Some assembly required

Computational simulations of dusty plasma

by

Isabel J. Rodriguez

An undergraduate honors thesis submitted in partial fulfillment of the requirements for the degree of

Bachelors of Science

in

University Honors

and

Physics

Thesis Adviser

Erik Sánchez

Portland State University

2018

Hey there! First off, thank you SO MUCH for taking the time to read this. Whether or not you're coming from a science background, you'll be able to help make this draft the best version of itself (in fact, those of you who are non-scientists can contribute some of the most useful insights!).

I'm looking to tell a story. This means that, putting the technical details aside, there's a flow that has to be followed in order to effectively deliver this narrative. (Keep in mind that conceptually I had to start somewhere– if you're curious, feel free to look things up or to reach out!)

This is a rough draft, and there will undoubtedly be issues with grammar and, hopefully to a lesser extent, consistency of voice. The big questions for me at this stage are: "Does this thesis tell a story?" and "Does that story make sense?".

To help answer those questions, below are some of the things I'm looking for feedback on:

* Flow/organization (thesis as a whole)

* Flow/organization (chapter level)

* Flow/organization (section level)

* Use of jargon

* Effectiveness of voice

* Am I missing anything?

Read as much or as little as you'd like.

Thanks again, Isa

Author's to-do list

* Add: Tables (experimental parameters and simulation parameters)

* Add: Figures (final draft)

* Add: Update references (Mendeley to .bib file)

* Expand: Theory (CH2, current/OML)

* Add: Links to git (code, footnote)

* Add: To appendix - teaching tool?

* Add: To results - more graphs!

* Review: equations (ALL, final draft)

* Consider: would the sections make sense w/o the equations?

* Formatting: consistency?

* Formatting: vectors in bold

* Update: Experimental results

Abstract

From astronomy to industry, dust's ubiquitous presence in plasma (so called "complex" or "dusty" plasma) makes it an interesting object of study for a number of different fields. In some cases, it plays a critical role in the progression of certain processes, such as the formation of complex molecules in interstellar clouds. In laboratory environments, it can play a more troublesome role– hindering, for instance, the efficiency of integrated circuits which are the foundation of our modern technological capabilities. Advances in computing power have enabled the utilization of simulations as a tool for exploring the transport of dust particles in these low-pressure radio-frequency discharges. In this work, results of a Particle-in-Cell (PIC) simulation used to study charging of individual dust particles immersed in a collisionless electron-ion plasma is presented.

Acknowledgments

For those in search of a gentle introduction to computational plasma physics.

To those without whom this thesis would not exist.

Mil gracias.

Contents

Abstract iii

Acknowledgements iv

1 Background 1

2 Characteristics of dusty plasma 4

2.1 Primer: What is a plasma? 4

2.2 Conditions for a dusty (complex) plasma 8

2.3 Debye shielding in a dusty plasma 9

2.4 Forces on dust particles 10

2.5 Charging mechanisms 10

3 Plasma modeling 13

4 Motivation 16

4.1 Dusty plasma in space 16

4.2 Dusty plasma in fusion devices 17

4.3 Plasma sources in semiconductor processing 18

5 Description of the model 20

5.1 Governing Equations 21

5.2 Particle in Cell Method 22

5.2.1 Weighting (charge deposition) 23
5.2.2 Solving for Φ 24
5.2.3 Field solvers 25
5.2.4 Add/Move Particles 25

6 Simulation results 27

6.1 Simulation parameters 27
6.2 2D PIC Results 28
6.2.1 Modification of plasma by dust particle 28
6.3 Why open source? 29
6.4 Future work 30

A

1D PIC Code in Python 33

A.1 Two-stream instability 33

B

Sample Python code 35

List of Figures

1.1 Examples of naturally-occuring dusty plasma: a) on Earth: noctilucent clouds are high-altitude clouds consisting of charged ice particles; b) in space: the Cassini mission confirmed the presence of spokes in Saturn's rings caused by interactions of charged particles; c) human-made: rocket exhaust. 2

2.1 A quasi-neutral plasma contains roughly equal amounts of ions, or positively charged particles, and electrons, or negatively charged particles. From some distance away, the charges cancel each other out and is considered to be neutral. In this plasma we can pick out a single charge, Q, and use it as a way to understand how this plasma behaves. 5

2.2 The solid line is the way the electric potential of a single (isolated) charge drops off over a given distance. The drop-off behavior of our Q embedded in an ambient plasma is shown in the dashed line. This is due to an effect called shielding. [ISA, lets make this one a two-parter figure. LEFT: Top: isolated particle, Bottom: reuse above figure. RIGHT: Graphs] 6

2.3 Components of a dusty plasma. Left to right: electron (e), ion (i), neutral atom (n), and dust particle (d). 8

2.4 Debye shielding of dust particles. A positively charged sheath with length λ_D results from the attraction of ions to the negatively charged dust particles that are separated by a distance a. This sheath shields the particles from 'seeing' the electric field generated by neighboring particles. 9

2.5 Left: Absorption of electrons incident to the dust particle's surface cause the dust to obtain a net negative charge. Right: An example of secondary emission as a result of highly energetic photons striking the dust particle (photoemission). . . 11

3.1 LEFT: Fluid models tend to dominate fields such as astrophysics and astronomy. Pictured: Simulation of a supernova explosion (REF). RIGHT: Kinetic models are often used for low-temperature, low-pressure plasmas such as those typically used in (WHICH) industry. Pictured: Simulation of Coulomb crystals (REF). . . 14

4.1 Dusty plasma in LEFT: interstellar environment (REF) RIGHT: Saturn's ring spokes (REF) 16

4.2 Dusty particles in thermonuclear fusion devices (REF). 18

4.3 Dusty plasma in semiconductor processing devices (REF). 19

5.1 1d vs 2d plasma on a grid. In 1-dimension, charges are infinite sheets while in 2-dimensions each macrocharge encompasses a grid cell. 22

5.2 A flow chart laying out the Particle-in-Cell algorithm. 23

5.3 Area weighting used to deposit charge. 24

5.4 An illustration of the leap-frog method. We time-center the force while advancing the particle's velocity, and likewise time-center the velocity while advancing a particle's position. 26

6.1 Ion density around a single dust charge in an ambient plasma for various initial conditions: (LIST) 28

6.2 Electric potential of the plasma for various initial conditions: (LIST). 29

A.1 Top: Two opposing electron streams moving in a background of ions. An instability will develop due to charge bunching as the streams move through each other. Bottom: Evolution of the electron-electron two stream instability in phase space at times a) t=0, b) t = 49, c) = 99, and d) t = 199. The initial velocities are ±0.2, the grid has 1000 cells, there are 20000 particles per beam, and the beams have an initial sinusoidal perturbation of mode 2. 33

Chapter 1

Background

Everything
Turns,
Rotates,
Spins...
Pulsates,
Resonates,
And
Repeats.

Suzy Kassem

As much as 99.999% of the matter in the observable universe is made up of plasma– an ionized or partially-ionized gas composed of electrons, ions, and neutral atoms– and much of it is laden with dust particulates with sizes ranging from submicrons to millimeters. This additional component in an otherwise typical plasma increases the behavioral complexity of the system, and is thus referred to as a “complex” or a “dusty” plasma.

Naturally-occurring complex plasma can be found in the interstellar medium, where it plays an important role in the formation of molecular hydrogen (Garscadden et al. 1994); it is embedded in protoplanetary disks, planetary rings, and in the tails

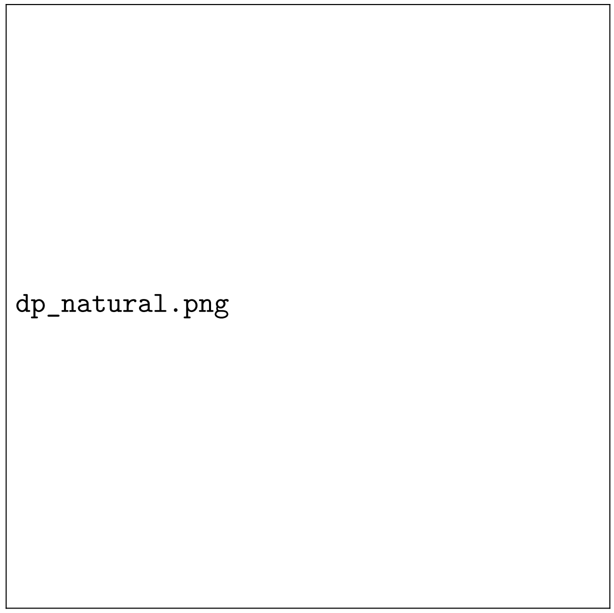

Figure 1.1: Examples of naturally-occuring dusty plasma: a) on Earth: noctilucent clouds are high-altitude clouds consisting of charged ice particles; b) in space: the Cassini mission confirmed the presence of spokes in Saturn's rings caused by interactions of charged particles; c) human-made: rocket exhaust.

of comets, and can even be observed as noctilucent clouds in the Earth's mesosphere.

In terrestrial environments, dusty plasma can be found everywhere from rocket exhaust to fusion devices, microelectronic fabrication and, of course, those created in the laboratory for scientific study. In industry, it can be regarded as technologically valuable for its applications in the medical field, where it can be used to treat wounds (Lacci et al., 2010); in microelectronics, where plasma processing has been responsible for innovation and growth (REF); and in aerospace, where ion propulsion is being pursued as a cheaper, faster and more efficient alternative to traditional chemical rockets (REF).

The computational exploration of physical systems is a unique niche that helps fill the expanse between the theoretical and experimental worlds. Analytic techniques are powerful, but on its own they can only fully solve special cases of problems. You eventually reach a stalling point as your system's complexity increases. On the other hand, some experiments as well as certain types of measurements are impractical due to lack of resources or (as in the case of fields such as astronomy) are simply impossible. Numerical techniques help pick up the slack where analytics falls short, and their results inform our knowledge and suggest new experimental directions to take.

In the remainder of this work, we will look at some of the basic theory behind plasma and what makes a dusty plasma, explore current modeling techniques, note some fields of interest, and explore how an electrostatic particle-in-cell algorithm was used to simulate the behavior of plasma in the presence of dust.

Chapter 2

Characteristics of dusty plasma

Physics depends on a universe
infinitely centered on an equals sign.

Mark Z. Danielenski

2.1 Primer: What is a plasma?

Here on Earth, we humans are intimately familiar with three phases of matter: solids, liquids, and gases. Our lived experience with water, for example, has endowed us with the understanding that adding energy (heat) to ice turns it into water, and that with enough heat that water will boil and evaporate into steam. Well, keep adding energy into that cloud of steam and at some point you will start to strip the atoms in that cloud of their electrons- a process called ionization- and create a plasma. The consequences resulting from this process is what really sets plasma apart from the other three states: its ability to produce electromagnetic (EM) radiation.

In its most basic definition, the plasma state is one in which a gas has become partially or fully ionized. To make this definition a bit more rigorous, there are three conditions that need to be satisfied, but before we go into what they are we need to

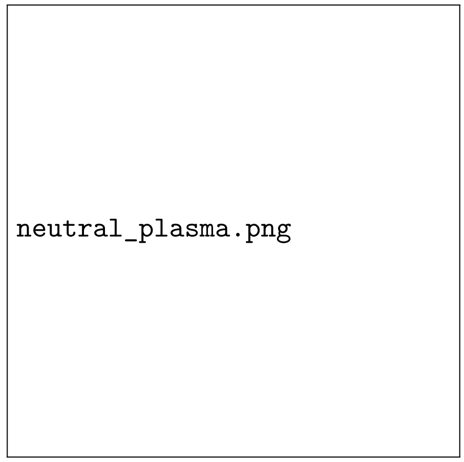

Figure 2.1: A quasi-neutral plasma contains roughly equal amounts of ions, or positively charged particles, and electrons, or negatively charged particles. From some distance away, the charges cancel each other out and is considered to be neutral. In this plasma we can pick out a single charge, Q, and use it as a way to understand how this plasma behaves.

first define a few important time and length scales. To do that, we will consider a very simple plasma system containing an equal number of ions and electrons (spoiler: a condition called quasineutrality).

If we pick out just one of the charges in this system, let's call Q our test particle [FIG], and look at its electric potential [1], we get the graph shown in [FIG]. This differs from a potential curve that would result if Q had been stranded alone in space, and is shown as the dashed curve in [FIG]. That's the Coulomb potential which drops off as one over the distance. Notice that while Q is embedded in the plasma, it has a potential which at first drops off with the Coulomb potential, but for larger distances drops off more sharply.

Recasting this in a mathematical form, we can treat the potential of an isolated charge as a sphere of charge whose potential is given by,

[1] Recall that electric potential is the work needed to be done in order to move that charge from one point to another.

$$\phi = \frac{q}{4\pi\epsilon_0 r}. \tag{2.1}$$

Compare this to the potential of our Q embedded in a plasma,

$$\phi = \frac{Qe^{-r/\lambda_D}}{4\pi\epsilon_0 r}, \tag{2.2}$$

where q is the charge, r is the distance, and λ_D is a quantity called the Debye length.

Figure 2.2: The solid line is the way the electric potential of a single (isolated) charge drops off over a given distance. The drop-off behavior of our Q embedded in an ambient plasma is shown in the dashed line. This is due to an effect called shielding. [ISA, lets make this one a two-parter figure. LEFT: Top: isolated particle, Bottom: reuse above figure. RIGHT: Graphs]

As can be observed in both [EQ 2.2] and [FIG], at points larger than λ_D the potential approaches zero. It is at this point that the test charge has been "screened" or "shielded" from other charges in the plasma, and with that we have introduced one of our most important characteristic length scales.

In a quasineutral plasma in which the ions are treated as immobile compared to the electrons, the Debye length is defined as,

$$\lambda_D = \sqrt{\frac{\epsilon_0 T_e}{n_e q_e}}. \tag{2.3}$$

Plasmas can oscillate with a frequency given by,

$$\omega_p = \sqrt{\frac{4\pi n_s Z^2 e^2}{m_s}}, \tag{2.4}$$

where s is the charge species (electron and ion).

The Debye length is related to this frequency by,

$$\lambda_D = \frac{v_{ts}}{\omega_p}, \tag{2.5}$$

were v_{ts} is the typical velocity of a given species as determined by its Maxwell-Boltzmann distribution.

Now that we have our time scale, time plasma frequency, and length scale, the Debye length, we can now talk about the three criteria for a plasma:

- Quasineutrality ($\lambda_d \ll L$)
- Collective effects dominate ($N\lambda_D^3 \gg 1$)
- Neutral collisions are negligible ($\omega\tau \gg 1$)

There are two other characteristic parameters for plasma left to briefly mention. One is the skin depth, or the depth at which plasma radiation can penetrate,

$$\lambda_{skin} = \frac{c}{\omega_p}. \tag{2.6}$$

The fourth parameter, the Larmour frequency, is necessary to in order to describe magnetic plasma however we will not consider it here. A more comprehensive look at plasma physics can be found in the work of Chen (1964).

dusty_plasma_species.png

Figure 2.3: Components of a dusty plasma. Left to right: electron (e), ion (i), neutral atom (n), and dust particle (d).

2.2 Conditions for a dusty (complex) plasma

The loose definition of dusty plasma is the presence of large [2] particles in an otherwise typical plasma (FIG) (REF). However, it is important to distinguish between simply having dust *in* a plasma and having a *dusty plasma*. This distinction is dependent on the ordering of three characteristic length scales specific to this type of plasma: the dust radius r_d, the distance between dust particles a, and the Debye length λ_D (Shukla et al., 2002). Debye shielding as it applies to dust particles will be discussed in the next subsection.

$$r_d \ll \lambda_D < a \tag{2.7}$$

$$r_d \ll a < \lambda_D \tag{2.8}$$

[2] Remember how we said that in an electron-ion plasma, the ions were massive and immobile compared to the electrons? Well compared to the ions, these dust partilces are billions of times more massive and thus can be considered immobile relative to the ions *and* the electrons. This will be important when we talk about charging of a dust partilce.

Put another way, if the distance between dust particles is smaller than the Debye length [EQ 2.8], the dust participates in the collective behavior of the system; conversely, if the interparticle distance is larger than the Debye length [EQ 2.7], you can regard it as having a collection of isolated screened grains, as was the case with our Q in the previous section. Furthermore, if the distance between dust particles is much smaller than the Debye length ($a \ll \lambda_D$), then we can treat the dust grains as if they were massive charged particles (Shukla et al., 2002).

2.3 Debye shielding in a dusty plasma

Figure 2.4: Debye shielding of dust particles. A positively charged sheath with length λ_D results from the attraction of ions to the negatively charged dust particles that are separated by a distance a. This sheath shields the particles from 'seeing' the electric field generated by neighboring particles.

Figure 2 illustrates the concept of Debye shielding in a dusty plasma. If we regard a dust particle as a ball of charge, we expect that it would attract particles of opposite charge (i.e, negative electrons if the dust has a net positive charge, or ions if the dust has net negative charge). This attraction creates a cloud surrounding the dust particle, called a sheath, which shields the electric field of our dust particle from the rest of the plasma. In a complex plasma, however, dust particles are not perfectly shielded due to the velocity distribution of ions and electrons in the sheath.

Another important consequence is the interplay of charges that allows particles to cluster/grow and create grains/larger particles.

2.4 Forces on dust particles

The basic governing equation describing the dynamics of a charged grain of dust with a mass m_d and velocity v_d is given to us by Newton's second law [3],

$$M_d \frac{dv_d}{dt} = \sum F = \vec{F}_E + \vec{F}F_G + \vec{F}_T + \vec{F}_D + \vec{F}_P, \tag{2.9}$$

where F_E is the electromagnetic force, F_G is the gravitational force, F_T is the thermophoretic force associated with a temperature gradient in the plasma, F_D is the drag force, and F_P is the radiation pressure force. The electromagnetic force is a combination of the Coulomb force and the Lorentz force, where E is the associated electric field, and B is the associated magnetic field.

$$\vec{F}_E = \vec{F}_C + \vec{F}_L = q_d \left(\vec{E} + \vec{v} \times \vec{B} \right). \tag{2.10}$$

For an electrostatic plasma, we would leave out the Lorentz force term.

2.5 Charging mechanisms

There are three basic processes by which dust particles immersed in an ambient plasma become charged:

- Interactions between dust and neutral particles
- Interactions between dust and energetic charged particles
- Interactions between dust and energetic light particles

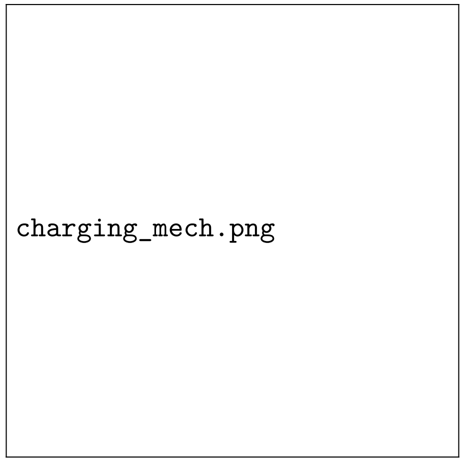

Figure 2.5: Left: Absorption of electrons incident to the dust particle's surface cause the dust to obtain a net negative charge. Right: An example of secondary emission as a result of highly energetic photons striking the dust particle (photoemission).

Don't be fooled, however! These elementary processes are actually quite complex and are difficult to understand, especially when trying to consider the different processes at one time as well as when looking at collections of dust particles. We will instead focus on the case of an isolated dust particle of finite size (that is, several λ_D in diameter).

A dust particle placed in a plasma acts as a probe that will collect a primary species in that plasma (REF). Absorption of the ions present will cause a dust particle to become positively charged. However, in laboratory plasmas it is often the case for dust particles to acquire a negative surface charge. This is because the thermal velocity of electrons is much greater than that of ions, thus the electrons will tend to reach the dust first.

The charge of a dust grain is described by the rate at which it absorbs or collects charged particles,

$$\frac{dq_d}{dt} = \sum_s I_s(q), \tag{2.11}$$

[3] F = ma

where I is the current and s is the particle species. For an electron-ion plasma, [EQ 2.11] becomes,

$$\frac{dq_d}{dt} = I_e + I_i. \tag{2.12}$$

At equilibrium, no additional charge is being collected and thus

$$\frac{dq_d}{dt} = \sum_s I_s = 0. \tag{2.13}$$

With no net current flow, the dust particle is left with a surface potential (which, again, tends to be negative for laboratory dusty plasmas). A [TABLE] can be found listing the typical surface potentials for the most commonly used plasma sources.

Lets take another look at the current.

$$I_s = \sum_s \int q_s f_s \sigma_s(v, q) v d\vec{v}, \tag{2.14}$$

Here $v \equiv |\vec{v}|$ is the absolute value of the speed of the particles, and σ_s is the charge-collection cross-section [4] (Vladimirov, 1997) given as

$$\sigma_s = \pi a^2 \left(1 - \frac{2q_s q_d}{am_s v^2}\right) \quad \text{if} \quad \frac{2q_s q}{am_s v^2} < 1, \tag{2.15}$$

and

$$\sigma_s = 0 \quad \text{if} \quad \frac{2q_s q_d}{am_s v^2} \geq 1. \tag{2.16}$$

A negatively charged dust particle can become positively charged through secondary electron emissions resulting from surface impacts with energetic electrons and ions [FIG], or through the process of photoemission, in which an energetic ultraviolet photon (found both in space and industry environments) incident on the dust's surface emits electrons [FIG].

[4] A cross-section is an area over which there is a probabilty for a process to occur (Here, it's the probabilty of a dust particle aborbing a charge on its surface.)

Chapter 3

Plasma modeling

> There is no need to ask the question, *Is the model true?*...The only question of interest is *Is the model illuminating and useful?*.
>
> *George Box*

Plasmas are dynamic, nonlinear, and can often be unstable, which is another way of saying they are too complicated to understand using pen and paper alone. Numerical modeling is essential for filling the gaps in our knowledge and is a critical tool for informing experimental diagnosis and experimental design (Bell, 1998). To be able to *fully* describe a problem in three dimensions, you'll actually need six: three in position space, and three in velocity space. This, as you might imagine, can be problematic in spite of our advances in computing power. Therefore, the goal (as is also the case for the theoretical and experimental branches of plasma physics) is to be able to get at the *essence* of plasma. The challenge is in the details, or rather in the ability to leave out just enough details to make it easier to simulate while at the same time maintaining the integrity of the information you later want to retrieve. Surprisingly, you can leave out a lot! Four dimensions, for instance (we would call this a 1D1V plasma which is a good place to start but may not necessarily be the

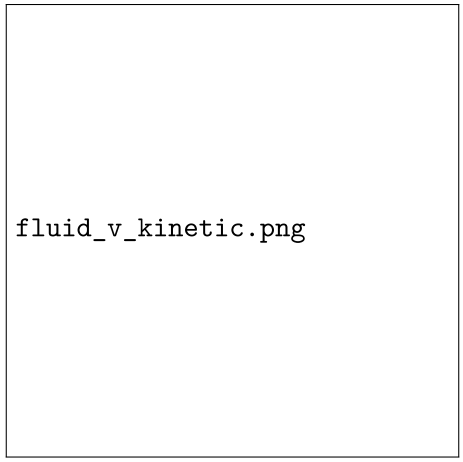

Figure 3.1: LEFT: Fluid models tend to dominate fields such as astrophysics and astronomy. Pictured: Simulation of a supernova explosion (REF). RIGHT: Kinetic models are often used for low-temperature, low-pressure plasmas such as those typically used in (WHICH) industry. Pictured: Simulation of Coulomb crystals (REF).

best way to go). But we can do this because plasma is a collective phenomena. Going back to the condition we laid out in SECTION 2.1, we are interested in plasma systems that are much longer than the Debye length ($\lambda_D \ll L$).

For a plasma that has magnetic fields, and for whose particles you can ignore quantum effects, we can fully describe it using Maxwell's equations [1]:

$$\nabla \cdot \vec{E} = -\frac{\rho}{\epsilon_0}, \tag{3.1}$$

$$\nabla \times \vec{E} = -\frac{\partial B}{\partial t}, \tag{3.2}$$

$$\nabla \cdot \vec{B} = 0, \tag{3.3}$$

$$\nabla \times \vec{B} = \mu_0 \left(J + \epsilon_0 \frac{\partial \vec{E}}{\partial t} \right). \tag{3.4}$$

[1]For more on electrodynamics, I refer you to Griffiths (YEAR).

Plasma simulations can be split into two main categories: fluid and kinetic [FIG 3.1]. A kinetic description is the more physically realistic of the two, as it focuses on the effects of motion of charged particles within a plasma (Callen, 2003). It is this type of model that dominates the literature [REFS]. Particle-in-cell (PIC) and cloud-in-cell (CIC) simulation methods are some of the more common examples of kinetic simulations. At the most basic level, PIC and CIC rely on a field solver and particle mover to add up behaviors and effects on an individual-particle level. While the physics here is more accurate, there are computational challenges to be considered: to simulate the trajectories billions of particles [2] in a fully three dimensional problem, or to take into account interparticle interactions (Monte Carlo Collisions modules often supplement advance models).

In contrast, fluid simulations reduce the computational complexity by focusing on macroscopic behaviors (e.g. density) of charged particles. This dispenses with phase space information, collapsing a 6d problem into a 3d velocity space. The payoff is the to simulate large systems for long periods of time (Bell, 1998).

In both the kinetic and fluid cases, the trade offs can be mitigated through the use of a hybrid approach. In a hybrid model, certain species (hot, thermal electrons for instance) would be modeled as a fluid, and others (the cooler, slower, and more massive ion, neutral species, and dust species) are given the kinetic treatment.

[2] That's 2^N bits of computer memory, where N can be upwards of 10^{12}!

Chapter 4

Motivation

Fascinating.

Mr. Spock

4.1 Dusty plasma in space

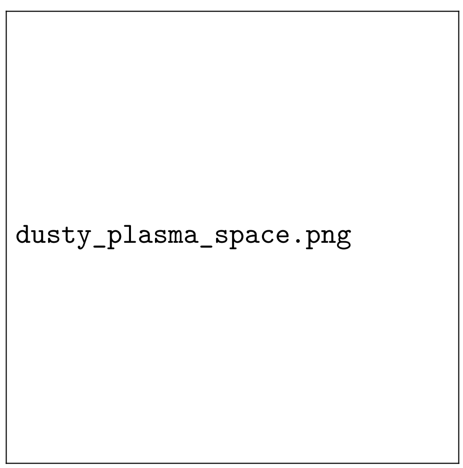

Figure 4.1: Dusty plasma in LEFT: interstellar environment (REF) RIGHT: Saturn's ring spokes (REF)

Interest in plasma as an astrophysical phenomena dates back to the 1980's, at a around the time most of the electromagnetic (EM) spectrum was being discovered

(Peratt, 1998). [1]. Examples in our own cosmic backyard include dusty cometary coma and Saturn's rings, both of which consist of tiny grains of charged ice and rock particles. The 1980 flyby of Saturn by Voyager 2 was the first time dynamic behavior was observed in Saturn's rings. Termed 'spokes', it's believed that these seasonal features are the result not of gravitational influences, but of the interplay between the charged particles and their electric fields (REF). The Voyager observations were followed up by the Cassini mission in 2005, and more than 10 years on are still considered an intriguing puzzle.

Leaving our neighborhood and venturing to the space between stars, the interstellar medium, we find that nebula (clouds of dust and ionized gases) are also full of complex organic molecules which may hold clues to the origin of life (REF). Collisional or collisionless shock waves [2] in these dust-molecular clouds are thought to be responsible for creating the density condensation necessary to accelerate gravitational collapse and form stars (Popel et al., 2001). Dust charging may also be important factors for such things as shocks in supernovae explosions and particle acceleration (Popel and Tsytovich, 1999).

4.2 Dusty plasma in fusion devices

Fusion devices, whose success as a sustainable alternative form of energy is perpetually a decade or two on the horizon, harness the power of plasma. In the case of thermonuclear fusion, tokamak reactors generate plasmas with energies upwards of 1000 eV (REF) – equivalent to 10 million degrees Celsius! These hot dense plasmas are confined using magnetic fields (so called 'magnetic confinement'), but confine-

[1] Early models of our universe supposed that all mass in the universe was like the type of mass found on our planet (Peratt, 1998). With the help of technology, we've since been able to extend our observational senses into other parts of the EM spectrum (x-rays, gamma rays, etc.) and have since come to the understanding that most of the observable matter in our universe is in the plasma state.

[2] Resulting from interparticle friction and wave-particle interactions, respectively.

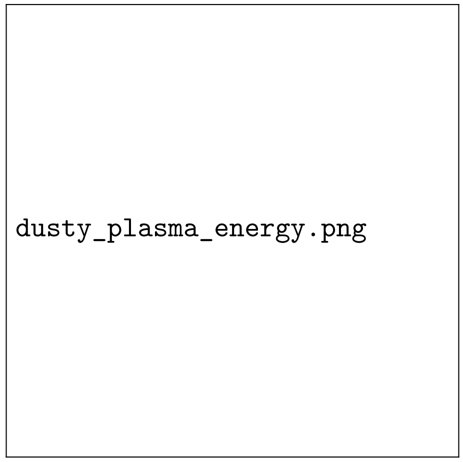

Figure 4.2: Dusty particles in thermonuclear fusion devices (REF).

ment is not perfect and this leads to damage to the damage of chamber walls and in-vessel ancillaries (REF). This not only has the potential to compromise structural integrity but the debris ends up entering the plasma at a rate of several kilograms per day in a device such as the International Thermonuclear Experimental Reactor (ITER) (Thomas Jr., 2015), and is evidenced by the particulates observed at the bottom of fusion devices post-operation (Winter, 1998) [FIG]. These particles can not only change the transport properties of a plasma, they also migrate and can wind up compromising components that require active cooling, or block necessary gaps within the chamber (Winter, 1998).

4.3 Plasma sources in semiconductor processing

Research of dusty plasmas in industrial applications took off during the 1990's and was driven by the formation of particles in reactors (Thomas Jr., 2015) which used radio frequency (RF) plasmas for semiconductor and thin-film processes. Capacitavely Coupled Plasmas are the most frequently used in the creation/development of the semiconductor wafers that are found in all of our electronic devices. Etching is of particular interest, as we attempt to scale down electronics from the micro to the

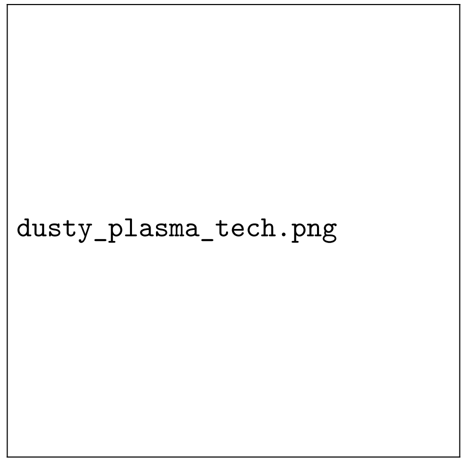

Figure 4.3: Dusty plasma in semiconductor processing devices (REF).

nano scale (Seo and Kim 2002). As we continue to downsize technology, challenges resulting from increased resistance and capacitance in integrated circuits (IC) have been met with the introduction of low-dielectric (low-k) materials (REFS). The fluorocarbon plasmas used to etch these materials leave behind a fluorocarbon polymer (CFx) residue which is subsequently cleaned using O2 and H2 plasmas. Plasma-induced damaged during this cleaning process changes the dielectric properties of the material (REFS), posing further challenges to IC manufacturing. Plasma modeling can play a role in the innovation of efficient and effective solutions by informing the most optimal combination of plasma characteristics such that interactions with the CFx residue minimizes damage to the low-k material (REFS).

Chapter 5

Description of the model

There is no problem more difficult to solve than that created by ourselves.

Felix Alba-Juez

A well-designed computational model can provide a wealth of information that is otherwise difficult to determine from experiments or observation alone. The Particle-In-Cell (PIC) technique, a powerful and widely used tool, simulates the motion of the charged particles in a plasma.

PIC simulations range from simple to complex depending on the initial assumptions made when appraising a problem. The simplest you can get is a 1D electrostatic plasma comprised of electrons and ions that are either stationary or moving so slowly that we can ignore dynamic effects. We can continue to add dimensions of motion, introduce magnetic fields, allow charges to move at appreciable rates, allow charges to interact with one another, or add massive objects such as dust particles into the mix. Each of these features adds to the complexity of the algorithm, and increases computational strain.

Some things we know about the problem at hand: we are working with a low-density glow discharge dusty plasma that has no applied or self-induced magnetic fields, and we are focusing in on a very small region of that plasma which contains a single dust particle that has already accumulated a net negative charge.

Relative to the massive ions, dust and neutral atoms, we can, as is often done in PIC code, consider the electrons to be a fluid [1] (Maiorov et al., 2000) (Other REFS). Furthermore, as we are working with a low density plasma, we can treat massive particles discrete particles (REFS). For simplicity, the dust grain will remain stationary and we will ignore particle interactions with neutral atoms.

With these validating assumptions in hand, we can make the choice to use an electrostatic model in which the Boltzmann relation can be applied to electron species. The latter choice will help to reduce computational time, as we will not have to follow the trajectories of every individual ion *and* electron– just the ions.

In the sections that follow, we will outline the equations that need to be solved in order to simulate our equation, and outline how the PIC algorithm does this.

5.1 Governing Equations

For an electrostatic plasma we need only concern ourselves with the electric potential and electric fields. These are given to us through [EQ 2.1], where the electric field is given as,

$$\vec{E} = -\nabla\phi. \tag{5.1}$$

This turns [EQ 3.1] into Poisson's equation,

$$\nabla^2\phi = -\frac{\rho}{\epsilon_0} \tag{5.2}$$

We can then pair these equations with a recasting of Newton's second law of motion (F = ma) to move the particles.

[1] This is due to the fact that, given their low mass, electrons have a much higher velocity than the other species.

$$\frac{d\vec{v}}{dt} = \frac{q}{m}\vec{E}, \tag{5.3}$$

$$\frac{d\vec{x}}{dt} = \vec{v}. \tag{5.4}$$

[NEEDS BIT ON BOLTZMANN RELATION FOR IONS]

5.2 Particle in Cell Method

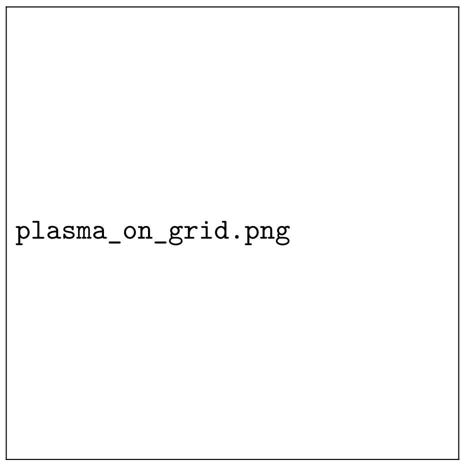

Figure 5.1: 1d vs 2d plasma on a grid. In 1-dimension, charges are infinite sheets while in 2-dimensions each macrocharge encompasses a grid cell.

As illustrated in [FIG 5.2], the standard outline of a PIC code proceeds as follows: deposit the particles across the grid, determine the charge density, solve for the potential and electric fields, and use the electric fields to determine the force that will then cause the particles to accelerate, update particle position and velocities:

$$(1)\ \vec{x}_{particle} \rightarrow (2)\ \rho_{grid} \rightarrow (3)\ \phi_{grid} \rightarrow (4)\ \vec{E}_{grid} \rightarrow (5)\ \vec{F}_{particle}.$$

This is looped until a steady state is reached.

Steps (1) to (2) and (4) to (5) are achieved using a weighting scheme, while the others require a recasting of the equations to a finite-difference form. The use of a

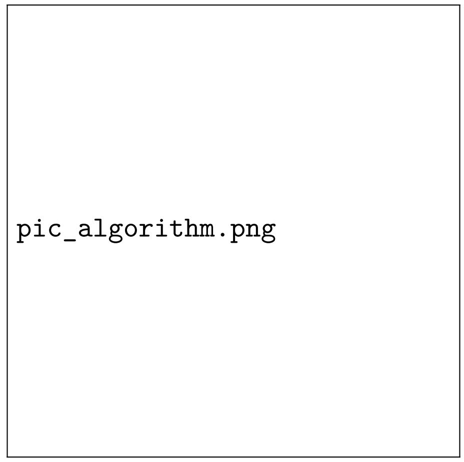

Figure 5.2: A flow chart laying out the Particle-in-Cell algorithm.

spatial grid in 2-dimensions leads to finite-sized square particles, and while they are fairly symmetric, this can lead to some unwanted effects due to the fact that the forces on the particles will depend on their position within the cell in addition to the distance to other particles.

A more comprehensive look at plasma simulations, including how to scale up to 3D, can be found in the seminal work Birdsall and Langdon (2005). Let's take a closer look at the individual steps.

5.2.1 Weighting (charge deposition)

Once we've established our spatial domain, the first thing we need to do is distribute our charged particles, a process called 'weighting'. There are several ways to go about this, but one of the common schemes is to use a first order weighting scheme [FIG], (also called area weighting or linear interpolation). This is given by the following,

$$w_1 = \frac{(\Delta x - x)(\Delta y - y)}{\Delta x \Delta y}, \tag{5.5}$$

$$w_2 = \frac{x(\Delta y - y)}{\Delta x \Delta y}, \tag{5.6}$$

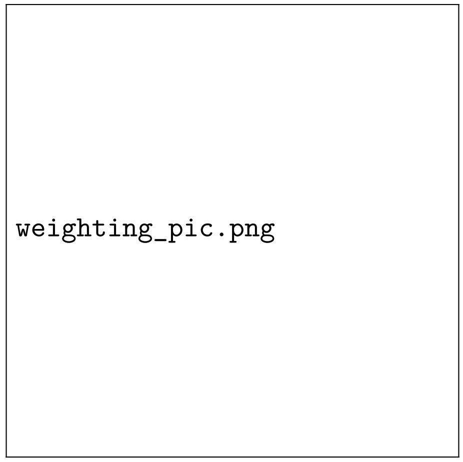

Figure 5.3: Area weighting used to deposit charge.

$$w_3 = \frac{(\Delta x - x)y}{\Delta x \Delta y}, \tag{5.7}$$

$$w_4 = \frac{xy}{\Delta x \Delta y}. \tag{5.8}$$

Here, Δx and Δy are fractional directions in the x and y directions respectively and $\Delta x \Delta y$ is the cell volume. At the boundaries of our domain, we have to account for the fact that only half of the grid cells are contributing (REF).

Once the charge has been distributed, the charge density, ρ, is then computed by dividing the total charge by the volume of each grid cell.

5.2.2 Solving for Φ

Once we have the charge density in hand, we can plug that into the right hand of [EQ 3.2] and put it into a finite-difference form using central differencing (also called a five-point difference),

$$\frac{\phi_{i-1,j} - 2\phi_{i,j} + \phi_{i+1,j}}{\Delta x^2} + \frac{\phi_{i,j-1} - 2\phi_{i,j} + \phi_{i,j+1}}{\Delta y^2} = -\frac{\rho_{i,j}}{\epsilon_0}. \tag{5.9}$$

The right hand side of [EQ 5.9] is recast in a way that allows us to use an iterative technique to solve for the potential which is based on the Gauss-Siedel method [2] (REF),

$$\phi = \frac{1}{M_{ii}}\left[b_i - \sum_{j=1}^{i-1} M_{ij}\phi_j - \sum_{j=1+1}^{n} M_{ij}\phi_j^k\right], \tag{5.10}$$

where M is a stencil matrix, and b is the density term which incorporates the Boltzmann treatment of the electrons,

$$b = \left(\frac{e}{\epsilon_0}\right) n_i - n_0 e^{\phi/T_e}, \tag{5.11}$$

where e is the elementary charge of the electron, $epsilon_0$ is the vacuum permittivity, ni is the ion density, n_0 is the background density, and T_e is the electron temperature (often measured in eV).

5.2.3 Field solvers

With the potential calculated, we can put that into [EQ 5.1] and use a two-point difference to calculate the electric field at those points,

$$\vec{E}_x = \frac{\phi_{i+1,j} - \phi_{i-1,j}}{2\Delta x}, \tag{5.12}$$

and

$$\vec{E}_y = \frac{\phi_{i,j+1} - \phi_{i,j-1}}{2\Delta y}. \tag{5.13}$$

5.2.4 Add/Move Particles

Using the force (EQ 5.3), we then advance our particles one time step Δt. To do this, the leapfrog method is commonly employed [FIG 5.4] by taking EQ. 5.3 and 5.4 and replacing them with a finite difference,

[2] Up on your linear algebra? The Gauss-Siedel method is one in which you are solving the matrix equation Ax = b for x.

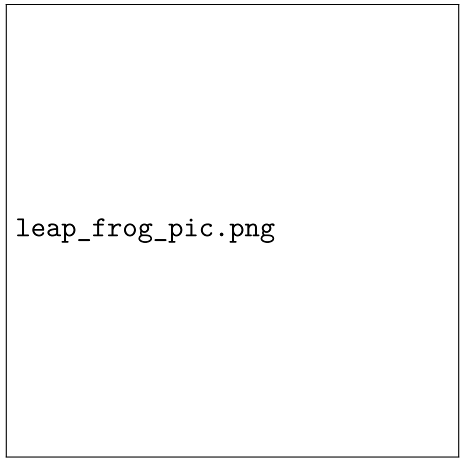

Figure 5.4: An illustration of the leap-frog method. We time-center the force while advancing the particle's velocity, and likewise time-center the velocity while advancing a particle's position.

$$\frac{v_{new} - v_{old}}{\Delta t} = \frac{q}{m}\vec{E} = \vec{F}_{old}, \tag{5.14}$$

and

$$\frac{x_{new} - x_{old}}{\Delta t} = v_{new}. \tag{5.15}$$

This requires the velocity to be pushed back to a negative half time step using the force calculated at $t = 0$, and for resulting calculations (i.e. the electric field) to be adjusted in such a way that they appear at the same time.

Chapter 6

Simulation results

> Physicists like to think that all you have to do is say, these are the conditions, now what happens next?
>
> *Richard Feynman*

[CHAPTER IN PROGRESS]

6.1 Simulation parameters

[TABLE: SIMULATION PARAMETERS (NX, NY, DT...)]

[TABLE: DUST PARAMETERS (d_D, V_D...)]

The presence of a charged dust particle influences the local electric potential and electric field in the background plasma which, on a larger scale, affects the properties of that plasma (REF).

In this simulation, a dust grain with a diameter ranging from micrometers to millimeters, and with a negative potential (implying that dust charging has already occurred) was placed in a low-temperature, low pressure argon plasma. Electrons were treated as a background fluid, allowing our focus to shift to the behavior of the flowing ions in the presence of this massive charged particle. Simulated ion macroparticles were introduced along the y-axis, and their paths were calculated through the integration of the equations of motion (EQ) and (EQ) with the force given by (EQ). (TABLE) summarizes the dust parameters while (TABLE) summarizes the plasma parameters.

Particle-particle interactions were not taken into account. Ions whose paths led them within the boundaries of the simulated dust particle, or past the boundaries were removed from the simulation.

6.2 2D PIC Results

6.2.1 Modification of plasma by dust particle

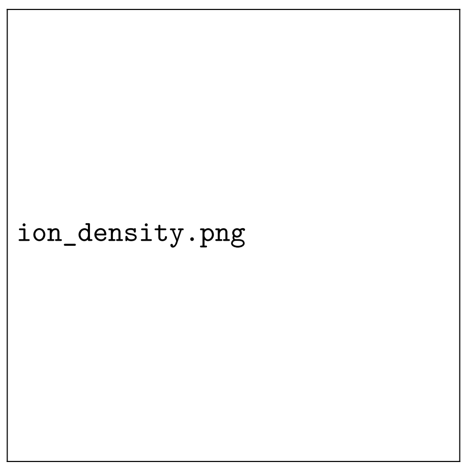

Figure 6.1: Ion density around a single dust charge in an ambient plasma for various initial conditions: (LIST)

[FIG] shows the resulting ion density after the simulation has reached a steady state (TIME?) for dust particles of differing sizes and potential.

[FIG] shows how the electrostatic potential of the local plasma changes.

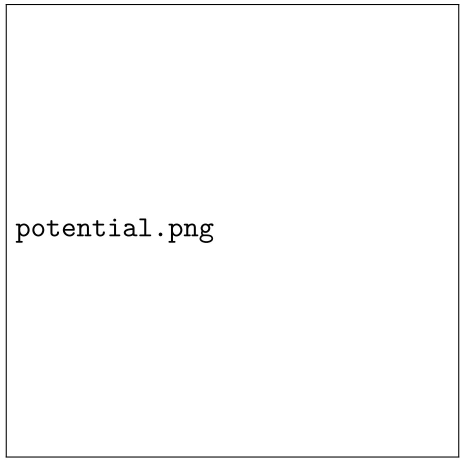

Figure 6.2: Electric potential of the plasma for various initial conditions: (LIST).

6.3 Why open source?

For those without the resources to access commercial products, (i.e. students and small research groups), free and open source tools offer a practical means by which to pursue research interests. From a philosophical standpoint, free and open source platforms offers a level of transparency that encourages community contribution. Software can be developed and shared with others either for critique, as an invitation for collaboration, or as a tool for others to take and modify for their individual needs.

Python, a popular open source programming language, is widely used in fields such as astronomy (REF), in part because of it's ease of use, versatility, and readability compared to other lower level programming languages. The community aspect of code developing and code sharing is reinforced through the use of web-based hosting services such as GitHub [1], and interactive projects such as Jupyter Notebook[2].

For all it's advantages, there are some disadvantages to consider; this is especially true when wading into under developed (or undeveloped!) programming territory. One of the main cost-benefit analyses to consider is time: commercially available

[1] https://github.com/

[2] http://jupyter.org/

software (e.g. MATLAB) have built-in features and functions that help you get to the science faster than trying to build them on your own (even with help from external sources).

While the above work was conducted in MATLAB, the Appendix contains 1D code that was developed in Python and includes a link to the GitHub repository where all of the source code (1D and 2D) can be found.

6.4 Future work

A fully 3D self-consistent simulation that includes molecular dynamics and that account dust charging is the type of model that would most closely capture the dynamics of a dusty plasma; however, a 2D model that takes interparticle interactions into account would be a natural place to start. Interparticle interactions (collisions) would be addressed through the addition of a Monte Carl Collisions handle which is an algorithm that checks each particle for a collision, determines the type of collision (inelastic, elastic) and assigns the appropriate reaction (ionization, excitation, etc.).

Using the simplified model as is, the effects of ion flow around dust particles of different shapes ("rough" vs "symmetric") could be investigated, as well as the dynamics resulting from collections of dust particles.

References

de Angelis, U., Tolias, P., Ratynskaia, S. (2012). Effects of dust particles in plasma kinetics: Ion dynamics time scales. Physics of Plasmas, 19(7), 073701.

Baklanov, M. R., de Marneffe, J. F., Shamiryan, D., Urbanowicz, A. M., Shi, H., Rakhimova, T. V., ... Ho, P. S. (2013). Plasma processing of low-k dielectrics. Journal of Applied Physics, 113(4), 4.

Blandón, J. S., Grisales, J. P., Riascos, H. (2017, June). Electrostatic plasma simulation by Particle-In-Cell method using ANACONDA package. In Journal of Physics: Conference Series(Vol. 850, No. 1, p. 012007). IOP Publishing.

Donnelly, V. M., Kornblit, A. (2013). Plasma etching: Yesterday, today, and tomorrow. Journal of Vacuum Science Technology A: Vacuum, Surfaces, and Films, 31(5), 050825.

Garscadden, A. (1994). Nucleation, growth, and morphology of dust in plasmas. Pure and applied chemistry, 66(6), 1319-1326.

Kanarik, K. J., Lill, T., Hudson, E. A., Sriraman, S., Tan, S., Marks, J., ... Gottscho, R. A. (2015). Overview of atomic layer etching in the semiconductor industry. Journal of Vacuum Science Technology A: Vacuum, Surfaces, and Films, 33(2), 020802.

Lacci, K. M., Dardik, A. (2010). Platelet-rich plasma: support for its use in wound healing. The Yale journal of biology and medicine, 83(1), 1.

Markidis, S., Henri, P., Lapenta, G., Ronnmark, K., Hamrin, M., Meliani, Z., Laure, E. (2013). The fluid-kinetic particle-in-cell solver for plasma simulations. arXiv preprint arXiv:1306.1089.

Pinhão, N. (2007). pypk-A Python extension module to handle chemical kinetics in plasma physics modeling.

Ricci, P., Lapenta, G., de Angelis, U., Tsytovich, V. N. (2001). Plasma kinetics in dusty plasmas. Physics of Plasmas, 8(3), 769-776.

Seo, H., Kim, S. B., Song, J., Kim, Y., Soh, H., Kim, Y. C., Jeon, H. (2002). Low temperature remote plasma cleaning of the fluorocarbon and polymerized residues formed during contact hole dry etching. Journal of Vacuum Science Technology B: Microelectronics and Nanometer Structures Processing, Measurement, and Phenomena, 20(4), 1548-1555.

Shoeb, J., Kushner, M. J. Comparison Of Cleaning And Damage Of Porous Low-k SiCOH In Ar/O2 And He/H2 Plasmas With UV/VUV Fluxes.

Shoeb, J., Wang, M. M., Kushner, M. J. (2012). Damage by radicals and photons during plasma cleaning of porous low-k SiOCH. I. Ar/O2 and He/H2 plasmas. Journal of Vacuum Science Technology A: Vacuum, Surfaces, and Films, 30(4), 041303.

Shukla, P. K., Mamun, A. A. (2002). Introduction to dusty plasma physics. IoP Publishing.

Somashekhar, A., Ying, H., Smith, P. B., Aldrich, D. B., Nemanich, R. J. (1999). Hydrogen Plasma Removal of Post-RIE Residue for Backend Processing. Journal of The Electrochemical Society, 146(6), 2318-2321.

Appendix A

1D PIC Code in Python

A.1 Two-stream instability

[TABLE: SIMULATION PARAMETERS (NX, NY, DT, N...)]

two_stream.png

Figure A.1: Top: Two opposing electron streams moving in a background of ions. An instability will develop due to charge bunching as the streams move through each other. Bottom: Evolution of the electron-electron two stream instability in phase space at times a) t=0, b) t = 49, c) = 99, and d) t = 199. The initial velocities are ±0.2, the grid has 1000 cells, there are 20000 particles per beam, and the beams have an initial sinusoidal perturbation of mode 2.

Building computational environments is a modular process. As such, it's critical to test as you go. We'll start with the 1D case which consists simply of electrons moving in a background of ions. To this program, we'll apply a benchmark test: the two-stream instability. This problem, illustrated in [FIG] models two opposing streams of electrons. [FIG] shows the time evolution of this stream, given an initial perturbation, in phase space (position vs velocity instead of the familiar position/velocity vs time).

What do we mean by perturbation? Well consider that a perfectly stationary beam of like charges will be equally spaced from each other in some equilibrium configuration. Disturbing a single particle from its equilibrium position is going to oscillate around that point at some frequency. To disturb *all* of the charged particles in this beam, as is the case in these simulations, we get a sinusoidal response where a particle's position is described as follows,

$$x = x_0 + x_1 cos\left(x_0 \frac{2\pi n}{L}\right), \tag{A.1}$$

Here L is the length of the length of the domain and n is the mode of excitation of the wave.

Allow that initial perturbation to run over time, and adding the second opposing beam in the mix, you get an instability that grows exponentially over time (Landon and Birdsall, 2005). We see this play out in [FIG].

Appendix B

Sample Python code

For the most current version of source code used in this project (in both 1D and 2D) visit my GitHub or copy and paste the url: https://github.com/space-isa/PIC. Included in the repository is an annotated notebook created using the Jupyter platform, as well as the .py and .m files that can be downloaded and run. Take it, reproduce it, break it, change it, share it! Below is a sample of code showing the main cycle used to run the simulations presented in Appendix A.

```
start = time.clock()


NP = 0
print("Calculating...")


for count in range(0, iterations):


    q = np.zeros((nx, ny))
    rho = np.zeros((ny, nx))


    for p in range (1, NP):
        fi = (1 + p_pos[p,0]) / (dh)
```

```
    i = np.floor(fi)
    hx = fi - i

    fj = (1 + p_pos[p,1]) / (dh)
    j = np.floor(fj)
    hy = fj - j

    q[i,j] = q[i,j] + (1-hx) * (1-hy)
    q[i+1, j] = q[i+1, j] + hx * (1-hy)
    q[i, j+1] = q[i, j+1] + (1-hx) * hy
    q[i+1, j+1] = q[i+1, j+1] + hx * hy

rho = (sw + q_mp * q) / (dh * dh)

rho[0,:] = 2 * rho[0,:]
rho[-1, :] = 2 * rho[-1, :]
rho[:, 0] = 2 * rho[:, 0]
rho[:, -1] = 2 * rho[:, -1]

rho = rho + (1 * 10 ** 4)
#print(rho)

#potential solver
V = solver_2d(rho, tol, Ti, n0, V_ref, QE)



#E field solver
Ex = np.zeros([nx, ny])
```

```
Ey = np.zeros([nx, ny])
E = np.zeros([nx, ny])

#internal nodes
Ex[1:nx-1, :] = V[0:nx-2,:] - V[nx-(nx-2):, :]
Ey[0: ,1:nx-1] = V[:, 0:ny-2] - V[:, 2:ny]

#boundaries
#multiplied by 2 to keep values equivalent to internal nodes
Ex[0,:] = 2* (V[0,:] - V[1,:])
Ex[nx-1, :] = 2 * (V[nx-2,:] - V[nx-1, :])

Ey[:, 0] = 2 * (V[:,0] - V[:,1])
Ey[:,ny-1] = 2 * (V[:, ny-2] - V[:, ny-1])

Ex = np.floor (Ex / (2 * dx) )
Ey = Ey / (2 * dy)



#generate particles

if NP + np_in >= N:
    np_in = N - NP

#insert particles
#(NOTE: save this for after 2d environment works)

#x position
```

```
p_pos[NP:NP+np_in, 1:] = np.random.rand(np_in,1) * dh

#y position
p_pos[NP:NP+np_in, 1:] = np.random.rand(np_in,1) * Ly

#sample Maxwellian in x,y
#add drift velocity in x

p_velo[NP:NP+np_in, 1:] = v_drift + (-1.5 + np.random.rand(np_in,1)
     + np.random.rand(np_in,1) + np.random.rand(np_in, 1)) * vth
p_velo[NP:NP+np_in, 1:] = 0.5 * (-1.5 + np.random.rand(np_in,1)
     + np.random.rand(np_in,1) + np.random.rand(np_in, 1)) * vth

#move particles
p = 1

while p <= NP:

    fi = 1 + p_pos[p,0]/dx
    i = np.floor(fi)
    hx = fi - i

    fj = 1+ p_pos[p,1]/dy
    j = np.floor(fj)
    hy = fj - j

    E = ([Ex[i, j], Ey[i,j]]) * (1-hx) * (1-hy)
    E = E + ([Ex[i + 1, j], Ey[i + 1, j]]) * hx * (1-hy)
```

```
    E = E + ([Ex[i, j + 1], Ey[i + 1, j]]) * (1-hx) * hy
    E = E + ([Ex[i + 1, j + 1], Ey[i + 1, j + 1]]) * hx * hy

    F = QE * E
    a = F/MI

    p_pos[p, :] = p_velo[p, :] + a * dt
    p_velo[p, :] = p_pos[p, :] + p_velo[p, :] * dt

    if p_pos[p,1] < 0:
        p_pos[p,1] = -p_pos[p,1]
        p_velo[p,1] = -p_velo[p,1]

    p = p + 1
    print(p_pos, p_velo)

print("Clocking in at %s seconds"  % (time.clock() - start))
```