_c}{v_{th} m_e} = \frac{-e}{m_e} \sqrt{\frac{m_e \epsilon_0}{n_0 e^2}} \sqrt{\frac{n_0 k_B T_e}{\epsilon_0}} \sqrt{\frac{m_e}{k_B T_e}} = -1$$

Donc l'équation II.1 devient :

$$\frac{d\bar{v}}{d\bar{t}} = -\bar{E}(x)$$

- Equation II.2 :

$$v = \frac{dx}{dt}$$

Remplaçant t et v par (1) et (2) et x par :

$$x = \bar{x} \lambda_D = \bar{x} \sqrt{\frac{\varepsilon_0 k_B T_e}{n_0 e^2}} \quad (4)$$

On obtient :

$$v_{th} \bar{v} = \frac{\lambda_D}{\omega_p^{-1}} \frac{d\bar{x}}{d\bar{t}} \Rightarrow \bar{v} = \frac{\lambda_D}{v_{th} \omega_p^{-1}} \frac{d\bar{x}}{d\bar{t}}$$

Calculons le rapport $\frac{\lambda_D}{v_{th} \omega_p^{-1}}$, on obtient :

$$\frac{\lambda_D}{v_{th} \omega_p^{-1}} = \frac{\sqrt{\frac{\varepsilon_0 k_B T_e}{n_0 e^2}}}{\sqrt{\frac{m_e}{k_B T_e}} \sqrt{\frac{n_0 e^2}{m_e \varepsilon_0}}} = 1$$

Donc l'équation II.2 devient :

$$\bar{v} = \frac{d\bar{x}}{d\bar{t}}$$

- Equation II.3 :

$$E(x) = - \frac{d\phi(x)}{dx}$$

Remplaçant E et x par (3) et (4) et ϕ par:

$$\phi = \bar{\phi} \phi_c = \bar{\phi} \frac{k_B T_e}{e} \quad (5)$$

On obtient :

$$E_c \bar{E} = - \frac{\phi_c}{\lambda_D} \frac{d\bar{\phi}}{d\bar{x}} \Rightarrow \bar{E} = - \frac{\phi_c}{\lambda_D E_c} \frac{d\bar{\phi}}{d\bar{x}}$$

Calculons le rapport $-\frac{\phi_c}{\lambda_D E_c}$, on obtient :

$$-\frac{\phi_c}{\lambda_D E_c} = - \frac{k_B T_e}{e} \frac{\sqrt{\frac{n_0 e^2}{\varepsilon_0 k_B T_e}}}{\sqrt{\frac{\varepsilon_0}{n_0 k_B T_e}}} = -1$$

Donc l'équation II.3 devient :

$$\bar{E} = - \frac{d\bar{\phi}}{d\bar{x}}$$

- Equation II.4 :

$$\frac{d^2\phi(x)}{dx^2} = - \frac{\rho}{\varepsilon_0}$$

Remplaçant x et ϕ par (4) et (5) et ρ par:

$$\rho = \bar{\rho} \rho_c = \bar{\rho} n_0 e \quad (6)$$

La dérivée seconde peut être écrite comme suit :

$$\frac{d}{dx} \left(\frac{d\phi(x)}{dx} \right) = -\frac{\rho}{\epsilon_0}$$

Commençant par la première dérivée, on obtient :

$$\frac{\phi_c}{\lambda_D} \frac{d}{dx} \left(\frac{d\bar{\phi}}{d\bar{x}} \right) = -\frac{n_0 e}{\epsilon_0} \bar{\rho} \Rightarrow \frac{d}{dx} \left(\frac{d\bar{\phi}}{d\bar{x}} \right) = -\frac{\lambda_D n_0 e}{\phi_c \epsilon_0} \bar{\rho}$$

Ensuite la deuxième dérivée :

$$\frac{1}{\lambda_D} \frac{d}{d\bar{x}} \left(\frac{d\bar{\phi}}{d\bar{x}} \right) = -\frac{\lambda_D n_0 e}{\phi_c \epsilon_0} \bar{\rho} \Rightarrow \frac{d^2 \bar{\phi}}{d\bar{x}^2} = -\frac{\lambda_D^2 n_0 e}{\phi_c \epsilon_0} \bar{\rho}$$

Calculons le rapport $-\frac{\lambda_D^2 n_0 e}{\phi_c \epsilon_0}$, on obtient :

$$-\frac{\lambda_D^2 n_0 e}{\phi_c \epsilon_0} = -\frac{\epsilon_0 k_B T_e}{n_0 e^2} \frac{e}{k_B T_e} \frac{n_0 e}{\epsilon_0} = -1$$

Donc l'équation II.4 devient :

$$\frac{d^2 \bar{\phi}}{d\bar{x}^2} = -\bar{\rho}$$

- Equation II.5 :

$$\frac{d^2 \phi(x)}{dx^2} = \frac{e}{\epsilon_0} (n(x) - n_0)$$

Cette équation est normalisée à partir de la normalisation de l'équation II.4, en remplaçant $\bar{\rho}$ par: $\frac{\rho}{\rho_c}$ ensuite on remplace ρ par $-e(n(x) - n_0)$, on obtient :

$$\frac{d^2 \bar{\phi}}{d\bar{x}^2} = -\frac{\rho}{\rho_c} \Rightarrow \frac{d^2 \bar{\phi}}{d\bar{x}^2} = \frac{e}{n_0 e} (n(x) - n_0)$$

Après simplification, l'équation II.5 devient:

$$\frac{d^2 \bar{\phi}}{d\bar{x}^2} = \frac{n(x)}{n_0} - 1$$

Remplaçant n par :

$$n = \bar{n} n_0$$

Finalement, l'équation II.5 devient :

$$\frac{d^2 \bar{\phi}}{d\bar{x}^2} = \bar{n} - 1$$

- Equation II.6 :

$$U = \frac{1}{2} m_e v^2 - e\phi$$

Remplaçant v et ϕ par (1) et (5) et U par:

$$U = \bar{U} U_c = \bar{U} k_B T_e \quad (7)$$

On obtient :

$$\bar{U} U_c = \frac{1}{2} m_e \bar{v}^2 v_{th}^2 - e \bar{\phi} \phi_c \Rightarrow \bar{U} = \frac{1}{2} \frac{m_e v_{th}^2}{U_c} \bar{v}^2 - \frac{e \phi_c}{U_c} \bar{\phi}$$

Calculons les deux rapports $\frac{m_e v_{th}^2}{U_c}$ et $-\frac{e \phi_c}{U_c}$ on obtient :

$$\frac{m_e v_{th}^2}{U_c} = \frac{m_e}{k_B T_e} \frac{k_B T_e}{m_e} = 1$$

$$-\frac{e \phi_c}{U_c} = -\frac{e}{k_B T_e} \frac{k_B T_e}{e} = -1$$

L'équation II.6 devient :

$$\bar{U} = \frac{1}{2} \bar{v}^2 - \bar{\phi}$$

Annexe A-3 :

Schémas aux différences finies

Développement de Taylor :

La formule de Taylor, du nom du mathématicien Brook Taylor qui est établie en 1712, permet l'approximation d'une fonction plusieurs fois dérivable au voisinage d'un point, par un polynôme dont les coefficients dépendent uniquement des dérivées de la fonction en ce point.

Théorème (Taylor-Young):

Soit I un intervalle ouvert non vide de \mathbb{R} et soit x_i un point de I . Soit $f : I \rightarrow \mathbb{R}$ une fonction et n un entier ≥ 0 . On suppose que f est n fois dérivable sur I (elle est de classe C^n). Alors, pour tout $h \in \mathbb{R}$ tel que $x_i + h$ appartienne à I , on peut écrire :

$$f(x_i + h) = f(x_i) + h \frac{df(x_i)}{dx} + \frac{h^2}{2!} \frac{d^2f(x_i)}{dx^2} + \dots + \frac{h^n}{n!} \frac{d^n f(x_i)}{dx^n} + h^n \varepsilon(h)$$

$$f(x_i + h) = \sum_{k=0}^n \frac{h^k}{k!} f^{(k)}(x_i) + h^n \varepsilon(h)$$

Avec $\varepsilon(x)$ est une fonction définie sur I , qui tend vers 0 quand h tend vers 0 ; et h est le pas de la subdivision de l'intervalle I tel que $x_{i+1} = x_i + h$.

Développement d'ordre 1 :

Si f est dérivable en $x_i \in I$, alors :

$$f(x_{i+1}) = f(x_i + h) = f(x_i) + h \frac{df(x_i)}{dx} + h^n \varepsilon(h)$$

On peut négliger le reste $\varepsilon(x)$, puisqu'il tend vers 0 lorsque h tend vers 0 .

Donc on obtient :

$$f(x_{i+1}) = f(x_i) + h \frac{df(x_i)}{dx}$$

Qui implique :

$$\frac{df(x_i)}{dx} = \frac{f(x_{i+1}) - f(x_i)}{h}$$

En faisant une approximation: $\begin{cases} f(x_i) \sim f_i \\ f(x_{i+1}) \sim f_{i+1} \end{cases}$ et sachant que $h = x_{i+1} - x_i = \Delta x$; on trouve :

$$\frac{df_i}{dx} = \frac{f_{i+1} - f_i}{\Delta x}$$

C'est le schéma aux différences finies pour la dérive première d'une fonction.

Développement d'ordre 2 :

Si f est deux fois dérivable en $x_i \in I$, alors :

Développement progressive :

$$f(x_{i+1}) = f(x_i + h) = f(x_i) + h \frac{df(x_i)}{dx} + \frac{h^2}{2!} \frac{d^2f(x_i)}{dx^2} + h^n \varepsilon(h)$$

Développement régressif :

$$f(x_{i-1}) = f(x_i - h) = f(x_i) - h \frac{df(x_i)}{dx} + \frac{h^2}{2!} \frac{d^2f(x_i)}{dx^2} + (-h)^n \varepsilon(h)$$

On peut négliger le reste $\varepsilon(x)$, puisqu'il tend vers 0 lorsque h tend vers 0 .

Donc on obtient :

$$\begin{cases} f(x_{i+1}) = f(x_i) + h \frac{df(x_i)}{dx} + \frac{h^2}{2!} \frac{d^2f(x_i)}{dx^2} \\ f(x_{i-1}) = f(x_i) - h \frac{df(x_i)}{dx} + \frac{h^2}{2!} \frac{d^2f(x_i)}{dx^2} \end{cases}$$

En faisant la somme entre les deux équations, on obtient :

$$f(x_{i+1}) + f(x_{i-1}) = 2f(x_i) + h^2 \frac{d^2f(x_i)}{dx^2}$$

Qui implique :

$$\frac{d^2f(x_i)}{dx^2} = \frac{f(x_{i+1}) - 2f(x_i) + f(x_{i-1}))}{h^2}$$

En faisant une approximation : $\begin{cases} f(x_i) \sim f_i \\ f(x_{i+1}) \sim f_{i+1} \end{cases}$ et sachant que $h = x_{i+1} - x_i = \Delta x$; on trouve:

$$\frac{d^2f_i}{dx^2} = \frac{f_{i+1} - 2f_i + f_{i-1}}{\Delta x^2}$$

C'est le schéma aux différences finies pour la dérive seconde d'une fonction.

Annexe A-4 :

Interpolation de Lagrange

Interpolation de Lagrange :

En analyse numérique, les polynômes de Lagrange, du nom de Joseph-Louis Lagrange, permettent d'interpoler une série de points par un polynôme qui passe exactement par ces points appelés aussi nœuds. Cette technique d'interpolation polynomiale a été découverte par Edward Waring en 1779 et redécouverte plus tard par Leonhard Euler en 1783.

Polynôme de Lagrange :

Soit x_0, x_1, \dots, x_n $n + 1$ points distincts. On appelle base de Lagrange associées aux points x_i les $n + 1$ polynômes pour $i = 0$ à n tel que :

$$l_i(x) = \frac{(x - x_0)(x - x_1) \dots (x - x_{i-1})(x - x_{i+1}) \dots (x - x_n)}{(x_i - x_0)(x_i - x_1) \dots (x_i - x_{i-1})(x_i - x_{i+1}) \dots (x_i - x_n)}$$

Interpolation de Lagrange :

Soit f une fonction donnée définie sur R et x_0, x_1, \dots, x_n $n + 1$ réels donnés distincts.

Interpoler la fonction f par un polynôme de degré n aux points x_0, x_1, \dots, x_n consiste à trouver un polynôme p de degré $\leq n$ tel que $p(x_i) = f(x_i), 0 \leq i \leq n$.

Le polynôme s'écrit de la manière suivante :

$$f(x) = \sum_{i=0}^n f(x_i) l_i(x)$$

De même :

$$f(x) = \sum_{i=0}^n f(x_i) \prod_{j=0: j \neq i}^n \frac{(x - x_j)}{(x_i - x_j)}$$

Pour deux points x_i et x_{i+1} :

$$f(x) = f(x_{i+1}) \frac{(x - x_i)}{(x_{i+1} - x_i)} + f(x_i) \frac{(x - x_{i+1})}{(x_i - x_{i+1})}$$

Abstract

In this work, we simulated the behavior of charged particles in a plasma initially subjected to a perturbation due to an external electric field, using the particle in Cell in 1D, neglecting the induced or external magnetic field. Therefore each particle is subjected only to the electrostatic force. We have assumed that the plasma is very dilute in order to neglect collisions between particles, and we have chosen electrons as the charges to be simulated.

We explained the various stages of the electrostatic PIC1D simulation, which consists in following the movement of the particles and this by solving the Poisson equation and the motion equations. As a result, we can plot the phase diagram relating the velocities to the particles positions, the evolution of the potential, the electrostatic field and the total energy of the particles. We were able to simulate an oscillating cold plasma, a hot plasma, an electron stream and the instability of two electrons streams moving in two different directions.

ملخص

في هذا العمل، قمنا بمحاكاة سلوك الجسيمات المشحونة في البلازما التي تعرضت في البداية لاضطراب يعود لمجال كهربائي خارجي باستعمال طريقة الجسيم في الخلية ذات البعد الواحد، مع إهمال المجال المغناطيسي المستحث أو الخارجي. نتيجة لذلك فإن الجسيمات تخضع فقط للقوة الكهروستاتيكية. لقد افترضنا أن البلازما مخففة جداً من أجل إهمال الاصطدامات بين الجسيمات وقد اخترنا الإلكترونات كشحنة لمحاكاتها.

قمنا بشرح مختلف مراحل المحاكاة PIC1D الكهروستاتيكية والتي تتمثل في متابعة حركة الجسيمات وذلك بحل معادلة بواسون ومعادلات الحركة. نتيجة لذلك، نستطيع رسم منحنى الطور الذي يربط السرعات بمواضع الجسيمات، وتطور الجهد الكهربائي، والمجال الكهروستاتيكي، وكذلك الطاقة الكلية للجسيمات. تمكنا من محاكاة تذبذب البلازما الاصطناعية بنوعيتها، تدفق الإلكترونات، وعدم استقرار تدفقين للإلكترونات تتحرك في اتجاهين مختلفين.

Résumé

Dans ce travail, nous avons simulé le comportement des particules chargées dans un plasma soumis initialement à une perturbation dû à un champ électrique extérieur en utilisant la méthode particule in Cell en 1D, en négligeant le champ magnétique induit ou extérieur. Par conséquent chaque particule est soumise uniquement à la force électrostatique. Nous avons supposé que le plasma est très dilué afin de négliger les collisions entre les particules et nous avons choisi comme charges à simuler des électrons.

Nous avons expliqué les diverses étapes de la simulation PIC1D électrostatique qui consiste à suivre le mouvement des particules et ceci en résolvant l'équation de Poisson et les équations de mouvement. Comme résultat, nous pouvons tracer le diagramme de phase reliant les vitesses aux positions des particules. L'évolution du potentiel, du champ électrostatiques ainsi que l'énergie totale des particules. Nous avons pu simuler un plasma froid oscillatoire, un plasma chaud, un flux d'électrons et l'instabilité de deux flux d'électrons se déplaçant dans deux directions différentes.