

Structure-Preserving Particle-In-Cell Methods for the Vlasov-Poisson-Landau System

by

Daniel S. Finn

May 04, 2023

A dissertation submitted to the
Faculty of the Graduate School of
the University at Buffalo, The State University of New York
in partial fulfillment of the requirements for the
degree of

Doctor of Philosophy

Computational and Data-Enabled Sciences Program

Copyright by
Daniel S. Finn
2023
All Rights Reserved

For Sabrina.

Acknowledgments

“At the moment of commitment the entire universe conspires to assist you.”

-Johann Wolfgang von Goethe

This quote was sent to me on the morning of my final defense. It immediately got me thinking about what (or who) it is that has been with me throughout this journey assisting me. If I were to restate this quote, I would say that at the moment of commitment *your* entire universe conspires to assist you. Each of us sees and experiences the Universe in a different way but it is these individual universes that assist us along our paths. We all depend on the people or institutions or something else in our lives at one point or another. Thus, we all have our own universe assisting us throughout our lives. So what is *my* universe? My family. At every stage of this journey, my family has been by my side, guiding me, listening to me talk about something I find interesting, or just being there for me when I needed it.

First and foremost, I would not be here without the constant love and support of my amazing wife, Sabrina. I am tremendously lucky to have someone like Sabrina to come home to every day and share all my life moments and achievements with. My parents, Ann and Marty, and my brothers, Ryan, Chris, and Jack, have always fostered my love of knowledge and exploration of the unknown, and for that, I am eternally grateful. They have each been role models to me in their own ways. I would also like to thank all of the many other family members who have encouraged me along this path. I am fortunate enough to have too many to thank individually here, but I am so thankful for the role each and every one of them

have played in my life.

This research would also not have been possible without the support of the newer additions to my universe; the friends, colleagues and mentors I have made along the way. I would like to thank my advisor, Professor Matt Knepley for his guidance and expertise in this research. Working with Prof. Knepley has not only greatly deepened my understanding of mathematics and physics as a whole but has helped me recognize the kind of scientist I hope to be. I feel lucky to have worked with an advisor who I consider not just a mentor but a friend. I would also like to thank my lab partner Dr. Joe Pusztay. This project was initiated prior to me joining the RELACS lab, and his grasp of the material and methodology has been hugely helpful throughout my time on the RELACS team. I would particularly like to thank both Prof. Knepley and Dr. Pusztay for the many hours they spent helping me debug my code when I most often just forgot how pointers work. Finally, I would like to thank my committee for their guidance in my research: Prof. Margarete Jadamec, Prof. David Salac, and Prof. Salvatore Rappoccio. Their deep knowledge of these topics along with their comments and suggestions helped me develop a piece of scientific research that I am tremendously proud of.

Abstract

Numerical solutions to the Vlasov-Poisson-Landau equations have important applications in the fields of plasma physics, solar physics and cosmology. The goal of this dissertation is to develop a structure-preserving particle-in-cell Vlasov-Poisson-Landau model using the Portable, Extensible Toolkit for Scientific Computation (PETSc). A toolkit for particle-in-cell methods has previously been implemented within PETSc, however, these tools are not accurate enough over long time scales to resolve the fine-scale plasma structures essential in understanding the overall dynamics within a plasma system. We extend many of the existing tools, and add new ones, to more accurately model fine-scale plasma structures in long-time simulations. In the particle-in-cell method, a hybrid discretization is constructed with (a) a grid of finitely supported basis functions to represent the electric, magnetic and/or gravitational fields and (b) a distribution of radial basis functions to represent the particle field. We test our collisionless Vlasov-Poisson solvers on the standard plasma test, Landau damping, which requires highly accurate methods to resolve the correct plasma oscillations. Discrete gradient dependent integrators are developed and used to solve the Landau collision operator. These integrators are fully conservative and monotonic in entropy, the latter being a property that previous collision operators have failed to achieve. We consider how this new toolkit could be used in modeling a variety of plasma systems, such as ultracold neutral plasma experiments, solar coronae, and galactic dynamics.

Table of Contents

Acknowledgments	iv
Abstract	vi
List of Tables	ix
List of Figures	x
1 Introduction	1
1.1 Solar Atmospheres	3
1.2 Ultracold Plasmas	6
1.3 Galactic Dynamics	10
1.4 An Overview of Numerical Methods	11
2 Software Infrastructure in PETSc	16
2.1 DMSwarm: Particles	17
2.2 DMPlex: Unstructured Grids	20
2.3 TS: Timesteppers	20
2.4 KSP: Linear Solvers	23
2.4.1 PC: Preconditioners	24
2.5 SNES: Nonlinear Solvers	25
2.6 Contributions to PETSc	26
3 Metriplectic Dynamics	28
3.1 Equations of Kinetic Theory	29
3.2 Metriplectic Formulation	33
4 Collisionless Kinetics	37
4.1 PETSc-PIC	38
4.1.1 Particle Methods	38
4.1.2 PETSc-FEM	40
4.1.3 Conservative Projections	41
4.1.4 Noise Reduction	42
4.2 A Survey of Relevant Timesteppers	44
4.2.1 Basic Symplectic Integrators	44
4.2.2 Discrete Gradient Integrators	50

4.2.3	Discrete Gradient Jacobian Considerations	53
4.3	A Mathematical Review of Landau Damping	55
4.3.1	A Numerical Approach to Landau's Dispersion Relation	59
4.4	Numerical Examples	62
4.4.1	Harmonic Oscillator	62
4.4.2	Landau Damping	66
4.5	Conclusions	74
5	Collisional Kinetics	77
5.1	Marker-Particle Discretization of the Collisional Bracket	82
5.2	Discrete Gradient Dependent Integrators	86
5.2.1	Singularities in DGDI	90
5.3	Numerical Examples	90
5.4	Conclusions	96
6	Conclusions	98
References		101

List of Tables

2.1	Table of existing timestepping methods in PETSc from [38].	23
4.1	Values for ω_r and γ for given values of \hat{k} from [23].	60
4.2	Table of convergence rates from TSConvest for a variety of timesteppers. . .	65

List of Figures

1.1	Various plasmas (or astrophysical systems) are charted with respect to how strongly or weakly coupled they are. We also consider which numerical regime (kinetic or hydrodynamic) each system typically falls into.	4
1.2	Strontium atomic and ionic energy levels from [75]. (A) Creation of an ultracold plasma starts with laser cooling and trapping neutral atoms using the $^1S_0 - ^1P_1$ transition at 461 [nm]. 1P_1 atoms are then photoionized with photons from a laser at ~ 412 [nm]. (B) Ions are optically imaged using the $^2S_{1/2} - ^2P_{1/2}$ transition in Sr+ at 422 [nm]. $^2P_{1/2}$ ions decay to the $^2D_{3/2}$ state 7% of the time, after which they cease to interact with the imaging beam. This does not affect most experiments because ions typically scatter fewer than one photon while the imaging beam is on.	8
1.3	The lifecycle and times scale of an ultracold neutral plasma. The electrons equilibrate first leading to the development of an ion space charge. Ion equilibration is the next process to occur, around the 1[μ s] time mark. Lastly, a “self-similar” expansion occurs in which the cloud retains (roughly) its gaussian shape.	8
1.4	The evolution of an ion distribution inside an ultracold neutral plasma experiment for (left) $T_e(t = 0) = 25$ [K] and (right) $T_e(t = 0) = 105$ [K]. Plots contain 1D slices through the density profile of experimental and simulation data. The hydrodynamic simulation overestimates the prominence of the central density enhancement as the plasma expands. Plots from [96].	9
2.1	DMSwarm Gaussian initialization of particle coordinates (left) and velocities (right). The particle coordinate Gaussian in this example was intentionally given a width wider than the cell size.	18
2.2	DMSwarm uniform initialization of particle coordinates (left) and velocities (right).	19
2.3	Phase space plot of particles initialized by DMSwarm API with Guassian (left) and uniform (right) coordinates and velocities.	19

2.4	DAG-based representation of a single tetrahedron in <code>DMPlex</code> from [85].	21
2.5	A chart of contributions made to the PETSc library as a part of this dissertation. All code listed in this graphic has been merged into PETSc/main.	27
4.1	Area preservation of a variety of common numerical integration methods for the harmonic oscillator [55].	46
4.2	A comparison of various approximations to the Landau dispersion relation and a numerical fit to data.	59
4.3	Complex plots of the Landau dispersion relation. The complex argument (phase) is represented by the color at a specific point and the magnitude is represented by the brightness. A box is added qualitatively to the plot which indicates the reduced region in which the algorithm finds roots.	60
4.4	A comparison of data from previous root finding methods [23] to the algorithm used in this project.	61
4.5	The energy of two particles in a harmonic oscillator for a variety of timestepping integrators. (Top) First through fourth order basic symplectic types are compared to implicit midpoint and discrete gradient integrators. (Bottom) Higher order integrators are zoomed in on.	63
4.6	The energy of two particles in a harmonic oscillator over time for a variety of timesteppers with time steps $\Delta t = 0.001$	64
4.7	The energy of two particles in a harmonic oscillator over time for a variety of timesteppers with time steps $\Delta t = 0.0001$	64
4.8	The maximum value of the electric field as a function of time for the one-dimensional linear Landau damping problem.	67
4.9	The total mass, momentum and energy error for the particle and finite element solve. The moment errors all converge to zero given a long enough time.	68
4.10	A comparison of basic symplectic integrators to a third order Runge-Kunge (RK) integrator for the Landau damping test with $N_x = 30$ and $N_v = 500$. RK methods are sensitive to the time step size, so two step sizes were run ($\Delta t = 0.01$ and $\Delta t = 0.001$). Both RK runs failed to capture any of the key dynamics in Landau damping while, even with reduced resolution, captures the correct oscillation and damping.	69

4.11 (Top) Particle per cell number convergence plots for $N_x = 100$ and (bottom) mesh convergence plots for $N_v = 8,000$	70
4.12 A comparison of various approximations for ω_r and γ to root finding results and numerical results from PETSc-PIC. Plots on the right are zoomed in on the region $0.0 \leq \hat{k} \leq 0.75$ to show the accuracy of each approximation before they diverge from the data.	71
4.13 Numerical results for varying charge densities, $\int f_0/V$, compared to zero finding data. The charge density is represented on the x-axis as the integral of the initial distribution over the domain volume.	72
4.14 A comparison of electric field oscillations given different charge densities. . .	73
5.1 Electron-positron temperature equilibration using the two collision operators: the DGDI framework without the Gonzalez expansion term and the existing particle basis operator with a theta method timestepper. The DGDI method without the Gonzalez term is equivalent to using an implicit midpoint integrator. In PETSc the implicit midpoint method is performed by the TSTHETA framework. These two methods are therefore mathematically equivalent. . .	92
5.2 Comparison of relative kinetic energy error for electron-positron temperature equilibration using (top) the DGDI framework without the Gonzalez expansion term and (bottom) the existing particle basis operator with a theta method timestepper.	93
5.3 The evolution of the entropy in electron-positron temperature equilibration using the DGDI framework without the Gonzalez expansion term.	93
5.4 Electron-positron equilibration test using the full DGDI framework.	95
5.5 Relative kinetic energy error and entropy evolution in electron-positron equilibration test using the full DGDI framework.	96

Chapter 1

Introduction

Magnetized plasmas are present anywhere in the Universe. This statement by Hans Goedbloed and Stefaan Poedts [50] is incredibly powerful when considering the formulation of numerical models for a variety of physical systems across the Universe. Ignoring dark matter, it is estimated that up to 90% of matter in the Universe exists in the plasma state. By developing accurate and scalable numerical methods for simulating plasma, one can study a multitude of problems in Astrophysics, Cosmology and laboratory Plasma Physics. These physical systems include, but are not limited to: thermonuclear fusion devices [4], solar atmospheres [20], galaxies [13], accretion disks [53], and the magnetospheres of planets and moons [133].

The word “plasma” was first used to describe an ionized gas by Irving Langmuir in the early 1920s while investigating electric gas discharges, as noted by his colleague Lewi Tonks [132]. However, it did not appear in publications until 1928 [87]. With the discovery of the ionosphere [63] and the development of numerous plasma models in astrophysical systems [15, 5], by the 1930s, the study of plasma physics had become an indispensable field of physics. The computational model we utilize today began to take shape in 1936 when Lev Landau first formulated a simple kinetic model for the description of charged particles in a plasma by performing Coulomb collisions [83]. This model is based on the Boltzmann

equation [18] and now referred to as the *Landau-Fokker-Planck equation*. Two years later, Anatoly Vlasov built on Landau’s formulation with a new description that included “long-range force” [135], which Vlasov determined to be essential for an adequate description of plasmas oscillations. Finally, ten years after Landau’s initial formulation of the problem, Landau predicted, for the first time, the damping of non-relativistic, collisionless plasma oscillations, or Langmuir waves in [84]. The basic concept proposed in this paper, that a conservative phenomenon exhibits irreversible behaviors, has since influenced hundreds of papers and become one of the guiding principles in plasma physics. Thus, the phenomenon is now referred to as *Landau damping*. In his 1946 paper [84], Landau used the solution to the Cauchy problem for the linearized Vlasov-Poisson equation around a spatially homogeneous Maxwellian equilibrium. Landau solved the equation, analytically, using Fourier and Laplace transforms and concluded that the electric field damps exponentially and that the decay is a function of the wave number, k . In [17], Bohm and Gross provide a simple explanation for the damping in plasmas. In essence, plasmas exhibit a tendency to remain approximately field free. Therefore, if electric fields are introduced, either by external disturbance or by an incomplete space charge neutralization, the newly introduced fields will be forced out by a reaction from the free charges. The explanation given by Bohm and Gross can be broadly applied to a variety of physical systems. For example, in the realm of solar physics, Landau damping has been considered a major source of dissipation of Alfvénic turbulence in the solar wind [37].

Vlasov and Landau’s work, combined with the work of James Clerk Maxwell¹ [92] in the 1860s, provided a framework for modeling collisionless and collisional electromagnetic plasmas that is still a pillar of plasma modeling today. This system has thus been labeled the *Vlasov-Maxwell-Landau* set of equations, with the non-magnetic *Vlasov-Poisson-Landau* simplification. Throughout the rest of the 20th century, physicists and computer scientists

¹This is of course a collection of previous work from Faraday, Ampère, Gauss and others that Maxwell organized into a standardized set of equations. The modern form of the equations was written for the first time by Heaviside in 1893 [62].

discovered a variety of other plasma phenomena, such as two-stream instability and plasma echoes. Many of these phenomena provided excellent test beds for verifying the accuracy of modern computational models.

Despite being vital in furthering our understanding of plasma systems throughout the universe, constructing long-time² accurate computational plasma methods remains a challenge. Preservation of key thermodynamic principles, such as conservation of moments and monotonicity of entropy, is often lost in the discretization of continuum equations. The primary goal of this research is to provide a numerical framework and set of implementations for the Vlasov-Poisson-Landau system of equations. Furthermore, there exists today an apparent lack of uniformity in notation and methodology between the different disciplines, even when they utilize the same core equations and numerical formulations. Therefore, we seek to apply the same techniques to a variety of scientific applications with the goal of unifying work into a single cohesive framework of kinetics. In the next sections, we consider a variety of “plasma” systems for which this research would be applicable.

1.1 Solar Atmospheres

Identifying and understanding the mechanisms responsible for heating the solar corona to several million degrees hotter than the photosphere, the next inward layer of the sun, is at the core of the *coronal heating problem* [77]. First suggested by Thomas Gold in a discussion at the 22nd International Astronomical Union (IAU) Symposium in 1965 [51], one common explanation for the observed coronal heating is small energetic bursts of order 10^{24} [*erg*] resulting from small-scale magnetic reconnection heating events. The idea of impulsive heating began to build real traction in the community after Eugene Parker introduced the idea of *nanoflares*, i.e. small scale heating events that can occur as a result of

²Here, the term “long-time” is in relation to the number of time steps, which is normally limited by either stability or accuracy concerns. We define “long-time” as the time required for global structures, such as runaway electrons, to build up. For the purposes of this dissertation, “long-time” refers to anything greater than 10^6 time steps.

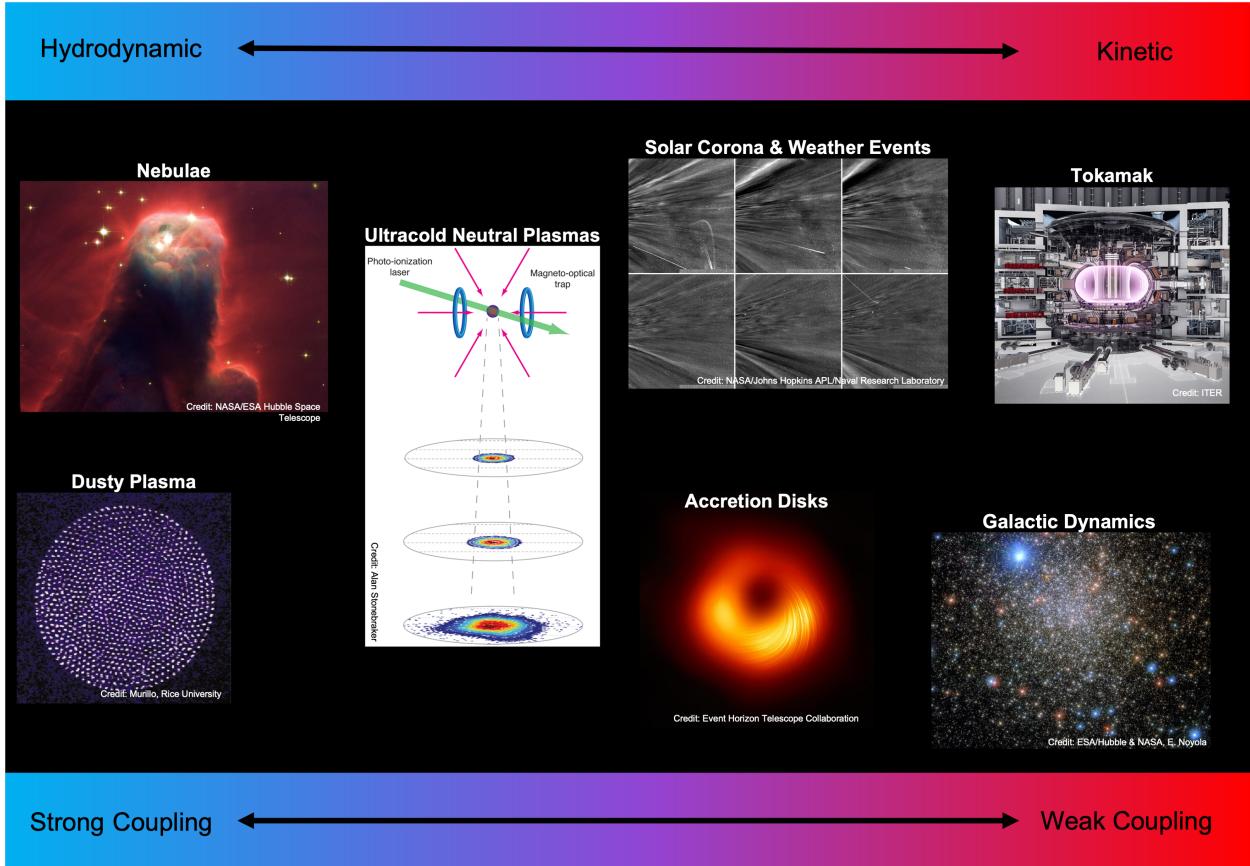


Figure 1.1: Various plasmas (or astrophysical systems) are charted with respect to how strongly or weakly coupled they are. We also consider which numerical regime (kinetic or hydrodynamic) each system typically falls into.

magnetic reconnection, as a unit of coronal energy release [111]. Understanding how magnetic reconnection operates in the solar atmosphere is an essential piece in this solar heating model. In addition to potentially heating the corona, magnetic reconnection manifests in nearly all solar phenomena, such as the formation and structure of the solar wind, flares and coronal mass ejections (CMEs).

The change in magnetic connectivity leads to the conversion of stored energy into bulk flow, thermal and non-thermal energy. At kinetic scales, collective fluid motions degenerate over time into heat. These diffusive, and dissipative, processes can be driven by either collisional [108] or collisionless [69, 109] properties of the plasma. For example, for standard magnetic reconnection to occur, both electrons and ions decouple from the magnetic field,

and the dissipation region develops a two-scale structure of diffusive electron and ion regions. Tearing of an initial plasma current sheet leads to the formation of a series of primary magnetic islands, or plasmoids.

An alternate model for the coronal heating problem has been proposed based on the predictions of Landau. Section 4.4.2 will briefly review a derivation of Landau damping from [88], however, the key concept behind Landau damping in the solar corona is as follows. If the majority of particles have velocity less than the phase velocity of the field wave, which in the case of the solar corona are ion-acoustic or Alfvén waves, energy is drawn from the wave to the particles leading to the damping of the field and a growth in the number of high energy particles [34]. When surface Alfvén waves propagate into an inhomogeneous plasma, they are first converted to kinetic Alfvén waves as the mode conversion occurs, and then the plasma is heated when the kinetic Alfvén wave energy is dissipated owing to the Landau damping [124]. Developing high-fidelity collisionless models, such as those considered in this project, is key to understanding this phenomenon.

The difficulty in proving the cause of coronal heating is twofold. The first obstacle has been the inability to make direct observations of the processes expected to be at the center of the heating mechanisms [113, 24]. This is primarily due to inadequate spectral coverage of instruments, the efficiency of thermal conduction, and non-equilibrium ionization [24, 140, 10]. With NASA’s 2018 launch of the Parker Solar Probe as well as similar missions, such as CuSP and the Solar Orbiter from ESA, there now exists a plethora of data and observations on solar activity that can be used in comparisons with larger scale numerical simulations. Further observations have shown the development of plasmoids in flare current sheets following coronal mass ejections (CMEs) [131, 72] inside the solar atmosphere, as well as in interactions between laboratory created laser-ablated copper and plastic plasmas [147].

The second difficulty in determining the heating mechanisms has been in developing high-fidelity numerical frameworks that can resolve small-scale particle interactions in plasmas over long time scales, while preserving conservation principles as well as other thermodynamic

principles. Magnetohydrodynamic (MHD) models have long been used to model the solar atmosphere. These models have been successful in modeling large scale solar plasma systems for decades, however, they fail to accurately capture the kind of small-scale interactions that are becoming an essential piece to the complete picture of solar heating, as well as other solar phenomena. Kinetic approaches are superior in these regimes but are limited in their potential for larger scale modeling. Therefore, one of the applications of this research could be to provide a set of kinetic implementations and tools capable of being coupled to existing, or later developed, MHD models.

1.2 Ultracold Plasmas

The quest for validation of computational models presents an interesting set of obstacles for the field of computational physics. Models of astrophysical systems, such as galactic collisions or accretion disks, can have time scales in the millions of years. Similarly, working thermonuclear fusion devices are still uncommon. For that reason, finding data to validate models is challenging. Ultracold neutral plasmas are a recent class of plasmas that are created in laser-cooling experiments. In contrast to conventional plasmas, which require temperatures of thousands of Kelvin or more to exist, ultracold neutral plasmas are created by photoionizing laser-cooled strontium atoms captured in a magneto-optical trap (MOT) near the ionization threshold with electron temperatures being in the range $1 - 1000 [K]$ and ion temperatures being around $1 [K]$. Furthermore, particles in ultracold neutral plasmas fall in, or near, the strongly coupled plasma regime [71], whereas conventional plasmas (at least those considered in this work) are weakly coupled. This is to say, the coupling constant,

$$\Gamma = \frac{e^2}{4\pi\epsilon_0 a k_B T}, \quad (1.1)$$

where e is the electron charge, ϵ_0 is the permittivity of free space, k_B is the Boltzmann constant, T is the electron temperature, and a characterizes the separation between particles

at a given density, is between 1 and 10^6 for ultracold neutral plasmas, while most other common plasma systems (e.g. Tokamaks, solar coronae, etc.) fall below $\Gamma = 1$. In these strongly coupled plasmas, considerable deviations in plasma properties from ideal conditions can occur, such as how strongly Maxwellian the velocity distribution remains. This is more clear when considering the timeline of the plasma processes in ultracold neutral plasma experiments, as described by Killian et. al. in [75].

The first ultracold plasma experiments were performed in 1993 at the National Institute of Standards and Technology using metastable xenon [136], but experiments today are run with other atoms, such as strontium [103], calcium [33], rubidium [137], and cesium [120]. In [75], Killian et. al. use strontium atoms in experiments, as shown in Figure 1.2. Trapped atoms are laser cooled using the ${}^1S_0 - {}^1P_1$ transition at 461 [nm]. A typical initial density distribution, $n(r)$, of the cloud is of spherical Gaussian shape,

$$n(r) = n_0 \exp\left(\frac{-r^2}{2\sigma^2}\right), \quad (1.2)$$

where σ is the distribution width, typically in the range $\sigma = 200 - 1500 [\mu\text{m}]$ for these experiments, n_0 is the distribution height, and r is the radius. 1P_1 atoms are then photoionized using a pulsed dye laser at 412 [nm]. The number of atoms ionized by the laser can be tuned by varying the energy of the photoionizing pulse. The evolution of the photoionized plasma is shown in Figure 1.3. The excess energy above the ionization threshold is distributed between the electrons and ions, with the majority going to the electrons. The electrons quickly equilibrate. However, as electrons escape the trap, a non-neutrality is formed creating a potential hole which re-traps the electrons in the plasma. The ions, being larger in mass than the electrons, remain largely stationary as the electrons equilibrate and the non-neutrality is formed. Ions reach equilibrium on the timescale $t\tilde{\omega}_{pi}^{-1}$, where ω_{pi}^{-1} is the ion plasma frequency. The final stage of the ultracold neutral plasma cycle is an adiabatic expansion, which can be accurately modeled as purely hydrodynamic.

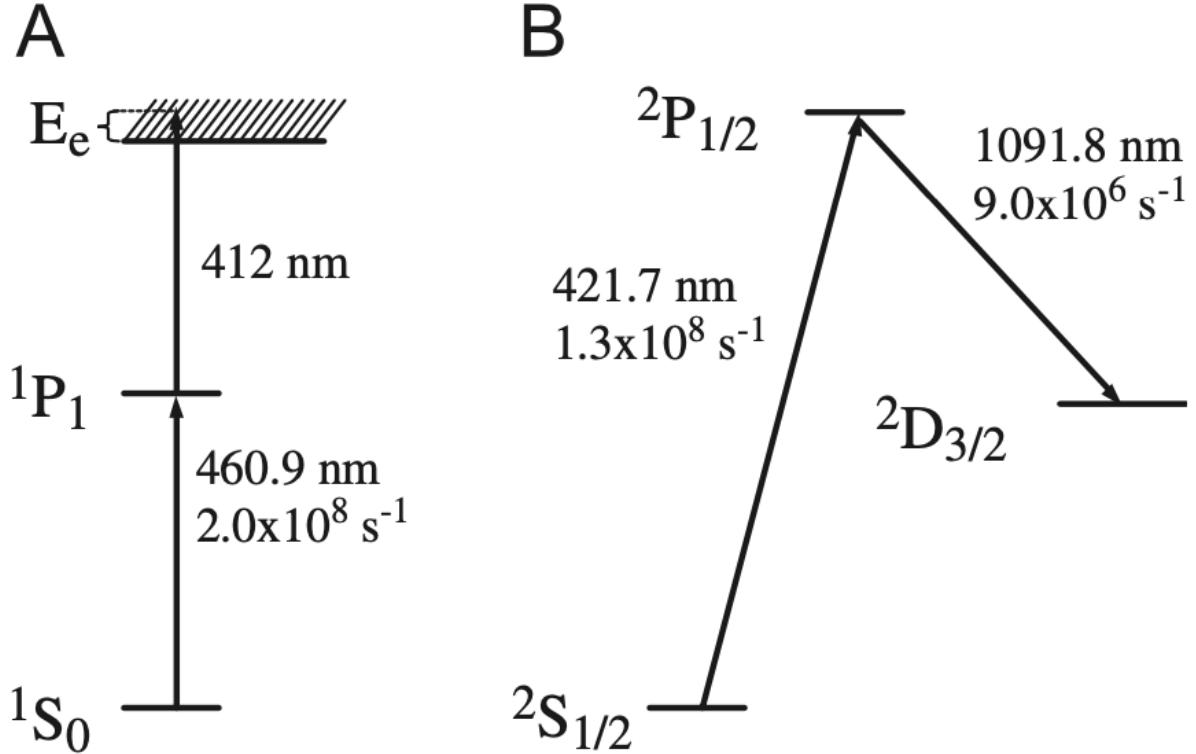


Figure 1.2: Strontium atomic and ionic energy levels from [75]. (A) Creation of an ultracold plasma starts with laser cooling and trapping neutral atoms using the $^1S_0 - ^1P_1$ transition at 461 [nm]. 1P_1 atoms are then photoionized with photons from a laser at ~ 412 [nm]. (B) Ions are optically imaged using the $^2S_{1/2} - ^2P_{1/2}$ transition in Sr+ at 422 [nm]. $^2P_{1/2}$ ions decay to the $^2D_{3/2}$ state 7% of the time, after which they cease to interact with the imaging beam. This does not affect most experiments because ions typically scatter fewer than one photon while the imaging beam is on.

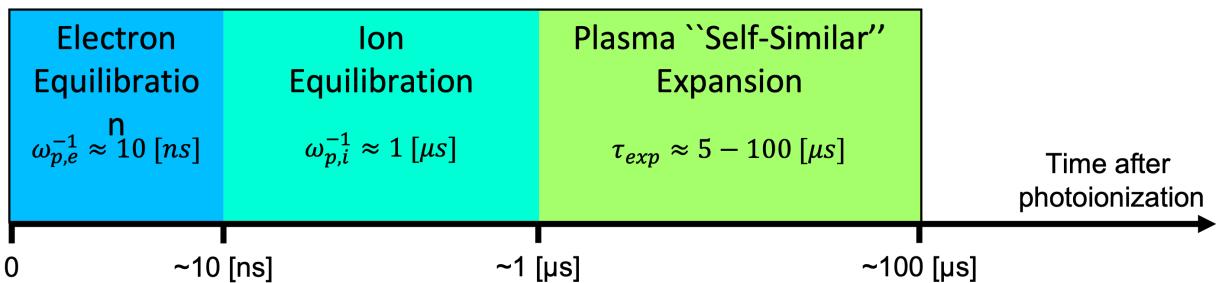


Figure 1.3: The lifecycle and times scale of an ultracold neutral plasma. The electrons equilibrate first leading to the development of an ion space charge. Ion equilibration is the next process to occur, around the $1[\mu\text{s}]$ time mark. Lastly, a “self-similar” expansion occurs in which the cloud retains (roughly) its gaussian shape.

We will compare various approaches to plasma modeling in more detail in future sections (see Section 1.4). However, it is pertinent to comment on the choice in numerical approaches for ultracold neutral plasmas here. Previous studies [75, 96, 97] have utilized hydrodynamic (MHD), or fluid, methods for modeling the hydrodynamic expansion phase. These methods are relatively accurate in the expansion phase, but fail in representations of earlier phases, i.e. electron and ion equilibration. Hydrodynamic methods cannot represent non-maxwellian plasma distributions, a problem associated with *non-locality* conditions. The earlier phases in the ultracold neutral plasma experiments can be modeled with molecular dynamic (MD) methods [57]. However, combining the two methods to obtain a complete picture of the plasma equilibration and expansion is a difficult task. Alternatively, kinetic methods can capture non-maxwellian (or non-local) plasmas with relative ease. Furthermore, the experimental plasmas are of a small enough scale that the more expensive kinetic approach will not be too costly.

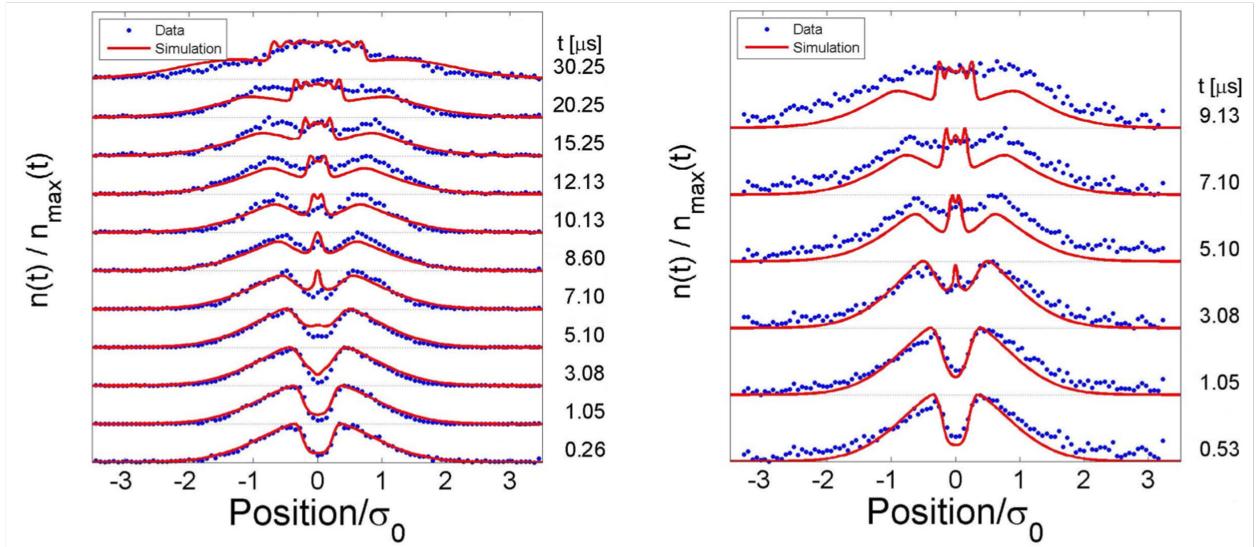


Figure 1.4: The evolution of an ion distribution inside an ultracold neutral plasma experiment for (left) $T_e(t = 0) = 25 \text{ [K]}$ and (right) $T_e(t = 0) = 105 \text{ [K]}$. Plots contain 1D slices through the density profile of experimental and simulation data. The hydrodynamic simulation overestimates the prominence of the central density enhancement as the plasma expands. Plots from [96].

Experimental and numerical studies of “streaming” plasmas [98, 96], i.e. ultracold neutral

plasmas with a central region removed, have shown that in certain cases, the ultracold neutral plasma contains noticeable kinetic dynamics during the expansion phase, as shown in Figure 1.4. The key parameter in this emergence of kinetic behavior is the ratio of the ion mean free path to the length scale of the plasma inhomogeneities as ions move through the gap and the populations collide. In the kinetic regime, the velocity distributions strongly deviate from local thermal equilibrium and show distinct populations of interpenetrating, counter streaming plasmas. The emergence of this kinetic behavior in streaming plasma experiments combined with an inability to model equilibration phases with hydrodynamic methods make ultracold neutral plasmas an excellent area for kinetic modeling.

1.3 Galactic Dynamics

Since the 1940s, the Vlasov-Poisson-Landau system (referred to as the *Boltzmann equation* or *collisionless Boltzmann equation*) has been used to build models of stellar and galactic systems. Chandrasekhar was the first to apply the concept of kinetics to stellar systems in 1942 [28]. In a series of papers, he described the collisional motion of a star in a stellar system using an analogy with Brownian motion [26]. However, he concluded that his description was lacking a dissipative term that he called *dynamical friction*. Incorporating the new dissipative term into his formulation leads to the Fokker-Planck equation, which Chandrasekhar used to estimate the rate of escape of stars from clusters [27]. Similar kinetic methods have since been applied to globular clusters by Spitzer [127] and Binney and Tremaine [13].

In 1962 [90], Donald Lynden-Bell proposed a phenomenon similar to Landau damping that he called *relaxation*, which occurs in an “artificial infinite uniform rotating system” of stars. Lynden-Bell claimed that a stellar system will reach equilibrium without the aid of any external dissipative mechanism. A few years later, in [89], Lynden-Bell applied this concept to observations of elliptical galaxies. He showed that the normal star-to-star interactions, which are one source of relaxation, are too weak to explain the “remarkable regularity” in

light distributions of the observed galaxies. Lynden-Bell hypothesized that some form of dissipation, or *violent collisionless relaxation*³ as he called it, must be driving the equilibration of the stellar distributions. More recent studies by Arad and Lynden-Bell [7] and Valluri et. al. [134] have shown that Lynden-Bell's original theory for the development of the equilibrium states of galaxies is incomplete.

Modern research in computational astrophysics seeks to build a more complex model of galactic equilibration as well as answer a variety of other questions about the kinetic behavior of self-gravitating systems throughout the universe. For example, many papers have been published on the topic of the origin of thin and thick disks within the Milky Way and other galaxies. The problem, first introduced by Gilmore and Reid in 1983 [48], revolves around the dual density profile within galactic structures, which remains largely unexplained by theoretical and computational models. Recent galactic surveys, such as GAIA-ESO [49], have sparked an increase of interest in this problem as they have provided key details about the makeup of these disks. Several mechanisms, such as galactic mergers and continuous heating of pre-existing thin disks, have been proposed to explain the existence of this dual profile structure. Like the other topics discussed in this section, kinetic approaches to modeling these gravitational systems are one of the best tools in advancing possible explanations [46, 110]. A toolkit containing a mix of accurate, scalable and efficient collisional Landau and collisionless Vlasov-Poisson solvers would prove extremely valuable in this field of research.

1.4 An Overview of Numerical Methods

Numerical methods for modeling plasmas, whether in space or laboratories, has traditionally been separated into three categories: N -body, kinetics and hydrodynamics. It is not accurate to say that any of these three approaches is superior to the other. Rather, each approach is efficient at modeling plasmas within a certain physical regime. These regimes can be defined by an assortment of parameters, such as the plasma coupling parameter (Γ)

³The assumption of collisionless behavior breaks down in galactic nuclei and globular clusters.

(1.1) used in Figure 1.1. Another key parameter that defines these regimes is the scale of interactions and activity that dominates the dynamics of a plasma, often characterized by the mean free path (λ).

The most basic model of a plasma, the N -body model, is derived by considering the motion of individual particles with Newtonian mechanics. N -body methods are the most matured numerical method of the three and thus have received decades of optimization and support leading to the highly accurate and scalable algorithms that exist today [6]. In the N -body model, the dynamics of each particle is characterized by the particle's position \mathbf{x} , velocity \mathbf{v} , mass m and charge q . The particles obey Newton's second law of motion,

$$\frac{d(\gamma m \mathbf{v})}{dt} = \sum F_{ext}, \quad (1.3)$$

where $\gamma = (1 - |\mathbf{v}|^2/c^2)^{-1/2}$ is the Lorentz factor, t is the time, c is the speed of light, and F_{ext} is the composition of all externally applied forces on the particle, which includes inter-particle forces. Newton's law can then be rewritten for each particle with the inclusion of the electromagnetic forces experienced in a typical plasma, e.g. the Lorentz force,

$$\frac{d(\gamma_i m_i \mathbf{v}_i)}{dt} = \sum_j q_i (\mathbf{E}_j + \mathbf{v}_i \times \mathbf{B}_j) + q_i (\mathbf{E}_{ext} + \mathbf{v}_i \times \mathbf{B}_{ext}), \quad (1.4)$$

with \mathbf{E}_j and \mathbf{B}_j denoting the electric and magnetic fields generated by particle j and \mathbf{E}_{ext} and \mathbf{B}_{ext} denoting external fields. Adding the relationship between the particle's position and velocity, we obtain the equations of motion for a general N -body model,

$$\frac{d\mathbf{x}_i}{dt} = \mathbf{v}_i, \quad (1.5)$$

$$\frac{d(\gamma_i m_i \mathbf{v}_i)}{dt} = \sum_j q_i (\mathbf{E}_j + \mathbf{v}_i \times \mathbf{B}_j) + q_i (\mathbf{E}_{ext} + \mathbf{v}_i \times \mathbf{B}_{ext}). \quad (1.6)$$

Alternatively, in gravitationally interacting models, the equations of motion can be writ-

ten as,

$$\frac{d\mathbf{x}_i}{dt} = \mathbf{v}_i, \quad (1.7)$$

$$\frac{d(\gamma_i m_i \mathbf{v}_i)}{dt} = G m_i m_j \frac{\mathbf{x}_j - \mathbf{x}_i}{||\mathbf{x}_j - \mathbf{x}_i||^3}, \quad (1.8)$$

where G is the gravitational constant.

N -body methods require the equations of motion (1.5) to be solved for every individual particle at each time step which, including the summation, means these methods are of order $\mathcal{O}(N^2)$. In typical N -body applications, such as inter-galactic or inter-stellar simulations, where N is on the order of 10^6 (maybe 10^9 in the largest cases), this level of computation complexity is acceptable. In typical plasmas, however, which have N on the order of 10^{12} ion-electron pairs per cm^3 and above [29], these methods become far too expensive. N -body models historically also struggle with the addition of particle collisions, drastically increasing computational loads.

A more computationally efficient method for denser, collisional plasma systems can be constructed by applying mean field electrostatics on a larger grid alongside a Landau collision operator. This is the foundational principle of metriplectic Particle-in-Cell (PIC) methods; overall accuracy of the simulation is traded for a computational speedup. New structure-preserving algorithms for simulating plasmas have been a major field of interest, due in part to the integral role these simulations play in fusion energy research and the International Thermonuclear Experimental Reactor (ITER) project [119]. These numerical frameworks, which we will consider in more depth in later sections, preserve invariants, respect thermodynamic inequalities and in general preserve the rich structure of the underlying model, while both reducing cost by, for instance, providing exact conservation with low resolution grids. Developments in the field of structure-preserving algorithms have been, to a large extent, due to the advances in geometric particle-in-cell (GEMPIC) methods [40, 82, 118, 122, 129, 143, 141]. GEMPIC algorithms are based on discretizing either the variational or Hamil-

tonian structure of the underlying kinetic models. These schemes are desirable not only because they preserve the energy of the system, but also because they typically guarantee a local algebraic charge conservation law, the preservation of the multisymplectic two-form and long-time fidelity and stability of simulations. Furthermore, the use of well understood methods such as adaptive mesh refinement [3], high-order accurate discretizations as well as the effective use of modern heterogeneous hardware can ameliorate computational challenges posed by the high dimensionality of PIC models.

While they are the chosen method for this research, kinetic methods are not without flaws. Primarily, while kinetic methods offer considerable computational speedup in comparison to N -body models, in large-scale plasma systems, there is still a high computational demand. Consider that the scale of magnetic reconnection in the solar atmosphere can be as large as $10^7 - 10^8$ [m], while time scales range from minutes to hours. Even within smaller systems, such as Tokamak fusion reactors, the number of particles required for complete models is on the scale of 10^{20} or larger. One solution to this problem, is to focus models on specific smaller scale plasma phenomena within larger systems. Another solution is to utilize a less-expensive fluid model of the plasma by assuming local thermal equilibrium and averaging the kinetic equations. Most commonly, the particle solver in a PIC code is more demanding than the field solve. Therefore, removing the particle solve can lead to a significant speedup of a plasma solver. Magnetohydrodynamics (MHD), an extension of simple hydrodynamic methods to electrically conducting fluids, have long been the primary choice in simulating large scale phenomenon with the local thermal equilibrium assumption, such as magnetic reconnection [20, 107] and other plasma phenomena.

The MHD equations of motion are derived from a combination of Newton's equation and Maxwell's equations applied in the fluid regime. In contrast to N -body and kinetic methods, in MHD models the spatial distributions of macroscopic quantities (mass density, velocity density, etc) are obtained by solving the moment equations for each particle species. A full description of this derivation can be found in [29, 50]. The end result of this is a set of

equations of motion for fluid plasma descriptions,

$$\frac{\partial \rho}{\partial t} + \nabla \cdot (\rho \mathbf{v}) = 0, \quad (\text{Continuity}) \quad (1.9)$$

$$\rho \left(\frac{\partial \mathbf{v}}{\partial t} + \mathbf{v} \cdot \nabla \mathbf{v} \right) + \nabla p - \frac{1}{\mu_0} (\nabla \times \mathbf{B}) \times \mathbf{B} = 0, \quad (\text{Momentum}) \quad (1.10)$$

$$\frac{\partial p}{\partial t} + \mathbf{v} \cdot \nabla p + \gamma p \nabla \cdot \mathbf{v} = 0, \quad (\text{Euler}) \quad (1.11)$$

$$\frac{\partial \mathbf{B}}{\partial t} - \nabla \times (\mathbf{v} \times \mathbf{B}) = 0, \quad (\text{Induction}) \quad (1.12)$$

$$\nabla \cdot \mathbf{B} = 0, \quad (\text{magnetic Gauss}). \quad (1.13)$$

where ρ is mass density, p is pressure and the other variables are as defined above. This set of equations is a specific case where we have neglected all dissipative processes, i.e. finite viscosity, electrical resistivity and thermal conductivity. There are a variety of other derivations of the MHD equations for varying plasma conditions.

The speedup gained in MHD methods comes at the cost of small scale accuracy, which is lost in the averaging of macroscopic quantities. In many regimes, these small scale phenomena can be largely ignored but, as we have discussed in this chapter, there is an increasing understanding that under certain plasma conditions, kinetic dynamics become vital. For example, collisionless kinetic simulations of magnetic reconnection in the solar atmosphere have shown that the process of field lines breaking and reconnecting in plasmas involves both ion and electron kinetic-scale features, with electron current layers forming nonlinearly during the onset phase and playing a key role in enabling field lines to break [114, 64, 35]. MHD algorithms also struggle with problems of *non-locality* [31], which put plainly means fast moving particles don't interact with their nearest neighbor. This issue of non-locality also leads to the requirement of local thermal equilibrium, a condition that is not always present in plasmas. For these reasons, we have chosen to move forward with a kinetic description of our plasma systems. We focus much of our efforts on building highly scalable and efficient solvers to alleviate excess computational costs.

Chapter 2

Software Infrastructure in PETSc

PETSc, the Portable, Extensible Toolkit for Scientific Computation [8, 38, 112], is a well-known library for numerical methods. It provides parallel data management, structured and unstructured meshes, linear and nonlinear algebraic solvers and preconditioners, optimization algorithms, time integrators and many more functions. A variety of PETSc modules are used frequently in this work. In this section we will briefly review the foundational principles of the PETSc modules that are most essential to this project.

The approach to constructing a computational model for plasma systems in this dissertation differs from that in previous literature in that the goal is to develop a toolkit of solvers instead of a monolithic application code. By building a suite of individual composable solvers, that are parallelized by construction, a wide range of simulations can be built with little effort. We may also avoid common pitfalls of over-specialization, such as hard coding aspects of solvers. The primary focus, and strength, of the PETSc project is to provide user level APIs that allows implementations to conform to the API with solver details left in the back end code base as well as to give users the ability to change solver details to conform to the discretization, problem regime, or hardware.

2.1 DMSwarm: Particles

The management of particles within a domain, in serial or parallel environments, is one of the most important requirements of plasma physics simulations. Key aspects in the management of these particles are: the need to attach data (or fields) to each particle; support for the advection of particles within some domain; dynamic insertion and deletion of particles; collection type operations to gather nearby particles; and methods to interpolate and restrict data back and forth between a set of particles and a background mesh. The `DMSwarm` module, provides a fully parallel solution for pure particle methods (e.g. DEM, SPH, EFG) and for particle-mesh methods (e.g. PIC [39], FLIP [19], MPM [130], GIMP [9]) [93]. `DMSwarm` is capable of generating delta function discretizations for particle distributions as well as any other user defined shape function, whilst still providing point track and point-cell locating. Particles are tracked within `DMSwarm`'s internal data structure, which also provides built-in fields, such as particle coordinates (`DMSwarmPICField_coor`), essential in PIC simulations. Particle coordinates are used by the `DMSwarm` module to perform point-cell location within an associated PETSc mesh, using either brute force scans or grid hashing [99].

A suite of functions aimed at creating a more accessible particle initialization environment has been added to the `DMSwarm` module as part of this project. Users specify distribution functions for the particle locations and velocities (uniform, Gaussian, etc.) as well as a target number of particles desired for the simulation. The API also sets the weight of each particle in the domain. The user may also input a locally written distribution function to the API which will in turn be used to generate the particles. In uniform initialization, particle coordinates and velocities are generated evenly across the (spatial and velocity) cellwise domain with the specified number of particles. In more complex initialization schemes, the API integrates the predefined distribution functions to obtain the number of particles in each domain cell. The particles are then given initial positions and velocities using pseudorandom

methods, such as the approximate inverse error function in one-dimension,

$$q = \text{erf}^{-1}(z) = \sum_{k=0}^{\infty} \frac{c_k}{2k+1} \left(\frac{\sqrt{\pi}}{2} z \right)^{2k+1}, \quad (2.1)$$

where q is a phase space variable describing either the particle's position x or velocity v , z is some randomly generated number, $c_0 = 1$, and c_k is,

$$c_k = \sum_{m=0}^{k-1} \frac{c_m c_{k-1-m}}{(m+1)(2m+1)}, \quad (2.2)$$

and the Box-Mueller transform for two-dimensional sampling,

$$q_1 = \sqrt{-2 \ln z_1} \cos \{2\pi z_2\}, \quad (2.3)$$

$$q_2 = \sqrt{-2 \ln z_1} \sin \{2\pi z_2\}. \quad (2.4)$$

Examples of the initial distributions created by the DMSwarm initialization API are shown in Figures 2.1, 2.2 and 2.3.

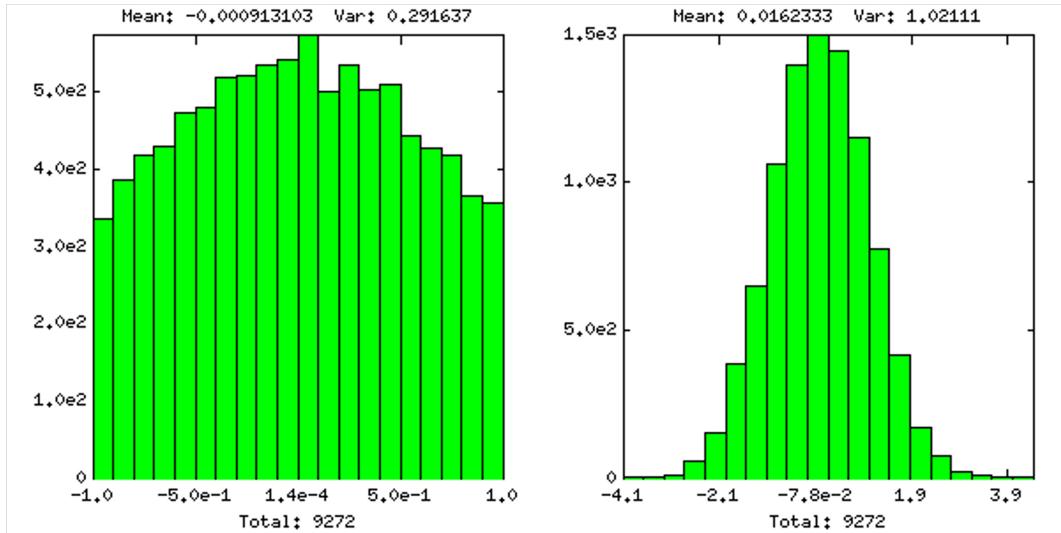


Figure 2.1: DMSwarm Gaussian initialization of particle coordinates (left) and velocities (right). The particle coordinate Gaussian in this example was intentionally given a width wider than the cell size.

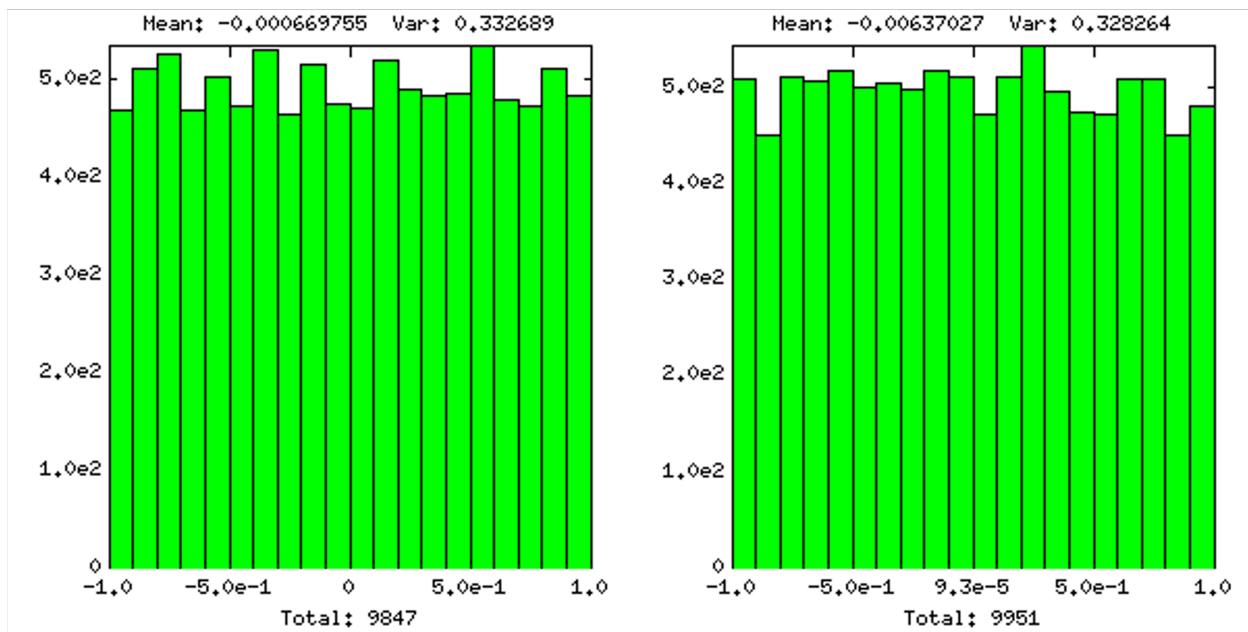


Figure 2.2: DMSwarm uniform initialization of particle coordinates (left) and velocities (right).

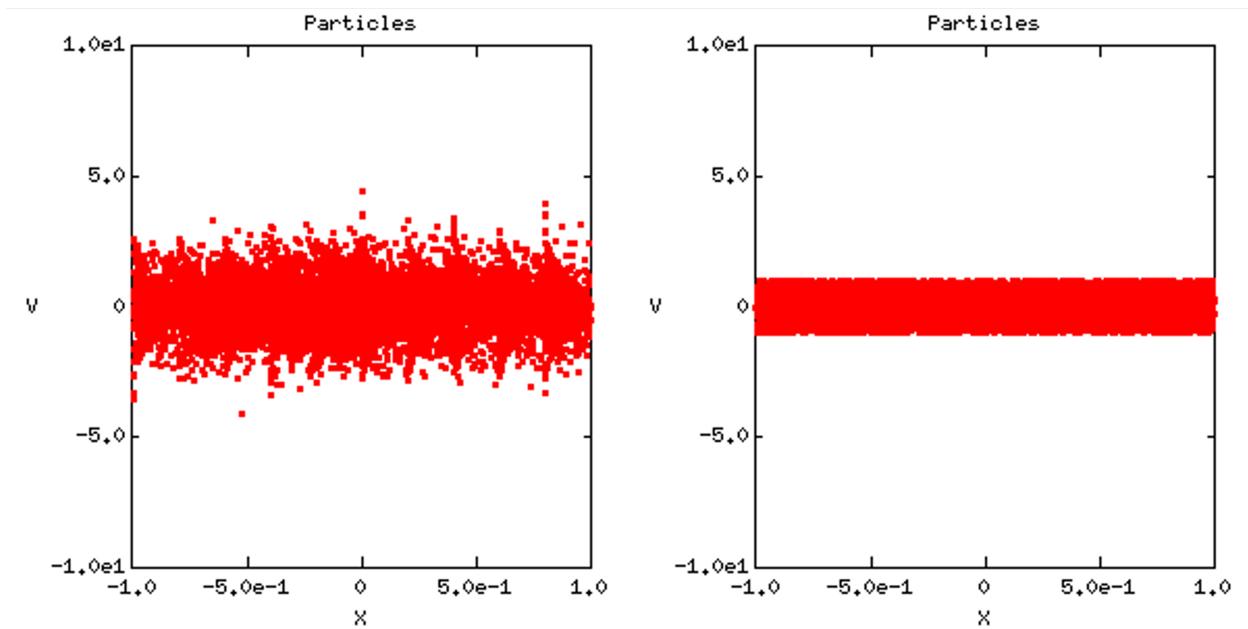


Figure 2.3: Phase space plot of particles initialized by DMSwarm API with Guassian (left) and uniform (right) coordinates and velocities.

2.2 DMPlex: Unstructured Grids

DMPlex is a PETSc module for generic unstructured mesh storage and operations [79, 86, 80, 73, 59]. It decouples user applications from the implementation details of common mesh-related utility tasks. The **DMPlex** module also handles the finite element discretization via the `PetscFE` object. Furthermore, parallelization is accomplished via *star forest* communication structures managed by `PetscSF`. PETSc does not contain its own mesh generation tools but rather utilizes external packages for this purpose. For example, the packages `tetgen` and `triangle` are often used for the generation of simplicial grids. The **DMPlex** API also allows users to load prebuilt mesh files. The internal representation of mesh topology provides a layer of abstraction that decouples the mesh from the underlying file format. This allows support for a variety of mesh file formats, including the GMsh and Exodus II formats.

DMPlex stores the connectivity of the associated mesh as a layered directed acyclic graph (DAG), where each layer, or *stratum*, represents a class of topological entities [85]. Mathematically, this is a graded poset known as a Hasse diagram [139]. An example of this DAG representation for a single tetrahedron is shown in Figure 2.4. Importantly, this representation is flexible enough to provide an abstract interface for the implementation of mesh management and manipulation algorithms using dimension-independent programming.

2.3 TS: Timesteppers

The integration of the Vlasov differential equation in this research is managed by the timestepping module, **TS** [1, 146]. The **TS** module is organized around differential-algebraic equations (DAE) of the form,

$$F(t, u, \dot{u}) = G(t, u), \quad u(t_0) = u_0. \quad (2.5)$$

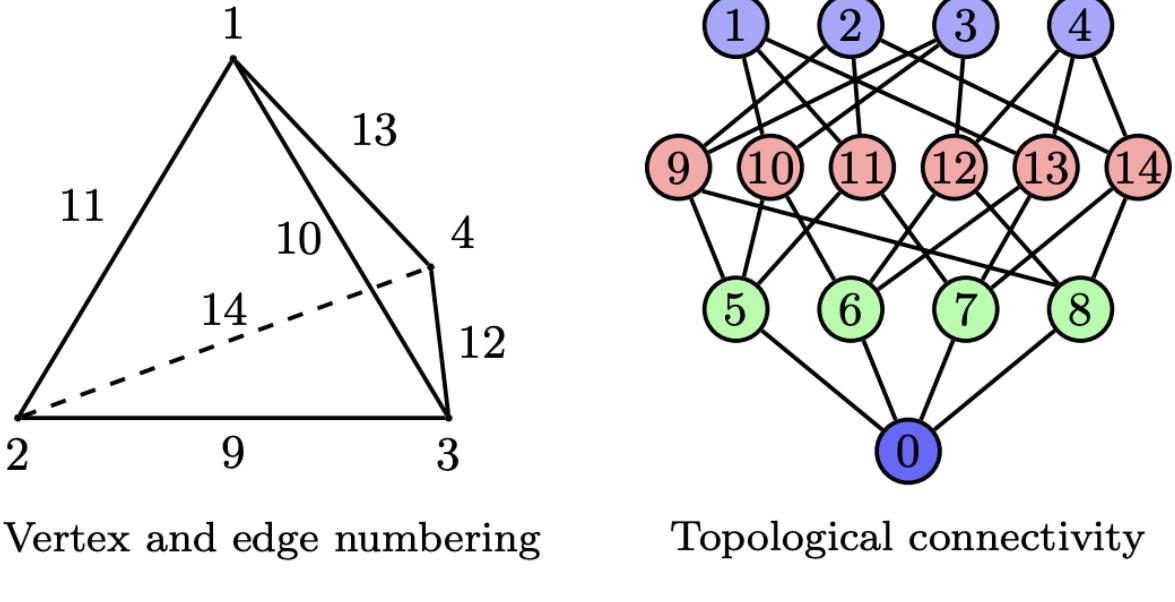


Figure 2.4: DAG-based representation of a single tetrahedron in DMPlex from [85].

In this equation, we refer to the function F as the I function, which depends on the current simulation time, t , the input state, u , and the input time derivative, \dot{u} . The function G is called the right-hand-side (RHS) function and depends only on the current time and input state. The equation becomes an ordinary differential equation (ODE) when the matrix $F_{\dot{u}}(t) = \partial F / \partial \dot{u}$ has full rank. In this case, the equation can be transformed to the standard explicit form,

$$\dot{u} = Q(t, u). \quad (2.6)$$

for some RHS function, Q . The implicit/DAE interface can significantly reduce the overhead in ODEs with nontrivial mass matrices to prepare the system for algebraic solvers by having the user assemble the correctly shifted matrix. These nontrivial matrices are often seen in finite element problems, which will play a key role in the PETSc-PIC algorithm. Within the TS API, the $F(t, u, \dot{u})$ and $G(t, u)$ functions are defined by the user with the functions `TSSetIFunction` and `TSSetRHSFunction` (assuming $G(t, u)$ is nonzero), respectively. In implicit, or semi-implicit methods, the user must also provide an appropriate Jacobian matrix of $F(t, u, \dot{u})$ using the `TSSetIJacobian` function. Furthermore, in fully implicit methods

the user also defines a Jacobian matrix for the right-hand-side, $G(t, u)$, with the function `TSSetRHSJacobian`. As correctly coding Jacobian matrices for a given problem is often a barrier in this process, PETSc provides the application of Jacobians via *matrix-free* differencing approaches, explicit computations of Jacobians via matrix coloring and differencing, and automatic testing of user-provided Jacobian computations to assist users in this process.

The Jacobian is defined by the chain rule as,

$$\frac{dF}{du^n} = \frac{\partial F}{\partial \dot{u}}|_{u^n} \frac{\partial \dot{u}}{\partial u}|_{u^n} + \frac{\partial F}{\partial u}|_{u^n}. \quad (2.7)$$

In ODE methods, the approximation of \dot{u} is linear in u^n , therefore, $\frac{\partial \dot{u}}{\partial u}|_{u^n} = \sigma$ where σ is the *shift* that depends on the ODE integrator and time step. We may rewrite the Jacobian function for $F(t, u, \dot{F})$ as,

$$\frac{dF}{du^n} = \sigma \frac{\partial F}{\partial \dot{u}}(t^n, u^n, \dot{u}^n) + \frac{\partial F}{\partial u}(t^n, u^n, \dot{u}^n). \quad (2.8)$$

This is the form of the Jacobian that the PETSc API requests as an input.

The classes of timestepper methods currently implemented in PETSc are shown in Table 2.1. Of particular interest in this research are the basic symplectic, theta and discrete gradient type timesteppers. We note here, and will discuss in more detail in Section 4.2, that the discrete gradient integrator is merely a theta method (with $\theta = 0.5$) with an added expansion term that preserves the first integral of the system.

When implicit and semi-implicit methods methods are used, additional PETSc linear and nonlinear solvers are added to the solver. These solvers will be discussed in more detail in the next few sections.

TS Name	Reference	Type	Order
euler	forward Euler	explicit	1
beuler	backward Euler	implicit	1
ssp	Multistage SSP	explicit	≤ 4
rk	Runge-Kutta	explicit	≥ 1
cn	Crank-Nicholson	implicit	2
theta	theta method	implicit	≤ 2
alpha	alpha method	implicit	2
gl	general linear	implicit	≤ 3
eimex	extrapolated IMEX	adaptive	≥ 1
arkeimex	IMEX Runge-Kutta	IMEX	1 – 5
rosw	Rosenbrock W-Schemes	linearly implicit	1-4
glee	GL Schemes with global error estimation	explicit and implicit	1-3
basicssymplectic	Basic Symplectic	depends on order	1-4
discgrad	Discrete Gradients	implicit	2

Table 2.1: Table of existing timestepping methods in PETSc from [38].

2.4 KSP: Linear Solvers

The KSP module in PETSc provides an easy-to-use API to the combination of Krylov subspace iterative methods and preconditioners, managed by the PC object. The KSP solvers are designed for solving linear systems of the form,

$$Ax = b, \quad (2.9)$$

where A is the matrix representation of a linear operator, b is the right-hand-side vector and x is the solution vector. In practice, the linear operator is set using the PETSc function `KSPSetOperators` and the right-hand-side vector is input when the `KSPSolve` function is called. The `KSPSetOperators` call also contains the matrix that will be used to construct the preconditioners, if it is different from the system matrix. The KSP module encapsulates a large assortment of common (and uncommon) Krylov methods, such as conjugate gradients (CG), GMRES, Richardson, and many others. It is an essential tool in the PETSc library and, as such, contains some of the most mature code in PETSc.

2.4.1 PC: Preconditioners

Preconditioning of the system is typically used in PETSc methods to alter the spectrum and, therefore, accelerate the convergence rate of iterative techniques. Preconditioners are commonly applied to the linear system by,

$$(M_L^{-1} A M_R^{-1})(M_R x) = M_L^{-1} b, \quad (2.10)$$

where M_L and M_R are left and right preconditioning matrices, respectively. In the case where the left preconditioning matrix is equal to the identity matrix, i.e. $M_L = \mathbb{I}$, the residual, r , of the linear system (2.9),

$$r \equiv b - Ax = b - AM_R^{-1}M_Rx \quad (2.11)$$

is preserved. Alternatively, if the right preconditioner is given by the identity matrix, $M_R = I$, the residual is altered for the left and symmetric preconditioning,

$$r_L \equiv M_L^{-1}b - M_L^{-1}Ax = M_L^{-1}r. \quad (2.12)$$

Left preconditioning is often chosen in KSP implementations because each approximate solution remains in the original space. The **PC** object is another mature module within PETSc, containing preconditioner types like block Jacobi, LU factorization, and singular value decomposition (SVD). The **PC** object also contains a native implementations of algebraic multi-grid (AMG) methods [94] and patch preconditioning (PCPATCH) [42]. PETSc also interfaces with three external AMG packages: **hyper**, **ML**, and **AMGx**.

2.5 SNES: Nonlinear Solvers

The **SNES** module provides a powerful suite of data-structure-neutral numerical methods for nonlinear problems of the form,

$$F(x) = 0, \quad (2.13)$$

where $F : \mathbb{R}^n \rightarrow \mathbb{R}^n$. At the core of the **SNES** module is Newton's method. The general form of the n -dimensional Newton's method is,

$$x_{k+1} = x_k - J(x_k)^{-1}F(x_k), \quad k = 0, 1, \dots, \quad (2.14)$$

where $J(x_k) = F'(x_k)$ is the Jacobian, which is nonsingular at each iteration. The method also requires some initial guess, x_0 , to the solution. Newton's method is commonly implemented in two steps: the first being an approximate solve of the system,

$$J(x_k)\Delta x_k = -F(x_k), \quad (2.15)$$

and the second being an update of the form,

$$x_{k+1} \leftarrow x_k + \Delta x_k. \quad (2.16)$$

While nonlinear solves are primarily handled by Newton solvers, a variety of other nonlinear solvers are implemented in the PETSc library, such as nonlinear CG, Richardson and quasi-Newton methods.

In the **SNES** module, the user must specify routines to form the residual function (**SNESSetFunction**) and the Jacobian (**SNESSetJacobian**). A strength of PETSc lies in the users ability to make changes to all the objects and solvers discussed in this chapter at the command line level. Functions, such as **KSPSetFromOptions**, **SNESSetFromOptions**, **DMPlexSetFromOptions**, as well as other similar ones, take in options from the command

line and adjust the program accordingly. This flexibility is a central component of PETSc programming.

2.6 Contributions to PETSc

Throughout this research, a number of test/tutorial files as well as backend functions have been written and added to the PETSc library. This research has also included adjustments or improvements to existing functions within the library. The contributions made to the PETSc library as a part of this dissertation have been summarized in Figure 2.5. This chart only includes code that has been officially merged into the PETSc/main branch and is available for use by the public.

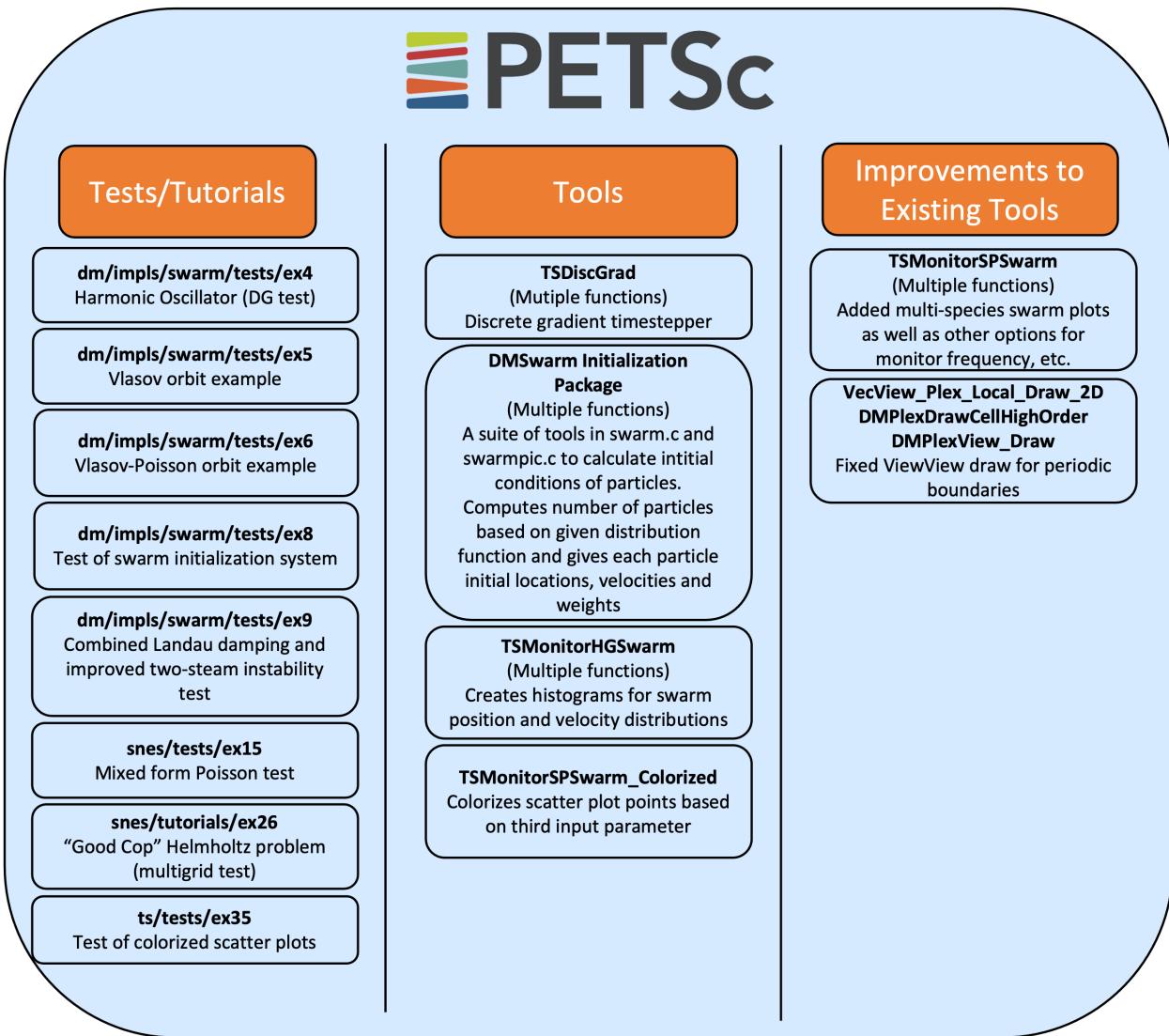


Figure 2.5: A chart of contributions made to the PETSc library as a part of this dissertation. All code listed in this graphic has been merged into PETSc/main.

Chapter 3

Metriplectic Dynamics

In recent years, structure-preserving algorithms for simulating plasmas have been a major field of interest. The goal of these algorithms is to provide a numerical framework for solving kinetic plasma equations, such as the Vlasov-Maxwell-Landau system and its simplifications, that preserves the basic laws of physics associated with plasmas. Most importantly, numerical frameworks must exhibit conservation of mass, momentum, kinetic energy, and the preservation of entropy monotonicity over long simulation times. Developments in the field of structure-preserving algorithms in the last decade or so have been, to a large extent, due to the advances in geometric particle-in-cell (GEMPIC) methods [40, 82, 118, 122, 129, 143, 141]. GEMPIC algorithms are based on discretizing either the variational or Hamiltonian structure of the underlying kinetic models. These schemes are desirable not only because they preserve the energy of the system but they typically also guarantee a local algebraic charge conservation law and the preservation of the multisymplectic two-form. Furthermore, the long-time fidelity and stability is especially important for kinetic simulations of magnetized fusion plasmas targeting macroscopic transport time scales.

3.1 Equations of Kinetic Theory

We must first derive a set of equations to describe a system of a large number of interacting charged particles in a plasma. The foundation of the kinetic theory of plasmas is built upon the idea that an accurate description can be made of charged particles in a plasma by considering their *collective behavior* rather than their *individual behaviors*. It is appropriate, therefore, to use a statistical approach for this analysis. In a statistical approach, the individuality of each particle is lost but we retain the physical information of the plasma as a whole by constructing distribution functions $f_s(\mathbf{x}, \mathbf{v}, t)$, which is a function of the positions, \mathbf{x} , and velocities, \mathbf{v} , of the particles in the plasma, as well as the time, t . The s subscript refers to the species of the specific distribution function: electron (f_e) or ion (f_i). The distribution function f_s is defined as the density of the representative points of particles in a six-dimensional phase space (\mathbf{x}, \mathbf{v}) . Discrete components of the distribution function, called “macro-” or “marker-” particles, represent larger groups of particles and are traced through the simulation. For simplicity, we commonly refer to these macroparticles as “particles”. The phase space consists of the subspace of \mathbb{R}^6 , containing all possible positions and velocities of the macroparticles. For any volume, V , the integral of the distribution function f_s is the average number of particles of the species, s whose position and velocity are in V .

The motion of a collection of representative points in phase space is described by applying the chain rule to the time derivative of the distribution function f_s ,

$$\frac{df_s}{dt} = \frac{\partial f_s}{\partial t} + \frac{\partial \mathbf{x}}{\partial t} \cdot \frac{\partial f_s}{\partial \mathbf{x}} + \frac{\partial \mathbf{v}}{\partial t} \cdot \frac{\partial f_s}{\partial \mathbf{v}}. \quad (3.1)$$

The terms $\partial \mathbf{x}/\partial t$ and $\partial \mathbf{v}/\partial t$ are equivalent to the velocity and acceleration of the particles, respectively. Recalling Newton’s second law of motion $F = ma$, for some force, F , mass, m , and acceleration, a , we can replace the acceleration term in (3.1) with the force, in this case

caused by the electric and magnetic fields of the plasma, acting on each particle divided by the mass of the particle,

$$\frac{df_s}{dt} = \frac{\partial f_s}{\partial t} + \mathbf{v} \cdot \frac{\partial f_s}{\partial \mathbf{x}} + \frac{F_s}{m_s} \cdot \frac{\partial f_s}{\partial \mathbf{v}}, \quad (3.2)$$

$$= \frac{\partial f_s}{\partial t} + \mathbf{v} \cdot \frac{\partial f_s}{\partial \mathbf{x}} + \frac{q_s}{m_s} (\mathbf{E} + \mathbf{v} \times \mathbf{B}) \cdot \frac{\partial f_s}{\partial \mathbf{v}}. \quad (3.3)$$

We note again that the subscript s , now attached to the force, F_s , mass, m_s , and charge, q_s , variables, refers to the specific species. It is important to note the difference in the notation for the total time derivative df_s/dt and the ordinary time derivatives $\partial \mathbf{v}/\partial t$. In the presence of binary interactions between particles, we can replace the total derivative of the distribution with a “collisional” term, $C_{s\bar{s}}$, such that,

$$\left(\frac{df_s}{dt} \right)_{coll} \equiv C_{s\bar{s}} = \frac{\partial f_s}{\partial t} + \mathbf{v} \cdot \frac{\partial f_s}{\partial \mathbf{x}} + \frac{q_s}{m_s} (\mathbf{E} + \mathbf{v} \times \mathbf{B}) \cdot \frac{\partial f_s}{\partial \mathbf{v}}. \quad (3.4)$$

This equation is commonly referred to as the *Boltzmann* or *Vlasov* equation. The electric and magnetic field terms, $\mathbf{E}(\mathbf{x}, t)$ and $\mathbf{B}(\mathbf{x}, t)$, respectively, include contributions from externally applied fields as well as from those originating from long-range particle interactions within the plasma. A method must be devised for which the electric and magnetic fields can be calculated from the particles in the plasmas, assuming the external contributions are previously defined. This is done by coupling (3.4) to the Maxwell equations which enables the calculation of the self-consistent electromagnetic field from the particle distribution,

$$-\frac{1}{c^2} \frac{\partial \mathbf{E}}{\partial t} + \nabla \times \mathbf{B} = \mu_0 \mathbf{J}, \quad (\text{Ampère}) \quad (3.5)$$

$$\frac{\partial \mathbf{B}}{\partial t} + \nabla \times \mathbf{E} = 0, \quad (\text{Faraday}) \quad (3.6)$$

$$\nabla \cdot \mathbf{E} = \frac{\rho}{\epsilon_0}, \quad (\text{Gauss}) \quad (3.7)$$

$$\nabla \cdot \mathbf{B} = 0, \quad (\text{magnetic Gauss}) \quad (3.8)$$

where the constants are c the speed of light, μ_0 the permeability of free space and ϵ_0 the permittivity of free space. The source terms in Maxwell's equations, the charge density, $\rho(\mathbf{x}, t)$, and the current density, $\mathbf{J}(\mathbf{x}, t)$, can be expressed as functions of the distribution functions $f_s(\mathbf{x}, \mathbf{b}, t)$,

$$\rho(\mathbf{x}, t) = \sum_s e_s \int f_s(\mathbf{x}, \mathbf{b}, t) d\mathbf{v}, \quad (3.9)$$

$$\mathbf{J}(\mathbf{x}, t) = \sum_s e_s \int f_s(\mathbf{x}, \mathbf{b}, t) \mathbf{v} d\mathbf{v}. \quad (3.10)$$

The “collisional” term $C_{s\bar{s}}$ represents the rate of change of the distribution function f_s due to short-range particle interactions. The collisions have the effect of bringing plasmas back to their thermodynamical equilibrium in which the velocity distribution is Maxwellian. A variety of collision operators for this system exist, which we will briefly review here. The earliest example of a collision operator, the *Boltzmann operator*, was derived by Ludwig Boltzmann in 1872 [18]. It is a nonlinear integral operator dependent on the scattering cross-section,

$$C_{s\bar{s}}(f_s, f_{\bar{s}}) := \int_{\mathbb{R}^d} \int_{\omega \in S^{d-1}} B(|v'_s - v_s|, \theta) (f(v'_s)f(v'_{\bar{s}}) - f(v_s)f(v_{\bar{s}})) dv_{\bar{s}} d\omega, \quad (3.11)$$

where v_s and $v_{\bar{s}}$ are the velocities of the two colliding particles prior to collision and the apostrophe denotes velocities of particles after collision, and ω is the velocity angle which presides in the unit sphere S^{d-1} , for some dimension, d . The scattering cross section B is a function of the angle between the two relative velocities $|v'_s - v_s|$ and $|v'_{\bar{s}} - v_{\bar{s}}|$, commonly referred to as the *deviation angle*. In the limit where the scattering angle goes to zero, we assume the particles undergo *grazing* collisions. These grazing collisions can be described by the Landau operator, introduced in [83], often referred to as the Fokker-Planck equation. The Landau equation is one of the most commonly used collision integrals as the grazing collisions regime has been found to be highly accurate. The basic formulation of the Landau

collision operator is given by,

$$C_{s\bar{s}}(f_s, f_{\bar{s}}) := \frac{\partial}{\partial v} \cdot \left[\int_{v_{\bar{s}}} \mathbb{Q}(v_s - v_{\bar{s}}) \left(\frac{\partial f(v_s)}{\partial v} f(v_{\bar{s}}) - \frac{\partial f(v_{\bar{s}})}{\partial v} f(v_s) \right) dv_{\bar{s}} \right], \quad (3.12)$$

where $\mathbb{Q}(v_s - v_{\bar{s}})$ is the collision kernel. In this formulation, the collision kernel is defined as,

$$\mathbb{Q}(\xi) = \frac{1}{|\xi|} \left(I - \frac{\xi^T \xi}{|\xi|^2} \right). \quad (3.13)$$

Note that ξ is in the nullspace of $\mathbb{Q}(\xi)$, i.e.,

$$\xi \mathbb{Q}(\xi) = 0. \quad (3.14)$$

This property of the collision kernel is exploited in obtaining conservation.

The Landau and Boltzmann collision operators can be further simplified into the Bhatnagar-Gross-Krook collision operator [12] under the assumption that the distribution functions relax towards a local thermal equilibrium or the Maxwellian. This BGK operator is defined as,

$$C_s(f_s) = \nu (f_s - f_s^*) \quad (3.15)$$

where f_s is the particle distribution function, f_s^* is the Maxwellian equilibrium distribution and ν is the collisional frequency. In its continuum form it conserves the moments. However, this conservation is broken when discretizing the operator for particle methods. The discrete operator also fails to preserve the monotonicity of entropy and dissipation of free energy. The $\mathcal{O}(N)$ complexity of this operator does offer an inexpensive option for models that are more tolerant of breaks in conservation. Therefore, an implementation of the BGK collision operator exists within PETSc and has been tested in previous works [115, 54]. One of the goals of this research, however, is to construct a collision operator that preserves the fundamental thermodynamic principles of the plasma system. Therefore, the more complex Landau operator will be the choice for modeling collisions moving forward.

The macroscopic quantities particle density, mean velocity, kinetic energy and temperature associated with each particle species can be defined in the following ways. The particle density, n_s , in physical space for species s is given by,

$$n_s(\mathbf{x}, t) = \int f_s(\mathbf{x}, \mathbf{v}, t) d\mathbf{v}. \quad (3.16)$$

The mean velocity, \mathbf{u}_s , then verifies the expression,

$$\mathbf{u}_s(\mathbf{x}, t) = \frac{1}{n_s(\mathbf{x}, t)} \int f_s(\mathbf{x}, \mathbf{v}, t) \mathbf{v} d\mathbf{v}. \quad (3.17)$$

The kinetic energy, E_s , is defined by,

$$E_s(\mathbf{x}, t) = \frac{m_s}{2n_s(\mathbf{x}, t)} \int f_s(\mathbf{x}, \mathbf{v}, t) |\mathbf{v}|^2 d\mathbf{v}. \quad (3.18)$$

Lastly, the temperature, T_s , is related to the kinetic energy, mean velocity and density, shown in the expression,

$$\frac{3}{2}k_B T_s(\mathbf{x}, t) = E_s(\mathbf{x}, t) - \frac{m_s}{2} \mathbf{u}_s^2(\mathbf{x}, t), \quad (3.19)$$

where k_B is the Boltzmann constant.

3.2 Metriplectic Formulation

Provided the full Vlasov-Maxwell-Landau system of equations, we seek a framework for the description of systems that display both Hamiltonian and dissipative dynamics. Therefore, we introduce the concept of metriplectic dynamics which provides such a framework. The dissipation-free Vlasov-Maxwell system has an infinite-dimensional non-canonical Hamiltonian structure culminating into a functional Poisson bracket $\{\cdot, \cdot\}$. The collisional Landau system can be formulated as an infinite-dimensional metric bracket (\cdot, \cdot) . Therefore, the evolution of an arbitrary functional of field variables \mathcal{U} in time, where we denote the

arbitrary field variables $u(z, t) = (u^1, u^2, \dots, u^m)^T$ is given by,

$$\frac{d\mathcal{U}}{dt} = \{\mathcal{U}, \mathcal{F}\} + (\mathcal{U}, \mathcal{F}). \quad (3.20)$$

where $\mathcal{F} = \mathcal{H} - \mathcal{S}$ is a generalized free energy functional and is equal to the difference of the Hamiltonian functional, \mathcal{H} , and the entropy functional, \mathcal{S} .

It is important to note here that in this formulation, we are considering the evolution of functionals \mathcal{U} in time rather than functions A . Functionals are used in measurements of certain quantities, e.g. temperature and kinetic energy, while functions, such as the velocity are used to evolve the system in time. The functional forms of these equations can be converted into function form with the inclusion of delta functions. The equations in functional form are more generalized and can be used to more easily showcase the structure-preserving qualities of the metriplectic formulations. For example, we can reexpress the particle number, mean velocity, and energy from the previous Section 3.1 as functionals for mass \mathcal{M}_s , momentum \mathcal{P}_s and kinetic energy \mathcal{E}_s ,

$$\mathcal{M}_s = m_s \int f_s(\mathbf{v}, t) dv, \quad (3.21)$$

$$\mathcal{P}_s = m_s \sum_s \int f_s(\mathbf{v}, t) \mathbf{v} dv, \quad (3.22)$$

$$\mathcal{E}_s = m_s \sum_s \int f_s(\mathbf{v}, t) |\mathbf{v}|^2 dv. \quad (3.23)$$

Returning to the metric bracket, we consider the continuous forms of the Poisson and Metric brackets. The Poisson bracket is a bilinear, anti-symmetric bracket of the form,

$$\{\mathcal{A}, \mathcal{B}\} = \int_{\Omega} \frac{\delta \mathcal{A}}{\delta u^i} \mathcal{J}^{ij}(u) \frac{\delta \mathcal{B}}{\delta u^j} dz \quad (3.24)$$

where \mathcal{A} and \mathcal{B} are arbitrary functionals, $\delta\mathcal{B}/\delta u^j$ is a functional, or *Fréchet*, derivative,

$$\frac{\partial}{\partial \epsilon} \Big|_{\epsilon=0} \mathcal{A}[f_s + \epsilon \delta f_s] = \int \frac{\delta \mathcal{A}}{\delta f_s} \delta f_s dz \equiv \delta \mathcal{A}[\delta f_s]. \quad (3.25)$$

and $\mathcal{J}^{ij}(u)$ is the bracket kernel. The kernel is defined such that the Poisson bracket satisfies the Jacobi identity,

$$\{\{\mathcal{A}, \mathcal{B}\}, \mathcal{C}\} + \{\{\mathcal{B}, \mathcal{C}\}, \mathcal{A}\} + \{\{\mathcal{C}, \mathcal{A}\}, \mathcal{B}\} = 0, \quad (3.26)$$

with \mathcal{C} being another arbitrary functional. If $\mathcal{J}^{ij}(u)$ has a non-empty nullspace, there exist Casimir invariants \mathcal{C} such that $\{\mathcal{A}, \mathcal{C}\} = 0$ for all functionals \mathcal{A} . We construct the metriplectic framework such that the Hamiltonian functional \mathcal{H} is a Casimir of the metric bracket, the entropy functional \mathcal{S} is a Casimir of the Poisson bracket and the metric bracket is negative semi-definite. Given these properties we can reproduce the First and Second Laws of Thermodynamics as well as the dissipation of free energy,

$$\frac{d\mathcal{H}}{dt} = \{\mathcal{H}, \mathcal{F}\} + (\mathcal{H}, \mathcal{F}) = \{\mathcal{H}, -\mathcal{S}\} = 0, \quad (3.27)$$

$$\frac{d\mathcal{S}}{dt} = \{\mathcal{S}, \mathcal{F}\} + (\mathcal{S}, \mathcal{F}) = -(\mathcal{S}, \mathcal{S}) \geq 0, \quad (3.28)$$

$$\frac{d\mathcal{F}}{dt} = \{\mathcal{F}, \mathcal{F}\} + (\mathcal{F}, \mathcal{F}) = -(\mathcal{F}, \mathcal{F}) \leq 0. \quad (3.29)$$

These properties also simplify the expression of the metriplectic system to,

$$\frac{d\mathcal{U}}{dt} = \{\mathcal{U}, \mathcal{H}\} + (\mathcal{U}, \mathcal{S}). \quad (3.30)$$

Following the requirement that the metric bracket is negative semi-definite, we may

express the metric bracket as,

$$(\mathcal{U}, \mathcal{S}) = - \sum_{s, \bar{s}} \frac{c_{s\bar{s}}}{2T} \int_{\Omega_v} \int_{\Omega_v} \left(\frac{1}{m_s} \frac{\partial}{\partial v_s} \frac{\delta \mathcal{U}}{\delta f_s} - \frac{1}{m_{\bar{s}}} \frac{\partial}{\partial v_{\bar{s}}} \frac{\delta \mathcal{U}}{\delta f_{\bar{s}}} \right) \quad (3.31)$$

$$\cdot Q(v_s, v_{\bar{s}}) f_{\bar{s}}(v_{\bar{s}}) f_s(v_s) \cdot \left(\frac{1}{m_s} \frac{\partial}{\partial v_s} \frac{\delta \mathcal{S}}{\delta f_s} - \frac{1}{m_{\bar{s}}} \frac{\partial}{\partial v_{\bar{s}}} \frac{\delta \mathcal{S}}{\delta f_{\bar{s}}} \right) dv_{\bar{s}} dv_s. \quad (3.32)$$

It is straightforward to show [81, 68] that the mass, momentum and kinetic energy functionals are all Casimirs of the metric bracket,

$$(\mathcal{M}_s, \mathcal{S}) = 0, \quad (3.33)$$

$$(\mathcal{P}_s, \mathcal{S}) = 0, \quad (3.34)$$

$$(\mathcal{E}_s, \mathcal{S}) = 0. \quad (3.35)$$

Therefore, we have a structure-preserving metriplectic formulation for the Vlasov-Maxwell-Landau system. In the following two chapters we will express these equations in discrete forms and show that these discrete forms are also structure-preserving.

Chapter 4

Collisionless Kinetics

If collisions are ignored, the Vlasov-Maxwell-Landau system can be simplified into the Vlasov-Maxwell, and the nonrelativistic zero-magnetic field Vlasov-Poisson, equations. Some of the most important phenomena in plasma physics occur in the non-magnetic, collisionless regime, such as Landau damping, and therefore it is a good place to start in the development of any kinetic solver. Each of these simplified systems, as well as other related systems [30], have seen significant progress in algorithms that satisfy energy conservation and also preserve other invariants present in the system, such as the momentum and charge conservation, and the divergence-free nature of the magnetic field [61, 128, 142].

As defined in the previous section, the Vlasov equation,

$$\frac{\partial f_s}{\partial t} + \mathbf{v} \cdot \frac{\partial f_s}{\partial \mathbf{x}} - \frac{q_s}{m_s} \mathbf{E} \cdot \frac{\partial f_s}{\partial \mathbf{v}} = 0, \quad (4.1)$$

describes the evolution of the phase space distribution, $f(\mathbf{x}, \mathbf{v}, t)$, defined over the domain $(\mathbf{x}, \mathbf{v}) \in \mathbb{R}^D \times \mathbb{R}^D$ where D is the spatial dimension. The electric field is obtained using Poisson's equation,

$$\Delta\phi(\mathbf{x}, t) = -\frac{\rho}{\epsilon_0}, \quad (4.2)$$

where ϕ is the electric potential, ρ is the charge density and $\mathbf{E} = -\nabla\phi$. The charge density

contains a neutralizing background term, σ , such that,

$$\rho(\mathbf{x}, \mathbf{v}, t) = \sigma - q_e \int_{\mathbb{R}^D} f_e(\mathbf{x}, \mathbf{v}, t) d\mathbf{v}. \quad (4.3)$$

This neutralizing background simulates the effect of ions on the electrons in the domain. The use of a stationary, uniform background charge is based on the assumption that the ions are much heavier than the electrons and thus feel little influence from them. With the inclusion of the background charge, we may drop the subscript s from the phase space distribution, as well as the mass and charge terms, the for the rest of the chapter as we are only evolving one species, i.e. the electrons, in time.

4.1 PETSc-PIC

The PETSc Particle-In-Cell (PETSc-PIC) algorithm is used to solve the collisionless Vlasov-Poisson system. PETSc-PIC uses symplectic integration schemes [1] for particle pushing, while conducting field solves with a finite element method [86, 78]. The solvers used in the PETSc-PIC algorithm fully conserve the moments, mass, momentum and energy, at each time step while also preserving entropy monotonicity. Recent advances in the PETSc-PIC code [117] also include conservative projections between the finite element and particle basis, a key step towards hybrid FEM-particle algorithms.

The PETSc-PIC algorithm primarily relies on two modules, discussed in more depth in Section 2.1 and 2.2, to handle the particle and mesh solves simultaneously: **DMplex** and **DMSwarm**.

4.1.1 Particle Methods

We start with discussion of the particle methods in the PETSc-PIC algorithm. A method must first be chosen to represent the particle space and for interpolation between the mesh and particle representations. There are numerous choices in shape functions for this purpose.

However, in our case a simple delta function representation of particles is chosen. Thus the approximation of the distribution function, f_p , is defined in the particle space as,

$$f_p = \sum_p \vec{\omega}_p \delta(\mathbf{x} - \mathbf{x}_p), \quad (4.4)$$

where $\vec{\omega}_p$ is the vector of weights, δ is the delta function, \mathbf{x} are the configuration space variables and \mathbf{x}_p represents the particle position and velocity, respectively. The finite element representation, f_{FE} , using a function space \mathcal{V} , is given by the weighted sum of basis functions,

$$f_{FE} = \sum_i f_i \psi_i(\mathbf{x}), \quad (4.5)$$

where $\psi_i \in \mathcal{V}$ denotes the basis functions and f_i the associated finite element coefficient.

The Vlasov equation is a linear hyperbolic equation which may be written in a simpler form,

$$\frac{\partial f}{\partial t} + \mathbf{z} \cdot \nabla_{\mathbf{q}} f = 0, \quad (4.6)$$

where $\mathbf{q} = (\mathbf{x}, \mathbf{v})$ is the phase space variable and $\mathbf{z} = (\mathbf{v}, -q_e \mathbf{E}/m_e)$ is the combined force on the electrons, denoted by subscript e . The force term $-q_e \mathbf{E}/m_e$ is independent of velocity, and therefore (4.6) may be written in the conservative form,

$$\frac{\partial f}{\partial t} + \nabla_{\mathbf{q}} \cdot (\mathbf{z} f) = 0. \quad (4.7)$$

Given this new advective form of the Vlasov equation, we can rewrite the equation for the characteristics $\mathbf{Q} = (\mathbf{X}, \mathbf{V})$,

$$\frac{d\mathbf{Q}}{dt} = \mathbf{z}, \quad (4.8)$$

which reexpressed with the original phase-space variables gives,

$$\begin{aligned}\frac{d\mathbf{X}}{dt} &= \mathbf{V}, \\ \frac{d\mathbf{V}}{dt} &= -\frac{q_e}{m_e} \mathbf{E}.\end{aligned}\tag{4.9}$$

Since particles follow characteristics, the Vlasov equation in the particle basis becomes

$$\begin{aligned}\frac{d\mathbf{x}_p}{dt} &= \mathbf{v}_p, \\ \frac{d\mathbf{v}_p}{dt} &= -\frac{q_e}{m_e} \mathbf{E}.\end{aligned}\tag{4.10}$$

The equations of motion are stepped forward in time using structure-preserving symplectic integrators which have been well studied [56], and will be further discussed in Section 4.2. The electric field is solved concurrently at each step using a finite element solver, discussed in the next section.

4.1.2 PETSc-FEM

At each step in the simulation, the Poisson equation is solved using the finite element method. The gradient of the potential, i.e. the electric field, is then interpolated across each cell at the particle locations. The interpolated electric field is then applied to the particles in the form of the Coulomb force.

The PETSc-FEM method is abstractly formalized by the *Ciarlet triple* [32, 76], such that a finite element is a triple $(\mathcal{T}, \mathcal{V}, \mathcal{V}')$, where,

- the domain \mathcal{T} is a bounded, closed subset of \mathbb{R}^d (for $d = 1, 2, 3, \dots$) with nonempty interior and piecewise smooth boundary;
- the space $\mathcal{V} = \mathcal{V}(\Omega)$ is a finite-dimensional function space on \mathcal{T} of dimension n ;
- the set of degrees of freedom (nodes) $\mathcal{V}' = \{l_1, l_2, \dots, l_n\}$ is a basis for the dual space,

that is, the space of bounded linear functionals on \mathcal{V} .

The cell \mathcal{T} together with the local function space \mathcal{V} and the set of rules for describing the functions in \mathcal{V} is the *finite element*. The discretization in PETSc is handled by the `PetscFE` object, which contains a `PetscSpace` (\mathcal{V}), `PetscDualSpace` (\mathcal{V}'), and `DMplex` (\mathcal{T}). `PetscFE` supports simplicial elements, tensor cells, and some special cells such as pyramids.

In general, the finite element solve for the Poisson equation can be accomplished using the standard H^1 function space. In the H^1 space, the weak form of the Poisson equation is,

$$\int_{\Omega} \nabla \psi_i \cdot \nabla \phi = \int_{\Omega} \psi_i, \quad (4.11)$$

where $\psi \in V$ and V is the set of basis functions on the cell. The elements are then constructed such that the zeroth derivatives of the basis functions are continuous across the cell boundaries. The H^1 finite element is sufficient for cell local Poisson solves. However, it enforces non-smooth continuity of the field at the cell boundaries. If a smooth continuity is desired, which is often true in plasma simulations, the Poisson equation can be reformulated using a mixed form and $H(\text{div})$ finite elements. Based on the results obtained in this work, shown in Section 4.4, the H^1 finite element was found to be accurate enough to capture the desired dynamics.

4.1.3 Conservative Projections

To preserve the conservation laws in a PIC simulation, a method must be constructed to conservatively project between the particle and grid representations. Weak equality of the representations,

$$\int_{\Omega} \psi_i f_{FE} = \int_{\Omega} \psi_i f_P \quad (4.12)$$

is enforced on the representations to achieve this [91, 117]. Restricting this equivalence to the finite-dimensional analogues gives the matrix-vector form,

$$M f_{FE} = M_p f_p, \quad (4.13)$$

where f_{FE} is a vector containing the finite element coefficients and f_p is the vector of particle weights, M is the finite element mass matrix,

$$M = \int_{\Omega} \psi_i \psi_j, \quad (4.14)$$

and M_p is the particle mass matrix,

$$M_p = \int_{\omega} \psi_i \delta(\mathbf{x} - \mathbf{x}_p). \quad (4.15)$$

The entries of M_p contain evaluations of the finite element basis functions at particle locations with rows being determined by the basis function index and columns being determined by the particle indices. Moving from the particle basis to the mesh, we must invert the finite element mass matrix, which is easily accomplished with CG/Jacobi [138]. In the other direction, we must invert a rectangular particle mass matrix, usually with LSQR [117].

4.1.4 Noise Reduction

In the regimes considered by kinetic plasma simulations, it is often the case that the energy density of the collective oscillations of the plasma is much smaller than the thermal or magnetic energy density. In these cases, noise created by the discreteness effects associated with the representation of the total distribution function by particles can completely mask important dynamics within the system. An important piece of PIC methods, therefore, is the reduction of noise producing methods. “Quiet start” [47, 22, 36] methods in the initialization phase can be a useful tool in reducing the signal-to-noise ratio experienced by many PIC

methods. The main goal of quiet start methods is to mimic the initial states of continuous methods.

Low-noise randomization, or “quasi-random”, methods for setting particle position and velocity has been one of the dominant approaches taken by previous PIC algorithms [14]. Sobol’ sequences [125] are quasi-random low-discrepancy sequences which are effective at low-noise randomization. Efficient algorithms for producing initial particle states with Sobol’ sequences are considered in [21] and [74]. Similar results can be achieved with “good lattices” [145] or Hammersley sequences [58]. Alternatively, in this work we mimic a “quiet start” continuum initialization by placing particles at the center of the spatial and velocity cells and weighting them based on the initial distribution function $f(x, v, t = 0)$.

Even with an efficient “quiet start” method, however, noise may still build up in long-time simulations. Particles being added or removed from the domain along with the mixing of the particles can degrade the accuracy of the simulation over time and increase the level of noise. Resampling, or remapping, is a common method to reduce noise further in such long-time runs. In resampling algorithms, particle weights are periodically adjusted to reform a more even distribution. The particles may also be mapped back to certain locations in x and v , often the cell center. Recent work [41] has shown that this process can be done while preserving the overall structure of the simulation as well. In [102], Myers et. al. utilize these methods to observed nonlinear damping effects in their plasma simulations on long time scales.

Reduction in noise over long time scales can be similarly achieved by significantly scaling up the size of simulations, in particular the number of particles per cell. Given an efficient and scalable algorithm, this method can be a more simple approach to noise reduction. The PETSc-PIC algorithm, therefore, effectively reduces noise by using “quiet start” algorithms for initialization and appropriately increasing the number of particles per cell.

4.2 A Survey of Relevant Timesteppers

A key piece to achieving structure-preservation in the collisionless regime lies in the formulation of conservative timestepping integrators. The PETSc-PIC algorithm primarily relies on basic symplectic type integrators, which preserve the symplectic structure of the phase space, to step forward in time. We will briefly review the foundation of basic symplectic integrators in this section and then consider an alternate integrator, the discrete gradient integrator, which preserves the first integral of a Hamiltonian system, H ,

$$H = p^T \dot{q} - L(q, \dot{q}), \quad (4.16)$$

where $q = (q_1, q_2, \dots, q_d)^T$ is the generalized coordinates set, $p = (p_1, p_2, \dots, p_d)^T$ is the generalized momenta and L is the Lagrangian of the system.

4.2.1 Basic Symplectic Integrators

Consider the symplectic space (V, S) with V open in m -dimensional Euclidian space \mathbb{R}^m with points denoted by $v = (v^1, \dots, v^m)$, and S is a bilinear form $S : V \times V \rightarrow \mathbb{R}$. Recall that S is skew-symmetric, i.e. $S(x, y) = -S(y, x)$, for any $x, y \in \mathbb{R}^m$. We attach the Hamiltonian, $H : V \rightarrow \mathbb{R}$, to the symplectic space (V, S) to form a Hamiltonian system (V, S, H) and give the continuous form of the Hamiltonian differential equations,

$$\dot{v} = X_H(v), \quad (4.17)$$

where $X_H : V \rightarrow \mathbb{R}$ is the Hamiltonian vector field associated with the Hamiltonian function H . Furthermore, we introduce the notion of a first integral, which is a smooth function $A : V \rightarrow \mathbb{R}$ that is constant along any solution and satisfies the orthogonality condition,

$$\nabla A(v) \cdot X_H(v) = 0. \quad (4.18)$$

Note that, given the skew-symmetry of matrix S , the Hamiltonian H is a first integral for the Hamiltonian system (V, S, H) .

Symplectic integrators, first developed by Ronald Ruth in 1983 [121], are canonical transformations forming a subclass of geometric integrators. A numerical scheme is considered a symplectic integrator if it conserves the symplectic 2-form $dp \wedge dq$. Symplectic integrators up to the fourth order are defined in PETSc. Each will be briefly reviewed here. The first order integrator is defined by the system,

$$p_{i+1} = p_i - \Delta t \nabla_q H(p_{i+1}, q_i), \quad (4.19)$$

$$q_{i+1} = q_i + \Delta t \nabla_p H(p_i, q_{i+1}), \quad (4.20)$$

with time steps Δt . This is commonly referred to as the *symplectic Euler* system. If the Hamiltonian is separable, i.e. $H(p, q) = T(p) - U(q)$, the symplectic Euler method becomes explicit.

Given a Hamiltonian of the form $H(p, q) = \frac{1}{2}p^T M^{-1} p + U(q)$, where $M \in \mathbb{R}^{d \times d}$ is a symmetric positive-definite matrix, the second order *Störmer-Verlet* integrator is defined by,

$$p_{i+1/2} = p_i - \frac{\Delta t}{2} \nabla U(q_i), \quad (4.21)$$

$$q_{i+1} = q_i + \Delta t M^{-1} p_{i+1/2}, \quad (4.22)$$

$$p_{i+1} = p_{i+1/2} - \frac{\Delta t}{2} \nabla U(q_{i+1}). \quad (4.23)$$

The implicit midpoint integrator is a similar second order symplectic integrator which has an implementation in PETSc. The area preservation quality of first and second order symplectic integrators in comparison to common non-symplectic integrators are shown in Figure 4.1. From this figure, we can see that the first (symplectic Euler) and second (Störmer-Verlet) order symplectic method are superior in area preservation than the non-symplectic integrators. However, we may also note that the second order integrator preserves the overall phase

space structure better than the first order method.

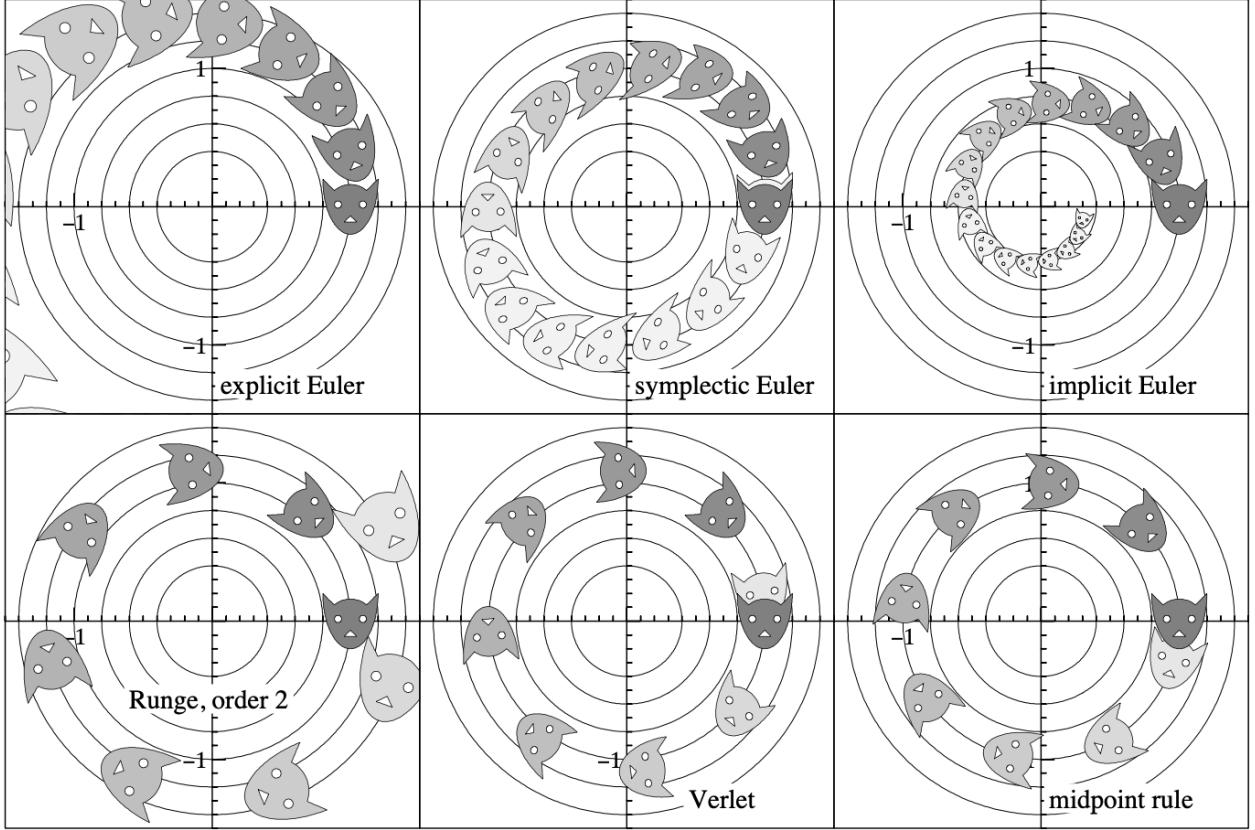


Figure 4.1: Area preservation of a variety of common numerical integration methods for the harmonic oscillator [55].

Ruth also considers symplectic integrators of order three in [121] for separable Hamiltonians of the form,

$$H(p, q) = T(p) + V(q). \quad (4.24)$$

However, the derivation is somewhat more involved for integrator orders higher than two.

The third order symplectic integrator is a three stage method that takes the form,

$$p_{n+1/3} = p_n - c_1 \Delta t \nabla_q H(q_n), \quad (4.25)$$

$$q_{n+1/3} = q_n + d_1 \Delta t \nabla_p H(p_{n+1/3}), \quad (4.26)$$

$$p_{n+2/3} = p_{n+1/3} - c_2 \Delta t \nabla_q H(q_{n+1/3}), \quad (4.27)$$

$$q_{n+2/3} = q_{n+1/3} + d_2 \Delta t \nabla_p H(p_{n+2/3}), \quad (4.28)$$

$$p_{n+1} = p_{n+2/3} - c_3 \Delta t \nabla_q H(q_{n+2/3}), \quad (4.29)$$

$$q_{n+1} = q_{n+2/3} + d_3 \Delta t \nabla_p H(p_{n+1}). \quad (4.30)$$

There are a variety of values for c_i and d_i that correctly solve the system. These constants must satisfy the following conditions,

$$c_1 + c_2 + c_3 = 1, \quad (4.31)$$

$$d_1 + d_2 + d_3 = 1, \quad (4.32)$$

$$c_2 d_1 + c_3 (d_1 + d_2) = \frac{1}{2}, \quad (4.33)$$

$$c_2 d_1^2 + c_3 (d_1 + d_2)^2 = \frac{1}{3}, \quad (4.34)$$

$$d_3 + d_2 (c_1 + c_2)^2 + d_1 c_1^2 = \frac{1}{3}. \quad (4.35)$$

For example, if we set $d_3 = 1$, we can obtain the following values for the constants c_i and d_i ,

$$c_1 = \frac{7}{24}, \quad (4.36)$$

$$c_2 = \frac{3}{4}, \quad (4.37)$$

$$c_3 = -\frac{1}{24}, \quad (4.38)$$

$$d_1 = \frac{2}{3}, \quad (4.39)$$

$$d_2 = -\frac{2}{3}, \quad (4.40)$$

$$d_3 = 1. \quad (4.41)$$

Shortly after publishing [121], Ruth went on to derive a fourth-order symplectic integrator during a sabbatical at CERN, a fact that comes from a review of Ruth's work by Étienne Forest in [44]. The fourth-order integrator was eventually published in 1990 in [45]. The integrator takes the form,

$$p_{n+1/4} = p_n - c_1 \Delta t \nabla_q H(q_n), \quad (4.42)$$

$$q_{n+1/4} = q_n + d_1 \Delta t \nabla_p H(p_{n+1/4}), \quad (4.43)$$

$$p_{n+2/4} = p_{n+1/4} - c_2 \Delta t \nabla_q H(q_{n+1/4}), \quad (4.44)$$

$$q_{n+2/4} = q_{n+1/4} + d_2 \Delta t \nabla_p H(p_{n+2/4}), \quad (4.45)$$

$$p_{n+3/4} = p_{n+2/4} - c_3 \Delta t \nabla_q H(q_{n+2/4}), \quad (4.46)$$

$$q_{n+3/4} = q_{n+2/4} + d_3 \Delta t \nabla_p H(p_{n+3/4}), \quad (4.47)$$

$$p_{n+1} = p_{n+3/4} - c_4 \Delta t \nabla_q H(q_{n+3/4}), \quad (4.48)$$

$$q_{n+1} = q_{n+3/4} + d_4 \Delta t \nabla_p H(p_{n+1}). \quad (4.49)$$

The constraint equations for c_i and d_i are quite complex to write out in full detail. For example, one such constraint is given as,

$$\begin{aligned}
& \left(d_1^2 c_1 c_2 + d_1^2 c_1 c_3 + d_1^2 c_1 c_4 - d_1^2 c_1 - 2 d_1 c_1 d_2 c_2 + 2 d_1 c_1 d_2 c_3 + 2 d_1 c_1 d_2 c_4 - \right. & (4.50) \\
& 2 d_1 c_1 d_2 - 2 d_1 c_1 c_2 d_3 - 2 d_1 c_1 c_2 d_4 + 2 d_1 c_1 c_2 - 2 d_1 c_1 d_3 c_3 + 2 d_1 c_1 d_3 c_4 - 2 d_1 c_1 d_3 - \\
& 2 d_1 c_1 c_3 d_4 + 2 d_1 c_1 c_3 - 2 d_1 c_1 d_4 c_4 - 2 d_1 c_1 d_4 + 2 d_1 c_1 c_4 - d_1 d_2 c_2^2 + 2 d_1 d_2 c_2 c_3 + \\
& 2 d_1 d_2 c_2 c_4 - 2 d_1 d_2 c_2 - d_1 c_2^2 d_3 - d_1 c_2^2 d_4 + d_1 c_2^2 - 2 d_1 c_2 d_3 c_3 + 2 d_1 c_2 d_3 c_4 - \\
& 2 d_1 c_2 d_3 - 2 d_1 c_2 c_3 d_4 + 2 d_1 c_2 c_3 - 2 d_1 c_2 d_4 c_4 - 2 d_1 c_2 d_4 + 2 d_1 c_2 c_4 - d_1 d_3 c_3^2 + \\
& 2 d_1 d_3 c_3 c_4 - 2 d_1 d_3 c_3 - d_1 c_3^2 d_4 + d_1 c_3^2 - 2 d_1 c_3 d_4 c_4 - 2 d_1 c_3 d_4 + 2 d_1 c_3 c_4 - \\
& d_1 d_4 c_4^2 - 2 d_1 d_4 c_4 + d_1 c_4^2 + c_1 d_2^2 c_3 + c_1 d_2^2 c_4 - c_1 d_2^2 - 2 c_1 d_2 d_3 c_3 + \\
& 2 c_1 d_2 d_3 c_4 - 2 c_1 d_2 d_3 - 2 c_1 d_2 c_3 d_4 + 2 c_1 d_2 c_3 - 2 c_1 d_2 d_4 c_4 - 2 c_1 d_2 d_4 + 2 c_1 d_2 c_4 + \\
& c_1 d_3^2 c_4 - c_1 d_3^2 - 2 c_1 d_3 d_4 c_4 - 2 c_1 d_3 d_4 + 2 c_1 d_3 c_4 - c_1 d_4^2 + d_2^2 c_2 c_3 + \\
& d_2^2 c_2 c_4 - d_2^2 c_2 - 2 d_2 c_2 d_3 c_3 + 2 d_2 c_2 d_3 c_4 - 2 d_2 c_2 d_3 - 2 d_2 c_2 c_3 d_4 + 2 d_2 c_2 c_3 - \\
& 2 d_2 c_2 d_4 c_4 - 2 d_2 c_2 d_4 + 2 d_2 c_2 c_4 - d_2 d_3 c_3^2 + 2 d_2 d_3 c_3 c_4 - 2 d_2 d_3 c_3 - d_2 c_3^2 d_4 + \\
& d_2^- c_3^2 - 2 d_2 c_3 d_4 c_4 - 2 d_2 c_3 d_4 + 2 d_2 c_3 c_4 - d_2 d_4 c_4^2 - 2 d_2 d_4 c_4 + d_2 c_4^2 + \\
& c_2 d_3^2 c_4 - c_2 d_3^2 - 2 c_2 d_3 d_4 c_4 - 2 c_2 d_3 d_4 + 2 c_2 d_3 c_4 - c_2 d_4^2 + d_3^2 c_3 c_4 - \\
& d_3^2 c_3 - 2 d_3 c_3 d_4 c_4 - 2 d_3 c_3 d_4 + 2 d_3 c_3 c_4 - d_3 d_4 c_4^2 - 2 d_3 d_4 c_4 + d_3 c_4^2 - \\
& \left. c_3 d_4^2 - d_4^2 c_4 \right) = 0.
\end{aligned}$$

This set of constraints does not have an obvious solution. However, it is possible to reduce the

system to a more simple set of analytic equations and calculate the values for the constants,

$$c_1 = c_4 = \frac{1}{2(2 - 2^{1/3})}, \quad (4.51)$$

$$c_2 = c_3 = \frac{1 - 2^{1/3}}{2(2 - 2^{1/3})}, \quad (4.52)$$

$$d_1 = d_3 = \frac{1}{2 - 2^{1/2}}, \quad (4.53)$$

$$d_2 = -\frac{2^{1/3}}{2 - 2^{1/3}}, \quad (4.54)$$

$$d_4 = 0. \quad (4.55)$$

An alternate derivation of these constants was presented by Yoshida in [144].

4.2.2 Discrete Gradient Integrators

In [52], Gonzalez developed the formalism for the design of conservative time-integration schemes for Hamiltonian systems with symmetry. The main result of which is that, through the introduction of discrete gradients, implicit second-order conservative schemes can be constructed for general systems which preserve the Hamiltonian along with first integrals that arise from symmetries. These discrete gradient integrators can, therefore, offer an alternative to the basic symplectic integrators and other conservative timesteppers, such as implicit midpoint, for solving the Vlasov-Poisson-Landau system. Furthermore, as we will show in Section 5.2, in collisional plasma systems, discrete gradient integrators will preserve entropy monotonicity.

Now we consider the framework necessary for the construction of numerical approximation schemes of the form,

$$\frac{u_{n+1} - u_n}{\Delta t} = \mathbf{X}_H(u_{n+1}, u_n), \quad (4.56)$$

that inherit the integral A , where $\mathbf{X}_H : V \times V \rightarrow \mathbb{R}^m$ is a two-point approximation to the exact vector field, X_H , e.g. $\mathbf{X}_H(u_{n+1}, u_n) \approx X_H(u_{n+\frac{1}{2}})$ and $u_{n+\frac{1}{2}} = \frac{1}{2}(u_{n+1} + u_n)$. We use

u here to denote an arbitrary variable to be evolved in time. Assume there exists a vector $\bar{\nabla}A(x, y) \in \mathbb{R}^m$, for any arbitrary $x, y \in \mathbb{R}$, that has the properties,

$$\bar{\nabla}A(x, y) \approx \nabla A\left(\frac{x+y}{2}\right) \text{ and,} \quad (4.57)$$

$$\bar{\nabla}A(x, y) \cdot (y - x) = A(y) - A(x). \quad (4.58)$$

We can therefore extend these properties to the approximation scheme such that,

$$A(u_{n+1}) - A(u_n) = \bar{\nabla}A(u_n, u_{n+1}) \cdot (u_{n+1} - u_n), \quad (4.59)$$

$$= \Delta t \bar{\nabla}A(u_n, u_{n+1}) \cdot \mathbf{X}_H(u_n, u_{n+1}). \quad (4.60)$$

It can be shown that if the approximate vector field \mathbf{X}_H satisfies the discrete orthogonality condition,

$$\bar{\nabla}A(x, y) \cdot \mathbf{X}_H(x, y) = 0, \quad \forall x, y \in V, \quad (4.61)$$

then A is an integral of (4.56). We can now make the following definition:

Theorem 4.2.1. *A discrete gradient for a smooth function $A : V \rightarrow \mathbb{R}$ is a mapping $\bar{\nabla} : V \times V \rightarrow \mathbb{R}^m$ with the following properties.*

1. *Directionality:* $\bar{\nabla}A(x, y) \cdot (y - x) = A(y) - A(x)$ for any $x, y \in V$.
2. *Consistency:* $\bar{\nabla}A(x, y) = \nabla A\left(\frac{x+y}{2}\right) + \mathcal{O}(\|y - x\|)$ for all $x, y \in V$ with $\|y - x\|$ sufficiently small.

We refer to the new system $(V, S, \bar{\nabla}, H)$ as a *discrete Hamiltonian system* and associate it with the difference equation (4.56), where the discrete Hamiltonian vector field, \mathbf{X}_H , is defined as

$$\mathbf{X}_H = S(x, y) \bar{\nabla}A(x, y), \quad (4.62)$$

Now that we have a framework for the construction of a discrete gradient, we can present

one potential form for the discrete gradient, often referred to as the midpoint discrete gradient.

Proposition 4.2.1. *Let $A : \mathbb{R}^m \rightarrow \mathbb{R}$ be a smooth function. Then a discrete derivative for A is defined by,*

$$\bar{\nabla}A(x, y) = \nabla A\left(\frac{x+y}{2}\right) + \frac{A(y) - A(x) - \nabla A\left(\frac{x+y}{2}\right) \cdot (y-x)}{\|y-x\|^2}(y-x), \quad (4.63)$$

for any two points $x, y \in \mathbb{R}^m$ where $\|\cdot\|$ denotes the standard Euclidian norm in \mathbb{R}^m .

Proof. (1) To verify the directionality condition in 4.2.1 we may apply $\bar{\nabla}A(x, y)$ to $v = (y-x)$ and get,

$$\bar{\nabla}A(x, y) \cdot v = \nabla A\left(\frac{x+y}{2}\right) \cdot v + \frac{A(y) - A(x) - \nabla A\left(\frac{x+y}{2}\right) \cdot v}{\|v\|^2} v \cdot v, \quad (4.64)$$

$$= A(y) - A(x). \quad (4.65)$$

(2) To verify the consistency condition we can examine what happens to the discrete derivative (4.63) as $(y-x)$ approaches zero. We first write $f(y)$ and $f(x)$ using Taylor's Theorem,

$$f(y) = f(z) + \frac{1}{2}\nabla A(z) \cdot v + \frac{1}{4}\nabla^2 A(z) \cdot (v, v) + \frac{1}{8}\nabla^3 A(z) \cdot (v, v, v) \quad (4.66)$$

$$+ \frac{1}{16}\nabla^4 A(z) \cdot (v, v, v, v) + \mathcal{O}(\|v\|^5),$$

$$f(x) = f(z) - \frac{1}{2}\nabla A(z) \cdot v + \frac{1}{4}\nabla^2 A(z) \cdot (v, v) - \frac{1}{8}\nabla^3 A(z) \cdot (v, v, v) \quad (4.67)$$

$$+ \frac{1}{16}\nabla^4 A(z) \cdot (v, v, v, v) + \mathcal{O}(\|v\|^5).$$

This implies,

$$f(y) - f(x) - \nabla A(z) \cdot v = \frac{1}{4}\nabla^3 A(z) \cdot (v, v, v) + \mathcal{O}(\|v\|^5). \quad (4.68)$$

Letting the vector $v = (y-x) = \alpha w$ where $\alpha > 0$ and $w \in \mathbb{R}^m$ is a unit vector, the last

expression can be written as,

$$f(y) - f(x) - \nabla A(z) \cdot v = \frac{1}{4} \alpha^3 \nabla^3 A(z) \cdot (w, w, w) + \mathcal{O}(\alpha^5). \quad (4.69)$$

Plugging this expression into (4.63) then gives the relation,

$$\bar{\nabla} A(x, y) = \nabla A(z) + \left(\frac{1}{4} \alpha^2 \nabla^3 A(z) \cdot (w, w, w) + \mathcal{O}(\alpha^4) \right) w. \quad (4.70)$$

This implies $\bar{\nabla} A(x, y)$ is well defined as $\alpha = \|y - x\| \rightarrow 0$ and, therefore, satisfies the consistency requirement. \square

Similar proofs may be conducted for the average value discrete gradient, defined as,

$$\bar{\nabla} A(x, y) = \int_0^1 \nabla A(\xi y + (1 - \xi)x) d\xi. \quad (4.71)$$

The average value discrete gradient, introduced in [60], is considered in [66] for constructing an entropy monotonic Landau operator. This will be discussed in more detail in Section 5.2, but it is worth noting at this point that the midpoint discrete gradient is simpler to evaluate at each time step and will, therefore, be the preferred choice in this work.

4.2.3 Discrete Gradient Jacobian Considerations

As we have previously discussed, the PETSc timestepping module `TS` attaches a `SNES` object when implicit methods are utilized. This `SNES` object, which conducts solves using a Newton-like method, often requires the definition of an appropriate Jacobian. In simple Hamiltonian systems, such as a Harmonic oscillator, deriving a Jacobian by differentiating the Hamiltonian function is a relatively simple task. In the plasma systems considered in this research, however, differentiating the equations of motion is not straightforward. For systems such as these, we may employ a finite difference approach to approximate the

Jacobian matrix, J ,

$$J_{ij} = \frac{\partial A_i}{\partial \mathbf{x}_j} \approx \frac{A_i(\mathbf{x} + \delta \mathbf{u}_j) - A_i(\mathbf{x})}{\delta}, \quad (4.72)$$

where \mathbf{u}_j is a unit vector and the choice of δ is made by PETSc internally. In most cases, δ is defined as the square root. In the case of the discrete gradient integrator, the finite difference method correctly approximates the Jacobian for the system,

$$\frac{x_{n+1} - x_n}{\Delta t} = S \nabla A, \quad (4.73)$$

but does not take into account the extra “Gonzalez” expansion term in $\bar{\nabla}A$. Thus, for an accurate Jacobian approximation, we must consider the derivative of the expansion term with respect to the $n + 1$ step and add that to the finite difference approximation during each solve. This process is similarly done for the implicit Runge-Kutta timestepper.

The differentiation of the discrete gradient (4.63) outright is a complex task but it may be simplified by breaking (4.63) into pieces via the product rule. Before differentiating, we rewrite the discrete gradient (4.63) in the form,

$$\begin{aligned} \bar{\nabla}A &= \nabla A \left(\frac{x_{n+1} + x_n}{2} \right) \\ &+ \left[\frac{A(x_{n+1}) - A(x_n) - \nabla A(\frac{x_{n+1} + x_n}{2}) \cdot (x_{n+1} - x_n)}{\|x_{n+1} - x_n\|} \frac{(x_{n+1} - x_n)}{\|x_{n+1} - x_n\|} \right]. \end{aligned} \quad (4.74)$$

We may then break the expansion term into three separate elements,

$$\begin{aligned} \bar{\nabla}A &= \nabla A \left(\frac{x_{n+1} + x_n}{2} \right) \\ &+ \left(A(x_{n+1}) - A(x_n) - \nabla A(\frac{x_{n+1} + x_n}{2}) \cdot (x_{n+1} - x_n) \right) \\ &\times \left(\frac{1}{\|x_{n+1} - x_n\|} \right) \left(\frac{(x_{n+1} - x_n)}{\|x_{n+1} - x_n\|} \right). \end{aligned} \quad (4.75)$$

The derivatives of each of these elements are given by,

$$\frac{d}{dx_{n+1}} \frac{(x_{n+1} - x_n)}{\|x_{n+1} - x_n\|} = 0, \quad (4.76)$$

$$\frac{d}{dx_{n+1}} \frac{1}{\|x_{n+1} - x_n\|} = \frac{(x_{n+1} - x_n)}{\|x_{n+1} - x_n\|^3}, \quad (4.77)$$

$$\begin{aligned} \frac{d}{dx_{n+1}} (\text{Numerator}) &= \left(\nabla F(x_{n+1}) - \nabla F\left(\frac{x_{n+1} + x_n}{2}\right) \right. \\ &\quad \left. - \frac{1}{2} H\left(\frac{x_{n+1} + x_n}{2}\right) \cdot (x_{n+1} - x_n) \right) \end{aligned} \quad (4.78)$$

We may then combine all of these derivatives using the product rule to obtain the full derivative of the discrete gradient,

$$\frac{d}{dx_{n+1}} \bar{\nabla} F = \frac{d}{dx_{n+1}} \nabla F \left(\frac{x_{n+1} + x_n}{2} \right) + W \frac{(x_{n+1} - x_n)^T}{\|x_{n+1} - x_n\|^2}, \quad (4.79)$$

where,

$$\begin{aligned} W &= \left(F(x_{n+1}) - F(x_n) - \nabla F\left(\frac{x_{n+1} + x_n}{2}\right) \cdot (x_{n+1} - x_n) \right) \frac{(x_{n+1} - x_n)}{\|(x_{n+1} - x_n)\|^2} \\ &\quad + \left(\nabla F(x_{n+1}) - \nabla F\left(\frac{x_{n+1} + x_n}{2}\right) - \frac{1}{2} H\left(\frac{x_{n+1} + x_n}{2}\right) \cdot (x_{n+1} - x_n) \right), \end{aligned} \quad (4.80)$$

and the derivative of $\nabla F\left(\frac{x_{n+1} + x_n}{2}\right)$ is approximated by the finite difference method. We attach the derivative of the expansion term when we form the jacobian matrix within each SNES solve.

4.3 A Mathematical Review of Landau Damping

We seek to first derive a set of equations to understand the damping of plasma oscillations in our system and to calculate expected values for the damping rate and electric field oscillation frequency. These expressions are found by first deriving the dispersion relation for a plasma. The derivation shown follows from [29]. Consider a uniform plasma with an

initial distribution $f_0(v)$ with zero initial electric and magnetic fields, $\mathbf{E}_0 = \mathbf{B}_0 = 0$. To first order, the perturbation in $f(x, v, t)$ is denoted by $f_1(x, v, t)$ such that,

$$f(x, v, t) = f_0(v) + f_1(x, v, t). \quad (4.81)$$

Plugging (4.81) in to (3.4) gives,

$$\frac{\partial f_1}{\partial t} + \mathbf{v} \cdot \frac{\partial f_1}{\partial \mathbf{x}} - \frac{q_e}{m_e} \mathbf{E}_1 \cdot \frac{\partial f_0}{\partial \mathbf{v}} = 0. \quad (4.82)$$

Assuming that the ions are massive and fixed and that the waves are one-dimensional plane waves $f_1 \propto e^{i(kx-\omega t)}$, (4.82) becomes,

$$f_1 = \frac{i q_e E_x}{m_e} \frac{\partial f_0 / \partial v_x}{\omega - k v_x}. \quad (4.83)$$

Recall the Poisson equation (4.2), with the potential ϕ replaced by the divergence of the electric field,

$$\begin{aligned} \nabla \cdot \mathbf{E} &= \nabla \cdot \mathbf{E}_1 = -\frac{\rho}{\epsilon_0} \\ &= -\frac{1}{\epsilon_0} \left(\sigma - q_e \int (f_0(v) + f_1(x, v, t)) dv \right). \end{aligned} \quad (4.84)$$

With zero initial electric field, the electric field vector is replaced by the electric perturbation, \mathbf{E}_1 , which takes the form $\mathbf{E}_1 = E_x e^{i(kx-\omega t)} \hat{\mathbf{x}}$. Furthermore, at equilibrium, the neutralizing background is equal to the total weight of the electron distribution, $\sigma = q_e \int f_0 dv$, leaving only the perturbation term f_1 in the Poisson equation. Thus we are left with,

$$ik\epsilon_0 E_x = -q_e \int f_1 dv. \quad (4.85)$$

Substituting (4.83) into (4.85) and dividing by $ik\epsilon_0 E_x$, we have,

$$1 = \frac{q_e^2}{km_e\epsilon_0} \int \frac{\partial f_0 / \partial v}{\omega - kv} dv. \quad (4.86)$$

The integral in (4.86) is a three-dimensional integral. However, for the Maxwellian or other factorable distribution, integration in the 2nd and 3rd dimension is simple. Evaluating the (4.86) integral in the 2nd and 3rd dimension, and substituting in the plasma frequency, $\omega_p = (n_e q_e^2 / m \epsilon_0)^{1/2}$, gives the dispersion relation,

$$1 = \frac{\omega_p^2}{k^2} \int_{-\infty}^{\infty} \frac{\partial f_0 / \partial v_x}{v_x - (\omega/k)} dv_x. \quad (4.87)$$

Landau showed that this problem can be solved rigorously by means of the Laplace transform method. Importantly, it is necessary to go around the singularity in the integrand in (4.87) in the complex plane. The solution to (4.87) takes the form,

$$\omega = \omega_r + i\gamma, \quad (4.88)$$

where ω_r represents the real oscillations of the plasma and γ the imaginary, which Landau showed to be the part of the solution driving the damping of the oscillations. Following Landau's method [29], an approximation for the oscillation and damping terms can be derived, given by,

$$\begin{aligned} \omega_r &= 1 + \frac{3}{2} \hat{k}^2, \\ \gamma &= -\sqrt{\frac{\pi}{8}} \frac{1}{\hat{k}^3} \exp -\frac{1}{2\hat{k}^2}. \end{aligned} \quad (4.89)$$

A normalized form of the wavenumber k has been introduced to simplify the equations going forward. The normalized wavenumber, \hat{k} , is given by,

$$\hat{k} = \frac{kv_{th}}{\omega_p} \quad (4.90)$$

where $v_{th} = \sqrt{k_B T/m}$ is the thermal velocity. For all examples, we non-dimensionalize so that $v_{th} = 1$. The real part of the solution to (4.87) was similarly derived by Vlasov in [135], however, Vlasov did not account for the imaginary damping term.

These approximations are valid for the case where $\hat{k} \ll 1$ but their accuracy degrades considerably as \hat{k} approaches 1 and higher. Even when $\hat{k} = 0.5$, the calculated values for ω_r and γ differ from the numerical results by at least 5%. In [95], McKinstry draws similar conclusions, electing to derive more accurate forms of (4.89) by expanding ω_r in powers of \hat{k} ,

$$\omega_r = 1 + \frac{3}{2}\hat{k}^2 + \frac{15}{8}\hat{k}^4 + \frac{147}{16}\hat{k}^6, \quad (4.91)$$

$$\gamma = -\sqrt{\frac{\pi}{8}} \left(\frac{1}{\hat{k}^3} - 6\hat{k} \right) e^{-\frac{1}{2\hat{k}^2} - \frac{3}{2} - 3\hat{k}^2 - 12\hat{k}^4}. \quad (4.92)$$

These new expressions are more accurate for \hat{k} up to 0.4 but still diverge from the correct values as \hat{k} increases further. Shalaby et. al. provided further refinements to these equations in [123], using a numerical fitting formula, taking the form,

$$\omega = 1 + \frac{3}{2}\hat{k}^2 + \frac{15}{8}\hat{k}^4 + \frac{147}{16}\hat{k}^6 + 736.437\hat{k}^8 - 14729.3\hat{k}^{10} \quad (4.93)$$

$$+ 105429\hat{k}^{12} - 370151\hat{k}^{14} + 645538\hat{k}^{16} - 448190\hat{k}^{18},$$

$$\gamma = -\sqrt{\frac{\pi}{8}} \left(\frac{1}{\hat{k}^3} - 6\hat{k} - 40.7173\hat{k}^3 + 3900.23\hat{k}^5 - 2462.25\hat{k}^7 - 274.99\hat{k}^9 \right)$$

$$\exp \left[-\frac{1}{2\hat{k}^2} - \frac{3}{2} - 3\hat{k}^2 - 12\hat{k}^4 - 575.516\hat{k}^6 + 3790.16\hat{k}^8 \right]$$

$$- 8827.54\hat{k}^{10} + 7266.87\hat{k}^{12} \right].$$

These equations give good estimates for ω_r and γ in the case where $\hat{k} = 0.5$, which is of particular interest in this paper. In fact, the values obtained from (4.93) in the case where $\hat{k} = 0.5$ and all other parameters (ω_p, v_{th}, q_e , etc.) are assumed to be 1.0 match those commonly listed as “analytic solutions” [29, 148, 102]. That being said, the accuracy of the numerical fit still decreases considerably for $\hat{k} > 0.6$.

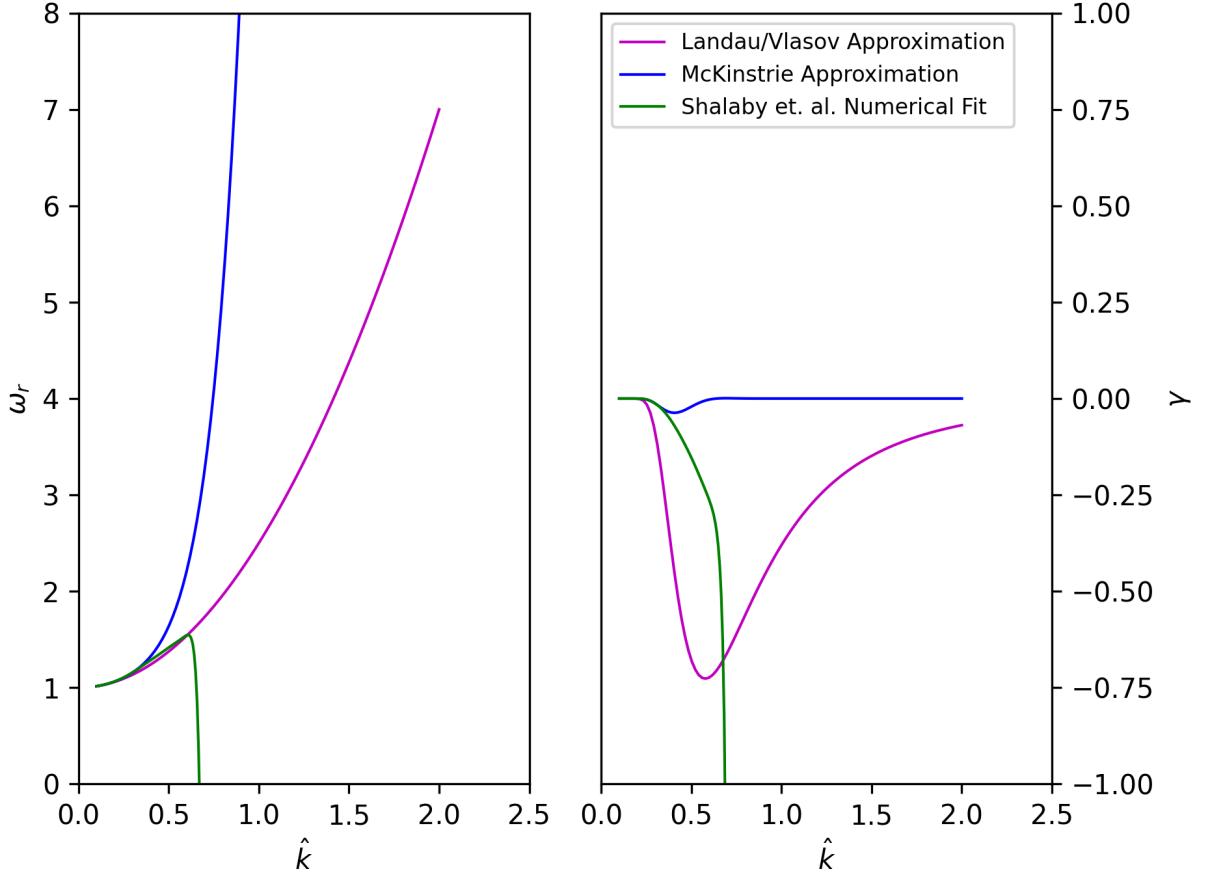


Figure 4.2: A comparison of various approximations to the Landau dispersion relation and a numerical fit to data.

4.3.1 A Numerical Approach to Landau’s Dispersion Relation

An alternate, and as we will show, more accurate way to calculate ω_r and γ for given values of \hat{k} is to find them by computing the zeros of (4.87). An early example of this came

from Josè Canosa in citeCanosa1973. Canosa used Muller's method [101], to calculate values for ω_r and γ , with five figure accuracy, for \hat{k} ranging from 0.25 to 2.0 in increments of 0.05 (see Table 4.1 for a selection of values).

\hat{k}	ω_r	γ
0.25	1.1056	-0.0021693
0.5	1.4156	-0.15336
0.75	1.7371	-0.46192
1.0	2.0459	-0.85134
1.5	2.6323	-1.7757
2.0	3.1891	-2.8272

Table 4.1: Values for ω_r and γ for given values of \hat{k} from [23].

We have implemented a numerical method for calculating values of ω_r and γ in Python. Using this algorithm, we may more parametrically study solutions to the dispersion relation. For example, it is of relative interest to study what effect variations in charge density have on the dynamics of the plasma. The first step of the algorithm is to plot the dispersion relation in the complex plane. An example of these plots is shown in Figure 4.3. A box within the

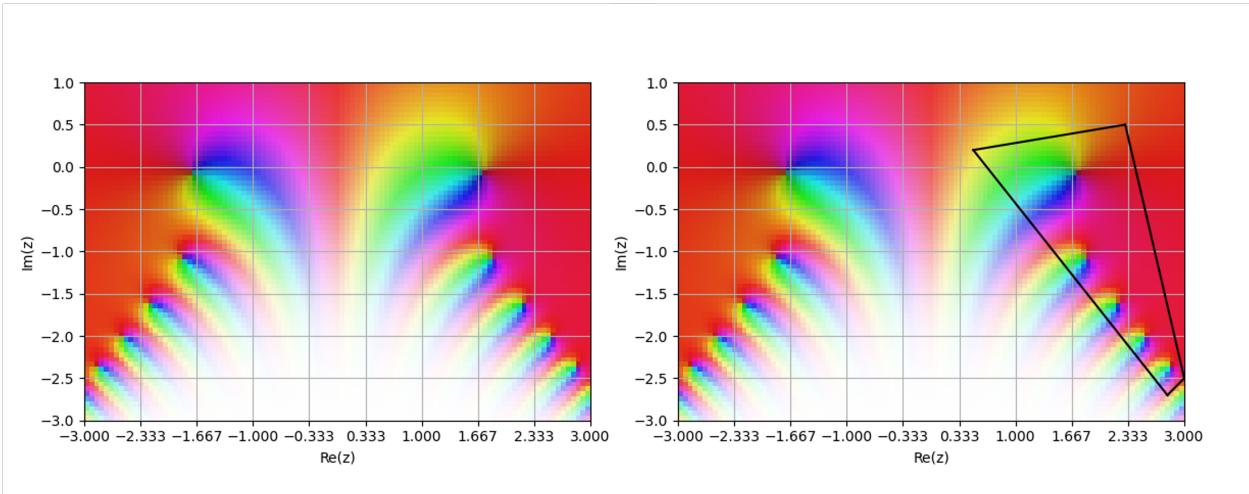


Figure 4.3: Complex plots of the Landau dispersion relation. The complex argument (phase) is represented by the color at a specific point and the magnitude is represented by the brightness. A box is added qualitatively to the plot which indicates the reduced region in which the algorithm finds roots.

domain is defined qualitatively around an area where zeros are expected. The algorithm

then creates a Hankel matrix with complex coordinates from the defined domain of interest. The eigenvalues of the Hankel matrices represent initial guesses for the roots (or zeros) [11] of the Landau dispersion relation and can be further refine with Newton iterations. The algorithm calculates four roots in each run, which represent multiple oscillation modes. The root with the largest imaginary part is chosen as it represents the Landau damping mode. The roots calculated by this algorithm are compared to those presented in [23] in Figure 4.4. A full comparison of the approximations by Landau, McKinstrie and Shalaby to the root finding results and numerical data is shown in Section 4.4.

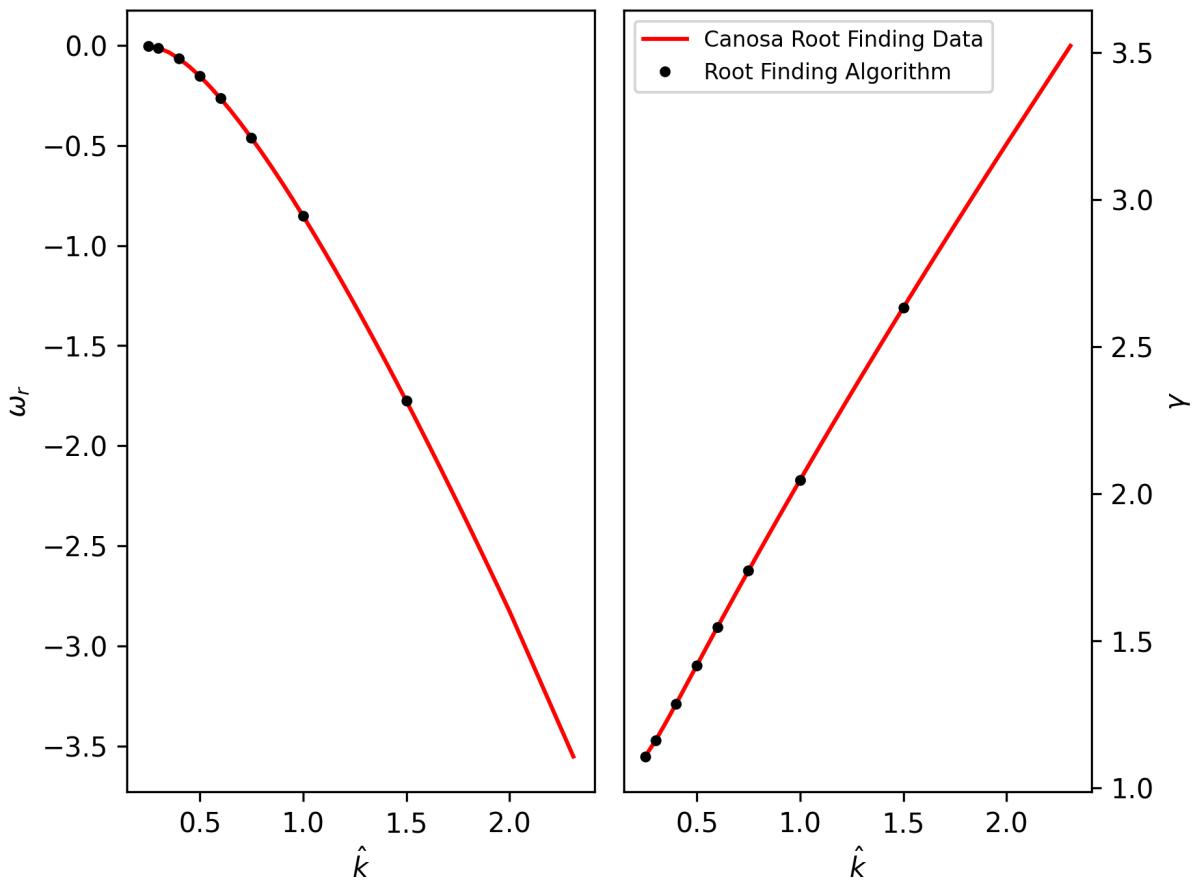


Figure 4.4: A comparison of data from previous root finding methods [23] to the algorithm used in this project.

4.4 Numerical Examples

4.4.1 Harmonic Oscillator

We begin our tests of the collisionless PETSc-PIC algorithm with the simple harmonic oscillator, a system consisting of n particles of mass, m , oscillating with some angular frequency, ω . The harmonic oscillator is a good starting point for testing a collisionless algorithm because it is conservative in the continuum and has easily derivable exact solutions. We can, therefore, use this system to test the accuracy and moment conserving qualities of the timestepping integrators in PETSc. The Hamiltonian of the one-dimensional system is,

$$H = \frac{1}{2}mv^2 + \frac{1}{2}m\omega^2x^2. \quad (4.94)$$

In higher dimensions, we simply replace the position, x , and velocity, v , terms with two-norms of the vectors, $|x|$ and $|v|$. Using Hamilton's equations,

$$\dot{q}_i = \frac{\partial H}{\partial p_i}, \quad (4.95)$$

$$\dot{p}_i = -\frac{\partial H}{\partial q_i} \quad (4.96)$$

we can obtain the equations of motion for this system,

$$\dot{x} = mv, \quad (4.97)$$

$$\dot{v} = -m\omega^2x. \quad (4.98)$$

Given the initial position and velocity of the particles, x_0 and v_0 , respectively, the exact solution of this system is,

$$x(t) = x_0 \cos(\omega t) + \frac{v_0}{\omega} \sin(\omega t), \quad (4.99)$$

$$v(t) = v_0 \cos(\omega t) - x_0 \omega \sin(\omega t). \quad (4.100)$$

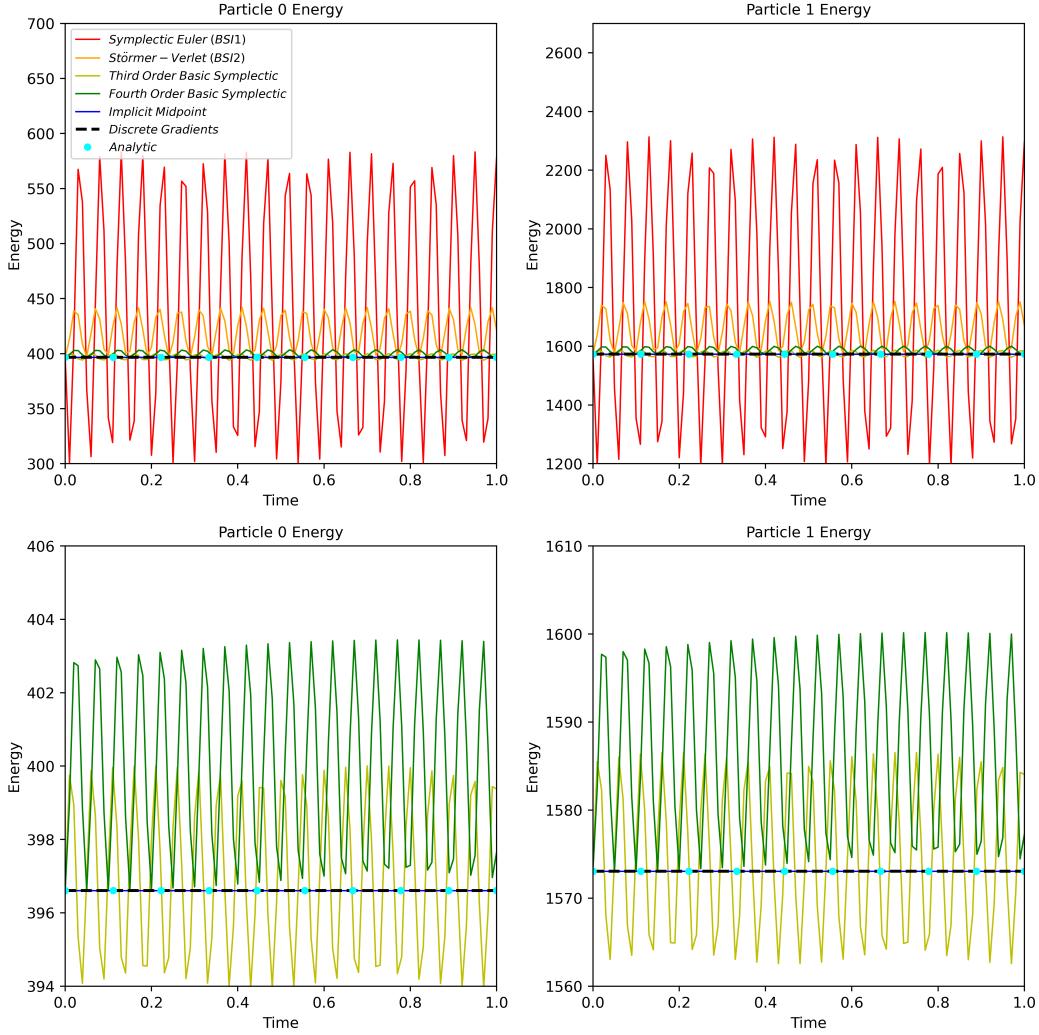


Figure 4.5: The energy of two particles in a harmonic oscillator for a variety of timestepping integrators. (Top) First through fourth order basic symplectic types are compared to implicit midpoint and discrete gradient integrators. (Bottom) Higher order integrators are zoomed in on.

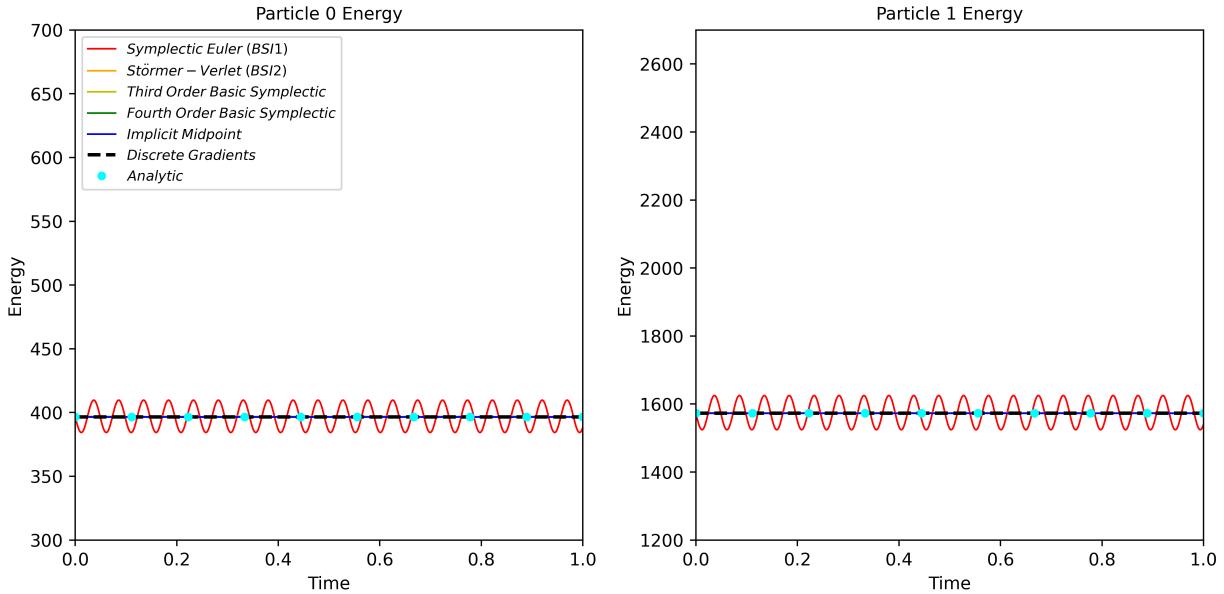


Figure 4.6: The energy of two particles in a harmonic oscillator over time for a variety of timesteppers with time steps $\Delta t = 0.001$.

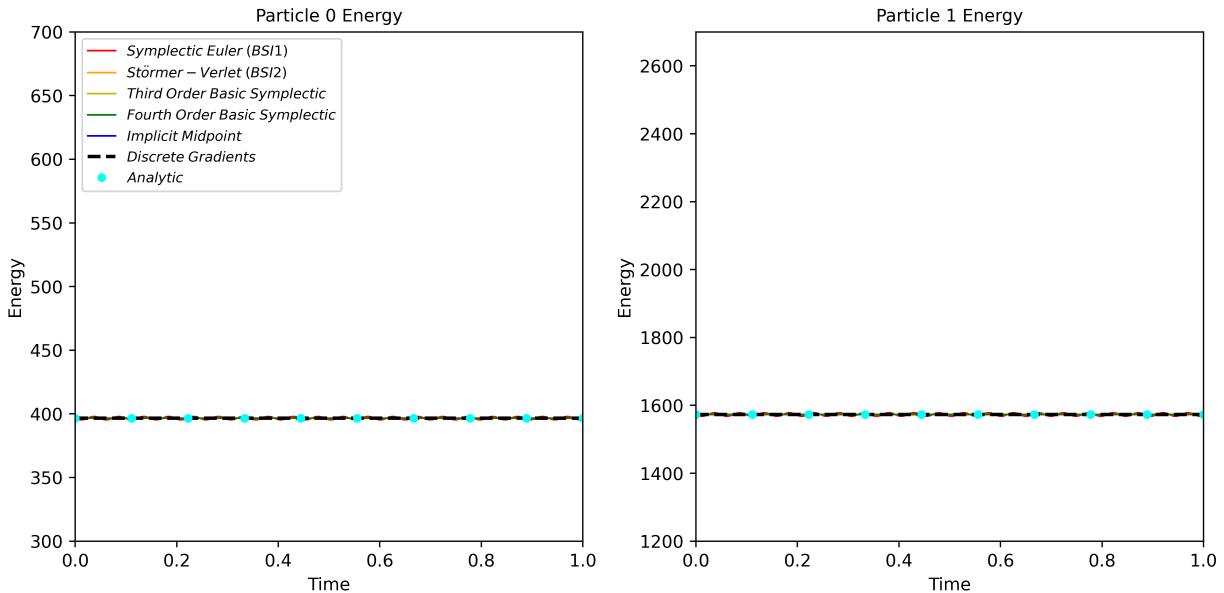


Figure 4.7: The energy of two particles in a harmonic oscillator over time for a variety of timesteppers with time steps $\Delta t = 0.0001$.

We consider the case where there are two particles oscillating inside the domain. The particles are non-interactive so considering any larger numbers of particles should have no

effect on the solver aside from increased computational time. The energy for each particle over the simulation time is shown in Figure 4.5 where the time steps are of size $\Delta t = 0.01$. The figure clearly displays a convergence in the oscillation amplitude of the energy as the symplectic integrator order is increased. When using the discrete gradients integrator, however, there is little to no oscillation in the energy over time. We also observe that the higher order methods, as well as implicit midpoint and discrete gradients converge on the analytic values for energy. The oscillations in energy may be reduced in amplitude for all integrators by decreasing the time step size, as shown in Figure 4.6 and 4.7. This is a product of the nature of basic symplectic integrators, which solve a modified system rather than the exact system, and thus may drift from the exact solution over time. This drift is a function of the time step, Δt . Therefore, reducing the step size reduces the drift.

Using the TSConvest, a convergence estimation tool in PETSc, we can obtain the order of accuracy for each timestepper. The results from TSConvest are shown in Table 4.2. As expected, the first through fourth order symplectic integrator have orders of accuracy. The implicit midpoint and discrete gradient method both have order two accuracies. This is similarly expected as the implicit midpoint method is an alternate symplectic integrator of order two and the discrete gradient method is simply implicit midpoint with an extra correction term.

Integrator	1D	2D	3D
Symplectic Euler (BSI1)	1.0	1.0	1.0
Störmer-Verlet (BSI2)	2.0	2.0	2.0
Third-Order Basic Symplectic	3.0	2.7	2.4
Fourth-Order Basic Symplectic	3.9	3.9	3.9
Implicit Midpoint	2.0	2.0	2.0
Discrete Gradient	2.0	2.0	2.0

Table 4.2: Table of convergence rates from TSConvest for a variety of timesteppers.

The results of the harmonic oscillator study show that the collisionless PETSc-PIC algorithm is sufficient in modeling simple Hamiltonian systems while conserving the energy of the system. We have also shown that the basic symplectic integrators perform as expected and that the discrete gradient integrator is second-order accurate and outperforms the symplectic integrators in conserving energy at all time step sizes. In the next section, we will extend the collisionless algorithm to a more complex Vlasov-Poisson test case, Landau damping, to show its effectiveness and accuracy in electrostatic solves.

4.4.2 Landau Damping

In order to study the linear Landau damping phenomenon, we consider the initial particle distribution,

$$f(x, v, t = 0) = \frac{1}{\sqrt{2\pi v_{th}^2}} e^{-v^2/2v_{th}^2} (1 + \alpha \cos(kx)) \quad (4.101)$$

$$(x, v) = [0, 2\pi/k] \times [-v_{max}, v_{max}]$$

where $v_{th} = \sqrt{KT_e/m}$, $\alpha = 0.01$, $k = 0.5$, $v_{max} = 10$ and the boundaries are periodic.

In this section, the results of this numerical study are presented [43]. We consider the one-dimensional (1X-1V) case of the Vlasov-Poisson system. According to (4.91), derived in Section 4.4.2, and the zero finding data from Canosa [23], the damping rate should be $\gamma = -0.153$ and the frequency of oscillations should be $\omega_r = 1.416$. All runs were conducted on a single 2.4 [GHz] 8-Core Intel Core i9 processor with 64 [GB] of memory.

To begin, we show results from the densest run of the PETSc-PIC simulation with 160 spatial cells and 8,000 particles per cell and a PIC timestep of $dt = 0.03$. We choose basic symplectic integrators for the timestepping method in these runs as they are conservative and explicit, whereas the discrete gradient methods, while conservative, require implicit solves. Figure 4.8 shows the maximum value of the electric field, $E_{max} = \max_{\Omega} |E|$, over time. The values for γ and ω_r were measured by fitting the peaks of the given data. The frequency of

oscillations describes the frequency of the electric field completing one full oscillation. Since each oscillation of E_{max} is the equivalent of one half of the electric field period, we count two E_{max} oscillations for each plasma oscillation. Values achieved by the PETSc-PIC algorithm, $\gamma = -0.1531$ and $\omega_r = 1.4124$ agree within 1% of the analytic values from Canosa and Shalaby et. al., which are assumed to be the most accurate for the case $k = 0.5$.

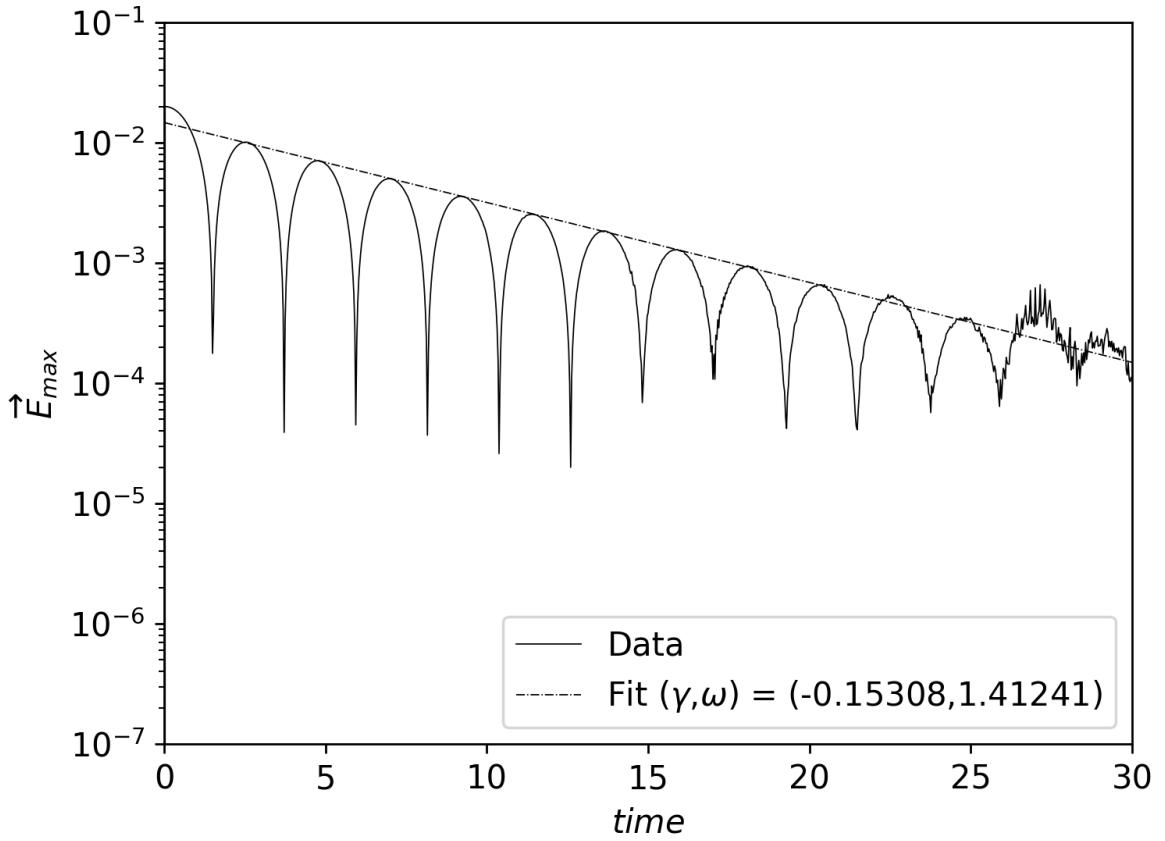


Figure 4.8: The maximum value of the electric field as a function of time for the one-dimensional linear Landau damping problem.

The total error in the moments, shown in Figure 4.9, was shown to be stable over the entire runtime. At early times, the error in momentum and energy fluctuate but each converges by $t = 20$. This convergence comes from the use of basic symplectic integrator in PETSc-PIC which guarantees the error does not grow over time. We also note that the error in the

particle solve and the finite element solve is exactly equal, apart from an increased level of noise in the particle solve. This confirms the effectiveness of the conservative projector used in PETSc-PIC. More detailed tests of the conservative projector can be found in [117].

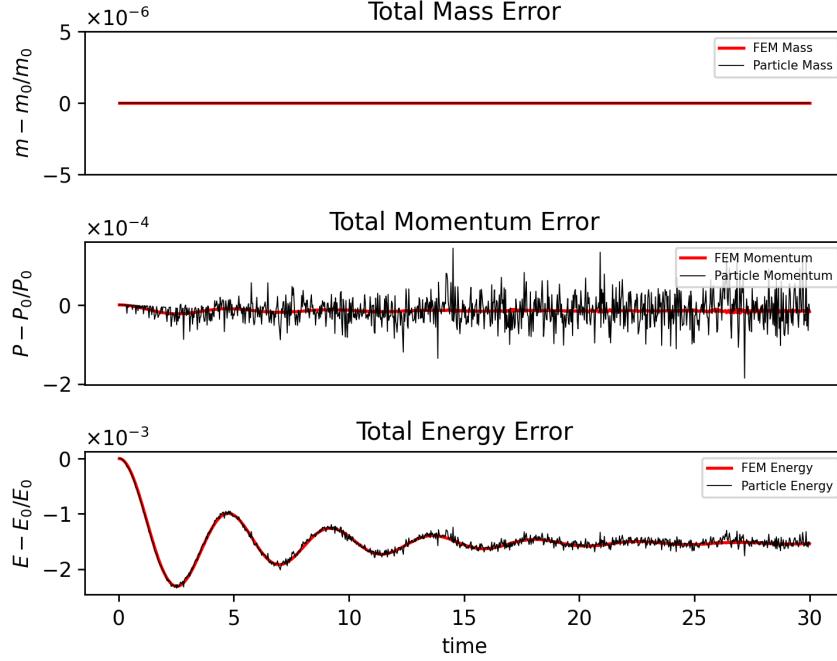


Figure 4.9: The total mass, momentum and energy error for the particle and finite element solve. The moment errors all converge to zero given a long enough time.

Lower resolution tests with a third order Runge-Kutta timestepper [16]¹ were conducted to compare the evolution of the electric field using a non-conservative timestepping method. Results of this, shown in Figure 4.10, confirm the non-conservative quality of the timestepper and show that the electric field does not oscillate as expected in this test. At this lower resolution, the basic symplectic integrator becomes noisy earlier but is still able to capture the oscillations and damping of the electric field.

Convergence studies were conducted in which either the mesh or the number of particles per cell were increased while the other was held constant. In the case of mesh convergence, the number of particles per cell was held at $N_v = 8,000$ while in the particle number convergence

¹-ts_type rk -ts_rk_type 3bs

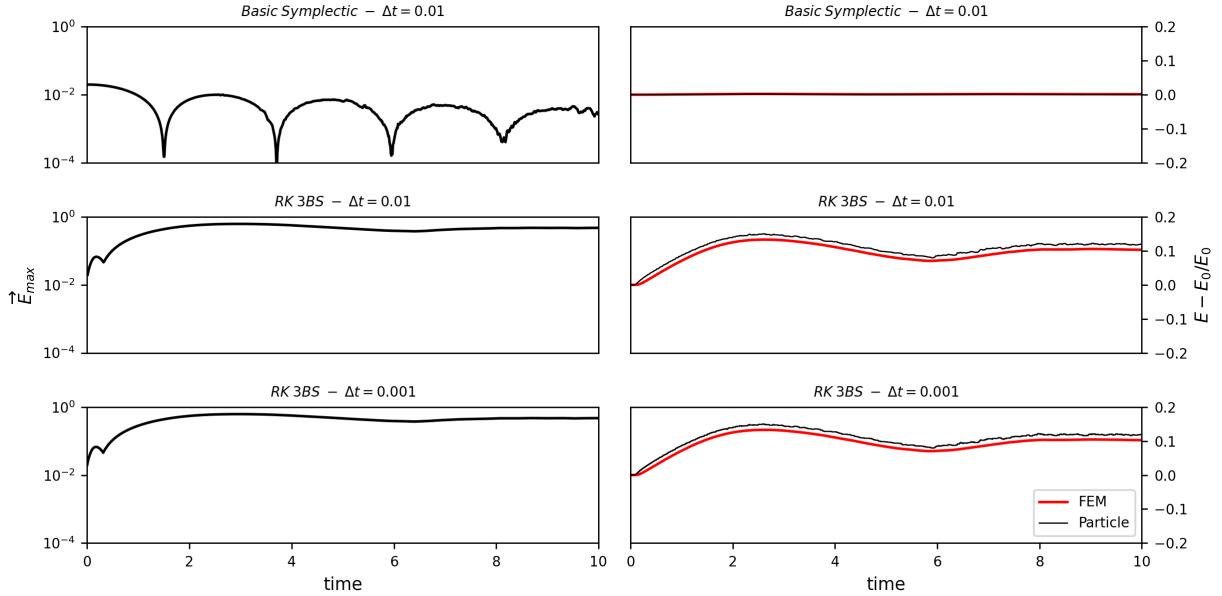


Figure 4.10: A comparison of basic symplectic integrators to a third order Runge-Kunge (RK) integrator for the Landau damping test with $N_x = 30$ and $N_v = 500$. RK methods are sensitive to the time step size, so two step sizes were run ($\Delta t = 0.01$ and $\Delta t = 0.001$). Both RK runs failed to capture any of the key dynamics in Landau damping while, even with reduced resolution, captures the correct oscillation and damping.

tests, the number of mesh cells was $N_x = 100$. We expect Monte Carlo convergence in particle number, $\mathcal{O}(1/\sqrt{N})$, and we indeed achieve this for ω_r in the upper left of Figure 4.11. Since we have such a regular initial particle distribution, we might hope to see Quasi-Monte Carlo convergence, $\mathcal{O}(1/N)$, and we do see this superconvergence in γ in the upper right of Figure 4.11. $\mathcal{O}(h^2)$ convergence is expected in the mesh resolution h since this controls the error in the electric field, and we see this in γ in the lower right of Figure 4.11. However, this convergence should quickly saturate as particle error begins to dominate, which we see in ω_r in the lower left of Figure 4.11.

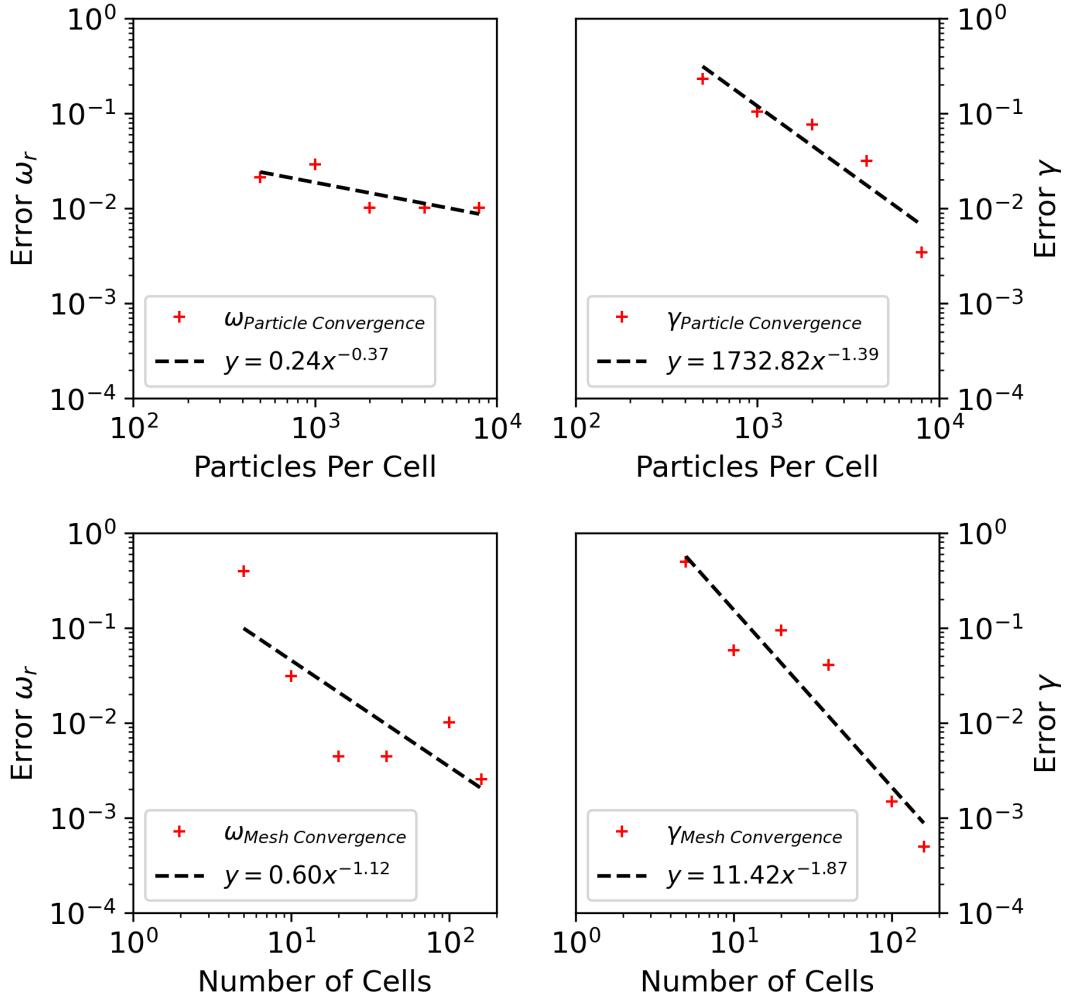


Figure 4.11: (Top) Particle per cell number convergence plots for $N_x = 100$ and (bottom) mesh convergence plots for $N_v = 8,000$.

4.4.2.1 Variations in Wavenumber and Charge Density

We have thus far shown that the PETSc-PIC algorithm is an accurate and structure-preserving method for modeling plasma systems. We next present results from tests in which the wavenumber, k , and consequently the domain size, and the charge density were varied. Varying either of these values impacts the value of the non-dimensional wavenumber \hat{k} . In the case of the wavenumber k , the calculated values for ω_r and γ were compared to the values obtained with the approximation equations (4.89) and (4.91), the numerical

fit (4.93) and the zero finding results from Table 4.1. The results from PETSc-PIC, shown in Figure 4.12, clearly show a strong deviation of the approximation equations and the numerical fit for $\hat{k} > 0.5$ while closely matching the zero finding data. This demonstrates that these approximations quickly break down outside of the small parameter range typically chosen in numerical studies of Landau damping. When considering real plasma systems in which values for k , ω_p , etc. are more dynamic, it is far more effective to use zero finding methods to calculate expected values for ω_r and γ .

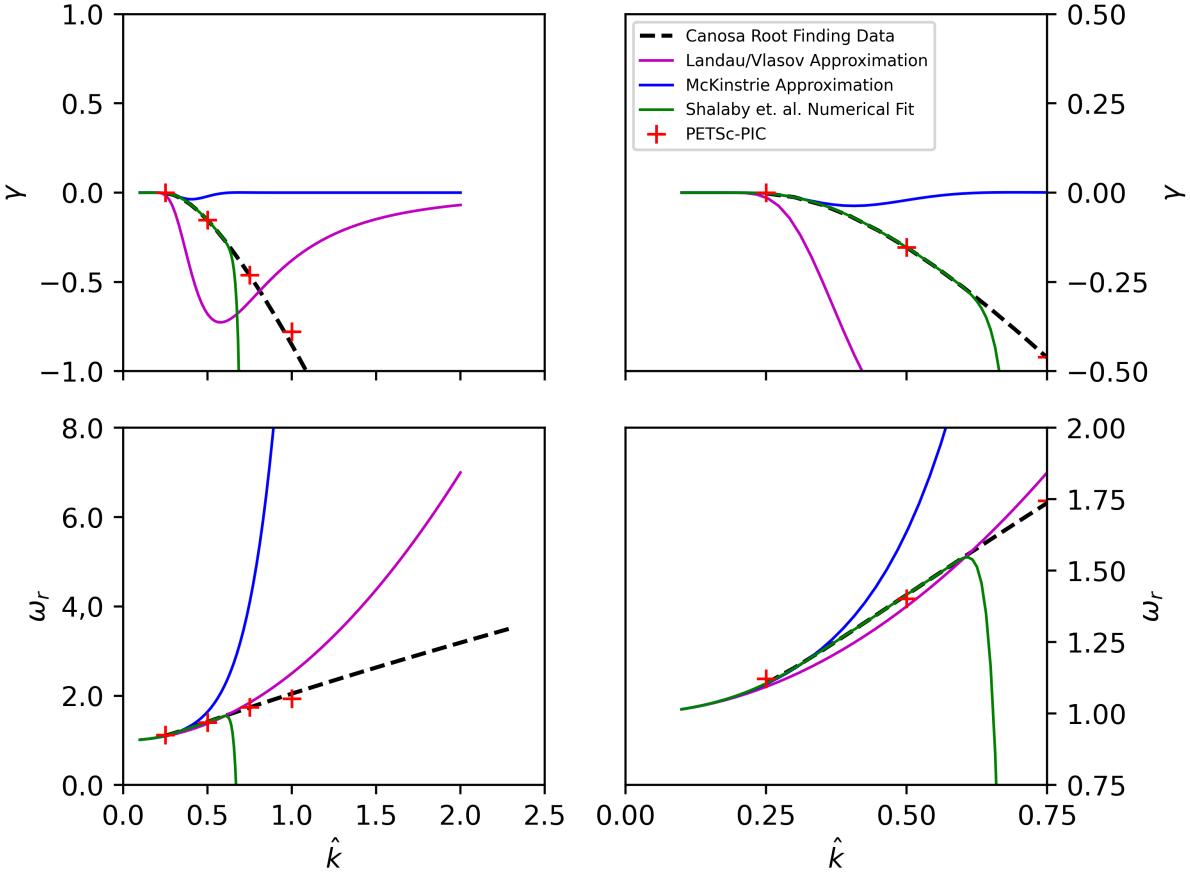


Figure 4.12: A comparison of various approximations for ω_r and γ to root finding results and numerical results from PETSc-PIC. Plots on the right are zoomed in on the region $0.0 \leq \hat{k} \leq 0.75$ to show the accuracy of each approximation before they diverge from the data.

It may be naively assumed that data from numerical tests with varying charge densities

will match the approximation equations (4.89), (4.91) and (4.93) or even the zero finding data from Canosa. However, these analytic results are based on an assumption of unchanging charge density. More specifically, these results are based on charge densities such that the plasma frequency, ω_p , is always unity. Therefore, to accurately compare analytic results to our data we must resolve the dispersion relation for varying charge densities. A zero finding algorithm, using Newton's method [126], was employed to calculate new analytic values for ω_r and γ with charge densities ranging from 0.1 to 2.0. The zero finding algorithm calculates multiple values for ω_r and γ . However, we select the solution containing the largest γ which corresponds to the smallest ω_r . Other solutions found by the algorithm represent less dominant modes which can be ignored for the purposes of this study. Figure 4.13 contains the results from the new zero finding algorithm along with data from numerical tests which agree perfectly. We observe that when the charge density is increased, the frequency of oscillations also increases. This matches the expected physical behavior of an electrically charged plasma.

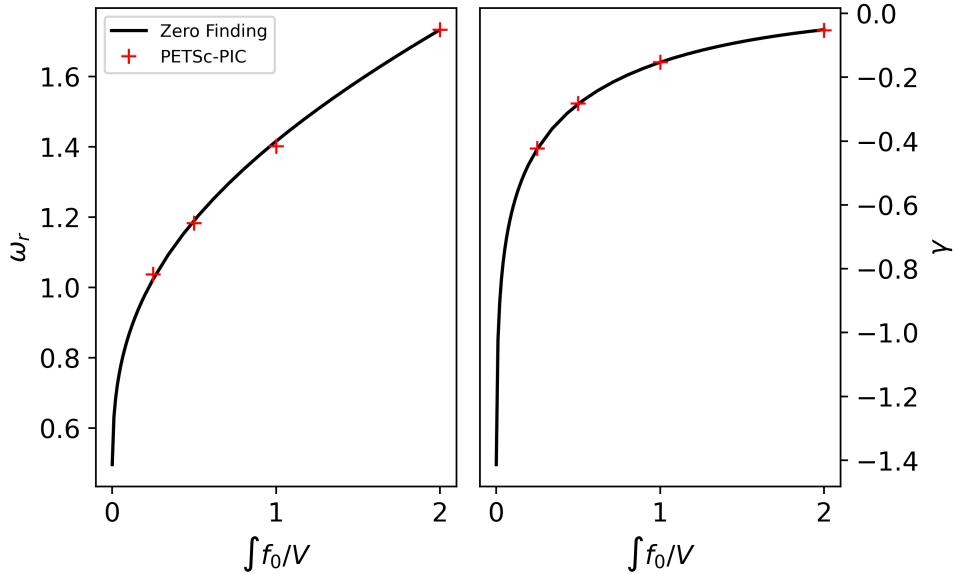


Figure 4.13: Numerical results for varying charge densities, $\int f_0/V$, compared to zero finding data. The charge density is represented on the x-axis as the integral of the initial distribution over the domain volume.

We have extended our zero finding algorithm to the case where the charge density approaches zero ($\hat{k} \rightarrow \infty$) to make note of an interesting phenomenon. At a charge density of zero, the dispersion relation has no solution. We capture this in Figure 4.13, where we observe that both ω_r and γ are asymptotic at $\int f_0/V = 0$. This can similarly be observed in the numerical results from our PETSc-PIC algorithm. As the charge density is decreased, the rate at which the electric field oscillations becomes too large to resolve numerically. In the case of $\int f_0/V = 0.25$, shown in Figure 4.14, we can only observe two full oscillations of the electric field before the simulation becomes too noisy. Theoretically, the charge density could be decreased asymptotically in our simulations to observe the damping rate and frequency trends but in practice there is too much noise to resolve any real processes in the plasma.

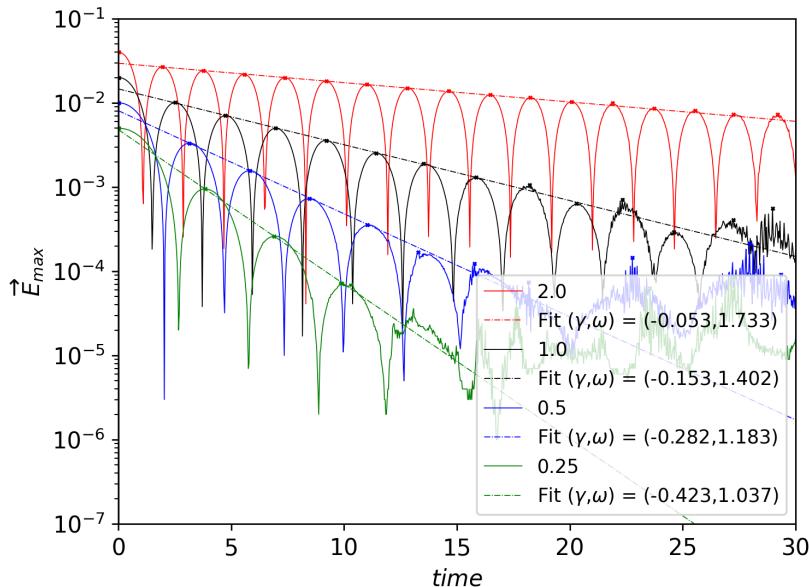


Figure 4.14: A comparison of electric field oscillations given different charge densities.

4.5 Conclusions

We have presented the collisionless PETSc-PIC algorithm, a structure-preserving Particle-in-Cell algorithm for solving the Poisson bracket in the metriplectic formulation of the Vlasov-Poisson-Landau system. The accuracy of our algorithm has been demonstrated using two test cases. In the first, the harmonic oscillator, a variety of timesteppers were used on a simple Hamiltonian system. Comparisons of the timesteppers ability to match expected energies as well as remain conservative over time were made. The order of accuracy of each timestepper was also calculated. While the symplectic timesteppers were effective in this test, in particular when the time step size was decreased, the discrete gradients integrator remained exactly conservative at all time step sizes while only requiring order two accuracy.

In the second test, Landau damping, the frequency of electric field oscillations and the damping rate of plasma oscillations in the PETSc-PIC model were compared to analytic values. The PETSc-PIC algorithm was able to closely match expected results while remaining fully conservative. We have also shown that the approximations for the frequency and damping rate break down outside of narrow ranges for the wavenumber and charge density. These approximations are often cited in numerical Landau damping studies without further context or reference to the equations used to compute the parameters, which can lead to complications in reproducing results. We have sought to provide a complete picture of Landau damping and the numerical methods we have used to simulate this phenomenon.

There are two primary areas in which this algorithm could be improved in future work. The use of a mixed form finite element method with $H(\text{div})$ elements, such as Raviart-Thomas or BDM elements, would provide a smooth electric field across cell boundaries. The mixed form method reduces the continuity of the electric field from C^1 to C^0 . We expect that C^0 electric fields will decrease the noise in our particle representation over time. PETSc currently includes support for the $H(\text{div})$ conforming finite elements Brezzi-Douglas-Marini (BDM) and Raviart-Thomas (RT) on simplicial grids. However, RT elements are

currently the only element type supported on tensor cells. Verification of the mixed form implementation in PETSc has been previously shown on a variety of other physical systems, such as droplet formation [104, 105, 106]. While we have not observed any major negative impacts from using H^1 finite elements in the linear regime and within the timescale chosen, when extending the times past $t = 30$, these smooth fields may be beneficial.

Nonlinear Landau damping is a more complex in that non-damping phenomenon, such as plasma echo, are present. Vitally, the linearization of the Vlasov equation, used as the fundamental approximation in the study of linear Landau damping, does not guarantee that the asymptotic behavior of the linear Vlasov equation is an approximation of the asymptotic behavior of the nonlinear Vlasov equation [100]. There are reasons to doubt that the study of the linearized equations gives any hint on the long-time behavior of the nonlinear equations. Therefore if an algorithm is desired that can accurately capture the long-time behavior of a plasma, the nonlinear case of Landau damping must also be considered. We will also replicate our tests in parallel, allowing us to increase the number of particles per cell by several orders of magnitude, reducing the largest source of error in the code.

We may also consider adding a resampling, or remapping, phase to the PETSc-PIC algorithm. Resampling ever few steps will reduce the statistical noise in the system over long-time runs. We observe in Figure 4.8 that around $t = 25$, considerable noise is introduced to the particles. We have thus far been able to reduce noise by reasonably scaling up the number of particles per cell chosen in the simulation, but with upscaling comes computational costs. Resampling is an efficient method for reducing noise without considerable computational cost.

Nonlinear Landau damping is a more complex in that non-damping phenomenon, such as plasma echo, are present. Vitally, the linearization of the Vlasov equation, used as the fundamental approximation in the study of linear Landau damping, does not guarantee that the asymptotic behavior of the linear Vlasov equation is an approximation of the asymptotic behavior of the nonlinear Vlasov equation [100]. There are reasons to doubt that the study of

the linearized equations gives any hint on the long-time behavior of the nonlinear equations. Therefore if an algorithm is desired that can accurately capture the long-time behavior of a plasma, the nonlinear case of Landau damping must also be considered.

Chapter 5

Collisional Kinetics

Introduced in Section 3.1, the full Vlasov-Maxwell-Landau system of equations, for the mass distribution function $f_s(\mathbf{z})$ of species s in phase space $\mathbf{z} = (\mathbf{x}, \mathbf{v}) \in \Omega \times \mathbb{R}^d$, is given by,

$$\begin{aligned}\frac{df_s}{dt} &= \frac{\partial f_s}{\partial t} + \frac{\partial \mathbf{x}}{\partial t} \cdot \nabla_{\mathbf{x}} f_s + \frac{\partial \mathbf{v}}{\partial t} \cdot \nabla_{\mathbf{v}} f_s \\ &= \frac{\partial f_s}{\partial t} + \mathbf{v} \cdot \nabla_{\mathbf{x}} f_s + \frac{q_s}{m_s} (\mathbf{E} + \mathbf{v} \times \mathbf{B}) \cdot \nabla_{\mathbf{v}} f_s = \sum_{\bar{s}} C_{s\bar{s}}(f_s, f_{\bar{s}}),\end{aligned}\tag{5.1}$$

where the right-hand side is the collision operator, which in our implementation is given by the Landau operator (3.12).

The Landau collision operator admits multiple conservative discretizations which makes it ideal for this implementation in which our goal is to provide a variety of accurate and diverse solvers for the full Vlasov-Poisson-Landau system. The admission of conservative discretizations, however, comes at the cost of computational complexity, which is $\mathcal{O}(N^2)$. There are two primary discretizations currently available in PETSc. The first, a finite element based discretization, was developed by Hirvijoki and Adams in 2017 [65, 4]. In this implementation, the magnetic field is ignored and the evolution of the distribution function is restricted to the velocity space $\mathbf{v} \in \mathbb{R}^D$, thus giving the continuous form of the Landau

equation,

$$\frac{\partial f_s}{\partial t} + \frac{e_s}{m_s} \mathbf{E} \cdot \nabla f_s = \sum_{\bar{s}} C_{s\bar{s}}(f_s, f_{\bar{s}}), \quad (5.2)$$

$$= \sum_{\bar{s}} \nu_{s\bar{s}} \frac{m_0}{m_s} \nabla \cdot \int_{\bar{\Omega}} d\bar{\mathbf{v}} \mathbb{Q}(\mathbf{v}, \bar{\mathbf{v}}) \cdot \left(\frac{m_0}{m_s} \bar{f}_{\bar{s}} \nabla f_s - \frac{m_0}{m_{\bar{s}}} f_s \bar{\nabla} \bar{f}_{\bar{s}} \right), \quad (5.3)$$

where the subscript s indicates the particle of interest and \bar{s} the colliding particle. Furthermore, m_0 is a reference mass, ϵ_0 is the vacuum permittivity and \mathbb{Q} is the Landau tensor,

$$\mathbb{Q}(\mathbf{v}, \bar{\mathbf{v}}) = \frac{1}{|\mathbf{v} - \bar{\mathbf{v}}|^3} \left(|\mathbf{v} - \bar{\mathbf{v}}|^2 \mathbb{I} - (\mathbf{v} - \bar{\mathbf{v}})(\mathbf{v} - \bar{\mathbf{v}})^T \right). \quad (5.4)$$

Overbar terms are evaluated on the $\bar{\mathbf{v}}$ grid that covers the domain $\bar{\Omega}$ of particle (or species) \bar{s} . As stated above, this implementation relies on finite elements. Therefore, a weak form for the Landau equation is derived, as discussed in more detail in [65, 4, 2]. Given a test function $\psi(\mathbf{x})$, the weak form is

$$\int_{\Omega} d\mathbf{v} \psi \cdot \left(\frac{\partial f_s}{\partial t} + \left(0, \frac{q_s}{m_s} \mathbf{E} \right) \cdot \nabla f_s \right) = \sum_{\bar{s}} (\psi, f_s)_{\mathbf{D}, s\bar{s}} + \sum_{\bar{s}} (\psi, f_s)_{\mathbf{K}, s\bar{s}}, \quad (5.5)$$

where $(\cdot, \cdot)_{\Omega}$ is a standard L^2 inner product in Ω . The inner products of the advective and diffusive parts of the Landau equation can be written as,

$$(\psi, \phi)_{\mathbf{D}, s\bar{s}} = - \int_{\Omega} d\mathbf{v} \nabla \psi \cdot \nu \frac{m_0}{m_s} \frac{m_0}{m_{\bar{s}}} \mathbf{D}(f_{\bar{s}}, \mathbf{v}) \cdot \nabla \phi, \quad (5.6)$$

$$(\psi, \phi)_{\mathbf{K}, s\bar{s}} = \int_{\Omega} d\mathbf{v} \nabla \psi \cdot \nu \frac{m_0}{m_s} \frac{m_0}{m_{\bar{s}}} \mathbf{K}(f_{\bar{s}}, \mathbf{v}) \phi, \quad (5.7)$$

with \mathbf{D} and \mathbf{K} being a tensor and vector, respectively, defined by,

$$\mathbf{D}(f, \mathbf{v}) = \int_{\bar{\Omega}} d\bar{\mathbf{v}} \mathbb{Q}(\mathbf{v}, \bar{\mathbf{v}}) f(\bar{\mathbf{v}}), \quad (5.8)$$

$$\mathbf{K}(f, \mathbf{v}) = \int_{\bar{\Omega}} d\bar{\mathbf{v}} \mathbb{Q}(\mathbf{v}, \bar{\mathbf{v}}) \cdot \bar{\nabla} f(\bar{\mathbf{v}}). \quad (5.9)$$

A finite-dimensional vector space is chosen to discretize the weak form in the velocity space. In [65], quadratic Q2-Lagrange finite elements are selected for the spatial discretization. Implicit time integrators, such as Crank-Nicolson, are then used to discretize the equation in time. This collision operator has been tested extensively and has proved to be highly scalable and conservative for a variety of plasma tests. The finite element discretization also allows for the use of adaptive mesh refinement methods, which are well implemented in PETSc.

The finite element Landau operator is, however, not without flaws. The primary draw-back to this grid-based method is the significant setup costs and $\mathcal{O}(N^2)$ overall complexity. Moreover, similar to the evaluation of the electrostatic field solve discussed in Section 4.1.3, a grid-based collision operator requires a conservative projection of the fields to and from the finite element basis along with a projection of altered weights from the finite element basis back to the particle basis at each time step. Scalable algorithms have been developed for these operations but their computational cost still remains non-negligible, particularly in higher dimensional systems. Lastly, while the finite element operator is conservative of moments, it does not guarantee entropy monotonicity. For these reasons, in certain applications, a particle based Landau operator may be preferable.

A second discretization of the Landau collision operator based in the particle basis was therefore implemented by Pusztay in 2023 [116]. Construction of a deterministic particle based discretization for the homogenous Landau collision operator was demonstrated for the first time by Carrillo et al. in [25]. In this paper, Carrillo et al. presented the idea that Coulomb collisions can be interpreted as compressible flow driven by an entropy functional enabling a natural introduction of marker particles with arbitrary weights. The implementation, however, uses an Eulerian time discretization which fails to preserve the conservation of energy achieved by the continuum representation. These methods were furthered in [67, 149] which presents the necessary discrete-time scheme for simulating the Landau operator while conserving the moments. A reformulation of the Landau operator is provided by Hirvijoki

in [66], in which the Landau equation is given by the expression,

$$C_{s\bar{s}}(f_s, f_{\bar{s}}) := -\frac{\nu_{s\bar{s}}}{m_s} \nabla_{\mathbf{v}} \cdot \left[\int \delta(\mathbf{x} - \bar{\mathbf{x}}) f_s(\mathbf{z}) f_{\bar{s}}(\bar{\mathbf{z}}) \mathbb{Q}(\mathbf{v} - \bar{\mathbf{v}}) \cdot \boldsymbol{\Gamma}_{s\bar{s}}(\mathcal{S}, \mathbf{z}, \bar{\mathbf{z}}) d\bar{\mathbf{z}} \right], \quad (5.10)$$

where the vector $\boldsymbol{\Gamma}_{s\bar{s}}(\mathcal{A}, \mathbf{z}, \bar{\mathbf{z}})$ is defined by the equation,

$$\boldsymbol{\Gamma}_{s\bar{s}}(\mathcal{A}, \mathbf{z}, \bar{\mathbf{z}}) := \frac{1}{m_s} \frac{\partial}{\partial \mathbf{v}} \frac{\delta \mathcal{A}}{\delta f_s}(\mathbf{z}) - \frac{1}{m_{\bar{s}}} \frac{\partial}{\partial \bar{\mathbf{v}}} \frac{\delta \mathcal{A}}{\delta f_{\bar{s}}}(\bar{\mathbf{z}}), \quad (5.11)$$

for some arbitrary functional \mathcal{A} . The matrix $\mathbb{Q}(\boldsymbol{\xi})$ is the redefined collision kernel,

$$\mathbb{Q}(\boldsymbol{\xi}) = \frac{1}{|\boldsymbol{\xi}|} \left(\mathbb{I} - \frac{\boldsymbol{\xi}\boldsymbol{\xi}}{|\boldsymbol{\xi}|^2} \right). \quad (5.12)$$

We will consider the formulation of a discretized metric bracket in more depth for this particle-basis Landau operator as the work conducted in this project is based upon this formulation.

The first step in deriving the metric bracket formulation for the Landau equation is to obtain the weak form of the collision operator (5.10). We do this by multiplying (5.10) by a test function, which in this case is an arbitrary species-dependent function $G_s(\mathbf{z})$, and integrating over the infinite velocity space,

$$\begin{aligned} \int G_s(\mathbf{z}) \frac{\partial f_s}{\partial t} d\mathbf{z} &= \sum_s \int G_s(\mathbf{z}) C_{s\bar{s}}(f_s, f_{\bar{s}}) d\mathbf{z}, \\ &= \sum_{s,\bar{s}} \iint \frac{1}{m_s} \frac{\partial G_s(\mathbf{z})}{\partial \mathbf{v}} \cdot \mathbb{W}_{s\bar{s}}(\mathbf{z}, \bar{\mathbf{z}}) \cdot \boldsymbol{\Gamma}_{s\bar{s}}(\mathcal{S}, \mathbf{z}, \bar{\mathbf{z}}) d\bar{\mathbf{z}} d\mathbf{z}, \\ &= \sum_{s,\bar{s}} \frac{1}{2} \iint \left(\frac{1}{m_s} \frac{\partial G_s(\mathbf{z})}{\partial \mathbf{v}} - \frac{1}{m_{\bar{s}}} \frac{\partial G_{\bar{s}}(\bar{\mathbf{z}})}{\partial \bar{\mathbf{v}}} \right) \cdot \mathbb{W}_{s\bar{s}}(\mathbf{z}, \bar{\mathbf{z}}) \cdot \boldsymbol{\Gamma}_{s\bar{s}}(\mathcal{S}, \mathbf{z}, \bar{\mathbf{z}}) d\bar{\mathbf{z}} d\mathbf{z}, \end{aligned} \quad (5.13)$$

where the positive semidefinite matrix $\mathbb{W}_{s\bar{s}}(\mathbf{z}, \bar{\mathbf{z}})$ is defined as,

$$\mathbb{W}_{s\bar{s}}(\mathbf{z}, \bar{\mathbf{z}}) = \nu_{s\bar{s}} \delta(\mathbf{x} - \bar{\mathbf{x}}) f_s(\mathbf{z}) f_{\bar{s}}(\bar{\mathbf{z}}) \mathbb{Q}(\mathbf{v} - \bar{\mathbf{v}}). \quad (5.14)$$

If we consider the function G_s expressed as a functional derivative,

$$\mathcal{G} = \sum_s \int G_s f_s d\mathbf{z}, \quad (5.15)$$

and likewise,

$$\frac{\delta \mathcal{G}}{\delta f_s} = G_s, \quad (5.16)$$

we can identify $\boldsymbol{\Gamma}_{s\bar{s}}(\mathcal{G}, \mathbf{z}, \bar{\mathbf{z}})$ in (5.13). With the introduction of the entropy functional,

$$\mathcal{S} = - \sum_s \int f_s \ln f_s d\mathbf{z}, \quad (5.17)$$

the collisional evolution of any arbitrary functional \mathcal{A} can then be expressed as a positive semidefinite metric bracket,

$$\left. \frac{d\mathcal{A}}{dt} \right|_{\text{coll}} = (\mathcal{A}, \mathcal{S}), \quad (5.18)$$

$$= \sum_{s,\bar{s}} \frac{1}{2} \iint \boldsymbol{\Gamma}_{s\bar{s}}(\mathcal{A}, \mathbf{z}, \bar{\mathbf{z}}) \cdot \mathbb{W}_{s\bar{s}}(\mathbf{z}, \bar{\mathbf{z}}) \cdot \boldsymbol{\Gamma}_{s\bar{s}}(\mathcal{S}, \mathbf{z}, \bar{\mathbf{z}}) d\bar{\mathbf{z}} d\mathbf{z}, \quad (5.19)$$

where $\boldsymbol{\Gamma}_{s\bar{s}}(\mathcal{A}, \mathbf{z}, \bar{\mathbf{z}})$ is still defined by (5.11).

The conservation of the moments is shown by way of the moment functionals being invariants of this bracket, i.e.,

$$(\mathcal{M}_s, \mathcal{A}) = 0, \quad (5.20)$$

$$(\mathcal{P}_s, \mathcal{A}) = 0, \quad (5.21)$$

$$(\mathcal{K}_s, \mathcal{A}) = 0. \quad (5.22)$$

It may also be shown that the positive semidefinite nature of the matrix $\mathbb{W}_{s\bar{s}}$ matrix ensures the preservation of entropy monotonicity,

$$(\mathcal{S}, \mathcal{S}) \geq 0. \quad (5.23)$$

We now consider the discretization of this bracket. For simplicity of indices, we reduce the system to one species going forward and thus drop the s and \bar{s} subscripts.

5.1 Marker-Particle Discretization of the Collisional Bracket

Consider a common discretization for particle-in-cell simulations, primarily based on the idea that the distributional density f can be represented by the expression,

$$f_h(\mathbf{z}, t) d\mathbf{z} = \sum_p w_p \delta(\mathbf{x} - \mathbf{x}_p(t)) \delta(\mathbf{v} - \mathbf{v}_p(t)) d\mathbf{z}, \quad (5.24)$$

where $\mathbf{x}_p(t)$, $\mathbf{v}_p(t)$ and w_p , are the position, velocity and weight of particle p , respectively. Before proceeding with the discretization, we must take a step back and address a few questions that will be raised in the next steps.

First, if the entropy functional is evaluated with respect to the marker-particle distribution (5.24), the entropy functional (5.17) is not well defined. To resolve this, Carrillo et al. [25], propose introducing a regularized entropy functional where the distribution function (5.24) is first convolved with a radial basis function, also called a *mollifier*, ψ_ϵ . In practice, the mollifier can be any radial basis function, but we will focus on the case where the mollifier is a Gaussian,

$$\psi_\epsilon(v) = \frac{1}{(2\pi\epsilon)^{\frac{d}{2}}} \exp\left(-\frac{|v|^2}{2\epsilon}\right). \quad (5.25)$$

for any $\epsilon > 0$. The entropy functional (5.17) can then be replaced by the regularized entropy functional,

$$\mathcal{S}[f] \Rightarrow \mathcal{S}[\psi_\epsilon * f] = \mathcal{S}[f], \quad (5.26)$$

which enables the evaluation of the entropy functional even for the delta distribution (5.24). Furthermore, the delta functions appearing in the entropy functional (5.26) can simply be

replaced with radial basis functions ψ_ϵ , centered at the particle locations, so that,

$$\mathcal{S} [\psi_\epsilon * f_h] = - \int \sum_p w_p \psi_\epsilon (\mathbf{z} - \mathbf{z}_p) \ln \left(\sum_{p'} w_{p'} \psi_\epsilon (\mathbf{z} - \mathbf{z}_{p'}) \right) d\mathbf{z}. \quad (5.27)$$

Secondly, we encounter a question when considering the evaluation of the functional derivatives (5.16) with respect to the distribution (5.24). Two similar solutions to this issue have been proposed, first by Carrillo et al. in [25] and second by Hirvijoki in [66]. We will refer to the solution in [66] and give a brief summary here. If the distribution function f is restricted to the specific parametrized form f_h given in (5.24), then the perturbation of f also becomes parametrized and takes the form,

$$\delta f_h d\mathbf{z} = - \sum_p w_p \left(\delta (\mathbf{v} - \mathbf{v}_p) \nabla \delta (\mathbf{x} - \mathbf{x}_p) \cdot \delta \mathbf{x}_p + \delta (\mathbf{x} - \mathbf{x}_p) \frac{\partial}{\partial \mathbf{v}} \delta (\mathbf{v} - \mathbf{v}_p) \cdot \delta \mathbf{v}_p \right) d\mathbf{z}. \quad (5.28)$$

Accordingly, the variation of the functional $\mathcal{A}[f]$, being restricted to distributions of type (5.24), then become,

$$\delta \mathcal{A} [\delta f_h] = \sum_p w_p \left(\nabla \frac{\delta \mathcal{A}}{\delta f} \Big|_{\mathbf{z}_p} \cdot \delta \mathbf{x}_p + \frac{\partial}{\partial \mathbf{v}} \frac{\delta \mathcal{A}}{\delta f} \Big|_{\mathbf{z}_p} \cdot \delta \mathbf{v}_p \right). \quad (5.29)$$

When evaluated with respect to the distribution f_h , the functional \mathcal{A} , however, becomes an ordinary function of the degrees of freedom. That is to say,

$$A [f_h] = A(\mathbf{X}, \mathbf{V}; W), \quad (5.30)$$

where $\mathbf{X} = \{\mathbf{x}_p\}_p$ and $\mathbf{V} = \{\mathbf{v}_p\}_p$ denote the collection of marker-particle degrees of freedom, and $W = \{w_p\}_p$ denotes the collection of particles fixed weights. Therefore, $\delta \mathcal{A} [\delta f_h]$ must

be equal to the variation of the function $A(\mathbf{X}, \mathbf{V}; W)$. This leads to the identities,

$$\nabla \frac{\delta \mathcal{A}}{\delta f} \Big|_{\mathbf{z}_p} = \frac{1}{w_p} \frac{\partial A(\mathbf{X}, \mathbf{V}; W)}{\partial \mathbf{x}_p}, \quad (5.31)$$

$$\frac{\partial}{\partial \mathbf{v}} \frac{\delta \mathcal{A}}{\delta f} \Big|_{\mathbf{z}_p} = \frac{1}{w_p} \frac{\partial A(\mathbf{X}, \mathbf{V}; W)}{\partial \mathbf{v}_p}. \quad (5.32)$$

Lastly, to move forward with the discretization, we must consider the local nature of the Landau operator. In practice, the two marker particles will never be at exactly the same location. Therefore, an approximation for the delta function in (5.14), which is responsible for the localization of the Landau operator, must be used. For this implementation, we will divide the spatial configuration domain into disjoint, so-called collision cells that share a boundary if adjacent. In this case, the delta function is replaced by an indicator function $\mathbf{1}(p, \bar{p})$ which is one if the particles p and \bar{p} are within the collision cell and zero otherwise. With these issues now resolved, we can proceed with the discretization of the metric bracket (5.19).

Substituting the discrete distribution function (5.24), the transformation of the functional derivatives (5.32), and the replacement of the localizing delta function with an indicator function into the single-species version of the metric bracket (5.19), we get a finite-dimensional formulation of the metric bracket,

$$(A, B)_h = \frac{1}{2} \sum_{p, \bar{p}} \boldsymbol{\Gamma}(A, p, \bar{p}) \cdot \mathbb{W}(p, \bar{p}) \cdot \boldsymbol{\Gamma}(B, p, \bar{p}), \quad (5.33)$$

where the vector $\boldsymbol{\Gamma}(A, p, \bar{p})$ is now given by,

$$\boldsymbol{\Gamma}(A, p, \bar{p}) = \frac{1}{mw_p} \frac{\partial A}{\partial \mathbf{v}_p} - \frac{1}{mw_{\bar{p}}} \frac{\partial A}{\partial \mathbf{v}_{\bar{p}}}, \quad (5.34)$$

and the matrix $\mathbb{W}(p, \bar{p})$ is,

$$\mathbb{W}(p, \bar{p}) = \nu w_p w_{\bar{p}} \mathbf{1}(p, \bar{p}) \mathbb{Q}(\mathbf{v}_p - \mathbf{v}_{\bar{p}}). \quad (5.35)$$

The finite-dimensional bracket (5.33) then leads to a new collisional evolution expression for functions that depend on particle degrees of freedom,

$$\frac{dA}{dt} \Big|_{\text{coll}} = (A, S_\epsilon) \quad (5.36)$$

where $S_\epsilon(\mathbf{X}, \mathbf{V}; W) = \mathcal{S}[f_h]$ is the regularized entropy functional, defined in (5.27).

To obtain the equation of motion for the particle coordinates, we consider the specific case where the arbitrary functional A is replaced by the function \mathbf{v}_p by means of a Riesz map. The general form of this equation of motion is,

$$\frac{d\mathbf{v}_p}{dt} \Big|_{\text{coll}} = \frac{1}{2} \sum_{\bar{p}} \boldsymbol{\Gamma}(\mathbf{v}_p, p, \bar{p}) \cdot \mathbb{W}(p, \bar{p}) \cdot \boldsymbol{\Gamma}(S_\epsilon, p, \bar{p}), \quad (5.37)$$

where we again have replaced the delta function with an indicator function $\mathbf{1}(p, \bar{p})$ inside the matrix \mathbb{W} . By exploiting the antisymmetry of the vector $\boldsymbol{\Gamma}(\mathbf{v}_p, p, \bar{p})$ and the symmetry of the matrix $\mathbb{Q}(\mathbf{v}_p - \mathbf{v}_{\bar{p}})$ this expression can be simplified to,

$$\frac{d\mathbf{v}_p}{dt} \Big|_{\text{coll}} = \frac{\nu}{m} \sum_{\bar{p}} w_{\bar{p}} \mathbf{1}(p, \bar{p}) \mathbb{Q}(\mathbf{v}_p - \mathbf{v}_{\bar{p}}) \cdot \boldsymbol{\Gamma}(S_\epsilon, p, \bar{p}). \quad (5.38)$$

To confirm the accuracy of this discretization, we can recheck the mass, momentum and kinetic energy conservation laws as well as the entropy monotonicity. We first redefine the moment functionals as ordinary functions of the degrees of freedom,

$$M(\mathbf{X}, \mathbf{V}; W) = \mathcal{M}[f_h] = \sum_p w_p m, \quad (5.39)$$

$$P(\mathbf{X}, \mathbf{V}; W) = \mathcal{P}[f_h] = \sum_p w_p m \mathbf{v}_p, \quad (5.40)$$

$$K(\mathbf{X}, \mathbf{V}; W) = \mathcal{K}[f_h] = \sum_p w_p \frac{m}{2} |\mathbf{v}_p|^2. \quad (5.41)$$

It is straightforward then to show that the moments are invariants of the discrete bracket. For mass and momentum this follows from the fact that $\boldsymbol{\Gamma}(M, p, \bar{p}) = 0$ and $\boldsymbol{\Gamma}(P, p, \bar{p}) = \mathbb{I} - \mathbb{I} = 0$.

For kinetic energy, $\boldsymbol{\Gamma}(K, p, \bar{p}) = \mathbf{v}_p - \mathbf{v}_{\bar{p}}$ and, consequently, $\boldsymbol{\Gamma}(K, p, \bar{p}) \cdot \mathbb{W}(p, \bar{p}) = 0$. Thus the collisional evolution equations of the moments are,

$$\frac{dM}{dt} \Big|_{\text{coll}} = (M, S_\epsilon)_h = 0, \quad (5.42)$$

$$\frac{dP}{dt} \Big|_{\text{coll}} = (P, S_\epsilon)_h = 0, \quad (5.43)$$

$$\frac{dK}{dt} \Big|_{\text{coll}} = (K, S_\epsilon)_h = 0, \quad (5.44)$$

and the moments are conserved. Lastly, we observe that the matrix $\mathbb{W}(p, \bar{p})$ in this finite-dimensional bracket is positive semidefinite, therefore, the regularized entropy is trivially dissipated,

$$\frac{dS_\epsilon}{dt} \Big|_{\text{coll}} = (S_\epsilon, S_\epsilon)_h \geq 0. \quad (5.45)$$

5.2 Discrete Gradient Dependent Integrators

We have thus far shown that it is possible to obtain a finite-dimensional metric bracket for the Landau equation, however, the final challenge of this derivation is to construct a time-discrete form of this bracket that preserves the fundamental thermodynamic principles. An implementation of one conservative time discretization exists already in PETSc [115]. This time discretization does not, however, guarantee entropy monotonicity. To guarantee the conservation of the moments as well as the preservation of entropy monotonicity¹, we follow from the work presented by Hirvijoki in [66] and introduce a new time stepping integrator that utilizes discrete gradients, defined by Gonzalez in [52],

$$\frac{\mathbf{v}_p^{n+1} - \mathbf{v}_p^n}{\Delta t} = \frac{\nu}{m} \sum_{\bar{p}} w_{\bar{p}} \mathbf{1}(p, \bar{p}) \mathbb{Q}(\overline{\boldsymbol{\Gamma}_n^{n+1}(K, p, \bar{p})}) \cdot \overline{\boldsymbol{\Gamma}_n^{n+1}(S_\epsilon, p, \bar{p})}. \quad (5.46)$$

¹The dissipation of the free energy functional is also preserved by way of the entropy monotonicity due to the relationship $\mathcal{F} = \mathcal{H} - \mathcal{S}$.

where the vector $\overline{\Gamma_n^{n+1}(A, p, \bar{p})}$ is defined by,

$$\overline{\Gamma_n^{n+1}(A, p, \bar{p})} = \frac{1}{mw_p} \bar{\nabla} A(v_p^n, v_p^{n+1}) - \frac{1}{mw_{\bar{p}}} \bar{\nabla} A(v_{\bar{p}}^n, v_{\bar{p}}^{n+1}). \quad (5.47)$$

Unlike the average discrete gradient used in [66], we define $\bar{\nabla} A(v_p^n, v_p^{n+1})$ using Gonzalez's midpoint discrete gradient (4.63). Further generalizing, the collisional discrete-time evolution of an arbitrary functional is given by,

$$\frac{A^{n+1} - A^n}{\Delta t} = \frac{1}{2} \sum_{p, \bar{p}} \overline{\Gamma_n^{n+1}(A, p, \bar{p})} \cdot \overline{W_n^{n+1}(p, \bar{p})} \cdot \overline{\Gamma_n^{n+1}(S_\epsilon, p, \bar{p})}. \quad (5.48)$$

where the matrix $\overline{W_n^{n+1}(p, \bar{p})}$ is defined as,

$$\overline{W_n^{n+1}(p, \bar{p})} = \nu w_p w_{\bar{p}} \mathbf{1}(p, \bar{p}) \mathbb{Q}\left(\overline{\Gamma_n^{n+1}(K, p, \bar{p})}\right). \quad (5.49)$$

Returning to the discussion of discrete gradient integrators, we now recognize that in (5.48), and specifically in (5.46), we have derived an alternate form to the numerical scheme (4.56), proposed by Gonzalez. For arbitrary functionals we see that the discrete Hamiltonian vector field becomes,

$$\mathbf{X}_H = \frac{1}{2} \sum_{p, \bar{p}} \overline{\Gamma_n^{n+1}(A, p, \bar{p})} \cdot \overline{W_n^{n+1}(p, \bar{p})} \cdot \overline{\Gamma_n^{n+1}(S_\epsilon, p, \bar{p})}, \quad (5.50)$$

and for the evolution of the vector field,

$$\mathbf{X}_H = \frac{\nu}{m} \sum_{\bar{p}} w_{\bar{p}} \mathbf{1}(p, \bar{p}) \mathbb{Q}\left(\overline{\Gamma_n^{n+1}(K, p, \bar{p})}\right) \cdot \overline{\Gamma_n^{n+1}(S_\epsilon, p, \bar{p})}. \quad (5.51)$$

This so-called discrete gradient dependent integrator (DGDI) is not itself a discrete gradient, as it does not pass the directionality test, but relies on the properties of the discrete gradient (4.63) (see (4.2.1)) to be conservative. We must therefore prove that our new integrator is consistent with the finite-dimensional form of the collisional evolution (5.18). We must also

show that the integrator conserves the moments and preserves the monotonicity of entropy.

Theorem 5.2.1. *The collisional discrete-time evolution equation, given by (5.48), is a consistent approximation to the finite-dimensional bracket (5.33).*

$$\frac{dA}{dt} \Big|_{coll} \approx \frac{A^{n+1} - A^n}{\Delta t}, \quad (5.52)$$

in the limit where $\|u_p^{n+1} - u_p^n\| \rightarrow 0$ and $\|v_{\bar{p}}^{n+1} - v_{\bar{p}}^n\| \rightarrow 0$.

Proof. First we consider what happens to (5.47) as $\|v_p^{n+1} - v_p^n\|$ and $\|v_{\bar{p}}^{n+1} - v_{\bar{p}}^n\|$ approach zero,

$$\begin{aligned} \overline{\Gamma_n^{n+1}(A, p, \bar{p})} &= \frac{1}{mw_p} \overline{\nabla A(v_p^n, v_p^{n+1})} - \frac{1}{mw_{\bar{p}}} \overline{\nabla A(v_{\bar{p}}^n, v_{\bar{p}}^{n+1})} \\ &= \frac{1}{mw_p} \left(\nabla A(v_p^{n+1/2}) + \mathcal{O}(\|v_p^{n+1} - v_p^n\|) \right) \\ &\quad - \frac{1}{mw_{\bar{p}}} \left(\nabla A(v_{\bar{p}}^{n+1/2}) + \mathcal{O}(\|v_{\bar{p}}^{n+1} - v_{\bar{p}}^n\|) \right) \\ &\approx \frac{1}{mw_p} \nabla A(v_p^{n+1/2}) - \frac{1}{mw_{\bar{p}}} \nabla A(v_{\bar{p}}^{n+1/2}) \\ &\approx \Gamma(A, p, \bar{p}) \end{aligned}$$

Plugging this into relation into (5.48) gives,

$$\begin{aligned} \frac{A^{n+1} - A^n}{\Delta t} &= \frac{1}{2} \sum_{p, \bar{p}} \overline{\Gamma_n^{n+1}(A, p, \bar{p})} \cdot \overline{W_n^{n+1}(p, \bar{p})} \cdot \overline{\Gamma_n^{n+1}(S_\epsilon, p, \bar{p})} \\ &\approx \frac{1}{2} \sum_{p, \bar{p}} \Gamma_n^{n+1}(A, p, \bar{p}) \cdot \left[\nu w_p w_{\bar{p}} \mathbf{1}(p, \bar{p}) \mathbb{Q}(\Gamma_n^{n+1}(K, p, \bar{p})) \right] \cdot \Gamma_n^{n+1}(S_\epsilon, p, \bar{p}) \end{aligned}$$

Realizing $\Gamma(K, p, \bar{p}) = v_p - v_{\bar{p}}$, we can therefore show that,

$$\begin{aligned}\frac{A^{n+1} - A^n}{\Delta t} &\approx \frac{1}{2} \sum_{p, \bar{p}} \Gamma_n^{n+1}(A, p, \bar{p}) \cdot [\nu w_p w_{\bar{p}} \mathbf{1}(p, \bar{p}) \mathbb{Q}(v_p - v_{\bar{p}})] \cdot \Gamma_n^{n+1}(S_\epsilon, p, \bar{p}) \\ &\approx \frac{1}{2} \sum_{p, \bar{p}} \Gamma_n^{n+1}(A, p, \bar{p}) \cdot \mathbb{W}(p, \bar{p}) \cdot \Gamma_n^{n+1}(S_\epsilon, p, \bar{p}) \\ &\approx \left. \frac{dA}{dt} \right|_{\text{coll}}\end{aligned}$$

□

Proving conservation of the moments follows similarly from the conservation proofs in previous sections. Conservation of mass is trivial to show for the DGDI as the discrete mass function is a function of the weights alone. Thus $\Gamma_n^{n+1}(M^n, p, \bar{p}) = 0$ and the conservation of (5.48) follows. Conservation of momentum again follows from the condition $\overline{\Gamma_n^{n+1}(P, p, \bar{p})} = \mathbb{I} - \mathbb{I} = 0$,

$$\begin{aligned}\frac{P^{n+1} - P^n}{\Delta t} &= \frac{\nu}{2} \sum_{p, \bar{p}} \overline{\Gamma_n^{n+1}(P, p, \bar{p})} \cdot \overline{W_n^{n+1}(p, \bar{p})} \cdot \overline{\Gamma_n^{n+1}(S_\epsilon, p, \bar{p})}, \\ &= \frac{\nu}{2} \sum_{p, \bar{p}} (\mathbb{I} - \mathbb{I}) \cdot \overline{W_n^{n+1}(p, \bar{p})} \cdot \overline{\Gamma_n^{n+1}(S_\epsilon, p, \bar{p})}, \\ &= 0.\end{aligned}$$

Conservation of kinetic energy is dependent on the fact that $\mathbb{Q}(\xi)$ has the null eigenvector ξ , thus $\overline{\Gamma_n^{n+1}(K^n, p, \bar{p})} \cdot \mathbb{Q}(\overline{\Gamma_n^{n+1}(K^n, p, \bar{p})}) = 0$. Therefore, the discrete-time collisional rate-of-change of kinetic energy follows as such,

$$\begin{aligned}\frac{K^{n+1} - K^n}{\Delta t} &= \frac{\nu}{2} \sum_{p, \bar{p}} \overline{\Gamma_n^{n+1}(K, p, \bar{p})} \cdot \overline{W_n^{n+1}(p, \bar{p})} \cdot \overline{\Gamma_n^{n+1}(S_\epsilon, p, \bar{p})} \\ &= \frac{\nu}{2} \sum_{p, \bar{p}} \overline{\Gamma_n^{n+1}(K, p, \bar{p})} \cdot [\nu w_p w_{\bar{p}} \mathbf{1}(p, \bar{p}) \mathbb{Q}(\overline{\Gamma_n^{n+1}(K, p, \bar{p})})] \cdot \overline{\Gamma_n^{n+1}(S_\epsilon, p, \bar{p})} \\ &= 0\end{aligned}$$

Lastly, we consider the preservation of entropy monotonicity. As a result of $\overline{W_n^{n+1}(p, \bar{p})}$ being a positive semidefinite matrix, the discrete-time entropy behaves as desired,

$$\begin{aligned} \frac{S_\epsilon^{n+1} - S_\epsilon^n}{\Delta t} &= \frac{\nu}{2} \sum_{p, \bar{p}} \overline{\Gamma_n^{n+1}(S_\epsilon, p, \bar{p})} \cdot \overline{W_n^{n+1}(p, \bar{p})} \cdot \overline{\Gamma_n^{n+1}(S_\epsilon, p, \bar{p})}, \\ &\geq 0. \end{aligned}$$

5.2.1 Singularities in DGDI

It is worthwhile to note one potential limiting case for the DGDI formulation. As plasma systems approach equilibrium the change in the positions and velocities decreases considerably. As a consequence of this decrease, the term $\|x_{n+1} - x_n\|^2$ in the denominator of (4.63) moves towards zero creating a potential singularity within the discrete gradients formulation. In collisionless systems, where the time step size is not too small, this singularity is unlikely to appear, however, to achieve acceptable conservation in collisional examples, the time step size must be decreased by multiple orders of magnitude. The inclusion of DGDI must, therefore, be carefully considered before being implemented in some plasma simulations. For this work, we conduct tests of the DGDI framework with and without the expansion term to verify the functionality of the implementation while trying to avoid any problems related to this singularity. For now, tests of the DGDI framework with the expansion term included are conducted on plasma systems not near equilibrium to mitigate any potential problems. Future work will be conducted to more clearly define regimes in which the various forms of DGDI are most accurate.

5.3 Numerical Examples

A variety of collisional tests have been previously run in [115, 116] verifying the accuracy and conservative quality of the particle-basis Landau operator. Thus we do not seek to prove the effectiveness or accuracy of the framework within PETSc that much of the DGDI relies

on. Discrete gradient (DG) methods (collisionless and collisional) utilize an alternate set of setup functions to the standard `RHSFunction` function required by other timesteppers in PETSc. The user must instead provide three functions to the `TS` module to construct essential pieces of (4.62): (1) the skew-symmetric matrix S , the function² F , and the gradient of F , ∇F . From these functions, the `TS` module can construct the DG (4.63) and the appropriate timestepper. The `TS` module also requires the user to defined a Jacobian for the equations of motion, as discussed in Section 2.3. However, for the Landau system this is quite a complex exercise. The Jacobian is therefore approximated using either a standard finite difference, or matrix free approach. The matrix free approach drastically improves runtime, particularly due preallocations of the Jacobian matrix being nonoptimal at the time of writing.

If the Gonzalez expansion terms are ignored within (4.63), the DG formulation reverts back to implicit midpoint. Thus, to prove the correctness of the DG formulation in the collisional case, as we have already done this for the collisionless case, we must first confirm that the DG framework does in fact match the existing particle-basis implementation when the Gonzalez expansion term is ignored. Then we must show that the addition of the expansion term leads to monotonic entropy, along with the already conserved moments. For this purpose, we consider a simple electron-positron equilibration test case in velocity space.

In these tests, particle management is once again performed by the PETSc object `DMSwarm`, which initializes the distribution functions, and provides data structure for particle management as well as algorithms for particle migration between MPI ranks in parallel runs. The initialization of the particles in velocity space is done in a similar fashion to the collisionless tests in Section 4.4. Particles are regularly laid out in a grid with their spacing determined by $\Delta x = 2L/N$ where the velocity domain is $v = [-L, L]^d$, and N is the number of particles along a line in one of the dimensions. Particle weights are then assigned using

²In simple Hamiltonian systems like the harmonic oscillator discussed in Section 4.4, the function F is often represented by the Hamiltonian H .

the integral,

$$w_p = \int_{v_p-h/2}^{v_p+h/2} \frac{1}{\sqrt{2\pi v_{th}^2}} e^{-v^2/v_{th}^2} dv, \quad (5.53)$$

where v_{th} is the same thermal velocity defined in Section 4.4. To construct multiple thermal populations, the thermal velocity is varied for given particles of each population and/or species. The system is constructed in a single spatial cell, omitting the spatial component entirely from the simulation. The entropy and entropy gradient terms, S and ∇S , respectively, are calculated using a sixth order Gauss-Hermite quadrature.

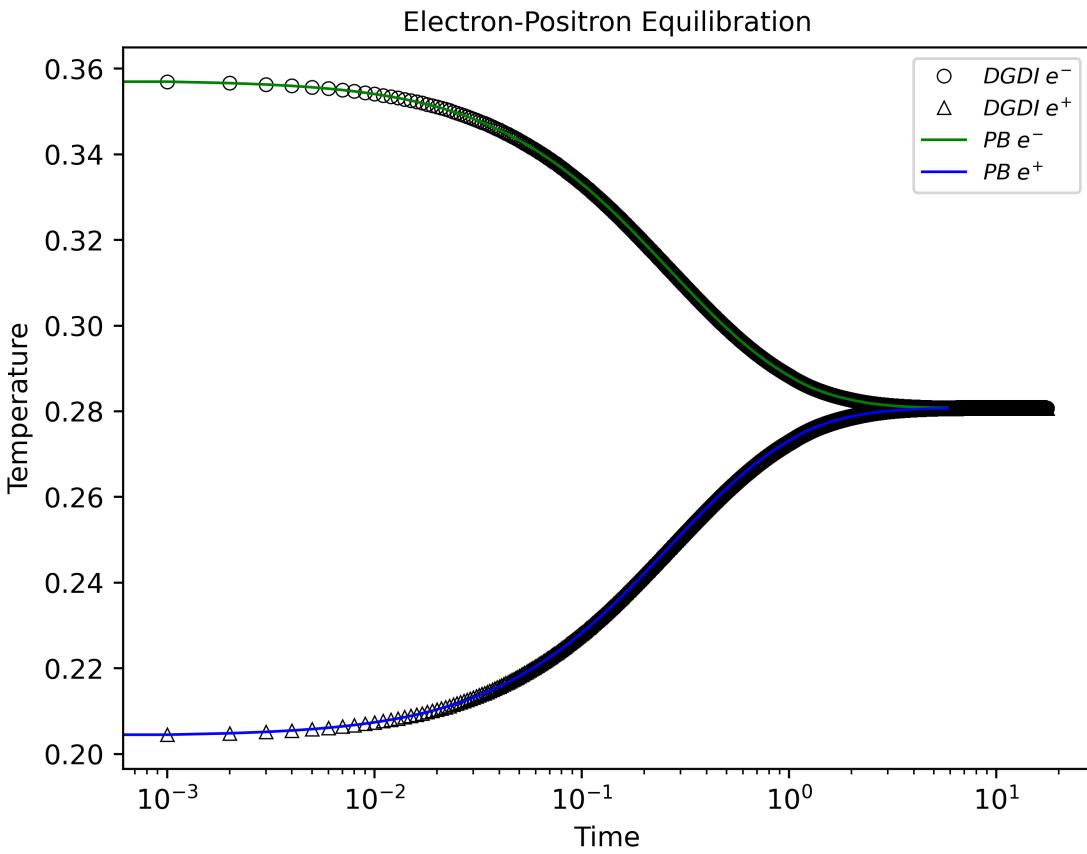


Figure 5.1: Electron-positron temperature equilibration using the two collision operators: the DGDI framework without the Gonzalez expansion term and the existing particle basis operator with a theta method timestepper. The DGDI method without the Gonzalez term is equivalent to using an implicit midpoint integrator. In PETSc the implicit midpoint method is performed by the `TSTHETA` framework. These two methods are therefore mathematically equivalent.

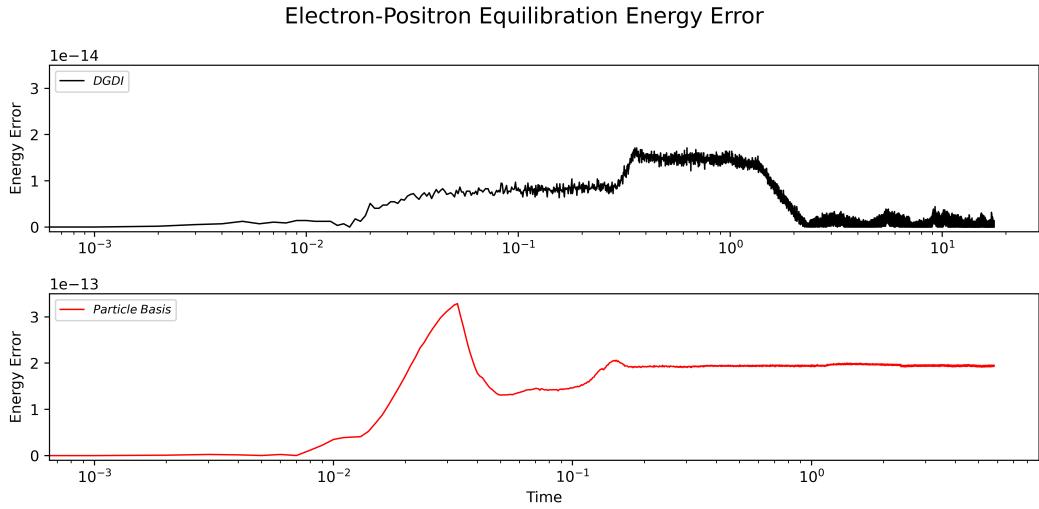


Figure 5.2: Comparison of relative kinetic energy error for electron-positron temperature equilibration using (top) the DGDI framework without the Gonzalez expansion term and (bottom) the existing particle basis operator with a theta method timestepper.

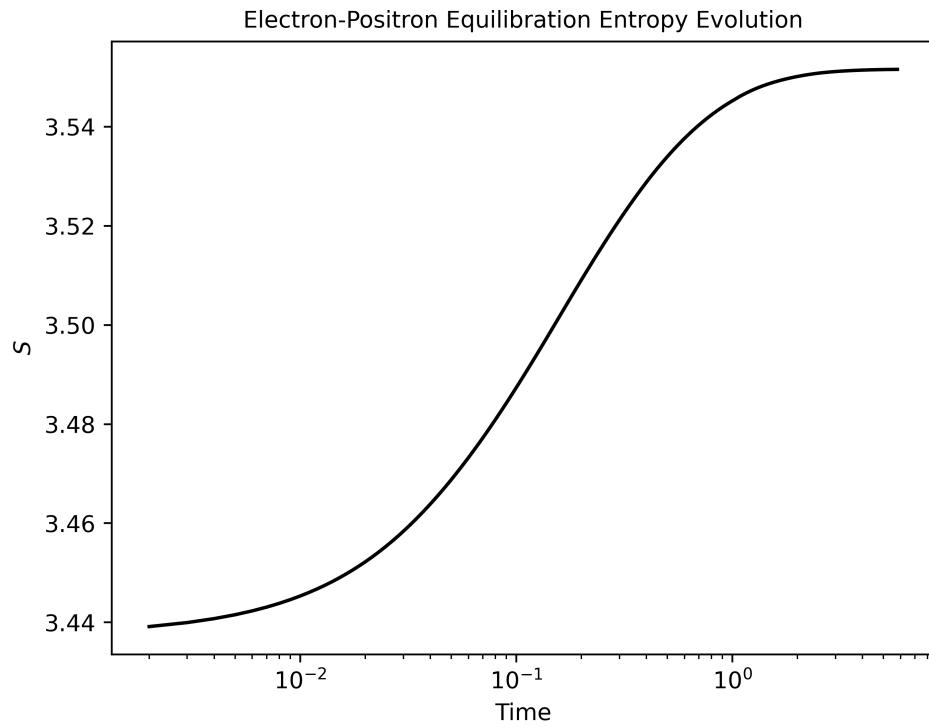


Figure 5.3: The evolution of the entropy in electron-positron temperature equilibration using the DGDI framework without the Gonzalez expansion term.

A comparison of the temperature evolution of the two particle populations from the equilibration tests using the DGDI framework without the Gonzalez expansion term and the existing particle basis framework, which relies on the `TSTHETA` timestepper in PETSc, is shown in Figure 5.1. These tests were conducted with 800 particles per population and a time step size of 0.001. The relative error in the kinetic energy in this system using both integration methods is then shown in Figure 5.2. Ignoring the Gonzalez term in the DGDI framework, these two approaches are mathematically equivalent. The existing particle basis algorithm has been previously verified against analytical equations, described in the Naval Research Lab Plasma Formulary [70]. Therefore, as the temperature evolution of the populations in both methods is equivalent, we have shown that the new DGDI framework correctly reduces to a conservative implicit midpoint scheme when ignoring the Gonzalez term. We note, however, that the error in the DGDI test is decreased tenfold from the error when using `TSTHETA`. Lastly, we observe in Figure 5.3 that, even without Gonzalez's expansion term in the DGDI formulation, we achieve entropy monotonicity (or near entropy monotonicity) in this test. As expected, the change in entropy between steps approaches zero as the system equilibrates. Similar entropy results have been found using these particle based collision operators in simple tests but in more complex collisional systems we expect the advantage of using DGDI to be more apparent.

Having confirmed the accuracy of the DGDI framework without the Gonzalez term, the electron-positron equilibration test was then run again with the full DGDI framework. The inclusion of the Gonzalez term in the DGDI equations drastically increases the number of nonlinear solver iterations required to achieve convergence at each time step. Therefore, to complete these tests the number of particles per population was reduced to 36 and the time step size was set to 0.01. We also use a nonlinear Guass-Seidel (NGS) method for the `SNES` solver, as opposed to the typical Newton method used in the other tests. The NGS method has proven to be the most effective way to consistently reach convergence in these full DGDI tests. Despite these changes, the nonlinear solve fails to converge around $t=0.6$ which may

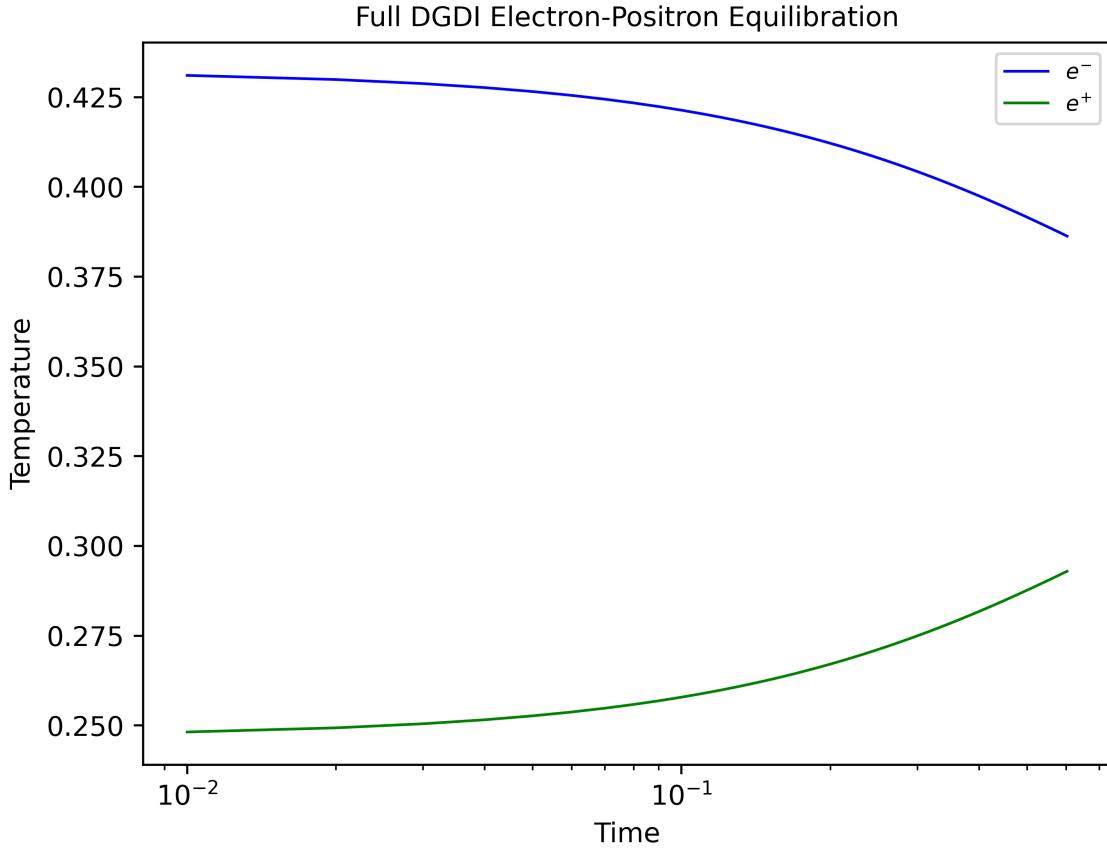


Figure 5.4: Electron-positron equilibration test using the full DGDI framework.

be the result of an inaccurate Jacobian approximation or singularities encountered in the DGDI equations. Further studies will need to be conducted to provide a more robust DGDI framework. Nonetheless, we may examine the results of the test up to the point of failure to confirm the accuracy of the DGDI methodology.

The temperature evolution of the electrons and positrons is shown in Figure 5.4 and the relative kinetic energy and entropy are shown in Figure 5.5. We note that the temperature calculations of the two populations differs from previous tests because of the low number of particles per population, which reduces the statistical accuracy of the particle representation. The results from the early stages of the test, however, show the two populations moving towards the expected thermalization as well as moment conservation and monotonic entropy.

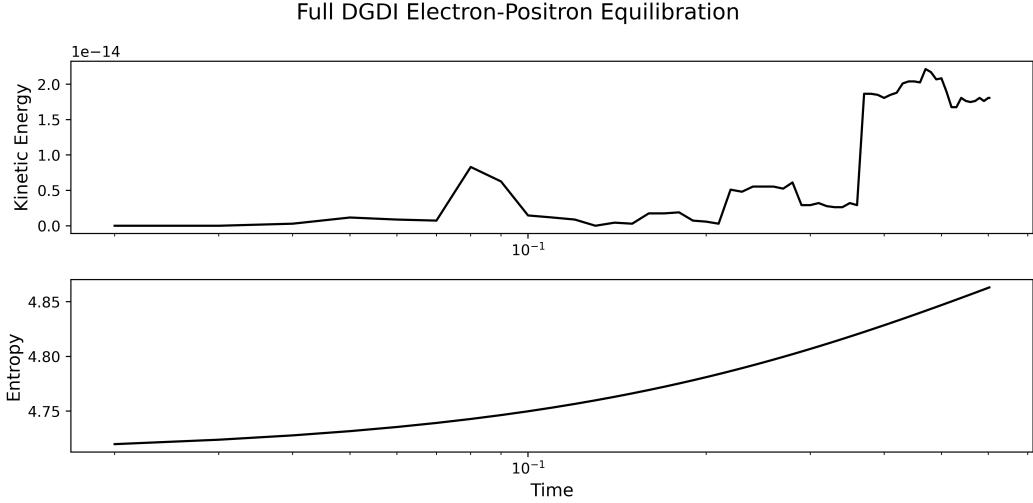


Figure 5.5: Relative kinetic energy error and entropy evolution in electron-positron equilibration test using the full DGDI framework.

Once again, the variation in kinetic energy is an order of magnitude smaller than that of the theta method test. Considering the increased size of the time step used in this test and the dependence of the moment conservation on the time step size, this result is promising. This confirms that, under the right conditions, the DGDI framework can model plasma systems while preserving fundamental thermodynamic principles.

5.4 Conclusions

We have considered an alternate form of the particle-basis Landau operator that includes discrete gradients. This new integrator, the Discrete Gradient Dependent Integrator (DGDI), adds entropy monotonicity to the already moment conservative collision operator. We have shown mathematically that the collisional evolution is a consistent approximation to the finite dimensional bracket and that the DGDI will preserve the thermodynamic principles we are considering. The framework of this new operator was then tested without the expansion terms that make it entropy monotonic using an electron-positron equilibration test. Results from this test were compared to the existing particle basis collision operator in PETSc,

which relies on the mathematically equivalent `TSTHETA` timestepper. The thermalization of the two populations matched between the two methods, however, despite the mathematical equivalence, the new DGDI framework was more conservative than the existing methodology. The DGDI framework was also shown to be monontonic, or very near monontonic, in entropy. This is somewhat unexpected as this test does not include the extra term necessary for mathematical preservation of entropy monotonicity. We expect future tests on more complex plasma systems to show a more clear advantage when including the expansion term.

The full DGDI framework was tested on the early stages of the same electron-positron equilibration test. As expected, the particle populations accurately move towards an equilibrium state while conserving the moments and entropy monontonicity until the nonlinear solver fails to converge. It is difficult to identify the specific reasons for the solver failure in this test. Therefore, in future work, we intend to conduct a more in depth study of the specific regimes in which the DGDI formulation is impacted by these singularities and alternate formulations of the DGDI that may not encounter these problems. We may also consider the average discrete gradient (4.71) instead of the midpoint discrete gradient used in this work.

We also plan to rewrite the PETSc implementation of DGDI, and the original particle-basis operator, for GPGPU optimization. This work, which will leverage kokkos on the backend, is already underway. A fully optimized implementation of the collision operators will provide the ability to conduct full-scale real world studies of plasma systems. A variety of other collisional tests, such as Spitzer resistivity, can be considered in the future as part of these optimization goals to further prove the accuracy of this new method.

Chapter 6

Conclusions

The goal of this dissertation was to develop a long-time accurate structure-preserving particle-in-cell Vlasov-Poisson-Landau model using the Portable, Extensible Toolkit for Scientific Computation (PETSc). Instead of writing a monolithic application code framework for this problem, we have approached each piece of this framework as a separate tool which can be used on its own or as a part of a complete Vlasov-Poisson-Landau test. This approach provides flexibility and extensibility in our new toolkit. As we have discussed throughout this work, a toolkit for particle-in-cell methods was previously constructed within PETSc, however, these tools are not accurate enough over long time scales to resolve the fine-scale plasma structures essential in understanding the overall dynamics within a system. We have introduced a number of alterations and improvements to this toolkit that have improved the overall accuracy of the simulations.

At the core of these improvements is the introduction of discrete gradient integrators for both the Poisson and metric bracket. In simple Hamiltonian systems, discrete gradient integrators preserve the first integral of the Hamiltonian. In the collisionless plasma systems of interest, driven by the Poisson bracket, this property provides conservation of the moments. We have shown that this holds true by implementing discrete gradient integrators in the PETSc TS module and testing it against the existing basic symplectic and Runge-Kutta

integrators on a simple harmonic oscillator system. In these simple Hamiltonian tests, the discrete gradient methods are superior as they solve the exact system rather than a modified system which leads to a more exact conservation of the Hamiltonian than basic symplectic methods. We have also made significant improvements in the overall accuracy of the PETSc-PIC algorithm, which has been extensively tested on the classic plasma system, Landau damping. Runge-Kutta integrators are non-conservative and therefore perform poorly in the Landau damping test. Basic symplectic integrators, however, are accurate and conservative enough to obtain the correct dynamics in the Landau damping test. With a sufficiently high order basic symplectic method, the advantage gained by using discrete gradient integrators is marginal. Additionally, the basic symplectic methods are explicit while the new discrete gradient integrators are implicit. Therefore, in the collisionless plasma regime we may often opt to use the basic symplectic integrators rather than the discrete gradient integrator.

The implementation of discrete gradient integrators is most impactful in the evolution of the metric bracket. Discrete gradient dependent integrators (DGDI) were developed and used to solve the Landau collision operator. These integrators are fully conservative and monotonic in entropy, the latter being a property that previous collision operators have failed to achieve mathematically. The new DGDI framework was tested on multi-species equilibration problems and compared to the previous particle basis collisional approach. We have confirmed the preservation of the expected thermodynamic qualities with the DGDI framework and shown that the new framework often outperforms the previous algorithm even without the inclusion of the monotonic expansion term.

We have therefore developed, and tested, a set of tools for solving the Vlasov-Poisson-Landau system of equations that are long-time accurate, scalable and preserve fundamental thermodynamic principles necessary to model real world plasmas. The tools for evolving the Poisson and metric bracket still remain implemented on separate test systems. For that reason, the next major step in this research is constructing a simulation of the full metriplectic system and testing this complete system on a variety of plasma systems. A

complete Vlasov-Poisson-Landau simulation using the tools described in this work would be a significant step in the field of computational plasma physics. We may use this toolkit to deepen our understanding of stellar atmospheres, thermonuclear fusion devices and many other plasma systems throughout the Universe.

References

- [1] Abhyankar, S. et al. *PETSc/TS: A Modern Scalable DAE/ODE Solver Library*. Preprint ANL/MCS-P5061-0114. ANL, 2014.
- [2] Adams, M. F. et al. “Landau collision operator in the CUDA programming model applied to thermal quench plasmas”. In: *2022 IEEE International Parallel and Distributed Processing Symposium (IPDPS)*. 2022 IEEE International Parallel and Distributed Processing Symposium (IPDPS). IEEE, 2022, pp. 115–123. ISBN: 978-1-6654-8106-9. DOI: 10.1109/IPDPS53621.2022.00020. URL: <https://ieeexplore.ieee.org/document/9820672/>.
- [3] Adams, M. F., Samtaney, R., and Brandt, A. “Toward textbook multigrid efficiency for fully implicit resistive magnetohydrodynamics”. In: *Journal of Computational Physics* 229.18 (2010), pp. 6208–6219. ISSN: 00219991. DOI: 10.1016/j.jcp.2010.04.024. URL: <https://linkinghub.elsevier.com/retrieve/pii/S0021999110002007>.
- [4] Adams, M. F. et al. “Landau Collision Integral Solver with Adaptive Mesh Refinement on Emerging Architectures”. In: *SIAM Journal on Scientific Computing* 39.6 (2017), pp. C452–C465. ISSN: 1064-8275. DOI: 10.1137/17M1118828. URL: <https://pubs.siam.org/doi/10.1137/17M1118828>.
- [5] Alfvén, H. *Cosmic Radiation as an Intra-galactic Phenomenon*. Arkiv för matematik, astronomi och fysik. Almqvist & Wiksell Boktryckeri, 1937. URL: <https://books.google.com/books?id=TPD3GwAACAAJ>.
- [6] Applebaum, E. et al. “Ultrafaint Dwarfs in a Milky Way Context: Introducing the Mint Condition DC Justice League Simulations”. In: *The Astrophysical Journal* 906.2 (2021), p. 96. ISSN: 1538-4357. DOI: 10.3847/1538-4357/abcafa. URL: <https://iopscience.iop.org/article/10.3847/1538-4357/abcafa>.
- [7] Arad, I. and Lynden-Bell, D. “Inconsistency in theories of violent relaxation”. In: *Monthly Notices of the Royal Astronomical Society* 361.2 (Aug. 2005), pp. 385–395. DOI: 10.1111/j.1365-2966.2005.09133.x. arXiv: astro-ph/0409728 [astro-ph].

- [8] Balay, S. et al. “Efficient Management of Parallelism in Object-Oriented Numerical Software Libraries”. In: *Modern Software Tools for Scientific Computing*. Boston, MA: Birkhäuser Boston, 1997, pp. 163–202. DOI: 10.1007/978-1-4612-1986-6_8. URL: http://link.springer.com/10.1007/978-1-4612-1986-6_8.
- [9] Bardenhagen, S. and Kober, E. “The Generalized Interpolation Material Point Method”. In: *CMES - Computer Modeling in Engineering and Sciences* 5 (June 2004).
- [10] Barnes, W. T., Cargill, P. J., and Bradshaw, S. J. “Inference of Heating Properties From “Hot” Non-Flaring Plasmas in Active Region Cores. I. Single Nanoflares”. In: *The Astrophysical Journal* 829.1 (2016), p. 31. DOI: 10.3847/0004-637x/829/1/31. URL: <https://doi.org/10.3847/0004-637x/829/1/31>.
- [11] Beckermann, B., Golub, G. H., and Labahn, G. “On the numerical condition of a generalized Hankel eigenvalue problem”. In: *Numerische Mathematik* 106.1 (2007), pp. 41–68. ISSN: 0945-3245. DOI: 10.1007/s00211-006-0054-x. URL: <https://doi.org/10.1007/s00211-006-0054-x>.
- [12] Bhatnagar, P. L., Gross, E. P., and Krook, M. “A Model for Collision Processes in Gases. I. Small Amplitude Processes in Charged and Neutral One-Component Systems”. In: *Phys. Rev.* 94 (3 May 1954), pp. 511–525. DOI: 10.1103/PhysRev.94.511. URL: <https://link.aps.org/doi/10.1103/PhysRev.94.511>.
- [13] Binney, J. and Tremaine, S. *Galactic Dynamics: Second Edition*. 2008. ISBN: 9781400828722.
- [14] Birdsall, C. K. and Langdon, A. B. *Plasma Physics via Computer Simulation*. 1991.
- [15] Birkeland, K. *The Norwegian Aurora Polaris Expedition 1902-1903*. The Norwegian Aurora Polaris Expedition 1902-1903 v. 1, pt. 1. H. Aschehoug & Company, 1913. URL: <https://books.google.com/books?id=UHZmtAEACAAJ>.
- [16] Bogacki, P. and Shampine, L. “A 3(2) pair of Runge - Kutta formulas”. In: *Applied Mathematics Letters* 2.4 (1989), pp. 321–325. ISSN: 0893-9659. DOI: [https://doi.org/10.1016/0893-9659\(89\)90079-7](https://doi.org/10.1016/0893-9659(89)90079-7). URL: <https://www.sciencedirect.com/science/article/pii/0893965989900797>.
- [17] Bohm, D. and Gross, E. P. “Theory of Plasma Oscillations. A. Origin of Medium-Like Behavior”. In: *Phys. Rev.* 75.12 (1949), pp. 1851–1864. DOI: 10.1103/PhysRev.75.1851. URL: <https://link.aps.org/doi/10.1103/PhysRev.75.1851>.
- [18] Boltzmann, L. “Weitere Studien über das Warmegleichgewicht unter Gasmolekülen”. In: vol. 66. Translation: Further studies on the thermal equilibrium of gas molecules, in

Kinetic Theory 2, 88–174, Ed. S.G. Brush, Pergamon, Oxford (1966). 1872, pp. 275–350.

- [19] Brackbill, J. and Ruppel, H. “FLIP: A method for adaptively zoned, particle-in-cell calculations of fluid flows in two dimensions”. In: *Journal of Computational Physics* 65.2 (1986), pp. 314–343. ISSN: 0021-9991. DOI: [https://doi.org/10.1016/0021-9991\(86\)90211-1](https://doi.org/10.1016/0021-9991(86)90211-1). URL: <https://www.sciencedirect.com/science/article/pii/0021999186902111>.
- [20] Bradshaw, S. J. and Cargill, P. J. “The Influence of Numerical Resolution on Coronal Density in Hydrodynamic Models of Impulsive Heating”. In: *The Astrophysical Journal* 770.1 (2013), p. 12. DOI: 10.1088/0004-637X/770/1/12. URL: <https://dx.doi.org/10.1088/0004-637X/770/1/12>.
- [21] Bratley, P. and Fox, B. L. “Algorithm 659: Implementing Sobol’s Quasirandom Sequence Generator”. In: *ACM Trans. Math. Softw.* 14.1 (Mar. 1988), pp. 88–100. ISSN: 0098-3500. DOI: 10.1145/42288.214372. URL: <https://doi.org/10.1145/42288.214372>.
- [22] Byers, J. A. “Noise Suppression Techniques in Macroparticle Models of Collisionless Plasmas”. In: *Proceedings of the Fourth Conference of Numerical Simulation of Plasmas*. Ed. by NTIS. 1970.
- [23] Canosa, J. “Numerical solution of Landau’s dispersion equation”. In: *Journal of Computational Physics* 13.1 (1973), pp. 158–160. ISSN: 0021-9991. DOI: [https://doi.org/10.1016/0021-9991\(73\)90131-9](https://doi.org/10.1016/0021-9991(73)90131-9). URL: <https://www.sciencedirect.com/science/article/pii/0021999173901319>.
- [24] Cargill, P. J. and Klimchuk, J. A. “A Nanoflare Explanation for the Heating of Coronal Loops Observed by Yohkoh”. In: *The Astrophysical Journal* 478.2 (1997), pp. 799–806. DOI: 10.1086/303816.
- [25] Carrillo, J. A. et al. “A particle method for the homogeneous Landau equation”. In: *Journal of Computational Physics: X* 7 (2020). ISSN: 25900552. DOI: 10.1016/j.jcpx.2020.100066.
- [26] Chandrasekhar, S. “Dynamical Friction. I. General Considerations: the Coefficient of Dynamical Friction.” In: *Astrophysical Journal* 97 (1943), p. 255. DOI: 10.1086/144517.
- [27] Chandrasekhar, S. “Dynamical Friction. II. The Rate of Escape of Stars from Clusters and the Evidence for the Operation of Dynamical Friction.” In: *Astrophysical Journal* 97 (1943), p. 263. DOI: 10.1086/144518.
- [28] Chandrasekhar, S. *Principles of Stellar Dynamics*. University of Chicago Press, 1942.

- [29] Chen, F. F. *Introduction to Plasma Physics and Controlled Fusion*. 2nd ed. Vol. 1. Plenum Press, New York, 1984.
- [30] Chen, G., Chacón, L., and Barnes, D. C. “An energy- and charge-conserving, implicit, electrostatic particle-in-cell algorithm”. In: *Journal of Computational Physics* 230.18 (2011), pp. 7018–7036. ISSN: 0021-9991. DOI: <https://doi.org/10.1016/j.jcp.2011.05.031>. URL: <https://www.sciencedirect.com/science/article/pii/S0021999111003421>.
- [31] Cho, J. “Non-locality of Hydrodynamic and Magnetohydrodynamic Turbulence”. In: *The Astrophysical Journal* 725.2 (2010), pp. 1786–1791. DOI: 10.1088/0004-637X/725/2/1786. arXiv: 1005.0878 [astro-ph.EP].
- [32] Ciarlet, P. G. *Numerical Analysis of the Finite Element Method*. Les Presses de l’Université de Montréal, 1976.
- [33] Cummings, E. A. et al. “Fluorescence Measurements Of Expanding Strongly Coupled Neutral Plasmas”. In: *Phys. Rev. Lett.* 95 (23 Nov. 2005), p. 235001. DOI: 10.1103/PhysRevLett.95.235001. URL: <https://link.aps.org/doi/10.1103/PhysRevLett.95.235001>.
- [34] D’Angelo, N. “Heating of the Solar Corona”. In: *Astrophysical Journal* 154 (1968), p. 401. DOI: 10.1086/149768.
- [35] Daughton, W. et al. “Role of electron physics in the development of turbulent magnetic reconnection in collisionless plasmas”. In: *Nature Physics* 7.7 (2011), pp. 539–542. DOI: 10.1038/nphys1965.
- [36] Denavit, J. and Walsh, J. M. *Comments on Plasma Physics and Controlled Fusion*. New York.
- [37] Dobrowolny, M. and Torricelli-Ciamponi, G. “Alfven wave dissipation in the solar wind”. In: *Astronomy and Astrophysics* 142.2 (1985), pp. 404–410.
- [38] Et. al., S. B. *PETSc/TAO Users Manual*. Argonne National Laboratory.
- [39] Evans, M. W. and Harlow, F. H. “The Particle-In-Cell Method For Hydrodynamic Calculations”. In: (June 1957). URL: <https://www.osti.gov/biblio/4326770>.
- [40] Evstatiev, E. G. and Shadwick, B. A. “Variational formulation of particle algorithms for kinetic plasma simulations”. In: *Journal of Computational Physics* 245 (2013). ISSN: 10902716. DOI: 10.1016/j.jcp.2013.03.006.
- [41] Faghihi, D. et al. “Moment preserving constrained resampling with applications to particle-in-cell methods”. In: *Journal of Computational Physics* 409 (2020), p. 109317.

ISSN: 0021-9991. DOI: <https://doi.org/10.1016/j.jcp.2020.109317>. URL: <https://www.sciencedirect.com/science/article/pii/S0021999120300917>.

- [42] Farrell, P. E. et al. “PCPATCH: software for the topological construction of multigrid relaxation methods”. In: *ACM Transaction on Mathematical Software* 47.3 (2021), pp. 1–22. eprint: <http://arxiv.org/abs/1912.08516>.
- [43] Finn, D. S. et al. *A Numerical Study of Landau Damping with PETSc-PIC*. 2023. arXiv: 2303.12620 [physics.plasm-ph].
- [44] Forest, É. “Geometric integration for particle accelerators”. In: *Journal of Physics A: Mathematical and General* 39.19 (Apr. 2006), p. 5321. DOI: 10.1088/0305-4470/39/19/S03. URL: <https://dx.doi.org/10.1088/0305-4470/39/19/S03>.
- [45] Forest, E. and Ruth, R. D. “Fourth-order symplectic integration”. In: *Physica D: Nonlinear Phenomena* 43.1 (1990), pp. 105–117. ISSN: 0167-2789. DOI: [https://doi.org/10.1016/0167-2789\(90\)90019-L](https://doi.org/10.1016/0167-2789(90)90019-L). URL: <https://www.sciencedirect.com/science/article/pii/016727899090019L>.
- [46] Fouvry, J.-B. et al. “Resonant thickening of self-gravitating discs: imposed or self-induced orbital diffusion in the tightly wound limit”. In: *Monthly Notices of the Royal Astronomical Society* 471.3 (Nov. 2017), pp. 2642–2673. DOI: 10.1093/mnras/stx1625. arXiv: 1605.03379 [astro-ph.GA].
- [47] Friedberg, J. P., Morse, R. L., and Nielson, C. W. “No Title”. In: *Proceedings of the Third Conference on Numerical Simulation of Plasmas*. 1969.
- [48] Gilmore, G. and Reid, N. “New light on faint stars - III. Galactic structure towards the South Pole and the Galactic thick disc.” In: *Monthly Notices of the Royal Astronomical Society* 202 (Mar. 1983), pp. 1025–1047. DOI: 10.1093/mnras/202.4.1025.
- [49] Gilmore, G. et al. “The Gaia-ESO Public Spectroscopic Survey”. In: *The Messenger* 147 (Mar. 2012), pp. 25–31.
- [50] Goedbloed, J. P. *Principles of magnetohydrodynamics : with applications to laboratory and astrophysical plasmas*. Cambridge, UK ; Cambridge University Press, 2004. ISBN: 0521623472.
- [51] Gold, T. “The dissipation of magnetic energy in the upper atmosphere of the sun”. In: *Stellar and Solar Magnetic Fields*. Ed. by R. Lust. Vol. 22. 1965, p. 390.
- [52] Gonzalez, O. “Time Integration and Discrete Hamiltonian Systems”. In: *Journal of Nonlinear Science* 6.5 (1996). ISSN: 09388974. DOI: 10.1007/BF02440162.

- [53] Groselj, D. et al. *Ab Initio Turbulent Comptonization in Magnetized Coronae of Accreting Black Holes*. 2023. DOI: 10.48550/ARXIV.2301.11327. URL: <https://arxiv.org/abs/2301.11327>.
- [54] Haack, J. et al. “Numerical schemes for a multi-species BGK model with velocity-dependent collision frequency”. In: *Journal of Computational Physics* 473 (2023), p. 111729. ISSN: 0021-9991. DOI: <https://doi.org/10.1016/j.jcp.2022.111729>. URL: <https://www.sciencedirect.com/science/article/pii/S0021999122007926>.
- [55] Hairer, E., Wanner, G., and Lubich, C. “Symmetric Integration and Reversibility”. In: *Geometric Numerical Integration: Structure-Preserving Algorithms for Ordinary Differential Equations*. Berlin, Heidelberg: Springer Berlin Heidelberg, 2006. ISBN: 978-3-540-30666-5. DOI: 10.1007/3-540-30666-8_5.
- [56] Hairer, E., Wanner, G., and Lubich, C. “Symplectic Integration of Hamiltonian Systems”. In: *Geometric Numerical Integration: Structure-Preserving Algorithms for Ordinary Differential Equations*. Berlin, Heidelberg: Springer Berlin Heidelberg, 2006, pp. 179–236. ISBN: 978-3-540-30666-5. DOI: 10.1007/3-540-30666-8_6. URL: https://doi.org/10.1007/3-540-30666-8_6.
- [57] Hamaguchi, S. and Farouki, R. T. “Thermodynamics of strongly-coupled Yukawa systems near the one-component-plasma limit. I. Derivation of the excess energy”. In: *The Journal of Chemical Physics* 101.11 (1994), pp. 9876–9884. DOI: 10.1063/1.467954. eprint: <https://doi.org/10.1063/1.467954>. URL: <https://doi.org/10.1063/1.467954>.
- [58] Hammersley, J. M. and Handscomb, D. C. *Monte Carlo Methods*. Dordrecht: Springer Netherlands, 1964. ISBN: 978-94-009-5821-0. DOI: 10.1007/978-94-009-5819-7. URL: <http://link.springer.com/10.1007/978-94-009-5819-7>.
- [59] Hapla, V. et al. “Fully Parallel Mesh I/O Using PETSc DMplex with an Application to Waveform Modeling”. In: *SIAM Journal on Scientific Computing* 43.2 (2021), pp. C127–C153. ISSN: 1095-7197. DOI: 10.1137/20m1332748. URL: <http://dx.doi.org/10.1137/20M1332748>.
- [60] Harten, A., Lax, P. D., and Leer, B. van. “On Upstream Differencing and Godunov-Type Schemes for Hyperbolic Conservation Laws”. In: *Upwind and High-Resolution Schemes*. Ed. by M. Y. Hussaini, B. van Leer, and J. Van Rosendale. Berlin, Heidelberg: Springer Berlin Heidelberg, 1997, pp. 53–79. ISBN: 978-3-642-60543-7. DOI: 10.1007/978-3-642-60543-7_4. URL: https://doi.org/10.1007/978-3-642-60543-7_4.
- [61] He, Y. et al. “Hamiltonian time integrators for Vlasov-Maxwell equations”. In: *Physics of Plasmas* 22.12 (2015). ISSN: 10897674. DOI: 10.1063/1.4938034.

- [62] Heaviside, O. *Electromagnetic Theory Volume 1*. v. 1. The Electrician Printing and Publishing Co, London, 1893.
- [63] Heaviside, O. “Telegraphy”. In: *Encyclopaedia Britannica*. Edinburgh, London: Adam, and Charles Black, 1902, p. 214.
- [64] Hesse, M. et al. “The Diffusion Region in Collisionless Magnetic Reconnection”. In: *Space Science Reviews* 160.1-4 (2011), pp. 3–23. DOI: 10.1007/s11214-010-9740-1.
- [65] Hirvijoki, E. and Adams, M. F. “Conservative Discretization of the Landau Collision Integral”. In: *Physics of Plasmas* 24.3 (2017). ISSN: 10897674. DOI: 10.1063/1.4979122.
- [66] Hirvijoki, E. “Structure-preserving marker-particle discretizations of Coulomb collisions for particle-in-cell codes”. In: *Plasma Physics and Controlled Fusion* 63.4 (2021). ISSN: 13616587. DOI: 10.1088/1361-6587/abe884.
- [67] Hirvijoki, E. and Burby, J. W. “Collisional gyrokinetics teases the existence of metriplectic reduction”. In: *Physics of Plasmas* 27.8 (2020). ISSN: 10897674. DOI: 10.1063/5.0011297.
- [68] Hirvijoki, E., Kraus, M., and Burby, J. W. *Metriplectic Particle-in-Cell Integrators for the Landau Collision Operator*. 2018. arXiv: 1802.05263 [physics.comp-ph].
- [69] Howes, G. G. et al. “Kinetic Simulations of Magnetized Turbulence in Astrophysical Plasmas”. In: *Phys. Rev. Lett.* 100.6 (2008), p. 65004. DOI: 10.1103/PhysRevLett.100.065004. URL: <https://link.aps.org/doi/10.1103/PhysRevLett.100.065004>.
- [70] Huba, J. “NRL Plasma Formulary”. In: (May 1995).
- [71] Ichimaru, S. “Strongly coupled plasmas: high-density classical plasmas and degenerate electron liquids”. In: *Rev. Mod. Phys.* 54 (4 Oct. 1982), pp. 1017–1059. DOI: 10.1103/RevModPhys.54.1017. URL: <https://link.aps.org/doi/10.1103/RevModPhys.54.1017>.
- [72] Innes, D. E. et al. “IRIS Si IV Line Profiles: An Indication for the Plasmoid Instability during Small-scale Magnetic Reconnection on the Sun”. In: *The Astrophysical Journal* 813.2 (2015), p. 86. DOI: 10.1088/0004-637X/813/2/86. eprint: 1509.08837 (astro-ph.SR).
- [73] Isaac, T. and Knepley, M. G. “Support for Non-conformal Meshes in PETSc’s DMFlex Interface”. In: *ACM Transaction on Mathematical Software* (2017). In review. eprint: 1508.02470.

- [74] Joe, S. and Kuo, F. Y. “Constructing Sobol Sequences with Better Two-Dimensional Projections”. In: *SIAM Journal on Scientific Computing* 30.5 (2008), pp. 2635–2654. DOI: 10.1137/070709359. eprint: <https://doi.org/10.1137/070709359>. URL: <https://doi.org/10.1137/070709359>.
- [75] Killian, T. et al. “Ultracold neutral plasmas”. In: *Physics Reports* 449.4 (2007), pp. 77–130. ISSN: 0370-1573. DOI: <https://doi.org/10.1016/j.physrep.2007.04.007>. URL: <https://www.sciencedirect.com/science/article/pii/S0370157307001937>.
- [76] Kirby, R. C. “Algorithm 839: FIAT, a new paradigm for computing finite element basis functions”. In: *ACM Transactions on Mathematical Software* 30.4 (2004), pp. 502–516. ISSN: 0098-3500. DOI: 10.1145/1039813.1039820.
- [77] Klimchuk, J. A. “On Solving the Coronal Heating Problem”. In: *Solar Physics* 234.1 (2006), pp. 41–77. DOI: 10.1007/s11207-006-0055-z. eprint: astro-ph/0511841 (astro-ph).
- [78] Knepley, M. G. et al. “Achieving High Performance with Unified Residual Evaluation”. In: *ArXiv e-prints* (2013). eprint: 1309.1204 (cs.MS).
- [79] Knepley, M. G. and Karpeev, D. A. “Mesh Algorithms for PDE with Sieve I: Mesh Distribution”. In: *Scientific Programming* 17.3 (2009), pp. 215–230. DOI: 10.3233/SPR-2009-0249. URL: <http://arxiv.org/abs/0908.4427>.
- [80] Knepley, M. G., Lange, M., and Gorman, G. J. “Unstructured Overlapping Mesh Distribution in Parallel”. In: (2017).
- [81] Kraus, M. and Hirvijoki, E. “Metriplectic integrators for the Landau collision operator”. In: *Physics of Plasmas* 24.10 (2017). ISSN: 10897674. DOI: 10.1063/1.4998610.
- [82] Kraus, M. et al. “GEMPIC: Geometric electromagnetic particle-in-cell methods”. In: *Journal of Plasma Physics* 83 (4 2017). ISSN: 14697807. DOI: 10.1017/S002237781700040X.
- [83] Landau, L. D. “Die kinetische Gleichung für den Fall Coulombscher Wechselwirkung”. In: *Phys. Zs. Sowjetunion* 10 (1936), pp. 154–164.
- [84] Landau, L. D. “On the Vibrations of the Electronic Plasma”. In: *Journal of Physics USSR* 10 (1946), pp. 25–34.
- [85] Lange, M., Knepley, M. G., and Gorman, G. J. “Flexible, Scalable Mesh and Data Management using PETSc DMPLex”. In: *Proceedings of the Exascale Applications and Software Conference*. Apr. 2015. URL: <http://www.easc2015.ed.ac.uk/sites/default/files/attachments/EASC15Proceedings.pdf>.

- [86] Lange, M. et al. “Efficient mesh management in Firedrake using PETSc-DMPlex”. In: *SIAM Journal on Scientific Computing* 38.5 (2016), S143–S155. DOI: 10.1137/15M1026092.
- [87] Langmuir, I. “Oscillations in Ionized Gases”. In: *Proceedings of the National Academy of Sciences* 14.8 (Sept. 1928), pp. 627–637. ISSN: 0027-8424. DOI: 10.1073/pnas.14.8.627. URL: <https://doi.org/10.1073/pnas.14.8.627>.
- [88] Lifshitz, E. M. and Pitaevski, L. P. *Physical Kinetics*. Ed. by E. M. Lifshitz and L. P. Pitaevski. Oxford: Butterworth-Heinemann, 1981. ISBN: 978-0-75-062635-4. DOI: <https://doi.org/10.1016/B978-0-08-057049-5.50015-1>. URL: <https://www.sciencedirect.com/science/article/pii/B9780080570495500151>.
- [89] Lynden-Bell, D. “Statistical Mechanics of Violent Relaxation in Stellar Systems”. In: *Monthly Notices of the Royal Astronomical Society* 136.1 (1967), pp. 101–121. ISSN: 0035-8711. DOI: 10.1093/mnras/136.1.101. URL: <https://doi.org/10.1093/mnras/136.1.101>.
- [90] Lynden-Bell, D. “The stability and vibrations of a gas of stars”. In: *Monthly Notices of the Royal Astronomical Society* 124 (1962), p. 279. DOI: 10.1093/mnras/124.4.279.
- [91] Maddison, J. R. and Farrell, P. E. “Directional integration on unstructured meshes via supermesh construction”. In: *Journal of Computational Physics* 231.12 (2012), pp. 4422–4432.
- [92] Maxwell, J. *A Treatise on Electricity and Magnetism*. A Treatise on Electricity and Magnetism. Cambridge University Press, 1873. ISBN: 9780511709340. DOI: <https://doi.org/10.1017/CBO9780511709340>.
- [93] May, D. A. and Knepley, M. G. “DMSwarm: Particles in PETSc”. In: *EGU General Assembly Conference Abstracts*. EGU General Assembly Conference Abstracts. 2017, p. 10133.
- [94] May, D. A., Brown, J., and Le Pourhiet, L. “A scalable, matrix-free multigrid preconditioner for finite element discretizations of heterogeneous Stokes flow”. In: *Computer Methods in Applied Mechanics and Engineering* 290 (2015), pp. 496–523. DOI: 10.1016/j.cma.2015.03.014.
- [95] McKinstry, C. J., Giaccone, R. E., and Startsev, E. A. “Accurate formulas for the Landau damping rates of electrostatic waves”. In: *Physics of Plasmas* 6.2 (1999), pp. 463–466. DOI: 10.1063/1.873212. URL: <https://doi.org/10.1063/1.873212>.
- [96] McQuillen, P. et al. “Emergence of kinetic behavior in streaming ultracold neutral plasmas”. In: *Physics of Plasmas* 22.4 (2015), p. 43514. DOI: 10.1063/1.4918705. URL: <https://doi.org/10.1063/1.4918705>.

- [97] McQuillen, P. et al. “Ion Temperature Evolution in an Ultracold Neutral Plasma”. In: *Physics of Plasmas* 22 (2015), p. 33513. DOI: 10.1063/1.4915135.
- [98] McQuillen, P. et al. “Ion holes in the hydrodynamic regime in ultracold neutral plasmas”. In: *Physics of Plasmas* 20.4 (2013), p. 043516. DOI: 10.1063/1.4802813. eprint: <https://doi.org/10.1063/1.4802813>. URL: <https://doi.org/10.1063/1.4802813>.
- [99] Mian, A., Bennamoun, M., and Owens, R. “Three-Dimensional Model-Based Object Recognition and Segmentation in Cluttered Scenes”. In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 28.10 (2006), pp. 1584–1601. DOI: 10.1109/TPAMI.2006.213.
- [100] Mouhot, C. and Villani, C. “On Landau Damping”. In: *Acta Mathematica* 207.1 (2011), pp. 29–201. DOI: 10.1007/s11511-011-0068-9. URL: <https://doi.org/10.1007/s11511-011-0068-9>.
- [101] Muller, D. E. “A Method for Solving Algebraic Equations Using an Automatic Computer”. In: *Mathematical Tables and Other Aids to Computation* 10.56 (1956), pp. 208–215. ISSN: 08916837. URL: <http://www.jstor.org/stable/2001916> (visited on 03/08/2023).
- [102] Myers, A., Colella, P., and Van Straalen, B. “A 4th-Order Particle-in-Cell Method with Phase-Space Remapping for the Vlasov-Poisson Equation”. In: (2016). DOI: 10.48550/ARXIV.1602.00747. URL: <https://arxiv.org/abs/1602.00747>.
- [103] Nagel, S. B. et al. “Magnetic trapping of metastable 3P_2 atomic strontium”. In: *Phys. Rev. A* 67 (1 Jan. 2003), p. 011401. DOI: 10.1103/PhysRevA.67.011401. URL: <https://link.aps.org/doi/10.1103/PhysRevA.67.011401>.
- [104] Nathawani, D. and Knepley, M. “One-dimensional modeling of pressure-driven droplet pinch-off”. In: *Bulletin of the American Physical Society* (2023).
- [105] Nathawani, D. K. and Knepley, M. G. “Simulating paraffin wax droplets using mixed finite element method”. In: *International Conference on Computational Fluid Dynamics (ICCFD 11)*. 2022.
- [106] Nathawani, D. K. and Knepley, M. G. “Droplet formation simulation using mixed finite elements”. In: *Physics of Fluids* 34.6 (2022), p. 064105. DOI: 10.1063/5.0089752.
- [107] Nicholeen, M. V. et al. *The Coronal Heating Problem*. 2020. DOI: <https://doi.org/10.1002/9781119815600.ch2>. URL: <https://doi.org/10.1002%2Fessoar.10502697.1>.

- [108] Numata, R. and Loureiro, N. F. “Electron and Ion Heating During Magnetic Reconnection in Weakly Collisional Plasmas”. In: *Proceedings of the 12th Asia Pacific Physics Conference (APPC12)*. Journal of the Physical Society of Japan, 2014. DOI: 10.7566/jpscp.1.015044. URL: <https://doi.org/10.7566%2Fjpscp.1.015044>.
- [109] Parashar, T. N. et al. “Kinetic dissipation and anisotropic heating in a turbulent collisionless plasma”. In: *Physics of Plasmas* 16.3 (2009), p. 32310. DOI: 10.1063/1.3094062. arXiv: 0801.0107 [physics.plasm-ph].
- [110] Park, M. J. et al. “Exploring the Origin of Thick Disks Using the NewHorizon and Galactica Simulations”. In: *The Astrophysical Journal Supplement Series* 254.1 (Apr. 2021), p. 2. DOI: 10.3847/1538-4365/abe937. URL: <https://dx.doi.org/10.3847/1538-4365/abe937>.
- [111] Parker, E. N. “Nanoflares and the Solar X-Ray Corona”. In: *The Astrophysical Journal* 330 (1988), p. 474. DOI: 10.1086/166485.
- [112] PETSc User Meetings. <http://www.mcs.anl.gov/petsc/meetings>. 2023.
- [113] Porter, L. J. and Klimchuk, J. A. *Soft X-Ray Loops and Coronal Heating*. 1995. DOI: 10.1086/176501.
- [114] Pritchett, P. L. “Geospace Environment Modeling magnetic reconnection challenge: Simulations with a full particle electromagnetic code”. In: *Journal of Geophysical Research: Space Physics* 106.A3 (2001), pp. 3783–3798. DOI: <https://doi.org/10.1029/1999JA001006>. URL: <https://agupubs.onlinelibrary.wiley.com/doi/abs/10.1029/1999JA001006>.
- [115] Pusztay, J. “A Particle Basis Vlasov-Poisson-Landau Solver for Plasma Simulation in PETSc”. PhD thesis. University at Buffalo, 2022.
- [116] Pusztay, J. et al. “Landau Collisions in the Particle Basis in PETSc”. In: *In Preparation* (2023).
- [117] Pusztay, J. V., Knepley, M. G., and Adams, M. F. “Conservative Projection Between Finite Element and Particle Bases”. In: *SIAM Journal on Scientific Computing* 44.4 (2022), pp. C310–C319. DOI: 10.1137/21M1454079. URL: <https://doi.org/10.1137/21M1454079>.
- [118] Qin, H. et al. “Canonical symplectic particle-in-cell method for long-term large-scale simulations of the Vlasov-Maxwell equations”. In: *Nuclear Fusion* 56.1 (2016). ISSN: 17414326. DOI: 10.1088/0029-5515/56/1/014001.

- [119] Reux, C. et al. “Runaway Electron Beam Generation and Mitigation During Disruptions at JET-ILW”. In: *Nuclear Fusion* 55 (2015), p. 93013. DOI: 10.1088/0029-5515/55/9/093013.
- [120] Robinson, M. P. et al. “Spontaneous Evolution of Rydberg Atoms into an Ultracold Plasma”. In: *Phys. Rev. Lett.* 85 (21 Nov. 2000), pp. 4466–4469. DOI: 10.1103/PhysRevLett.85.4466. URL: <https://link.aps.org/doi/10.1103/PhysRevLett.85.4466>.
- [121] Ruth, R. D. “A Canonical Integration Technique”. In: *IEEE Transactions on Nuclear Science* 30.4 (Aug. 1983), p. 2669. DOI: 10.1109/TNS.1983.4332919.
- [122] Shadwick, B. A., Stamm, A. B., and Evstatiev, E. G. “Variational formulation of macro-particle plasma simulation algorithms”. In: *Physics of Plasmas* 21.5 (2014). ISSN: 10897674. DOI: 10.1063/1.4874338.
- [123] Shalaby, M. et al. “SHARP: A Spatially Higher-order, Relativistic Particle-in-cell Code”. In: *The Astrophysical Journal* 841.1 (2017), p. 52. DOI: 10.3847/1538-4357/aa6d13. URL: <https://dx.doi.org/10.3847/1538-4357/aa6d13>.
- [124] Sharma, R. P. and Kumar, S. “Landau Damped Kinetic Alfvén Waves and Coronal Heating”. In: *Journal of Plasma Physics* 76.2 (2010), pp. 239–246. DOI: 10.1017/S0022377809990195.
- [125] Sobol’, I. “On the Distribution of Points in a Cube and the Approximate Evaluation of Integrals”. In: *USSR Computational Mathematics and Mathematical Physics* 7.4 (1967), pp. 86–112. ISSN: 0041-5553. DOI: [https://doi.org/10.1016/0041-5553\(67\)90144-9](https://doi.org/10.1016/0041-5553(67)90144-9). URL: <https://www.sciencedirect.com/science/article/pii/0041555367901449>.
- [126] Sonnendrücker, E. *Numerical Methods for the Vlasov–Maxwell equations (Book in preparation)*.
- [127] Spitzer, L. *Dynamical Evolution of Globular Clusters*. Princeton University Press, 1987.
- [128] Squire, J., Qin, H., and Tang, W. M. “Geometric integration of the Vlasov-Maxwell system with a variational particle-in-cell scheme”. In: *Physics of Plasmas* 19.8 (2012). ISSN: 1070664X. DOI: 10.1063/1.4742985.
- [129] Stamm, A. B., Shadwick, B. A., and Evstatiev, E. G. “Variational formulation of macroparticle models for electromagnetic plasma simulations”. In: *IEEE Transactions on Plasma Science* 42.6 (2014). ISSN: 00933813. DOI: 10.1109/TPS.2014.2320461.

- [130] Sulsky, D., Chen, Z., and Schreyer, H. L. “A particle method for history-dependent materials”. In: (June 1993). DOI: 10.2172/10177049. URL: <https://www.osti.gov/biblio/10177049>.
- [131] Takasao, S. et al. “Simultaneous Observation of Reconnection Inflow and Outflow Associated with the 2010 August 18 Solar Flare”. In: *Astrophysical Journal Letters* 745.1 (2012). ISSN: 2041-8205. DOI: 10.1088/2041-8205/745/1/L6. URL: <https://www.osti.gov/biblio/22048090>.
- [132] Tonks, L. “The Birth of “Plasma””. In: *American Journal of Physics* 35.9 (1967), pp. 857–858. DOI: 10.1119/1.1974266. eprint: <https://doi.org/10.1119/1.1974266>. URL: <https://doi.org/10.1119/1.1974266>.
- [133] Tóth, G. et al. “Extended magnetohydrodynamics with embedded particle-in-cell simulation of Ganymede’s magnetosphere”. In: *Journal of Geophysical Research: Space Physics* 121.2 (2016), pp. 1273–1293. DOI: <https://doi.org/10.1002/2015JA021997>. URL: <https://agupubs.onlinelibrary.wiley.com/doi/abs/10.1002/2015JA021997>.
- [134] Valluri, M. et al. “On Relaxation Processes in Collisionless Mergers”. In: *The Astrophysical Journal* 658.2 (Apr. 2007), pp. 731–747. DOI: 10.1086/511298. arXiv: [astro-ph/0609612 \[astro-ph\]](https://arxiv.org/abs/astro-ph/0609612).
- [135] Vlasov, A. “Über die Schwingungseigenschaften des Elektronengases”. In: *Zh. Èksper. Teor. Fiz.* 8 (1938), pp. 291–318. ISSN: 0044-4510.
- [136] Walhout, M. et al. “Magneto-optical trapping of metastable xenon: Isotope-shift measurements”. In: *Phys. Rev. A* 48 (2 Aug. 1993), R879–R882. DOI: 10.1103/PhysRevA.48.R879. URL: <https://link.aps.org/doi/10.1103/PhysRevA.48.R879>.
- [137] Walz-Flannigan, A. et al. “Cold-Rydberg-gas dynamics”. In: *Phys. Rev. A* 69 (6 June 2004), p. 063405. DOI: 10.1103/PhysRevA.69.063405. URL: <https://link.aps.org/doi/10.1103/PhysRevA.69.063405>.
- [138] Wathen, A. J. “Realistic eigenvalue bounds for the Galerkin mass matrix”. In: *IMA Journal of Numerical Analysis* 7.4 (1987), pp. 449–457.
- [139] Wikipedia. *Hasse Diagram*. http://en.wikipedia.org/wiki/Hasse_diagram. 2015. URL: http://en.wikipedia.org/wiki/Hasse_diagram.
- [140] Winebarger, A. R. et al. “Defining the “Blind Spot” of Hinode EIS and XRT Temperature Measurements”. In: *The Astrophysical Journal* 746.2 (2012), p. L17. DOI: 10.1088/2041-8205/746/2/117. URL: <https://doi.org/10.1088/2041-8205/746/2/117>.

- [141] Xiao, J., Qin, H., and Liu, J. “Structure-preserving geometric particle-in-cell methods for Vlasov-Maxwell systems”. In: *Plasma Science and Technology* 20.11 (2018). ISSN: 20586272. DOI: 10.1088/2058-6272/aac3d1.
- [142] Xiao, J. et al. “Explicit high-order non-canonical symplectic particle-in-cell algorithms for Vlasov-Maxwell systems”. In: *Physics of Plasmas* 22.11 (2015). ISSN: 10897674. DOI: 10.1063/1.4935904.
- [143] Xiao, J. et al. “Explicit high-order noncanonical symplectic algorithms for ideal two-fluid systems”. In: *Physics of Plasmas* 23.11 (2016). ISSN: 10897674. DOI: 10.1063/1.4967276.
- [144] Yoshida, H. “Construction of higher order symplectic integrators”. In: *Physics Letters A* 150.5 (1990), pp. 262–268. ISSN: 0375-9601. DOI: [https://doi.org/10.1016/0375-9601\(90\)90092-3](https://doi.org/10.1016/0375-9601(90)90092-3). URL: <https://www.sciencedirect.com/science/article/pii/0375960190900923>.
- [145] Zaremba, S. K. “Good lattice points, discrepancy, and numerical integration”. In: *Annali di Matematica Pura ed Applicata* 73.1 (Dec. 1966), pp. 293–317. ISSN: 0373-3114. DOI: 10.1007/BF02415091. URL: <http://link.springer.com/10.1007/BF02415091>.
- [146] Zhang, H., Constantinescu, E. M., and Smith, B. F. “PETSc TSAdjoint: A Discrete Adjoint ODE Solver for First-Order and Second-Order Sensitivity Analysis”. In: *SIAM Journal on Scientific Computing* 44.1 (2022), pp. C1–C24. DOI: 10.1137/21M140078X.
- [147] Zhao, Z. et al. “Laboratory observation of plasmoid-dominated magnetic reconnection in hybrid collisional-collisionless regime”. In: *Communications Physics* 5.1 (2022), p. 247. DOI: 10.1038/s42005-022-01026-7. eprint: 2202.12471 (physics.plasm-ph).
- [148] Zhou, T., Guo, Y., and Shu, C. W. “Numerical study on Landau damping”. In: *Physica D: Nonlinear Phenomena* 157.4 (2001), pp. 322–333. ISSN: 01672789. DOI: 10.1016/S0167-2789(01)00289-5.
- [149] Zonta, F., Pusztay, J., and Hirvijoki, E. “Multispecies structure-preserving particle discretization of the Landau collision operator”. In: *Physics of Plasmas* 29.12 (2022), p. 123906. DOI: 10.1063/5.0105182. URL: <https://doi.org/10.1063/5.0105182>.