Particle-in-cell simulation of nonlinear plasma instabilities

Paul Hughes

Dept of Physics and Astronomy
The University of Manchester

Fourth Year Master's Project Report

January 2007

This experiment was performed in collaboration with Daniel Martin

Supervised by Dr Philippa Browning

Abstract

A "Particle In Cell" (PIC) code was written to model a one dimension plasma which neglected magnetic effects. This program was initially used to simulate cold plasma oscillations as a test of its accuracy and then run to simulate two stream, four stream and weak beam – plasma instabilities. The resulting dispersion relation graph for the cold plasma oscillation, and the time evolution of the phase space plots for the instabilities, were used to confirm that these simulations were accurate.

Contents

1. Introduction 3
2. Method 4
2.1. Particle equations of motion 4
2.2. Weighting the charge density 6
2.2.1. Zeroth order weighting[5] 6
2.2.2. First Order Weighting[6] 6
2.3. Calculating the electric field 7
2.3.1. Fourier Transforms[7] 7
2.4. Weighting the electric field 8
3. Cold Plasma Oscillations 8
3.1. Definition of a cold plasma 8
3.2. Explanation of the origin of cold plasma oscillations[9] 8
3.3. Methodology of cold plasma oscillations 9
3.4. Analysis of the dispersion relation graph for the cold plasma oscillation[11] 9
4. Two Stream instability 10
4.1. Explanation of the origin of the two stream instability[12] 10
4.2. Methodology of two stream instability 11
4.3. Analysis of the time evolution of phase space during the two stream instability 11
4.4. Four Stream Instability 11
5. The Weak Beam-Plasma Instability 12
5.1. Explanation of the origin of the weak beam-plasma instability[15, 16] 12
5.2. Methodology of the weak beam-plasma instability[15] 12
5.3. Analysis of the time evolution of phase space during the beam-plasma instability[15] 12
6. Conclusion and Evaluation 13
7. Bibliography 14
Appendix A: Relative c++ program code…………………………………………………….. 15
Appendix B: Time Evolution of phase space during the two stream instability simulation…..…17
Appendix C: Time evolution of phase space during the four stream instability simulation……..19
Appendix D: Time evolution of phase space for the weak beam–plasma instability simulation 21

1. Introduction

The study of plasmas is a broad topic of great interest to the physics community and beyond. The most obvious application of plasma physics is in nuclear fusion, be it gravitationally confined fusion in a star or magnetically confined in experiments such as ITER[1]. However, other technological uses include fluorescent lighting, the construction of semiconductors and also plasma screen applications.

To begin to understand plasma physics, knowledge of the defining characteristics of a plasma is required. This knowledge can then be used to create computer simulations under different defined conditions, the findings of which can be used by experimentalists in their quest for technical innovation and practical applications.

In an ordinary gas, the electrons are bound to ions and so the particles are neutrally charged atoms moving about the gas. The interactions between these particles are due to the strong force and so the atoms effectively collide and recoil from each other in a manner very similar to billiard ball collisions.

In basic terms, a plasma is an ordinary gas which has been heated to such an extent that the electrons have completely disassociated from the ions. This 'soup' of ions and electrons moves in a manner which is no longer constrained as it was in the case for the ordinary gas. A plasma is therefore different from an ordinary gas in that it is a collection of charged particles with the interactions between particles being predominantly due to the Coulomb force. The Coulomb interaction is simply due to the fact that charged bodies attract or repel each other.

There are two major differences between the Coulomb interaction and the strong interaction previously mentioned. The first difference is that whilst in an ordinary gas particles 'collide', the long range nature of the Coulomb interaction allows particles in a plasma to affect each other's motion even at comparatively large distances. The second difference is that whilst atoms in a gas simply collide and (if we ignore the possibility of any chemical processes) recoil; the interaction between particles in a plasma can draw the particles together or repel the particles away from each other depending on the charges involved.

As a result of these differences, it is possible to see behaviours and patterns in plasmas which cannot be seen in ordinary gasses. However, as such situations are complex (involving at least a few thousand particles for very simple models) they cannot be solved analytically and so a computer simulation is required to make an approximation.

This is a Particle In Cell (PIC) simulation which, for the purposes of this project, will discretise space into a one dimensional grid and then compute the microscopic characteristics such as each particles individual position, velocity and the electric field at each individual grid point. These quantities are required to give the motion of each individual particle in the simulation. PIC codes treat plasmas as a collection of charged particles moving in electromagnetic fields. However the PIC code calculates the motion of computational particles which are actually superparticles that represent many real electrons or ions. An alternative method would be to use a magnetohydrodynamic[2] approach which treats the collection of particles as a fluid and computes the motion of that fluid.

PIC methods have the advantage that they are comparatively simple to implement, however they typically suffer from a large amount of statistical noise and they are usually used to simulate only a small number of particles $\left(\leq 10^{6}\right)$. A final disadvantage of PIC simulations is that they tend to not model charged particle simulations very accurately as the short range electric (Coulomb) fields of the colliding particles tend to be cancelled out by the electric fields of the rest of the particles in the system. Whilst PIC codes are relatively simple to manipulate, they do involve many calculations when compared with other methods such as the magnetohydrodynamic method mentioned previously and also hybrid models[3].

This project only considered a one dimensional plasma and neglected any magnetic effects. Using a PIC code, a simple plasma behaviour known as cold plasma oscillations was simulated and then more complicated observations including a two stream instability, a beam plasma instability and a four stream instability were undertaken.

My personal involvement in the project was researching the simulations and then supplying the relevant knowledge to my partner for the project Daniel Martin, who was then able to write the computer program. Once the program was operating appropriately, I used it to generate various graphs, notably the dispersion relation graphs for the cold plasma oscillations mentioned before.

In the next sections, the computational method is described along with the computer code used, then each simulation is individually discussed, the output of the simulation reviewed, finally the project is concluded and the results reiterated.

2. Method

2.1. Particle equations of motion

The motion of charged particles in electromagnetic fields is described by a combination of the Lorentz force and Maxwell's Laws. For the purpose of this project magnetic fields was neglected $\left(\overline{B} = \overline{0}\right)$ and so the Lorentz force, for a particle of charge q in an electric field $\overline{E}$, reduces to;

$$\overline{F} = q\left(\overline{E} + \overline{v} \times \overline{B}\right) = q\overline{E} \qquad [1]$$

This is the only force acting upon the particles and so the acceleration can be found from this;

$$m\,\overline{a} = q\overline{E} \qquad [2]$$

Also, as only one dimension is under consideration, Maxwell's first law reduces to;

$$\overline{\nabla}.\overline{E} = \frac{d\overline{E}}{dx} = \frac{\rho}{\varepsilon_0} \qquad [3]$$

Computer programs use discrete mathematics (that is to say they break continuous quantities such as lengths, velocities and electric fields up into small quanta) and so these equations become;

$$m\left(\frac{\overline{v}_{i+1} - \overline{v}_i}{\Delta t}\right) = q\overline{E} \qquad [4]$$

$$\rightarrow \overline{v}_{i+1} = \overline{v}_i + \frac{\Delta t\, q\, \overline{E}}{m} \qquad [5]$$

The velocities themselves must be calculated using what is known in mathematics as finite differences;

$$\frac{\overline{x}_{i+1} - \overline{x}_i}{\Delta t} = \overline{v}_{i+1} \qquad [6]$$

$$\rightarrow \overline{x}_{i+1} + \overline{x}_i + \overline{v}_{i+1}\Delta t \qquad [7]$$

Therefore to calculate the position that a particular particle will move to, Equation 5 must be solved and the result substituted into Equation 6. However, to solve Equation 5, the electric field at the current location of the particle must be found. This electric field is obtained by solving Equation 3, this is done by making use of the electric potential to give Poisson's Equation;

$$\overline{E} = -\overline{\nabla}\phi \qquad [8]$$

$$\rightarrow \nabla^2\phi \equiv \frac{\partial^2\phi}{\partial x^2} = -\frac{\rho}{\varepsilon_0} \qquad [9]$$

Discretising this yields;

$$\frac{\phi_{i-1} - 2\phi_i + \phi_{i+1}}{(\Delta x)^2} = -\frac{\rho_i}{\varepsilon_0} \qquad [10]$$

The upshot of this is that the charge density distribution can be tabulated, at a given instant in time, by simply counting the number of particles associating charge density with each grid point (or a more sophisticated variation of this idea which will be discussed in section 2.2). Then the electric potential can be found and from this the electric field can be used to give the motion of all of the particles.

Obviously this leads to a logistic loop and so the process is repeated to iterate the behaviour of the plasma through time, this is illustrated in Figure 1.

This logistic loop leads to an issue of when to use which force and when to use which velocity. For example, in order to find a particles velocity for the next iteration in time, Equation 4 must be used, however this requires a definite force $(q\overline{E})$ to be known.
Similarly a particles position in the next time-iterate requires the use of Equation 6 which in turn necessitates a definite velocity.

This leads to what is known as the leap-frog method[4]. This method involves using a 'time centred' force whilst calculating the new value of the velocity and then using a time centred velocity whilst calculating the new value of the position.
'Time centred' is a term that refers to using the value of the quantity required (be it the force or the velocity) at a time half way between the time iterations used for the velocity and position. The error associated with this method is proportional to the time step, Δt, and so provided the time step is sufficiently small, the leap frog method will provide adequately accurate results.

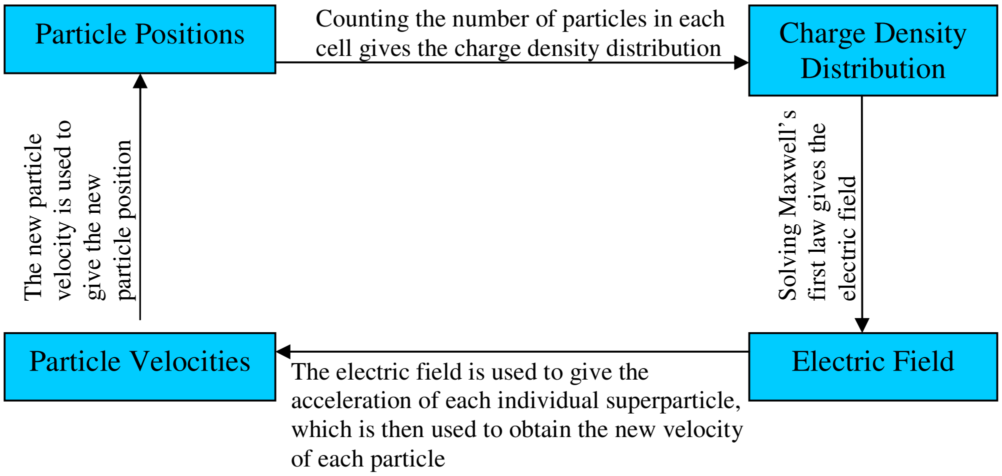

Figure 1: The loop of steps used to iterate the motion of the plasma through time

2.2. Weighting the charge density

The PIC code calculates a charge density distribution (discretised onto a finite grid) from the positions of the individual charged particles. As the Coulomb potential extends over all space, the electric potential at a given point would ideally be made up of contributions from all charged particles. However, in practice this is neither feasible nor necessary as the contributions from all but the closest particles (to the point in space under consideration) are negligible. As a result of this, the program considers how far each particle's contribution to the electric potential extends. This is done by allowing the contribution to the charge density to take certain forms which are described below.

2.2.1. Zeroth order weighting[5]

In zeroth order weighting, the particles are point like bodies which only contribute to the charge density of one grid point (namely the nearest grid point to the particle). This is rather unphysical as the particles can be very close to the boundary between two grid points and will still only contribute to one of those grid points. As particles move between grid points, zeroth order weighting produces discontinuities in the charge density distribution (see Figure 2). This is not desirable not only because it produces errors in the simulations but also as it prevents certain phenomena, such as cold plasma oscillations, from being simulated at all.

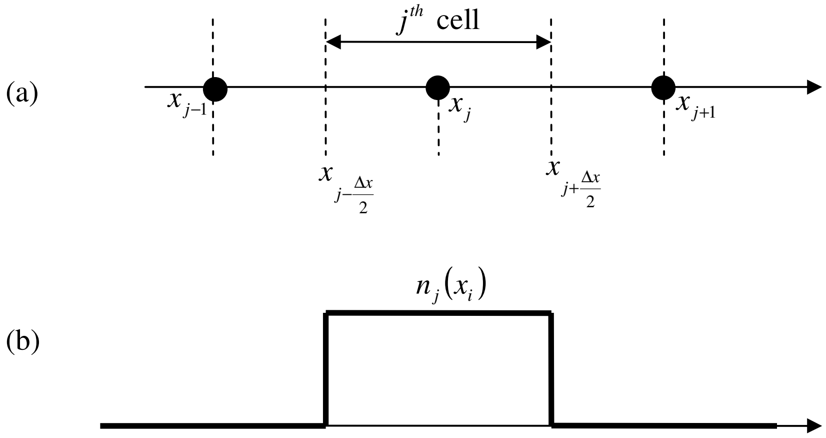

Figure 2: (a) Zeroth order particle and field weighting. Particles in the j^{th} cell are assigned to the point x_j. The charge density is simply calculated by counting the number of particles in each cell and dividing by the size (or in one dimension the length) of the cell. The particles in the j^{th} cell are acted upon by the electric field at the point x_j (i.e.$E(x_j)$). (b) The density, $n_j(x_i)$ at point x_j due to a particle located at x_i as the particle moves through the cell centred on x_j.

2.2.2. First Order Weighting[6]

In first order weighting, the charge of an individual particle is uniformly 'spread' over a region which is equal to the length between grid points. The effect of this weighting is that as a particle moves towards a grid point the contribution to the charge density of that point forms a linearly increasing function which peaks as the particle reaches the grid point and then decreases linearly as the particle moves away (see Figure 3).

First order weighting produces a smoother charge density distribution than the zeroth order weighting and so produces a better approximation of the electric field.

It is possible to use further order weighting. The advantage of doing so is that the density acquired becomes a smoother function, however this is at the cost of an increasing number of calculations and the loss of the point-like nature of the particles being simulated. For the purposes of this project, only zeroth and first order weightings were investigated. It was found that these weightings were adequate for our purposes.

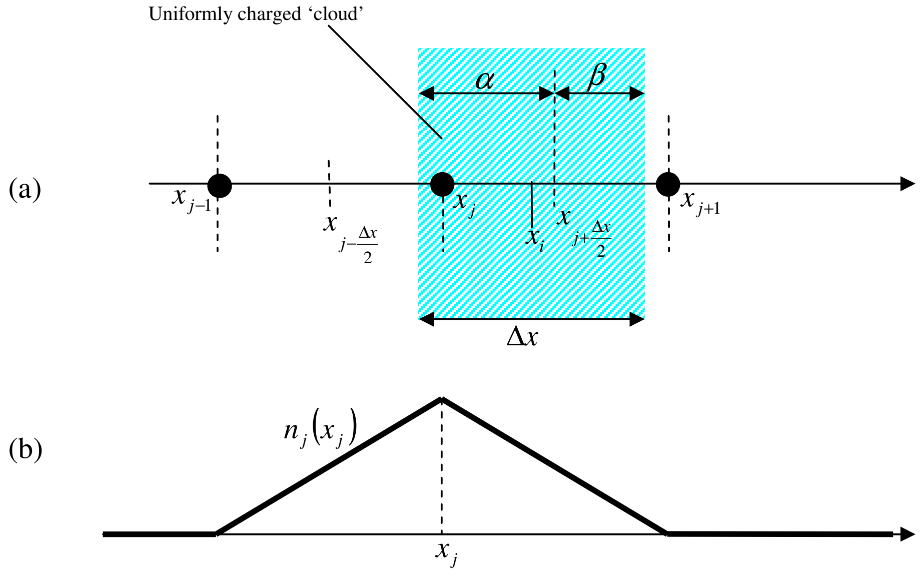

Figure 3: (a) First order particle weighting. The particle, or cloud, is one cell wide with centre at x_i. This weighting puts that part of the could which is in the j^{th} cell (labelled by α) at x_j and the part which is in the $(j+1)^{th}$ cell (labelled by β) at the point x_{j+1}. (b) The density, $n_j(x_i)$ at point x_i as the particle moves past x_j.

2.3. Calculating the electric field

2.3.1. Fourier Transforms[7]

The task is to solve the discretised Poisson's equation (i.e. Equation 10) numerically, given that the charge density on the grid points is known. The method described in section 2.1 can be simplified further by considering a periodic space. This is advantageous, as it represents an infinitely large space which is more physical than a confined region.

The reason that this simplifies the task is that it allows for the use of a Fourier transform when calculation the electric potential from the charge density distribution. The implementation of a Fourier transform replaces the second order derivative with square of the wavenumber;

$$\frac{\partial^2}{\partial x^2} \rightarrow -k^2 \qquad [11]$$

And so the expression for the electric potential simplifies;

$$\phi(k) = \frac{\rho(k)}{\varepsilon_0 k^2} \qquad [12]$$

By transforming the charge density distribution into Fourier space, then solving for the electric potential and finally inverse Fourier transforming to give the electric potential in x - coordinate space, the computational requirements are greatly reduced.

An alternative method is to use what is know as a Poisson solver. This method uses the fact that the space is periodic to solve a large collection of simultaneous equations[8].

2.4. Weighting the electric field

In a similar fashion to the density weighting of the particles (whilst calculating the charge density distribution) so to must the electric field be weighted when calculating the field at a particles location. The methods for weighting the electric field are analogous to the charge density weighting; the simplest and computationally least demanding is zeroth order weighting which assigns the electric field of the nearest grid point to a particle. As a result of this weighting sharp discontinuities occur in the electric field as a particle moves through the middle of a cell. First order weighting uses a linear combination of the electric fields associated with the two closest grid points which gives a smoother electric field and thereby provides more reliable results.

The code used to weight both the charge density and the electric field is outlined in Appendix A

3. Cold Plasma Oscillations

3.1. Definition of a cold plasma

The term "cold plasma" refers to a simplification made to the model discussed so far. This simplification is that motions due to the plasma's temperature are neglected. It is advantageous to test a plasma simulation program by modelling a cold plasma as the initial conditions are simple to input and the expectations are easily understood so that the outputted data can easily be analysed.

3.2. Explanation of the origin of cold plasma oscillations[9]

If the electrons in a plasma are displaced from a stable equilibrium (whilst fixing the ions in place), then the electric fields built up will be such that the electrons are forced back to equilibrium (as it is a stable equilibrium). However, upon reaching equilibrium, the electrons will have a kinetic energy that allows them to 'overshoot'. This process is then repeated with the electrons oscillating about their equilibrium positions. The angular frequency with which the electrons oscillate is known as the plasma frequency, ω_P.

By making the assumptions of neglecting magnetic fields, ignoring thermal motions, fixing the ions in space, using a plasma which is infinite in space and by only considering a one dimensional plasma, the following expression for the plasma frequency can be derived;

$$\omega_P = \left(\frac{n_0 e^2}{\varepsilon_0 m} \right)^{\frac{1}{2}} \qquad [13]$$

Where n_0 is the equilibrium particle density, e is the charge of an electron, m is the mass of the electron and ε_0 is the permittivity of free space.

3.3. Methodology of cold plasma oscillations

For the purposes of this project a one dimensional space of 2π was used. For the cold plasma oscillations this one dimensional space was divided up into 512 equally sized cells and a time step of $0.04s$ was used.

The program produced for this project outputs data such as the time evolution of the electric field, the charge density, the phase space, the kinetic energy and the potential energy, into a number of text files. GNUplot[10] was then used to display this data and to complete any analysis such as function fitting to the data outputted by the plasma simulator.

3.4. Analysis of the dispersion relation graph for the cold plasma oscillation[11]

As described in section 3.2, during the cold plasma simulation, oscillations in the time evolving kinetic, $T(t)$, and potential, $U(t)$, energies are expected. The frequency at which these oscillate is the plasma frequency and so by fitting these oscillations with an appropriate function (which was decided to be $T(t) = A\sin^2(\omega_p t)$, where A is an amplitude that is not of current interest) the plasma frequency can be obtained. By choosing to perturb different modes and observing the variation in the plasma frequency the dispersion relation can be found. Comparing the theoretical dispersion relation with that given by the simulation is the most appropriate numerical analysis of the cold plasma oscillations.

The plasma frequency is constant with respect to the perturbed mode, k, and so one would expect the dispersion relation to be of little interest. However the weightings discussed in section 2.2 have an associated shape factor which does depend upon the perturbed mode and so the dispersion relation is of interest as theoretical working can give the shape factors.

The theoretical dispersion relation for cold plasma oscillations (for first order weighting) is given below;

$$S_0(k) = \left[\frac{\sin\left(\frac{k\Delta x}{2}\right)}{\frac{k\Delta x}{2}}\right]^2 \qquad [14]$$

The argument of this theoretical prediction can be calculated for the grid used;

$$\frac{\Delta x}{2} = \frac{L}{2N} = \frac{\pi}{512} = 0.00613 \qquad [15]$$

Figure 4 shows the above sinc function fitted to the data given by the program. Constructing this dispersion relation graph requires each individual mode to be perturbed and the plasma frequency acquired by fitting an appropriate function. For the fitting of this function to make physical sense, the perturbed mode must be low enough to allow the oscillations to be resolved and so only the first 75 modes were used as the fitting was seen to break down for higher modes.

Fitting the sinc function of Equation 14 to the data, see Figure 4, gave an argument of $\frac{\Delta x}{2} = 0.00759$. This is adequately close to the theoretical prediction to be acceptable.

As described in section 2.1, the ‘leap frog’ method has an intrinsic error which is proportional to the time step, Δt, therefore using a smaller time step should give an argument of the sinc function even closer to the theoretical value. Another improvement would be to use higher order weighting. Both of these methods would require more computational resources and so, as this result is adequate for the project, they were not pursued.

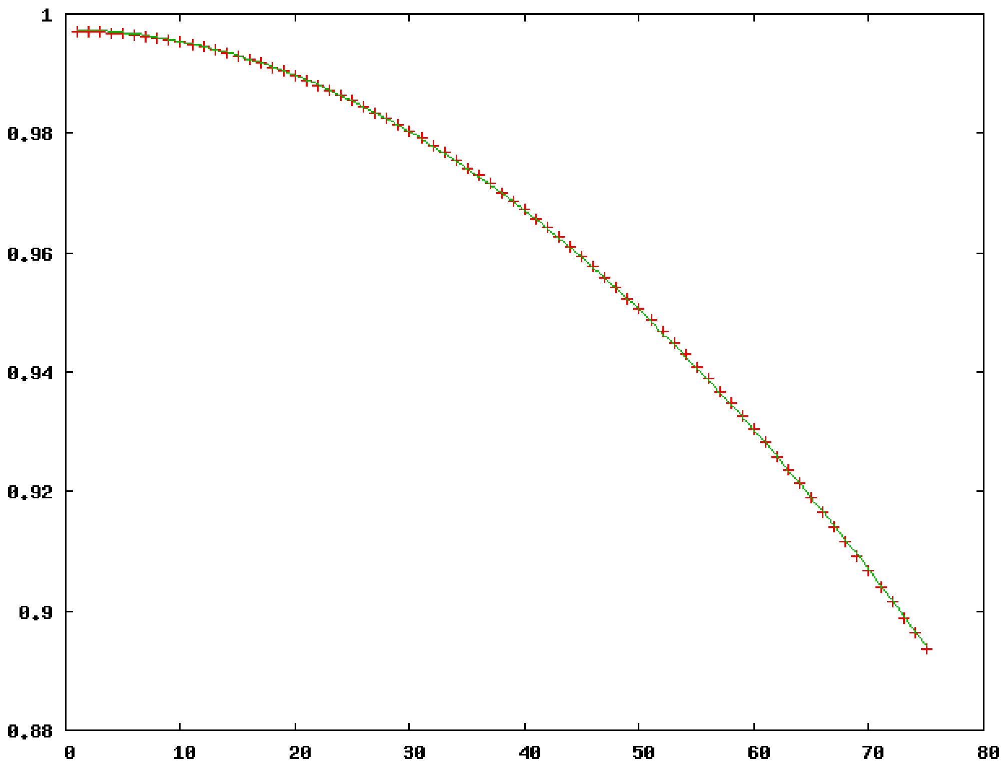

Figure 4: Dispersion Relation Plot for the Cold Plasma Oscillation, the red crosses are the outputted data from the program and the green line is Equation 14 fitted to the data with an argument of 0.00759

4. Two Stream instability

4.1. Explanation of the origin of the two stream instability[12]

The two stream instability involves two species of particles. This instability can be induced in a number of ways including injecting a stream into a stationary plasma and by setting a current along a plasma so that differently charged species have different drift velocities. However for the purposes of this project we adopted a conceptually simpler approach of simply using two streams of particles moving in opposite directions through the same region of space.

The electric field can have one of three possible effects upon a charged particle. Firstly, if the particle is moving slower than the velocity given by the electric field, then it will be accelerated by that field. In this case energy is transferred from the electric field to the particles. Conversely, if the particle is moving faster than the velocity given by the electric field, it will be decelerated by the field. Obviously in this situation energy is transferred from these more energetic particles to the electric field. Finally the particle could be moving at precisely the same speed as given by the electric field, in which case the particle’s velocity remains unchanged.
If there is a greater amount of energy being transferred to electric field than that extracted from the electric field, then exponential wave growth can occur. This is the source of the two stream instability.

4.2. Methodology of two stream instability

As stated in section 4.1, the two stream instability was simulated by constraining the initial conditions to be two streams of identical particles with opposite velocities moving through the same region of space.

Both streams contained 2048 particles of equal mass and charge, but opposite velocities. Exactly as for the cold plasma oscillations, a space of total size 2π was used which was divided into 512 equally sized cells. In order to obtain adequate and interesting results, after some experimentation, we decided to use a time step of $0.5s$ and a total of one hundred iterations through time (i.e. starting at $t = 0$ up to $t = 49$).

Once these initial conditions were inputted, the program iterated the motion of the particles in accordance with the discussion of previous sections of this report (namely by iterating through the logistic loop of Figure 1).

The behaviour of the two stream instability was investigated for variations in the excited mode (i.e. the value of k). It was found that the first mode $(k = 1)$ built up to dominate the other modes such that it was the main source of the two stream instability. The simulation was then run with no initial excitation at all, and the two stream instability was again observed, however this is believed to be a consequence of numerical error intrinsic to the weighting used. This is also the reason why zeroth order weighting can be used to simulate the two stream instability but not the cold plasma oscillations seen before.

4.3. Analysis of the time evolution of phase space during the two stream instability

The time evolving phase space plots shown in Appendix B are the best means of displaying the results of the two stream instability simulation (please note the nature of the time steps selected). These phase space plots agree with theoretical prediction; two vortices are formed by the majority of the particles and only the most energetic of the particles (i.e. those that have been accelerated by the instability) remain outside of these vortices. These phase space plots are also in accord with other simulations[13], as are the oscillations in the kinetic and potential energies. Rather than generate dispersion relation plots for the two stream instability[14], which would require a more complicated and lengthy approach than doing so for the cold plasma oscillations, it was decided to extend the project to observe other instabilities.

4.4. Four Stream Instability

The most obvious instability to simulate next was the four stream instability as this is an extension of the two stream instability. As before, a one dimensional space of length 2π was divided into 512 equal cells and each species contained 2048 particles. One hundred time iterations were calculated however, unlike the two stream instability, a time interval of $1s$ was used.

As its name suggests the four stream instability involves four species. Whilst the species used were again identical, each stream had a different velocity. As in the two stream instability, one stream was given an initial velocity in one direction whilst a second stream was given an equal but opposite velocity. However, in addition to this, two ‘slower’ streams were also propagated (again one slower stream for both directions). The magnitude of the initial velocity of the slower streams was chosen to be 0.3 that of the original ‘faster’ streams.

Appendix C shows the time evolving phase space plots for the four stream instability. In a similar fashion to those for the two stream instability, vortices can be seen. However whilst the two stream instability only had two vortices moving in opposite directions, the four stream instability has four vortices: two moving in each direction with one moving faster than the other. Again the majority of the particles are associated with these vortices and very few particles have been accelerated.

5. The Weak Beam-Plasma Instability

5.1. Explanation of the origin of the weak beam-plasma instability[15, 16]

The final instability to be simulated during this project was the weak beam-plasma instability. This instability involves two species, one moving and one stationary. The beam – plasma instability can be described as being strong or weak, which refers to how much smaller the beam species' plasma frequency, ω_{pb}, is than the stationary species' plasma frequency, ω_{pp}.

As previously described, the two stream instability can be simulated by injecting a beam into a stationary plasma. It is for this reason why the weak beam – plasma instability was chosen as the final observation, as the strong beam – plasma instability (especially the case where $\omega_{pb} = \omega_{pp}$) is very similar to the two stream instability observed before.

5.2. Methodology of the weak beam-plasma instability[15]

As usual a space of length 2π was used, however only 256 cells were used as it was found that the difference in accuracy was acceptable whilst the difference in computational requirement was desirable. Time intervals of $10s$ were used to iterate one hundred times.

For the weak beam-plasma instability, the beam is required to have a much smaller plasma frequency than the stationary plasma, the species that was initially moving was given a plasma frequency of $0.1 rads^{-1}$ whilst the stationary plasma was given a plasma frequency of $1 rads^{-1}$.

5.3. Analysis of the time evolution of phase space during the beam-plasma instability[15]

Appendix D shows the phase space plots iterated through time for the beam-plasma instability. The phase space plots generated by the simulation are agree quite well with other simulations[16]. As expected from the theoretical predictions, the stationary plasma is relatively unaffected with only slight perturbations from its initial state being visible. The moving beam is heavily affected however forms the expected pattern for the beam – plasma instability.

6. Conclusion and Evaluation

The original project was to write a computer program that would simulate the behaviour of a one dimensional plasma neglecting magnetic effects. This was done and a simulation of a cold plasma oscillation was used as a test as the behaviour of this simulation was easily predictable. After resolving any issues, the program was the used to simulate a two stream instability by using two streams initially moving in opposite directions. The time evolution of the phase space plots for this simulation were observed and found to agree with the current literature. In addition to this, the behaviour of the two stream instability was investigated for different excited modes and it was found that the first mode dominate over all other modes given a long enough time. Also it was found that numerical errors caused by the weighting of the charge density and the electric field could initiate a two stream instability, however this was not believed to be physically accurate. The two stream instability was then extended to a four stream instability and similar phase space plots were observed over a one hundred $0.5s$ time iterates. These phase space plots were similar in nature to the two stream results as expected. Finally the weak beam plasma instability was observed and simple behaviour was found to agree with other experiments[16].

This was an interesting project as it not only gave an insight to areas of plasma physics, but also as it served as an introduction to Particle In Cell techniques.

The project could easily be extended. Producing dispersion relation graphs for the two stream, four stream and beam-plasma instabilities may highlight any computational errors and would hopefully provide further evidence for the accuracy of the program. Also the effect of a greater number of cells, smaller time steps, different sized spaces and higher order weightings to the particles could be investigated. In addition to this other simulations such as the gravitational and beam – cyclotron instabilities could be investigated. All of these results could be compared with hybrid or mangetohydrodynamic simulation results. Finally the program could be altered to include magnetic effects and a higher number of dimensions.

7. Bibliography

1: www.iter.org – viewed 11/01/07

2: **P.A.Davidson**, *An Introduction to Magnetohydrodynamics*, **Cambridge University Press**

3: http://farside.ph.utexas.edu/teaching/329/lecture - viewed 06/01/07

4: **C K Birdsall & A B Langdon**, *Plasma Physics via Computer Simulation*, Section 2-4 Integration of the equation of motion, **Institute of Physics Publishing**

5: **C K Birdsall & A B Langdon**, *Plasma Physics via Computer Simulation*, Section 2-6 Particle and force weighting, Figure 2-6a, **Institute of Physics Publishing**

6: **C K Birdsall & A B Langdon**, *Plasma Physics via Computer Simulation*, Section 2-6 Particle and force weighting, Figure 2-6b, **Institute of Physics Publishing**

7: **C K Birdsall & A B Langdon**, *Plasma Physics via Computer Simulation*, Section 2-5 Integration of the field equations, **Institute of Physics Publishing**

8: **C K Birdsall & A B Langdon**, *Plasma Physics via Computer Simulation*, Appendix D Direct finite-difference equation solutions, **Institute of Physics Publishing**

9: **Francis F. Chen,** *Introduction to plasma physics and controlled fusion*, 2nd Edition, Section 4.3: Plasma Oscillations**, Plenum Press, New York & London**

10: http://www.gnuplot.info/ - viewed 11/01/07

11: **C K Birdsall & A B Langdon**, *Plasma Physics via Computer Simulation*, Section 4-6 Particles as seen by the grid, **Institute of Physics Publishing**

12: http://en.wikipedia.org/wiki/two_stream_instability - viewed 3/10/06

13: **C K Birdsall & A B Langdon**, *Plasma Physics via Computer Simulation*, Section 5-9 Two-stream instability: selected results, **Institute of Physics Publishing**

14: **C K Birdsall & A B Langdon**, *Plasma Physics via Computer Simulation*, Section 5-6 Two-stream instability; Linear analysis, Figure 5-6b **Institute of Physics Publishing**

15: **C K Birdsall & A B Langdon**, *Plasma Physics via Computer Simulation*, Section 5-10 Beam – plasma instability: Linear analysis, **Institute of Physics Publishing**

16: **C K Birdsall & A B Langdon**, *Plasma Physics via Computer Simulation*, Section 5-12 Beam – plasma instability: Project, **Institute of Physics Publishing**

18: **R.W. Hockney, J.W. Eastwood**, “Computer Simulation Using Particles”, 1988,
IOP Publishing

17: General correspondence and guidance from Dr. Philippa Browning