High order semi-Lagrangian numerical solutions to the plasma kinetic equation in the edge of magnetic fusion devices

David Sirajuddin

A preliminary report submitted in partial fulfillment of
the requirements for the degree of

Doctor of Philosophy

(Electrical and Computer Engineering)

at the

UNIVERSITY OF WISCONSIN − MADISON

December 12, 2014

Contents

List of Figures

List of Tables

Chapter 1

Introduction

Magnetic confinement fusion aims to confine hot plasma by the use of carefully designed magnetic coils. The basic strategy is to apply an external field so that the strongly magnetized plasma particles have large parallel field transport relative to small perpendicular excursions towards the material boundaries of the device. The two most researched devices are the two-dimensional, toroidally symmetric *tokamak* (DIII-D, JET, Alcator C-mod, ...) and the three-dimensional *stellarator* (HSX, LHD, W7-AS, ...) whose particular magnetic coil shapings and geometry provide for good confinement properties. Both of these systems are devised with two distinct regions: a high temperature core (millions of degrees Kelvin) whose volume is limited by a set of nested magnetic flux surfaces, and an outer section of open-field lines that are made to intersect either designated target plates in the case of divertors or conducting inserts known as limiters. This latter region is termed the *edge* of a magnetic confinement vessel, and is the concentration of this research.

The edge region of magnetic fusion devices is seen to be a complex, dynamic, and significant controller of the global confinement of high temperature plasmas. This preliminary work proposes a research regimen to develop high-order computational kinetic simulations of electrostatic plasma transport in the edge of fusion devices with a relevant boundary geometry subject to an intended host of collisional processes (i.e. solutions to the Boltzmann-Poisson sytem). The work accomplished thus far establishes a foundation for high-order numerical solutions to advection equations using a forward-trajectory semi-Lagrangian method known as *convected scheme* (CS). Recognizing advection equations present a reduced case of the Boltzmann transport equation, natural steps are prescribed to extend this foundational CS model for advection equations to ultimately address the full complexity of this so-called *plasma kinetic equation* for species in the edge region. While the intention is to obtain high order numerical solutions to the Boltzmann-Poisson system in the vicinity of the edge, the groundwork in reaching this point may permit further progress in modeling as discussed in the latter points just below.

The above developments will lend themselves towards the following goals:

1. *High order (> 2) accuracy for open systems*: A symplectic splitting technique will be advanced and combined with a remap correction to reduce numerical diffusive error. This first step will be applied to the simplest case of an electrostatic plasma which is described by the Vlasov-Poisson system. We will address open systems (i.e. periodic plasmas) in 1D-1V as our first pass.

2. *Boundaries*: The plasma coexists with a material boundary (plasma-facing components), which amounts to discontinuities in quantities such as temperature. A limiter (*Hartman filter*) will be designed to simulate this sharp transition.

3. *Collisions*: Moving onward from the Vlasov-Poisson equation, we will apprehend collisions among charged particles in a Boltzmann-Poisson formalism. In order to make this high order accurate, we will preliminarily investigate a method known as *defect correction*.

4. *Sheath and presheath modeling*: The strong gradients present in edge plasmas present a vast range of scales in plasma parameters that place significant restriction on mesh resolution as well as enforcing limitations on the physical time step required to account for all necessary physics. The developed high order tools above will be applied to edge modeling.

5. *Higher dimensions*: If progress permits, the above developed tools provide an opportunity to model higher dimensional systems to capture more of the physics involved in edge plasmas in magnetic fusion devices. In particular, a first step would be to add an additional velocity dimension (1D-2V). This will allow the effect of a magnetic field to be incorporated, so that the system to solve is the Boltzmann-Maxwell system.

This document is organized as follows. In chapter 2, a two-part literature survey is given. The former portion presents a brief exposition of the concept of magnetic confinement fusion and discusses relevant theory pertaining to the physics of the edge. Here, we focus specific attention to representative values of plasma parameters in typical devices for this region to emphasize the breadth of scales. Thus, the unique modeling challenges of plasma simulation in this region are highlighted. The latter part compares various computational methods used in the solution to the same system this preliminary research is working towards addressing; that is, the collisionless Boltzmann transport equation coupled with Poisson's equation for self-consistent electric field calculations (the *Vlasov-Poisson* system). Chapter 3 describes and motivates the preliminary work accomplished thus far, which consists of arbitrarily high-order convected scheme solutions of the single speed advection equations in one-dimension subject to periodic boundary conditions. Convergence analysis and representative test cases demonstrating the efficiency and accuracy are provided. Chapter 4 establishes a roadmap for future work, including the aim of apprehending the Vlasov-Poisson system. The numerical solution to this system is a stepping stone towards including not only boundaries, but also collisions which are encompassed in the Boltzmann-Poisson system. Finally, in chapter 5 we conclude with a summary of the overall proposition this preliminary document puts forth, and concretize both its importance and intellectual merit in the scope of not only progressing accurate and efficient numerical solutions in the context of fusion plasmas, but suggest additional applications where this computational foundation may find natural utility.

Chapter 2

Literature review

This chapter is segmented into two main categories. The first presents an exposition of magnetic confinement fusion wherein we find natural opportunity to formally introduce the edge region and to highlight its distinction from the confined core. Here, we introduce the governing equations for the kinetic plasma system, and cements its foundations in mathematics as well as present a physical derivation of the Boltzmann equation directly from Hamilton's equations. The second category constitutes the bulk of this chapter, which details the particular semi-Lagrangian method furthered in this work known as the *convected scheme* (CS). We begin with a review of this *method of characteristics* solution whereafter a prescription is carefully developed to render it accurate up to any order $N \in \mathbb{N}$. The chapter closes with the important topic of split operator methods, which forms a necessary bridge in apprehending high order solutions to more complex equations in our graduated pursuit of handling the Boltzmann equation.

2.1 Motivation

In this section, we begin with a brief overview of the idea of magnetic confinement fusion, formally define the edge region and detail its distinguishing characteristics. The key physics and unique issues of the edge are reviewed (e.g. edge localized modes), especially in the scope of fusion engineering and design. To this end, the concept of the divertor/limiter will find significant motivation. It will be seen that the edge plays a significant role in the overall confinement of a magnetized plasma. Particular attention is paid to distinguishing the physics associated with the edge as compared to the hot plasma core. In this way, fluid models will contend with kinetic descriptions, and it will be argued a kinetic treatment is the appropriate treatment required for accurate edge physics calculations.

2.1.1 Magnetic confinement devices

In a magnetic confinement fusion (MCF) reactor, hot plasma is contained by strong magnetic fields with the aim to produce more energy from fusion than is used to create it. As mentioned, the most investigated candidate devices for confinement are the tokamak or the stellarator. In the tokamak (Fig. 2.1), planar external coils set up magnetic fields that direct driven charged particles toroidally from a pulsed transformer, whose current adds a twist in the field lines by its produced poloidal field. This results in a magnetic field that is helical which is necessary to counter particle drifts. In a stellarator, the coils are modular (nonplanar) so that the helical twist can be accomplished with zero plasma current.
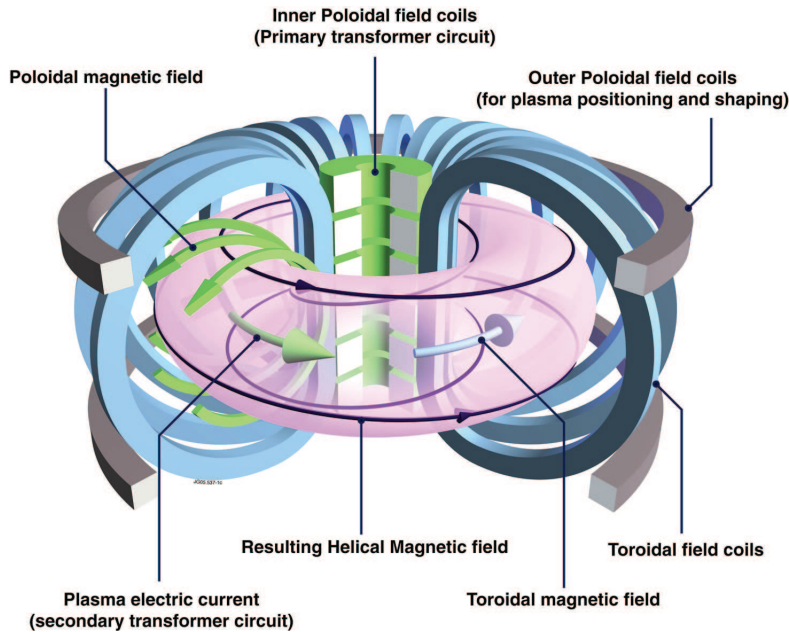
Figure 2.1: A typical tokamak setup [16].

A simple power balance analysis leads to a consolidated figure of merit in terms of reactor parameters known as *Lawson's criterion*, which stipulates the basic minimum requirement for any viable fusion reactor. A related, but more telling, criterion that is more commonly regarded is the assembled triple product $nT\tau_E$, involving the plasma number density n, fuel ion temperature T, and energy confinement time τ_E. Separately, these quantities indicate a three-dimensional parameter space whose bounding contour surface corresponds to breakeven conditions (independent of device), and above which marks a region where self-sustained fusion burn can take place in steady state known as *ignition*, defined in analog to fossil fuels. Using a empirical

approximation for fusion cross-sections in the vicinity of operable temperature ranges, a direct calculation gives the following condition for ignition in a Deuterium-Tritium (DT) fuel:

$$nT\tau_E \gtrsim 5 \times 10^{21}\,\mathrm{m}^{-3} \cdot \mathrm{s} \cdot \mathrm{keV} \qquad \underline{\text{The Lawson Criterion}}, \text{DT fuel}$$

where it is noted that while the triple product $(nT\tau_E)$ is not the original quantity of interest $(n\tau_E)$ put forth by Lawson in 1955, it has notwithstanding inherited the same name and is perhaps more commonly what is meant by the Lawson criterion . Including more physics gives refinements to this figure; however, more detailed analysis is not seen to adjust this baseline appreciably [63]. Thus, Lawson's criterion is still one of the two most cited (general) benchmarks needed to be achieved for a reactor. Experimental devices are able to achieve appropriate levels for these parameters separately; however, meeting all three marks in the triple product simultaneously is a step yet to be obtained in the magnetic confinement community (TFTR was the first to meet density and energy confinement time requirements, but could not produce them at the needed temperatures).

The other benchmark is a measure of success that can be straightforwardly defined as the standard energy gain factor Q [63]. The value $Q = 1$ is identified as breakeven, where external heating is matched by produced output energy. Practically, the stricter condition of ignition is sought after. At $Q = 1$ for a DT fuel, 20% of the produced fusion energy is reinvested in subsequent fusion reactions from α particle heating (the remainder predominantly escapes with the high energy neutrons), thus $Q = 5$ is understood as the point at which fusion power equals external input, and $Q \to \infty$ marks the condition for ignition (no input energy is needed to produce an output). A fusion reactor need not achieve ignition, but in the context of electrical power generation either $Q \sim 30$ or $Q \sim 70$ are projected to be sufficient for magnetic and inertial fusion power plants, respectively [37]. However, the quality of confinement needed to achieve such high Q values is almost as strict as that needed for ignition [18], so ignition remains the most discussed condition with the tacit understanding that this ultimately need not be achieved exactly. ITER aims to demonstrate a Q value of $5 \sim 10$, JET has aspirations to achieve $Q \sim 20$.

As concerns the state of the fusion, present day devices keep plasma density n sufficiently low to discourage unrecoverable losses through Bremsstrahlung radiation ($P_{Brem} \propto n^2$), but high enough to produce the required fusion energy density. In most devices the required plasma number density is approximately a million times less than the density of air (i.e. $10^{20}\,\mathrm{m}^{-3}$).

The ideal temperature range is around $100 - 200$ million Kelvin (about $7 \sim 14$ keV) for a DT plasma, where a compromise has been made between minimizing Bremsstrahlung radiation ($P_{Brem} \propto T_e^{1/2}$) [33], and maximizing the fusion cross-section, which exhibits a maximum at a much higher temperature (~ 70 keV) [25]. To achieve temperatures in the keV range a multi-levelled approach is undertaken. Collisional, or Ohmic, heating occurs preferentially for lower temperatures (i.e. the conductivity $\kappa \propto T^{-3/2}$), which raises the temperature up partway. To increase plasma temperatures higher, other methods are employed (NBI, ICRH, ECRH). Decades of fusion research have developed techniques that readily achieve appropriate temperatures (NBI, RF) and densities [18], leaving the remaining task to accomplish the required energy confinement time τ_E. This time is strongly influenced by the edge.

2.1.2 Defining the *edge*

The *edge*, or *scrape-off layer* (SOL), we formally define as the plasma that exists outside of the magnetically confined core (figure 2.2). In the tokamak configuration shown, the last closed flux surface (*separatrix*) is depicted as the flux surface that contains the locus of null points (X-points) in the magnetic field. Thus, a two-dimensional (e.g. poloidal) cross-section exhibits a characteristic ribbon shape, whose magnetic field lines terminate at divertor target plates. This surface can be used to classify distinct regions of the plasma. The region inside of the separatrix is confined in closed flux surfaces, constituting the core (or *main*) plasma whose

	Alcator C-Mod	DIII-D	NSTX	ITER
Parameters				
Major radius R [m]	$0.61 - 0.74$	$1.49 - 1.88$	$0.8 - 1.0$	6.2
Minor radius a [m]	$0.169 - 0.264$	$0.331 - 0.752$	$0.5 - 0.787$	2
Aspect ratio R/a	$2.8 - 3.6$	$2.5 - 4.5$	$1.27 - 1.6$	3.1
Max plasma volume [m^3]	1	24	14	700
Max T_i, T_e [keV]	5.6, 6.0	27.0, 16.0	2.5, 4.1	30, 30
Max current I [MA]	2.05	3	1.5	15
Max power density [MW/m^3]	6.7	1.3	1.1	0.7
Shot length [s] at B_{max}	1	6	1.5	400
PFCs	Mo, W	C	CFC/Graphite	W, C and Be
			Li coating	

Table 2.1: Parameters for four tokamaks. Reproduced from [56]

maximum width can be on the order of a meter. Drifts and anomalous transport enable plasma to access the open flux surface region outside of the separatrix, which forms a centimeter thick region designated as the edge or scrape-off layer. Inside the separatrix arms is the private plasma, a domain that is formed by drift and anomalous transport from both the main plasma as well as from the scrape-off layer. Table 2.1 provides a comparison of several tokamak parameters.

Among the features that characterize the edge is the presence of steep gradients in the density and temperature profiles which constitute what is known as the *pedestal*, the height of which is seen to provide a gauge for the quality of plasma confinement (Figure 2.3) [66]. While the mechanisms are not well understood, consistent correlations are seen to exist between power flux through the last closed flux surface and the spontaneous transition from a mode of low (L) to high (H) confinement [66, p. 4], a regime marked by an approximate two-fold increase in energy confinement time τ_E [58, p. 358]

The pedestal is formed due to conditions resulting from the presence of a material boundary and because the charged particles are characterized by a distribution of velocities. In the transition to a high mode (H-mode) of confinement (section 2.1.3), an internal transport barrier must develop that further amplifies the difference of scales between the core and the more sparse edge region. Notwithstanding, the transition layer occurs even in unconfined systems as Bohm [6] demonstrated without the context of fusion in mind.

Bohm showed the presence of a material boundary (wall) sets up simple electrodynamical interplay between the highly mobile (less massive) electrons and the less mobile (higher mass) positive ions. The narrative is explained simply as the presence of a the wall constitutes a sink for the mobile electrons which foremostly charge it negative. The negative potential from the accumulation of charge decreases the electron current to the wall by raising the energetic bar required to reach it. Meanwhile, any positive ions in sufficient vicinity are accelerated by the negative wall potential in accordance with the Debye shielding effect which increases positive ion effluxes towards it. The transport is enhanced by the parallel pressure gradient. For ions, the accelerating potential and the pressure gradient encourage ions to flow to the walls, whereas for electrons the electric field set up inhibits incoming fluxes though its parallel pressure gradient notwithstanding helps it (figure 2.4).
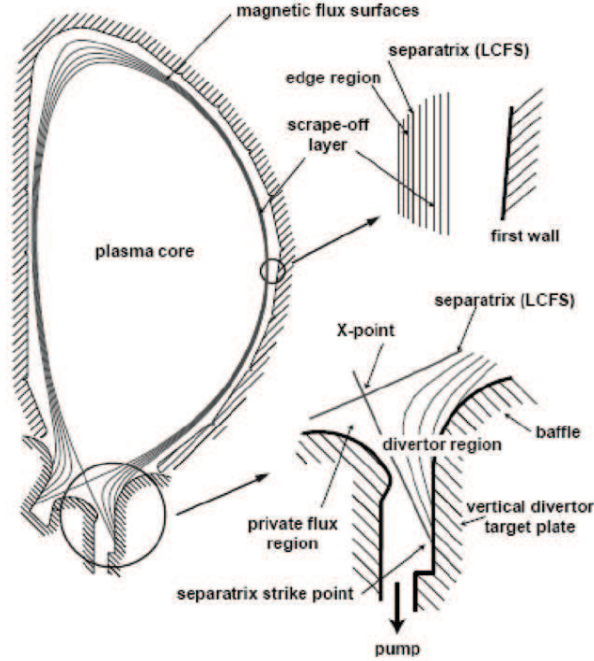
Figure 2.2: The taxonomy of terms used to describe the edge are shown for the case of a divertor tokamak configuration; limiter terminology follows similarly. External coils combined with the toroidally driven plasma current produce a helical magnetic field that traces out closed flux surfaces. This region of closed field lines constitute the core of a magnetic confinement device. A (poloidal) divertor is shown here that diverts the poloidal field so that a magnetic X-point appears on poloidally projected plane (shown). The magnetic surface that contains the X-point is the *separatrix*, and the region outboard of this surface is the *edge*. Reprinted under free-use policy from the IAEA [31].

The ion current makes the potential at the wall less negative; however, this simultaneously permits a larger range of electrons to reach the wall which lends itself to make the potential more negative. The two competing processes interplay until an equilibrium is reached, marked by the condition of equal *fluxes*, known as *ambipolar* flow, which creates a region of positive space charge near the wall known as the *sheath* ($n_i > n_e$, figure 2.4).

By seeking the potential profile $\phi(x)$ for cold ions ($T_i = 0$) in a fluid plasma as a solution to Poisson's equation ($\varepsilon_0 \partial_x^2 \phi(x) = -\rho(x)$) and matching the solutions obtained from the multiple scales analysis (one solution is obtained for the region pertaining to the core extending to the edge, and the other is analyzed from the edge reaching inward to the core side), Bohm showed that only physical (non-oscillatory) potentials were possible if the monoenergetic ions were accelerated exactly to the sound speed c_s near the plasma wall. This sharp step up in the velocity profile for ions in space defines a clear *sheath edge*. The generalization to a physical plasma which is characterized not by a monoenergetic source of ions, but rather by a distribution results in a smoothing of this sheath edge so that this transition layer is known as the *pedestal* (figure 2.5).

Properly, the Bohm criterion refers to one side of the inequality from the aforementioned multiple scales stepthrough. It expresses that ions from the core are accelerated to at least the sound speed c_s at the sheath edge (*se*) when "exiting" the plasma to reach the material boundary:

$$v_{se} \geq c_s, \qquad \underline{\text{Bohm criterion } (T_i = 0)} \qquad (2.1)$$

Thus, the Bohm criterion sets a lower bound of the ion exit velocity [58]. The requirement obtained when
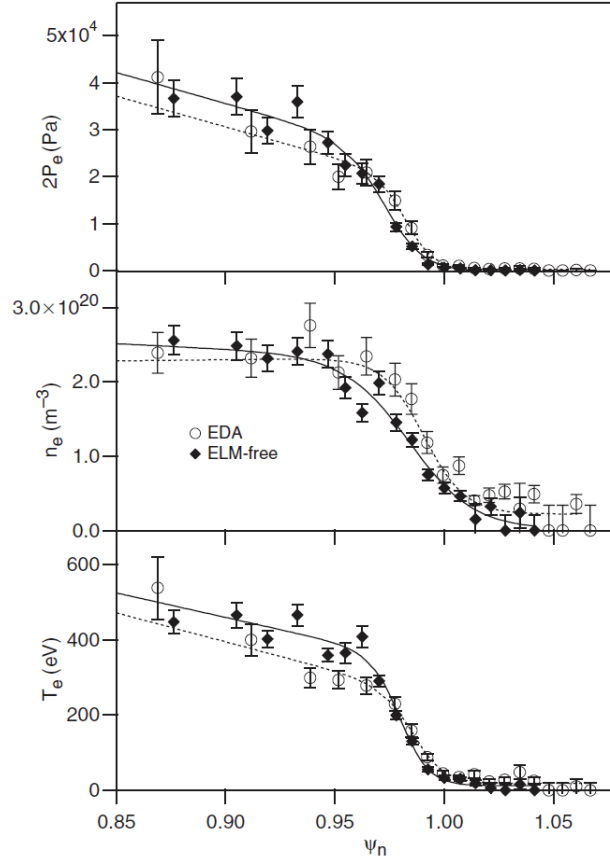
Figure 2.3: Typical profiles in the edge region. The sharp gradients in plasma parameters (electron pressure P_e, temperature T_e, and electron density n_e) form an abrupt step in the traces with respect to distance (normalized poloidal flux label ψ_n) towards the edge known as the *pedestal*. Here, an experiment on Alcator C-Mod demonstrates two kinds of H-modes: enhanced D_α (EDA) mode, and an ELM-free mode [47].

considering the problem from the core-side enforced that $v_{se} \leq c_s$, so that the equality is converged on when the two solutions are matched. The generalization to $T_i \neq 0$ produces a similar, yet distinct condition:

$$\int_0^\infty \frac{f_{se}^i(v)dv}{v^2} \leq \frac{m_i}{kT_e}, \qquad \underline{\text{Bohm criterion } (T_i \neq 0)} \tag{2.2}$$

where f_{se}^i is the distribution function for the ions at the sheath edge ($x = x_{se}$). Thus, the edge region in a magnetic confinement device is characterized by the the region of open field lines that extends outboard to the material boundary where invariably we encounter *sheath physics* whose sheath thickness is dictated by details contained only in a kinetic description.

2.1.3 Unique issues in the edge

The confinement improvement when operating in H-mode is seen to be *too good* in the sense that the enhanced confinement of the core leads to extreme gradients in the scrape-off layer which ultimately can lead to the generation of unique problems that pervade the edge region [23]. Two such problems are edge localized instabilities and the power dilution of the plasma by impurity influxes produced from the plasma interaction with the plasma facing components. Peaked fluctuations (figure 2.7) in both plasma and field quantities are

15
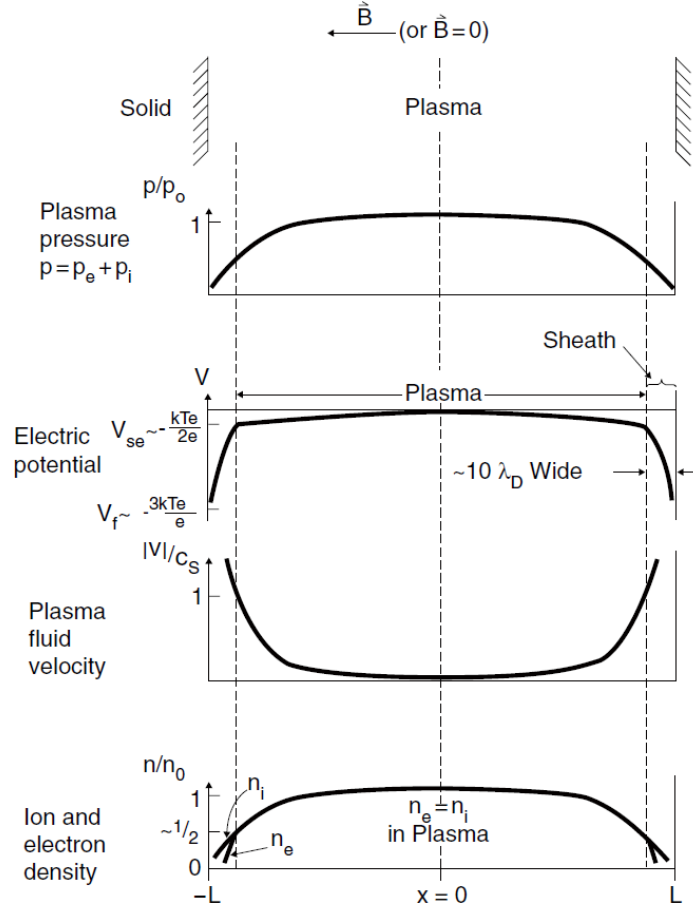
Figure 2.4: Traces of plasma parameters when walls are present

observed in this region which are regarded as signatures of the microturbulent (anomalous) transport that significantly amplify cross-field outfluxes of particles as compared to neoclassical predictions. Further still, the aforementioned instabilities deliver more high energy fluxe to the material boundary. The repeated stresses on the materials are not a concern for present day devices; however, the power will be sufficiently higher for larger devices such as ITER, and these repeated bursts of power effluxes to target plates will likely be a limiting factor for reactor lifetime. In this way, it is understood that an acceptably low plasma temperature at the wall must be maintained in order for the plasma wetted targets (divertor plates or limiters) to preserve their material integrity.

Multi-faced asymmetric radiation from the edge (MARFEs)

With respect to the concern just outlined above, it is possible to engineer a cushion between the target plates and the core plasma with injected neutral populations through puff valves. This limits the power fluxes to the target materials significantly; however, it has been seen that total *detachment* of the plasma volume in this way encourages the formation of persistent stationary structures due to *multi-faceted (poloidally) asymmetric radiation from the edge* (MARFE), particularly from recycled or injected neutral impurity atoms whose mean free path is large and can significantly penetrate into the plasma. Thus, a high amount of power is diluted from fuel ions and and is released through subsequent radiation. This presents a risk in that the plasma suffering a global disruption due to thermal collapse. In tokamaks, these structures manifest as toroidally
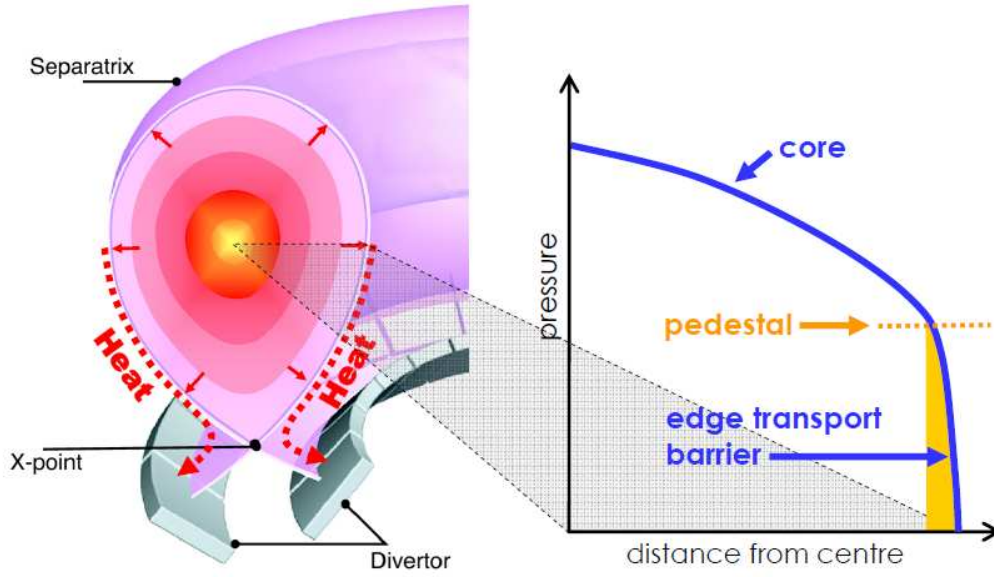
Figure 2.5: A poloidal cross-section of the magnetic configuration is shown on the left, whereas the plasma pressure along a radial arm from the core center to the edge is shown where the pedestal is identified in comparison with the core and edge [14].

symmetric aggregates of high density cold plasma at the inboard side of the last closed flux surface. In this way, temperature and density gradient driven transport that are enhanced by the presence of MARFEs will rapidly lead to global disruptions whenever a density limit is exceeded (Greenwald or Murakami limit for tokamaks, or the Sudo limit in stellarators). At densities approaching such a limit, the impurity radiation power is proportional to the square of its density so that it increases until the impurity radiation power becomes equal with the power input. Above this limit the plasma suffers thermal collapse by a corresponding decrease in temperature, thereby increasing resistivity which causes the current to be quenched. Thus, a compromise between the extent of detachment and power fluxes to plasma wetted targets must be settled on so that heat loads to the plasma facing components (PFCs) can be reduced while simultaneously discouraging the formation of these radiation precipitates. The recommendation for ITER is to operate in the *partially detached regime* wherein the inner divertor plate is completely detached, but the outer plate is only partially shielded [17]. In this way, the detached regions correspond to plasma particles whose connection length is longest and whose strike point is most remote from the plasma core. Impurity atoms from the recycling process will then not penetrate too far into the SOL in order to stave off disruption events of this kind.

Edge localized modes (ELMs)

A unique collective phenomenon that occurs in edge of H-mode plasmas are the so-called *edge localized modes* (ELMs). Such wave phenomena occur with regular periodicity whose appearance and quenching show provisional dependence on the power flux through the last closed flux surface (LCFS) in limiter systems, or equivalently the separatrix in divertor configurations. ELMs present with periodic tone bursts of directed plasma density in certain regimes that can deleteriously bombard the plasma facing components with high power fluxes. Further, the presence of ELMs is known to decreases the extent of confinement as shown generically in figure 2.6.
Notwithstanding, the elimination of ELMs is not desirable as it has utility in sweeping helium ash and impurites
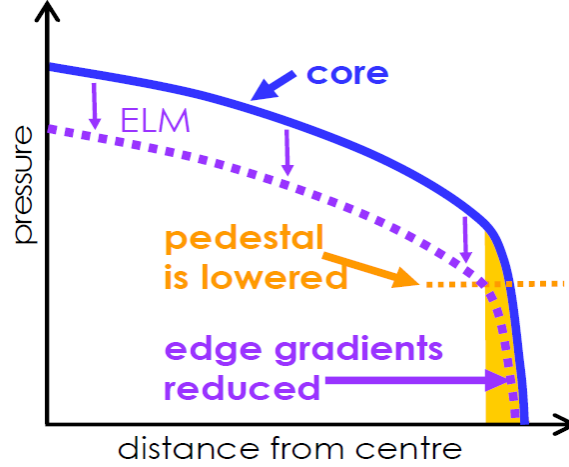
Figure 2.6: When an ELM removes density from the edge region, the edge gradients decrease and so does the pedestal which directly reduces the confinement of the core [14].

out of the plasma that would otherwise accumulate without a means to remove it. ITER intends to operate in a so-called *ELMy H-mode*.

Since ELMs are within the scope of the engineering of a fusion reactor design, we first note briefly the means by which their presence is dealt with or otherwise handled. While occassional ELMs are desired, their manifestation must be managed or minimized. Designs incorporating an ergodized edge and/or using error field correction coils (EFCC) have demonstrated the plasma effluxes can be distributed and ELMs can be mitigated [48]. Ergodicity in the edge region is accomplished by external coils that induce resonant magnetic perturbations with respect to q rational. We define *ergodicity* to mean introduced radial excursions of a field-line in its otherwise helical pathway as compared to the confined core. The result is enhanced radial transport in the presence of already strong parallel transport. Such confluent flows flatten the temperature profile with the aim of increasing the area of particle fluxes on the target plates to make the power deposition on the PFCs more reasonable. Combining an ergodic edge with an injection of an impurity (neon or argon are most common) also assists to reduce the power loads as the chaotic region of magnetic field lines spreads the exiting particles over a larger three-dimensional volume so that the chance of momentum exchange with impurity ions is more likely. In doing so, a radiation mantle can form with respect to the divertor plates and the resulting charged particle fluxes to the PFCs is reduced further. Lastly, more targeted stochastization of the edge magnetic field using resonant magnetic perturbations (RMPs) has also proven viable for active ELM control in devices such as KSTAR [35], and most recently by DIII-D with ITER-like shaping and collisionality [19]; JET has been in the process of updating its system to accommodate an RMP attachment.

Types of ELMs and regimes of operation

Edge localized modes are categorized by an enumerated type (I, II, and III), whose existence show clear delineations with respect to power flows through the LCFS. While the dependence is observed on the power through this flux surface, ELMs are classified most usually in terms of power input given it is a directly controlled experimental parameter. A general feature of these instabilities are their repetitiveness; ELMs appear as disturbances of the H-mode plasma periphery whose periodic relaxation mechanism results in pulses of directed high energy density plasma towards the walls of the confinement device. It is thought that ELMs originate as MHD instability (peeling-ballooning mode) caused by gradients in either pressure or current. Below, we step through each type in the order they are encountered for increasing values of the power input.
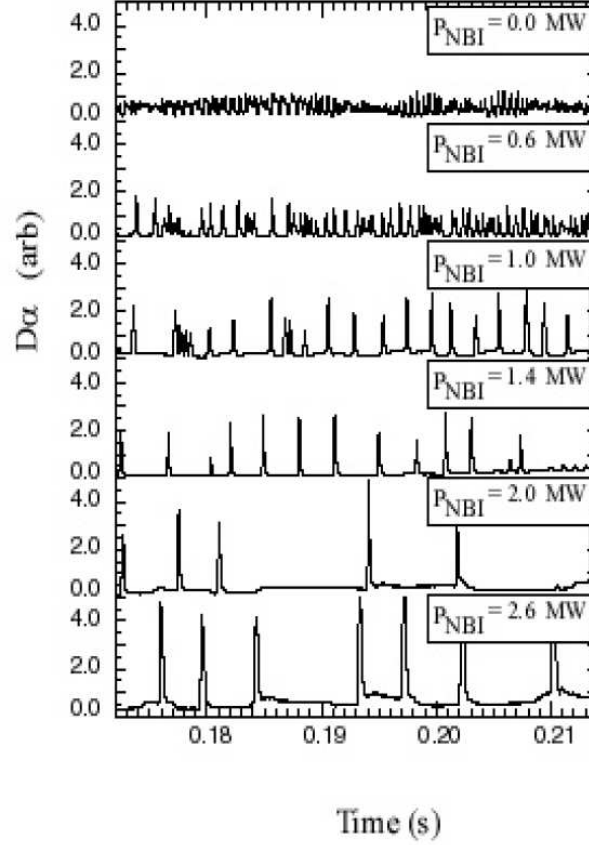
Figure 2.7: D_α measurements for different neutral beam injection (NBI) power inputs P_{NBI}. As the power increases, a regularity is achieved that emblemizes type III ELMs. Higher still, the type III ELM is exchanged for the giant type I ELMs shown in the final bottom two plots [14].

Type III ELMs develop at the lowest power inputs, decreasing in their frequency of recurrence with increasing power input until they become fully damped. A similar, yet distinct, phenonemon is observed as rapid interchanges between (L and H) modes near the L-H mode threshold power input (the *dithering H-modes*). These events are distinguishable from type III ELMs only insofar as that a high frequency magnetic field fluctuation is observed in the edge for type III ELMs as a precursor to its onset; this marker is absent in the dithering between modes. As the power input is increased further an intermediate range is reached, devices with highly shaped plasmas (triangulation and elongation) have witnessed generation of type II (*grassy*) ELMs whose repetition frequency ostensibly show no significant dependence on power input. As the power is increased higher still, type II ELMs are traded for type I, or *giant*, ELMs whose frequency is seen to increase with power (figure 2.7). Since the grassy (type II) ELMs are observed only in highly shaped plasmas, it is inviting to consider designing less shaped systems that then will exhibit quiescent regimes for intermediate power input. Thus, it seems possible to operate in an ELM-free regime to minimize damaging power loads to the plasma facing components (*QH modes*). This had first been achieved at DIII-D by counter-injection of neutral beams (against plasma current) alongside cryopumping for density management [8, 10], but has since been shown to be attainable with either co- or counter-injection with respect to the sign of edge rotation so long as the magnetic shear in this region is sufficiently high [9, 10]. QH modes had later been demonstrated in tokamak experiments such as JT-60U and the ASDEX-Upgrade [60], as well as NSTX [50]. The disadvantage to such modes originally was outpumping of impurity densities at the edge without affecting energy transport

in the core; however, characteristic edge harmonic oscillations (EHOs) were seen to develop in tandem with the onset of the QH mode which helped remove impurity buildup naturally as the EHOs enhance edge transport which permits outpumping of impurity accumulations as needed without affecting the core confinement. It has been suggested that this regime likely can maintain acceptable energy confinement times at the requisite densities [40]. Like ELMs, EHOs suggest their mechanism lies in MHD. However, they are seen to occur in an operating space below the peeling-ballooning instability limit.

It is alternatively seen that one may forego the aforementioned concern in favor of operating at higher power inputs by highly shaping the plasma. It has been demonstrated for multiple devices that while highly shaped plasmas exhibit grassy ELMs at intermediate power, type I ELMs at higher power inputs are damped out or otherwise evaded at the power injections desired. The role of shaping can be understood as raising the peeling-ballooning mode stability limit on power flow through the separatrix (or LCFS) so that systems can be designed to operate at higher desired power injections but still below type I ELM range. To this end, it is noted that puffing neutral populations to dilute some of the power through impurity radiation in the edge also directly accomplishes the same result [66]. Pursuant of this pathway allows the so-called *very high* (VH) mode of confinement to be achieved. For example, by modulating the triangularity of the confined plasma, JET was the first device to accomplish this, which is exemplified by an increase in energy confinement time by a factor of 1.5 above the conventional H-mode in the same device (JET) [32]. Notwithstanding, the problem of impurity accumulation has precluded goals of the VH-mode of operation in favor of either QH modes with EHOs or ELMy H-mode operation in type I ELM range given both their opportune utility in vacating impurity accumulations. Thus, while the absence of ELMs is seen as a positive with respect to the elimination of pulsed stresses on the plasma facing components, for devices such as ITER the overwhelming recommendation is to operate at a type I ELMy H-mode (although QH investigations are ongoing such as DIII-D and Alcator C-Mod's *enhanced D-alpha* (EDA) mode [14]).

Additional topics in the edge

Beyond the local structures that develop in the edge, the crosstalk between the core and the edge plasma is not well understood. Heat and particle transport from the confined plasma into the edge region is not adequately modelled outside of massively parallel multi-physics PIC or gyrokinetic codes (e.g. GENE), as well as within the scrape-off layer itself. To develop the predictive capability needed to understand the heat effluxes bombarding the divertor targets, for example, is a major task of the fusion computation community as the answers to such questions are needed to aid engineering design in the assessment of lifetime of materials. Other topics include the interaction with the wall (recycling of plasma as well as impurity transport back into the core from sputtering off the PFCs). Since the plasma existing in the edge is often non-Maxwellian, full-f kinetic treatments are required which preclude simplifications such as that used by the GENE code which is based on the premise that the distribution function can be linearly superposed as an averaged Maxwellian and a fluctuating component. Such a formulation is not valid in the nonlinear, nonlocal thermodynamic equilibrium region of the edge.

2.1.4   Kinetic versus Fluid descriptions

We are beginning to see the arguments that advocate a kinetic treatment for numerical simulations of plasma systems in the edge region. For example, we note that this generalized Bohm result (2.1) is inherently a *kinetic* boundary condition that resulted from a fluid model. Thus, in the interest of accurate simulations, the sheath edge can only be properly tracked by computational models that address the kinetic description of plasmas. Further, the presence of a boundary complicates parameter profiles given the physical discontinuity; a detail that cannot be adequately captured by fluid models (figure 2.8).

Finally, the vastness of scales present in the edge increases the computational burden (figures 2.9, tables
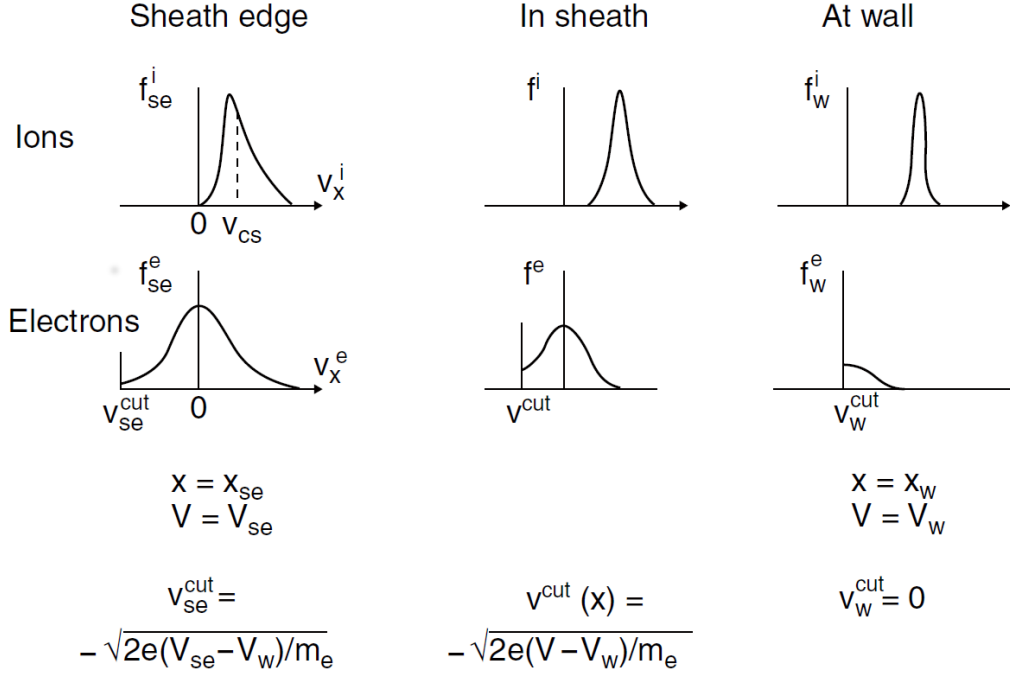
Figure 2.8: Distributions for the charged particle species at various positions near the wall [58]

2.1 and 2.2). Fluid models cannot capture the physics contained over different velocity regimes (although so-called *kinetic corrections* are applied to capture these effects to some extent). A kinetic treatment is most apt to apprehend these matters directly, especially given the significant role played by the population of charged particles in the high energy tail of the distribution function. Pursuing high order accurate solutions is ultimately constrained by the practical computational cost that can be afforded. As will be argued in a later section, this necessarily motivates methods that have significant finesse so that computational resources can be used as effectively as possible. Thus, a deterministic solution is advocated as more advantageous versus statistical methods given the latter requires simulating large numbers of events in order to capture sparse regions of phase space $(\vec{x}, \vec{v})$. The scales span orders of magnitude in densities, lengths, energies, and time scales given the edge region is the transition from the main core to the material boundary. Further, the parameters in the edge can differ significantly depending on one shot to the next, or one *device* to the next. Table 2.2 juxtaposes edge parameters for the Joint European Torus (JET) and Alcator C-mod at Massachusetts Institute of Technology, which emphasizes the large differences possible. For example, it is commonly argued that high collisionality validates fluid approaches; this table shows in particular that the edge is not necessarily collisional (cf. the normalized collision frequencies ν_e^* as well as the mean free paths $\lambda_{ee}, \lambda_{ii}$ between the two devices), and that a fluid approach might not capture as much of the required physics as hoped. In consulting this table, we also note the stark difference between these parameters as compared to the core. For example, typical core densities (including those designed for ITER) are an order of magnitude larger $(10^{20} - 10^{21}\,\mathrm{m}^{-3})$ than predicted at the edge $(10^{19} - 10^{20}\,\mathrm{m}^{-3})$, and that the temperatures in the core (keV) are at least three orders of magnitude as compared to those found in the edge (eV). Additional parameters were given for four devices in table 2.1.

2.2 Survey of computational schemes

In this section we present a brief overview of a selection of alternate schemes applied to the solution of equations in the same family as the Boltzmann-Maxwell set. The two general categories are statistical
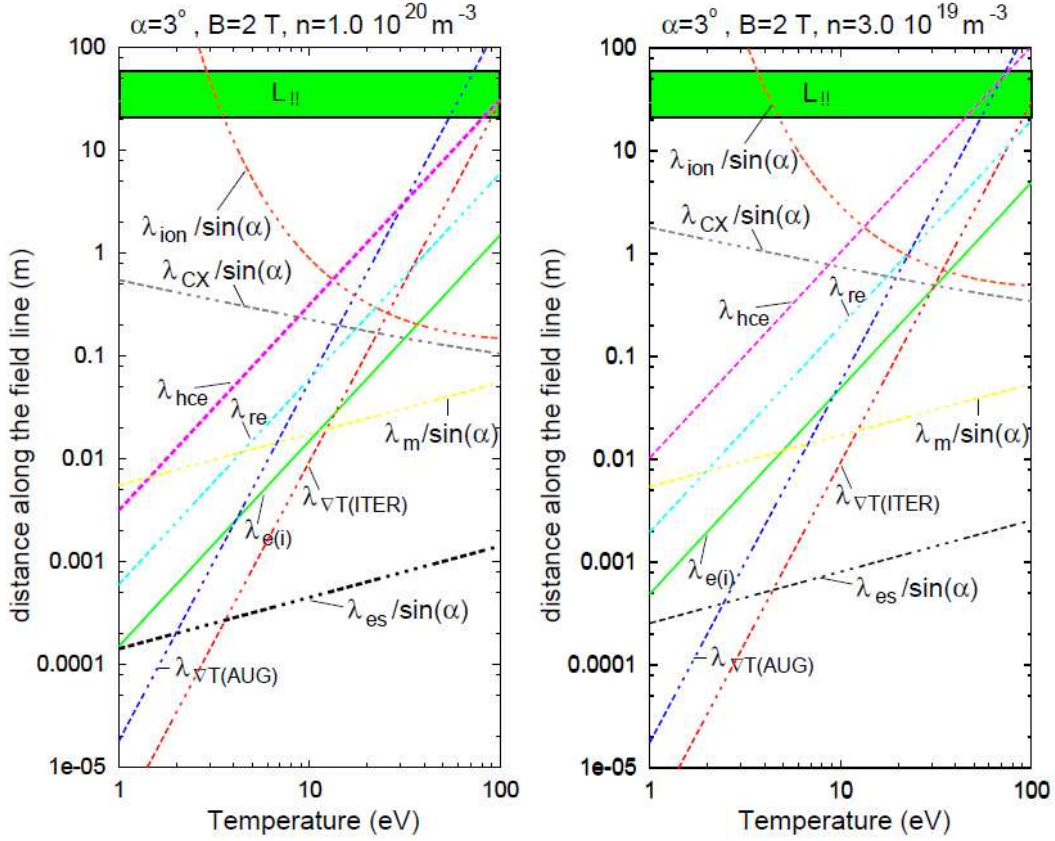
Figure 2.9: Length scales in the edge for a pitch angle of the $\vec{B}$-field of $\alpha = 3°$.

solutions and deterministic solutions.

2.2.1   Statistical solutions

Statistical methods (Monte Carlo and particle-in-cell) converge on the distribution function statistically by tracking particle trajectories as integrations of the equations of motion. Particle-in-cell (PIC) techniques constitute special cases of particle-mesh (PM) methods in that interactions of particles occur through averaged fields on a fixed mesh. The distribution function is discretized into shaped clouds (or macro*particles*) occupying *cells* in phase space that are evolved in time on a moving (Lagrangian) grid according to electromagnetic fields which are described on a fixed (Eulerian) mesh. The general method is built on the flexibility for arbitrary shapes ascribed to each cell through the implementation of a chosen normalized shape function with compact support, which furnishes the distinct nomenclature: *particle*-in-cell for Dirac delta functions, or *cloud*-in-cell for arbitrary shapes, e.g. *b-splines* [36]. Given that particle pushing occurs as Lagrangian trajectories, the method inherits the advantage of having no time step restrictions (e.g. Courant-Friedrichs-Lewy [CFL] condition) that would otherwise need to be addressed in other methods to ensure stability and accuracy, such as pure Eulerian methods (cf. finite difference and finite volume methods, below). Many well-known PIC methods conserve momentum, but in doing so sacrifice maintaining energy conservation [3, 30], which presents a trepidatious pathway for the solution of plasma systems whose behavior is often hand-in-hand with instabilities driven by energy exchanges and injections. In light of this concern, PIC methods that aimed to conserve total energy were first introduced by Lewis in the 1970s [39], and recently Markidis et. al have proposed a method that conserves it exactly [43]. The challenge in these methods is to simulate perhaps $10^5 \sim 10^6$ particles when there physically

	JET	Alcator C-mod
Parameters		
electron density n_e [m^{-3}]	10^{19}	$10^{20} - 10^{21}$
T_e [eV]	50	10
Connection length L [m]	40	8
$\nu_e^* = \nu_e/\nu_{e,bounce}$	25	1000
self-collisional mean free paths $\lambda_{ee}, \lambda_{ii}$ [m]	2.5	0.01
SOL dwell time $\tau_{SOL} = L/c_s$ [ms]	0.6	0.3

Table 2.2: Parameters in the scrape-off layer (SOL) for two tokamak devices. Reproduced from [58, p.21].

might be greater than 10^7 particles in a Debye sphere while retaining the essential physics. The aforementioned modeling of particles as macroparticles provides the workaround for this goal to be accomplished. In Monte Carlo, a random number generator is used to simulate the results of collision events. Convergence is obtained when a sufficient number of particles have been simulated. In order to resolve sparser regions of phase space, the burden of obtaining good statistics is increased and the computational expense is a factor. In plasma systems, where the high energy tail is such a significant controller of the physics, statistical methods are met with substantial challenge.

2.2.2 Deterministic solutions

The other category of solvers are direct solutions of the kinetic equation. These methods include Eulerian, Lagrangian, and semi-Lagrangian methods. Developing high order methods is highly specific to each scheme, as is the computational expense involved.

Eulerian methods include finite difference and finite volume methods constitute purely Eulerian approaches to the solution of the kinetic equation. The phase space grid is discretized, and partial differential equations are represented as point-wise differences whose forms are arrived at from Taylor expansions. It is seen that whether the method is *explicit* versus *implicit* with respect to the time-marching scheme greatly affects the stability of the numerical solution. In particular, explicit schemes (e.g. Lax-Wendroff) are subject to a Courant-Friedrichs-Lewy (CFL) condition whereas implicit schemes can often be constructed such that either more relaxed criteria arise or otherwise are unconditionally stable (e.g. Crank-Nicolson).

Finite volume methods are finite difference schemes applied to the conservative form of a given partial differential equation whereupon integration of the PDE permits replacing state variables with either volume-averaged or surface-averaged flux quantities at the cell centers or faces, respectively, by means of the divergence theorem. In this way, the tracked quantities are cell-centered volume averages in phase space that are related to specified or calculated flux-surface averages on the cell boundaries. An inherent advantage of such a scheme is evident in that the presence of flux terms specified on boundaries directly admits the result that global conservation laws are automatically satisfied (e.g. conservation of mass). An additional benefit is that the values that need to be specified are reduced; the grouped quantity of a flux is specified instead of multiple state variables individually. In this way, overspecification of boundary conditions is of less risk. The flux values on boundaries can also be naturally incorporated for problems that present either physical or transport boundaries. An anchor in these methods is that the stability is ensured only for certain values of the Courant-Friedrichs-Lewy (CFL) parameter. This results in severe time restrictions with fine phase space meshes. While these

schemes are straightforward to implement, there are a limited number of options to ameliorate or to mitigate this restriction so the methods can often be costly. The most developed of these methods is the weighted essentially non-oscillatory (WENO) schemes which is built on an interpolation method to calculate in-between values on a grid. The interpolation uses well-defined weights in a convex combination of interpolations of the same point using stencils involving a smaller number of points to create the effect of higher order. The weights that are obtained are then seen to dampen numerical oscillations in the solutions.

Lagrangian methods find the solution by following particles in phase space; however, pursuing such a method requires discerning the Green's function (propagator or kernel) for a particular problem, which may not be obtainable. Otherwise, the method can require some care to ameliorate the shapes of evolved Lagrangian cells, which can contort or otherwise filament. This filmanetation can introduce unwanted physics through means such as artificial (numerical) dissipation. However, the benefit of such methods is that they inherently have no CFL limit given the solution is solved in a frame moving with the state variables.

The semi-Lagrangian approach is a hybrid method that retains the advantage of having no time step restriction by convecting solutions in a Lagrangian mesh, but also evades the requirement of having a Green's function by working with a second, Eulerian, grid. One such scheme is the discontinuous Galerkin (DG) method, which partitions phase space into a broken finite element space, and assigns local basis functions (e.g. Legendre polynomials) with compact support that give a continous form to the cell-centered values over the extent of the cell. The method seeks the solution to the variational problem by finding the solution which minimizes the averaged error. In pursuit of this, a set of matrix equations are arrived at whose solution gives the result after each time step. However, DG methods have not been extended to obtain higher order solutions, and it is difficult to implement.

The final example of a semi-Lagrangian scheme is the method employed in this body of work. The convected scheme is a forward-trajectory method of characteristics solution. Particles are evolved on a Lagrangian grid along their characteristics, and remapped to the Eulerian grid for field calculations and updating the inventory of the distribution function. This method will be developed and presented in full in subsequent sections. The benefits of this method are clear: ease of implementation, able to be made higher order, and computational efficiency.

2.3 The Boltzmann-Maxwell system

A plasma system is described by the Boltzmann-Maxwell, which describes the evolution of the distribution function f_α for a particle species α in phase space $(\vec{x}, \vec{v})$ over time $(t \in \mathbb{R}^+)$:

$$f_\alpha = f_\alpha(t, \vec{x}, \vec{v}) \; : \; \mathbb{R}^+ \times \mathbb{R}^{d_x} \times \mathbb{R}^{d_v} \mapsto \mathbb{R}$$

where the dimension $d_x, d_v = \{1, 2, 3\}$ considered in configurational and velocity spaces need not be identical, and we regard the direct product above as acting over the sets which make up the domain in each variable of phase space. At times, it proves convenient to speak of (d_x+d_v)-dimensional phase space $\Omega = \mathbb{R}^{d_x} \times \mathbb{R}^{d_v}$. The zeroeth and first velocity moments of the distribution function furnish definitions for the number density $n_\alpha(t, \vec{x})$, as well as the charge and current density, $\rho(t, \vec{x})$ and $\vec{J}(t, \vec{x})$, respectively.

$$n_\alpha(t, \vec{x}) \; := \; \int f_\alpha(t, \vec{x}, \vec{v}) d^3\vec{v} \tag{2.3}$$

$$\rho(t, \vec{x}) \; := \; \sum_\alpha q_\alpha \langle n_\alpha \rangle \;\; = \;\; \sum_\alpha q_\alpha \int f_\alpha(t, \vec{x}, \vec{v}) d^3\vec{v} \tag{2.4}$$

$$\vec{J}(t, \vec{x}) \; := \; \sum_\alpha q_\alpha \langle n_\alpha v_\alpha \rangle \;\; = \;\; \sum_\alpha q_\alpha \int \vec{v} f_\alpha(t, \vec{x}, \vec{v}) d^3\vec{v} \tag{2.5}$$

These quantities, involving the distribution function f_α, appear in the Maxwell equations as well as the Boltzmann equation, altogether constituting a nonlinear set of coupled integro-partial differential equations:

$\boxed{\text{Boltzmann-Maxwell system}}$

$$\frac{\partial f_\alpha}{\partial t} + \vec{v} \cdot \frac{\partial f_\alpha}{\partial \vec{x}} + \frac{q_\alpha}{m_\alpha}\left(\vec{E} + \vec{v} \times \vec{B}\right) \cdot \frac{\partial f_\alpha}{\partial \vec{v}} = \left(\frac{\partial f_\alpha}{\partial t}\right)_{coll} \tag{2.6}$$

$$\vec{\nabla} \cdot \vec{E} = \frac{\rho}{\epsilon_0} \quad , \quad \vec{\nabla} \times \vec{B} = \mu_0 \vec{J} + \frac{1}{c^2}\frac{\partial \vec{E}}{\partial t} \tag{2.7}$$

$$\vec{\nabla} \times \vec{E} = -\frac{\partial \vec{B}}{\partial t} \quad , \quad \vec{\nabla} \cdot \vec{B} = 0 \tag{2.8}$$

for a plasma species of charge and mass q_α and m_α, respectively . The speed of light $c = 1/\sqrt{\mu_0 \epsilon_0}$ is given in terms of permeability (μ_0) and permittivity (ϵ_0) of free space. It is noted that the time-dependent electric field $\vec{E} = \vec{E}(t, \vec{x})$ and the magnetic induction $\vec{B} = \vec{B}(t, \vec{x})$ depend on the position in configurational space, which appear in the Lorentz force term in the *Boltzmann equation* (2.6). The namesake attributed to Boltzmann follows from statistical mechanics, whereas in plasma physics literature the same equation is arrived at through different pathways (e.g. BBGKY hierachy) and is known as the *plasma kinetic equation*. The inhomogeneity on the right-hand side of the Boltzmann equation captures the point-wise change in the distribution function due to collisions, known as the *collision integral*, whereas the *collision operator* refers to the entity acting on f_α. The details of the collision operator depend on both the considered physics and the level of detail treated. While technically distinct, the inhomogenity is often referred to as simply the collision operator.

Addressing the full Boltzmann-Maxwell system is expected to be beyond the scope of the intended research; however, we aim to establish a robust foundation that can be built upon in subsequent endeavors to approach the solution to this system in steps. As a first step we put forth in this preliminary work a case where the effect of collisions ($\partial_t f_\alpha|_{coll} = 0$) are not included and address particles that experience zero acceleration (i.e. $\vec{a} \equiv q_\alpha/m_\alpha(\vec{E} + \vec{v} \times \vec{B}) = 0$). The neglect of force fields decouples the system, leaving the governing equation for the distribution function as only the Boltzmann equation which suffers a significant reduction given the aforementioned assumptions. This result is recognized as the *advection equation*, and high order solutions to this equation are the building blocks in this work used to approach the handling of more complicated physics (e.g. Vlasov-Poisson system).

$\boxed{\text{Advection equation}}$

$$\frac{\partial f_\alpha}{\partial t} + \vec{v} \cdot \frac{\partial f_\alpha}{\partial \vec{x}} = 0 \tag{2.9}$$

The next order of increasing complexity to be apprehended in future work (Chapter 4) is the single species electrostatics case ($\vec{B} \equiv 0$). Here, the neglect of collisions and the magnetic induction reduces the Boltzmann equation (2.6) to the *Vlasov equation* (2.10a) below. Under electrostatics, Faraday's law (2.8) permits the introduction of a scalar potential $\vec{E}(t, \vec{x}) = -\vec{\nabla}\phi(t, \vec{x})$ such that $\vec{\nabla} \times \vec{E} = \vec{\nabla} \times (-\vec{\nabla}\phi) \equiv 0$ is satisfied by construction, whereas Gauss' law (2.7) in terms of this potential takes the form of *Poisson's equation* (2.10b). Thus, the coupled set of equations is known as the *Vlasov-Poisson system*:

Vlasov-Poisson system

$$\frac{\partial f_\alpha}{\partial t} + \vec{v} \cdot \frac{\partial f_\alpha}{\partial \vec{x}} + \frac{q}{m}\vec{E} \cdot \frac{\partial f_\alpha}{\partial \vec{v}} = 0 \tag{2.10a}$$

$$-\nabla^2 \phi = \frac{\rho}{\epsilon_0} \tag{2.10b}$$

The Vlasov equation in itself is a hyperbolic equation that evolves the distribution function in time, whereas Poisson's equation is an elliptic partial differential equation whose calculation depends on instantaneous values of its first moment. Plasma particles subject to these coupled equations interact only through the summative influence of long-ranged electrostatic forces generated from all charged particles in the system. The consequence of omitting the collisional term restricts the time scale of physical significance. That is, the Vlasov equation can only describe plasmas whose time scales of interest are shorter than the collision time ($\omega \gg \nu_{coll}$).

The family of Boltzmann equations have their foundations in the mathematics of Lie groups, which provides the framework needed to assemble and manipulate objects such as the Hamiltonian and develop Hamilton's equations. Because the backbone of the Boltzmann equation is the Hamiltonian, it shares many elegant properties associated Hamiltonian systems (e.g. symplecticity). Further, in the pursuit of numerical solutions to these equations, we will see this requires forming mathematical objects (exponentiated differential operators) that can only be understood as part of the Lie algebraic language. Thus, it is worthwhile to discuss the foundations of these equations in the higher abstraction of their equipped mathematics, so that when these objects are encountered we may proceed with the comfort in understanding they are well-defined and understood.

2.3.1 Lie groups and Lie algebras

In the same way that analytical mechanics is based on earlier developments in the theory of differential equations, and that general relativity was built under the framework of non-Euclidean differential geometry, so does the Hamiltonian formulation of classical mechanics find natural foothold in its own relevant area of mathematics. Specifically, the Boltzmann differential equation acts on state variables to produce an associative group of elements whose algebra is such that group elements are specified by one (or more) continuous actions on variable(s). These relationships among group elements are special in that the functions themselves are necessarily smooth and differentiable. These requirements fit under the Lie group. The Lie group is such that its group operations are rooted in both conventional algebra and have the flexibility to encompass notions of calculus; its group operations are compatible with a smooth structure. In fact, Sophus Lie first introduced them as "infinitesimal groups." The problem domain Ω sits on a $C^\infty(\Omega)$ ("smooth") manifold and is equipped with a product which obeys a Jacobi identity. As mentioned in the previous section, understanding that our foundation starts in Lie groups (which, in turn, may be analyzed using a Lie algebra) will enable us to form and work with robust and powerful operators to evolve our problem in time simply. These objects show up in the development of operator splitting methods (section 2.6.1). Further, the tools and forms available in Lie algebra will make the intrinsic characteristics of the Hamiltonian system more obvious, and provide the validation that the objects we arrive at are well-defined and that its properties are adequately characterized.

Often, the phase space of the problem domain Ω is the cotangent bundle $T^*\mathbb{R}^n$ of the configuration space $\mathbb{R}^n$ (e.g. when $\vec{B} \neq 0$) where $n = \{1, 2, 3\}$. In the force-free cases or otherwise when a force is conservative (e.g. electrostatics), the canonical momentum in the Hamiltonian formulation is the classical particle momentum which is always tangent to the trajectory of a particle in configurational space. In such cases, the phase space is the tangent bundle, $T\mathbb{R}^n$. Both possibilities are in the same dual space so that the phase space of the problem domain is the cohomology $\mathrm{Hom}(T\mathbb{R}^n, T^*\mathbb{R}^n)$. In general, the only requirement is that the space itself

is homeomorphic to the Euclidean space. In this way, the appropriate abstraction is a differentiable topological space on $C^\infty(\Omega)$, i.e. a *smooth manifold* $\mathcal{M}$.

The definition of a (co)tangent bundle is motivated as follows. Consider a vector field X on an open subset $U \subset \tilde{\mathcal{M}}$ as a map that assigns to each point $p \in U$ a tangent vector $X(p)$. The set of all possible tangent vectors through a point p on this subset of the manifold $\tilde{\mathcal{M}}$ forms a tangent space $T_p\tilde{\mathcal{M}}$. We can alternatively define the tangent space at p by using the notion of the ordinary derivative in an embedding in Euclidean space, furnished by any chart $\varphi \colon U \to \mathbb{R}^n$. Representing all such curves $\gamma_i(s)$ $(i \in \mathbb{N})$ for a suitable parametrization s in the subspace U that passes through p (typically defined such that $\gamma_i(0) = p$ for convenience), we operate with the ordinary derivative $\frac{d}{ds}(\varphi \circ \gamma_i)(0)$ to establish an equivalence relation with a unique tangent vector $X_i(p) \equiv X(p)$ as mapped onto the Euclidean space. Note, the collection of tangent vectors obtained does not depend on the choice of chart φ, and it is obvious that any such chart is bijective given the manifold is by definition point-wise homeomorphic to the Euclidean space. Thus, an inverse operation can be used to transfer the vector space in $\mathbb{R}^n$ back onto the manifold $\tilde{\mathcal{M}}$, in total assembling the same tangent space $T_p\tilde{\mathcal{M}}$.

The disjoint union of all such spaces defined at each $p \in \tilde{\mathcal{M}}$ $(\dim(\tilde{\mathcal{M}}) = \dim(T_p\tilde{\mathcal{M}}) = n)$ defines the $2n$ dimensional tangent bundle $T\tilde{\mathcal{M}} = \bigcup_{p \in \tilde{\mathcal{M}}}(p, T_p\tilde{\mathcal{M}})$. Alternatively, the set of all linear functionals spanning the vector space constitutes the dual bundle $T^*\tilde{\mathcal{M}} = (T\tilde{\mathcal{M}})^*$. The cohomology of both is a $2n$-dimensional manifold $\mathcal{M} = \mathrm{Hom}(T\mathbb{R}^n, T^*\mathbb{R}^n)$ that constitutes the phase space . An integral curve that constitutes a local solution to Hamilton's equations is such a curve $\gamma \colon t \mapsto x(t)$ which passes through a point in the configuration x (i.e. $\varphi \circ \gamma(0) = x$) whose derivative $\frac{dx(t)}{dt} = X(x(t))$ where X is an element of the bundle for all time t. In this sense, we establish a correspondence of vectors (resp. covectors) with the derivatives which correspond to momentum (resp. canonical momentum) so that ordinary differential equations (each one-form) on $\mathcal{M}$ are point-wise correspondent to vector fields on $\mathcal{M}$. That is, the geometric manifestation of the solution space for the Hamiltonian system *is* this (symplectic) manifold $\mathcal{M}$, which geometrically gives the state values, i.e. each unique trajectory ((co)tangent vector defining the canonical momentum value) at each position in configurational space.

In other words, understanding that the derivatives may be uniquely assigned to tangent vectors on the surface of the Cartesian product space $\mathbb{R}^n$ (or, again, in general a manifold $\mathcal{M}$), the set of Hamilton's (differential) equations can be viewed as specifying the trajectories of phase space coordinates on the space spanned by the (co)tangent bundle of a manifold $\tilde{\mathcal{M}}$. The solution space of the density function f_α lies on the $2n$-dimensional *differentiable* manifold $\mathcal{M}$, which is traced out by integral curves of a Hamiltonian vector field X_H defined as $H \mapsto X_H \colon \omega(X_H, \cdot) = i_{X_H}\omega = -dH$, where d is the exterior derivative and i_{X_H} is the contraction of the derivation of the differential two-form ω with the field X_H. The differential two-form ω allows a pairing between vectors and covectors whose action is to generate the Hamiltonian vector field X_H of which the integral curves give the trajectories of the phase space coordinates.

The vector field in this representation is equivalent to a coordinate-free version of Hamilton's equation. The smoothness of the manifold on which the solution sits suggests a natural Lie group representation (a manifold in C^∞ obeying group properties and operators which communicate the meaning of infinitesimal, smooth, changes). In Vlasov-Poisson, we understand the derivatives and the momentum correspond to the tangent bundle of the Hamiltonian manifold, whereas when the magnetic field is incorporated the generalized conjugate momentum is no longer tangent, but cotangent. Thus, the space corresponds to the cotangent bundle, which is notwithstanding part of the vector space dual $T^*\mathbb{R}^n = (T\mathbb{R}^n)^*$. Recall that $T\mathbb{R}^n \cong T^*\mathbb{R}^n \cong \mathbb{R}^{2n}$ where $\mathbb{R}^{2n}$ is the trivial manifold, but where we emphasize with the $\cong$ operator to show that each space is isomorphic to another, i.e. the tangent/cotangent bundles are also manifolds as these isomorphisms show these vector fields are homeomorphic to the Euclidean space which is the defining feature of a manifold. We may consider the governing equations in terms of the cotangent bundle in all generality of the manifold pertaining to the integral curves of the Hamiltonian vector field. This realization invites the equipment with a Lie algebra and designates the general Lie group as the module from which we operate. In fact, a Lie group G equipped with

an algebra $\mathfrak{g}$ on $\mathcal{M}$ is defined as the tangent space at identity, id, that is $\mathfrak{g} \coloneqq T_{\mathrm{id}}G$, where id $\in G$. Each element in $\mathfrak{g}$ represents an infinitesimal displacement away from id in G, and since id $\in G$ acts as an identity map to $\mathcal{M}$, this amounts to differential measures in $\mathcal{M}$; That is, this algebra describes exactly a vector field. Thus, we see that the Lie algebra completely characterizes the vector field X_H of the Hamiltonian phase space.

2.3.2   The connection between Hamilton's equations and the Boltzmann equation

The equations of motion for a system can be interpreted as a consequence of Hamilton's principle of stationarity. That is, minimizing the action functional of a scalar surrogate known as the Lagrangian L amounts to satisfying conditions called Euler-Lagrange equations which produce the on shell equations of motion in terms of the quantity $L = L(t, q, \dot{q})$. Here, the configuration variable is q and the overdot indicates its time derivative (the physical velocity). Hence, it is obvious that the Lagrangian always acts on the tangent bundle of the configuration space. The Hamiltonian is the Legendre (involutive) transform of the Lagrangian, where the velocity variable $\dot{q}$ is traded for generalized momentum p. An equivalent set of conditions to the Euler-Lagrange equations can be translated in terms of the Hamiltonian to give Hamilton's equations. By examining the time evolution of a defined density function F_α in terms of Hamiltonian phase space, the governing equation for the distribution function f_α can be found by a straightforward change of variables and is completed by invoking Hamilton's equations. Thus, Hamilton's equations produce the Boltzmann transport equation, which will bridge our later focus of building symplectic integrators (section 2.6.1) based on Hamilton's equations and the overarching goal of deterministically solving the kinetic equation for a plasma system. In most introductory texts, the Boltzmann equation is often arrived at from the Bogoliubov-Born-Green-Kirkwood-Yvon (BBGKY) hierarchy starting from the Klimontovich-Dupree equation. This section presents a derivation of the Boltzmann equation through Hamilton's equations [46] directly to show the connection of the basis for focusing our numerical efforts on properties of Hamiltonian systems.

Defining the canonical momentum $p = \partial L / \partial \dot{q}$, the Hamiltonian $H = H(q, p)$ of an autonomous system along with Hamilton's equations (2.11) describe particle trajectories:

$$\dot{q}_i = \frac{\partial H(q_j, p_j, t)}{\partial p_i}, \qquad \dot{p}_i = -\frac{\partial H(q_j, p_j, t)}{\partial q_i} \tag{2.11}$$

The total number of particles dN contained within a phase volume $d^3\vec{q}\,d^3\vec{p}$ furnishes the definition of a density function F, i.e. $dN = F(q_i, p_i, t)d^3\vec{q}\,d^3\vec{p}$. In the absence of collisions the particle number is conserved, so that the total derivative is a stationary point in phase space:

$$\frac{dF}{dt} = \frac{\partial F}{\partial t} + \sum_i \left(\frac{\partial F}{\partial q_i}\dot{q}_i + \frac{\partial F}{\partial p_i}\dot{p}_i \right) = \frac{\partial F}{\partial t} + \sum_i \left(\frac{\partial F}{\partial q_i}\frac{\partial H}{\partial p_i} - \frac{\partial F}{\partial p_i}\frac{\partial H}{\partial q_i} \right) = 0$$

by Hamilton's equations (2.11). including the effect of collisions amounts to including an inhomogeones term that captures the point-wise change in the distribution function with respect to time,

$$\frac{\partial F}{\partial t} + \sum_i \left(\frac{\partial F}{\partial q_i}\frac{\partial H}{\partial p_i} - \frac{\partial F}{\partial p_i}\frac{\partial H}{\partial q_i} \right) = \left(\frac{\partial F}{\partial t} \right)_{coll} \tag{2.12}$$

Recalling the Boltzmann equation (2.6) is defined in terms of the distribution function $f = f(\vec{x}, \vec{v}, t)$, we aim to prove through a straightforward change of variables $F(\vec{q}, \vec{p}, t) \to f(\vec{x}, \vec{v}, t)$, the above equation (2.12) is given by the Boltzmann equation (2.6), which is repeated here for convenience for a magnetized plasma:

$$\frac{\partial f}{\partial t} + \vec{v} \cdot \frac{\partial f}{\partial \vec{x}} + \frac{q}{m}\left(\vec{E} + \vec{v} \times \vec{B} \right) \cdot \frac{\partial f}{\partial \vec{v}} = \left(\frac{\partial f}{\partial t} \right)_{coll} \tag{2.13}$$

The Hamiltonian and its associate canonical momentum p_i $(i = \{1, 2, 3\})$ is given, as usual, by:

$$
\begin{aligned}
H &= \frac{1}{2m}(\vec{p} - q\vec{A})^2 + q\phi \\
p_i &= mv_i + qA_i \\
q_i &= x_i \\
\dot{x}_i &= v_i = \frac{\partial H}{\partial p_i} \\
\dot{p}_i &= -\frac{\partial H}{\partial x_i} = \sum_k \frac{(p_k - qA_k)}{m} q \frac{\partial A_k}{\partial x_i} - q \frac{\partial \phi}{\partial q_i} = q \left(\sum_k v_k \frac{\partial A_k}{\partial x_i} - \frac{\partial \phi}{\partial x_i} \right)
\end{aligned}
$$

Inserting the above definitions into (2.12), we have:

$$
\frac{\partial F}{\partial t} + \sum_i \left(\frac{\partial F}{\partial q_i} \frac{\partial H}{\partial p_i} - \frac{\partial F}{\partial p_i} \frac{\partial H}{\partial q_i} \right) = \left(\frac{\partial F}{\partial t} \right)_{coll}
$$

$$
\frac{\partial F}{\partial t} + \sum_i \frac{\partial F}{\partial x_i} v_i - \sum_i \frac{\partial F}{\partial p_i} q \left(\sum_k v_k \frac{\partial A_k}{\partial x_i} - \frac{\partial \phi}{\partial x_i} \right) = \left(\frac{\partial F}{\partial t} \right)_{coll} \tag{2.14}
$$

Now, noting that the density $F(q_i, p_i, t) = dN/(d^3\vec{q}d^3\vec{p}) = [\mathrm{L}^{-3} \cdot (\mathrm{ML/T})^{-3}]$, where the dimensional quantities are L = length, T = time, M = mass, we understand $f(x_j, v_j, t) = dN/(d^3\vec{x}d^3\vec{v}) = [\mathrm{L}^{-3}(\mathrm{L/T})^{-3}] = F(x_i, p_i, t)/m^3$. We can formally prove this through a change of variables facilitated by the relevant Jacobian. That is, the particle number dN within the infinitesimal phase volume is equal independent of if we choose to measure in terms of F or f. That is,

$$
\begin{aligned}
dN &= dN \\
F(x_i, p_i, t) d^3\vec{x}d^3\vec{p} &= f(x_j, v_j, t) d^3\vec{x}d^3\vec{v} \\
F(x_i, p_i, t) d^3\vec{x}d^3\vec{p} &= f(x_j, v_j, t)|\mathcal{J}| d^3\vec{x}d^3\vec{p}
\end{aligned} \tag{2.15}
$$

where the Jacobian

$$
\mathcal{J} = \frac{\partial(x_1, x_2, x_3, v_1, v_2, v_3)}{\partial(x_1, x_2, x_3, p_1, p_2, p_3)} =
\begin{vmatrix}
\frac{\partial x_1}{\partial x_1} & \frac{\partial x_1}{\partial x_2} & \frac{\partial x_1}{\partial x_3} & \frac{\partial x_1}{\partial p_1} & \frac{\partial x_1}{\partial p_2} & \frac{\partial x_1}{\partial p_3} \\
\frac{\partial x_2}{\partial x_1} & \cdots & \cdots & \cdots & \cdots & \frac{\partial x_2}{\partial p_3} \\
\frac{\partial x_3}{\partial x_1} & \vdots & \ddots & & & \frac{\partial x_3}{\partial p_3} \\
\frac{\partial v_1}{\partial x_1} & \vdots & & \ddots & & \frac{\partial v_1}{\partial p_3} \\
\frac{\partial v_2}{\partial x_1} & \vdots & & & \ddots & \frac{\partial v_2}{\partial p_3} \\
\frac{\partial v_3}{\partial x_1} & \frac{\partial v_3}{\partial x_2} & \frac{\partial v_3}{\partial x_3} & \frac{\partial v_3}{\partial p_1} & \frac{\partial v_3}{\partial p_2} & \frac{\partial v_3}{\partial p_3}
\end{vmatrix}
$$

It is easy to see from the definitions listed above (2.14) that

$$
\begin{aligned}
\frac{\partial x_i}{\partial x_j} &= \delta_{ij} \\
\frac{\partial v_j}{\partial x_i} &= 0
\end{aligned}
$$

29

$$\frac{\partial x_j}{\partial p_i} = \left(\frac{\partial p_i}{\partial x_j}\right)^{-1} = \left[\frac{\partial(mv_i + qA_i)}{\partial x_j}\right]^{-1} = \left(q\frac{\partial A_i}{\partial x_j}\right)^{-1} = \frac{1}{q}\left(\frac{\partial A_i}{\partial x_j}\right)^{-1}$$

$$\frac{\partial v_i}{\partial p_j} = \left(\frac{\partial p_i}{\partial v_j}\right)^{-1} = \left[\frac{\partial(mv_i + qA_i)}{\partial v_j}\right]^{-1} = \left(m\frac{\partial v_i}{\partial v_j}\right)^{-1} = \frac{1}{m}\delta_{ij}$$

so the 6×6 Jacobian matrix reduces to the triangular block matrix

$$\mathcal{J} = \begin{pmatrix} \boldsymbol{I}_{3\times 3} & \boldsymbol{\Lambda}_{3\times 3} \\ \boldsymbol{0}_{3\times 3} & \frac{1}{m}\boldsymbol{I}_{3\times 3} \end{pmatrix}$$

Where the block matrices have the subscripted dimensions and $\boldsymbol{I}$ and $\boldsymbol{0}$ denote the identity and zero matrices, respectively. The matrix $\boldsymbol{\Lambda}$ contains the nonzero derivatives corresponding to $\partial x_i/\partial p_j$ as given just above, but ultimately the calculation of which is not necessary as it does not enter the determinant of this *triangular* block matrix, $|\mathcal{J}|$.

$$|\mathcal{J}| = \underbrace{\det(\boldsymbol{I}_{3\times 3})}_{=1}\det(m^{-1}\boldsymbol{I}_{3\times 3}) = m^{-3}\underbrace{\det(\boldsymbol{I}_{3\times 3})}_{=1} = (1/m)^3$$

Then, eq. (2.15) informs $F(x_i, p_i, t)d^3\vec{x}d^3\vec{p} = f(x_j, v_j, t)|\mathcal{J}|d^3\vec{x}d^3\vec{p} = f(x_j, v_j, t)(1/m)^3 d^3\vec{x}d^3\vec{p}$, or equivalently $f(x_j, v_j(x_i, p_i, t), t) = m^3 F(x_i, p_i, t)$, where we explicitly emphasize the dependence of the velocity variable on the Hamiltonian set.

It is then a straightforward matter to multiply eq. (2.14) by m^3 in order to replace $m^3 F = f$, and to calculate the required derivatives via chain rule to fully carry out the changeover $(x_i, p_i, t) \rightarrow (x_i, v_i, t)$ [46, p.249]. Finally, identifying $\vec{\nabla} \times \vec{A} = \vec{B}$ and $\vec{E} = -\vec{\nabla}\phi - \partial\vec{A}/\partial t$, we directly obtain the Boltzmann equation (2.13). *Thus, we see that solutions to Hamilton's equations equivalently give solutions to Boltzmann-like equations.* In particular, this correspondence amounts to solutions of the family of Boltzmann equations inheriting the properties and structure of the Hamiltonian: (*i*) symplecticity of the flow, and (*ii*) conservation of the Hamilton (total energy) in the case of autonomous (time independent) systems.

2.3.3 Important properties of Hamiltonian systems

It is well-known that a numerical integrator cannot be designed to preserve both the symplecticity and the constancy of the Hamiltonian (energy) [64]. Thus, the choice is whether to construct integrators that are symplectic or energy-conservative. Here, the former is pursued. It has been seen that symplectic integrators, that is, integrators that preserve a nondegenerate antisymmetric bilinear differential two-form $\omega = dp \wedge dq$ perform well in long-time integrations. Further, while symplectic integrators cannot conserve energy, it is still possible to design integrators that do not contain a secular increase in energy with time. Such methods are said to be *energy stable*. Two key properties of autonomous Hamiltonian systems are introduced below.

Constancy of the Hamiltonian

For an autonomous N-body system with masses m_i (e.g. a plasma), we have the Hamiltonian

$$H = H(q_1, q_2, \ldots, q_N, p_1, p_2, \ldots, p_N)$$

$$H = \sum_{i=1}^{N} \frac{p_i^2}{2m_i} + V(q_1, q_2, \ldots, q_N)$$

where $q_i \in \mathbb{R}^3$, $p_i \in T_{q_i}^*\mathbb{R}^3$. Analyzing the time derivative of the Hamiltonian shows

$$\frac{dH}{dt} = \sum_{i=1}^{N} \frac{\partial H}{\partial q_i} \dot{q}_i + \frac{\partial H}{\partial p_i} \dot{p}_i = \sum_{i=1}^{N} \frac{\partial H}{\partial q_i} \frac{\partial H}{\partial p_i} + \frac{\partial H}{\partial p_i} \left(-\frac{\partial H}{\partial q_i} \right) = 0$$

Thus, the Hamiltonian is constant is time. This can also be seen by casting Hamilton's equations in terms of the $2N$ square Poisson matrix J:

$$J = \begin{pmatrix} \mathbf{0} & \mathbf{I} \\ -\mathbf{I} & \mathbf{0} \end{pmatrix}$$

where each block matrix above is of dimensions $N \times N$, $\mathbf{I}$ is the identity matrix and $\mathbf{0}$ is the matrix of zeroes. Defining the $6N$-dimensional gradient $\vec{\nabla}_{6N} = \sum_{i=1}^{N} (\vec{\nabla}_{q_i}, \vec{\nabla}_{p_i})$ for the system of N-bodies, and the phase space coordinate vector $\vec{z} = (q_1, q_2, \ldots, q_N, p_1, p_2, \ldots, p_N)$, Hamilton's equations can be written in compact form

$$\dot{\vec{z}} = J\vec{\nabla}_{6N} H \tag{2.16}$$

from which it is obvious that the Hamiltonian is constant as the structure of the Poisson matrix shows the above is in skew-gradient form.

Symplecticity of the Hamiltonian flow

The skew gradient form (2.16) implies the phase volume is preserved as the divergence of the Hamiltonian vector field vanishes

$$\vec{\nabla}_{6N} \cdot \dot{\vec{z}} = I\vec{\nabla}_{6N} \cdot J\vec{\nabla}_{6N} H = \sum_{i} \sum_{j} J_{ij} \frac{\partial^2 H}{\partial z_i \partial z_j} = 0$$

where I is the $2N \times 2N$ identity matrix. Since J is skew-symmetric and the Hessian matrix of the Hamiltonian is symmetric, it is clear the divergence is zero. Thus, the volume integral over phase space $\mathcal{M}$ is constant,

$$\vec{\nabla}_{6N} \cdot \dot{\vec{z}} = 0 \Rightarrow \int_{\mathcal{M}} (\vec{\nabla}_{6N} \cdot \dot{\vec{z}})\, d^{6N}\vec{z} = \int_{\partial\mathcal{M}} \dot{\vec{z}} \cdot d^{6N-1}\vec{z} = 0$$

where we have invoked the n-dimensional divergence theorem for the $(k-1)$-form ω:

$$\int_{\mathcal{M}} d\omega = \int_{\partial\mathcal{M}} \omega$$

since the volume element is nonzero, this requires $d\vec{z}/dt = 0$, or $\int d\vec{z} = \text{const}$. The trajectories $\vec{z}$ evolve according to the Hamiltonian flow $\varphi_t(A)$ where A represents the surface punctured by the bundle of phase space trajectories considered on the manifold $\mathcal{M}$, so that the surface area of the phase space manifold $\partial\mathcal{M}$ are initial conditions for a bundle of trajectories in the neighborhood circumscribed by the integration range A that propagates according to the Hamiltonian flow (recall that points on the phase space manifold correspond to state values of the Hamiltonian system). Then we have [45],

$$\int_{\varphi_t(A)} d\vec{z} = \int_{A} \delta(\varphi_t(\vec{z}) - \vec{z}')d\vec{z}'$$

Taking the limit of a vanishingly small integration volume, we recover the robust result that

$$dz = d\vec{p} \wedge d\vec{q} = \text{constant} \tag{2.17}$$

which states that Hamiltonian systems preserve their basic differential two-form exactly so that the evolution $(q(t), p(t)) \mapsto (q(t+\tau), p(t+\tau))$ is a canonical transformation. It can also be demonstrated [61] by tracking the evolution of this differential phase space volume, i.e. symplecticity requires $|\mathcal{J}| = 1$ in the statement $d\vec{p}_0 \wedge d\vec{q}_0 = |\mathcal{J}| d\vec{p}(t) \wedge d\vec{q}(t)$. It can be shown that

$$\frac{d|\mathcal{J}|}{dt} = |\mathcal{J}|(I\vec{\nabla}_{6N} \cdot J\vec{\nabla}_{6N}H) = 0$$

where the result is recalled that the Hessian matrix D^2H of the Hamiltonian is zero. Thus,

$$\frac{d|\mathcal{J}|}{dt} = 0 \Rightarrow |\mathcal{J}(t)| = |\mathcal{J}(0)| = 1$$

since at the start time $t = 0$, the Jacobian determinant is by definition $|\mathcal{J}| = 1$ which proves (2.17). Integrators that share these properties of the physical system can furnish additional diagnostics on the accuracy of the solution by monitoring these conserved quantities at each time step.

This concludes the first category of this chapter. The remainder focuses on addressing the computational mathematics involved in the numerical solution to these equations.

2.4 Convected scheme solutions to advection equations

The computational method furthered in this research is a forward-trajectory semi-Lagrangian method known as the convected scheme (CS). First, we describe the classic (first order accurate) CS. A corrective approach is then reviewed per Güçlu that establishes a prescription to reduce the numerical diffusion so that the overall method can be made accurate up to arbitrary order N in principal.

2.4.1 Classic convected scheme

The convected scheme (CS) is a conservative, positivity-preserving, second order accurate, forward semi-Lagrangian method. The domain spanned by the independent variables $(\vec{x}, \vec{v})$ of the distribution function f_α is partitioned into a discrete set of cells each allocated with an inventory of particles whose position and velocity correspond to a particular cell-center. The numerical solution implements a time marching scheme that convects initial cells from the fixed (Eulerian) grid along their characteristics; each convected cell is referred to as a *moving cell* (MC). The MCs are subsequently remapped to the fixed Eulerian grid to update trajectory kinematics for all cells which concludes the time step. The terminology arises in that the name ascribed to the collection of MCs is termed a Lagrangian mesh, which are remapped to the Eulerian grid according to a particular rule that is described in section 2.4.2.

To motivate the method in general, we report the Boltzmann equation (2.6) in terms of a generic force term $\vec{F} = \vec{F}(t, \vec{x}, \vec{v})$, and discuss the mesh-based solution of the distribution function $f_\alpha \equiv f$ on a (3+3)-dimensional phase space evolving in time as per [26]:

$$\frac{\partial f}{\partial t} + \vec{v} \cdot \frac{\partial f}{\partial \vec{x}} + \frac{\vec{F}}{m} \cdot \frac{\partial f}{\partial \vec{v}} = \left(\frac{\partial f}{\partial t}\right)_{coll} \tag{2.18}$$

where we have suppressed the label α of particle species for brevity. Equation (2.18) provides an Eulerian description of the evolution of the distribution function. Characteristics are naturally defined by considering the total derivative:

$$\frac{df}{dt} := \frac{\partial f}{\partial t} + \frac{d\vec{x}}{dt} \cdot \frac{\partial f}{\partial \vec{x}} + \frac{d\vec{v}}{dt} \cdot \frac{\partial f}{\partial \vec{v}} \tag{2.19}$$

where we physically identify the acceleration $\vec{a} := \vec{F}/m$. Comparing the definition (2.19) with the left-hand side of equation (2.18) above permits the equivalent formulation to be written down:

$$\frac{df}{dt} = \left(\frac{\partial f}{\partial t}\right)_{coll} \tag{2.20}$$

provided the following definitions are made

$$\frac{d\vec{x}}{dt} = \vec{v} \tag{2.21a}$$

$$\frac{d\vec{v}}{dt} = \frac{1}{m}\vec{F}(t, \vec{x}, \vec{v}) \tag{2.21b}$$

In general, the force term $\vec{F}(t, \vec{x}, \vec{v})$ may be tied to auxillary equations. In the case of the Lorentz force, we have a coupling to the Maxwell field equations (2.7) and (2.8). The semi-Lagrangian stepthrough of the numerical solution of the Boltzmann equation (2.20) be summarized as follows [26]:

1. *Eulerian step*: Collision events affect only particle velocities. This amounts to a point-wise velocity rearrangement in configurational space. That is, we solve

$$\frac{\partial f}{\partial t} = \left(\frac{\partial f}{\partial t}\right)_{coll} \tag{2.22}$$

2. *Lagrangian step*: The particles are advected along their characteristics according to the velocities $\{\vec{v_i}\}$ at locations $\{\vec{x_i}\}$ obtained in step (1) per the mapping prescribed by the characteristics (2.21a) and (2.21b). This is equivalent to integrating the homogeneous version of (2.20):

$$\frac{df}{dt} = 0 \tag{2.23}$$

The convected cells are then remapped to the Eulerian grid according to the CS remapping rule (discussed in section 2.4.2) whereafter the process is repeated for each time step in marching through the entire duration of the simulation.

The classic CS scheme [29] traces characteristics in the full phase space. The first order in space remapping to the Eulerian grid can amount to appreciable numerical diffusion (discussed later, figure 2.12) over longer simulation times. Since the convected scheme follows Lagrangian trajectories, it adheres to no Courant-Friedrichs-Lewy (CFL) constraint. Thus, an inviting pathway to reduce the diffusion is to minimize the number of remaps whenever possible. We note that remapping to the Eulerian grid is only necessary as a means to reappropriate the inventory of particles whose kinematics change, i.e. through collisions or acceleration terms. The implementation then consists of remapping only those updated particles to the Eulerian mesh, while holding onto the remaining (uncollided) densities in the same moving cell for multiple time steps. This leads to the *long-lived moving cells* (LLMC) version of the convected scheme [13, 20]. To this end, it has been shown that an estimate for this diffusion can be calculated by means of modified equation analysis to render the method 4th order [26, p.3294].

What is pursued in this work is an operator splitting strategem (e.g. Strang, section 2.6.2), which is an avenue that has received much attention especially within the past decade as applied to Vlasov-Poisson systems. Such techniques can be used to repurpose this implementation as a stepthrough of two equivalent advection equations: one in configurational space, and one in velocity space [27]. That is, the solution of equation (2.23) can be recast into the same problem of seeking high order solutions to advection equations, as

is the principal foundation developed thus far, at the cost of introducing an error that scales with the time. However, the nature of time splitting methods permits an obvious pathway to arrive at higher order in time methods (section 2.6.1). Thus, it is our aim to couple high order time splitting methods with the high order accurate in space CS to obtain efficient and accurate solutions to the plasma kinetic equation.

2.4.2   Convected scheme remapping rule

By definition, the number of particles $N = \int f(t, \vec{x}, \vec{x}) d^3 \vec{x} d^3 v$ of each cell $C = C(t)$ at any time t is preserved during the advection stage 2 of the above implementation, a statement which follows directly from its integral form.

$$\int_{C(t)} f(t, \vec{x}, \vec{v}) d^3 \vec{x} d^3 \vec{v} = \int_{C(t_0)} f(t_0, \vec{x}_0, \vec{v}_0) d^3 \vec{x}_0 d^3 \vec{v}_0 \qquad (2.24)$$

where the control volume at time $t = t_0$ pertains to a cell in phase space $(\vec{x}_0, \vec{v}_0)$. The definition of the moving cell $C(t)$ is particularly transparent from identity (2.24), which informs it is a cell $C_0 \mapsto C(t)$ of constant particle number which evolves according to the Lagrangian characteristics (2.21a) and (2.21b). In general, we have the freedom to ascribe the shape of the profile f by a suitable choice of basis functions, whose overall form is only mathematically restricted by the requirement of total particle number in the cell $N_{MC} = \int f_{MC}(t, \vec{x}, \vec{v}) d^3 \vec{x} d^3 \vec{v}$. This profile need not have compact support. For example, the effect of MCs whose profile could extend outside of the cell was investigated by Birdsall and Fuss [4] in the development of a *cloud-in-cloud* method, which was an extension to electrostatic plasma systems from a similar approach whose origins lie in meteorological applications.

In this work we are focused only on distribution functions parametrized with compact support such that each f_{MC} representing each MC is only nonzero inside its cell. The effect of choosing higher order moving cell profiles has been used as a keynote to develop methods such as the semi-Lagrangian discontinuous Galerkin (SL-DG) scheme [52], where moving cells are given a differentiable basis structure (e.g. Legendre polynomials), but do not have the requirement of continuity from one cell to the next (figure 2.10).
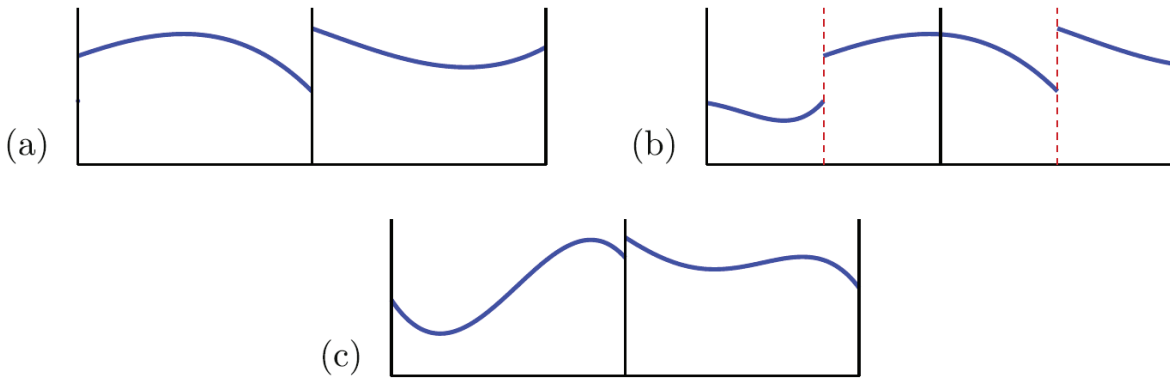
Figure 2.10: In a discontinuous-Galerking (DG) method, cells (finite elements) are ascribed a functional form by chosen basis functions which are weighted by a set of coefficients that describe the density f; there is no enforcement of continuity from one cell to the next. In panel (a), initial data is shown for two cells. Panel (b) illustrates the exact advection of the solution, and panel (c) is the final processed result after a time step where the exact evolution in panel (b) is re-projected back onto the basis in for each cell [52].

After the advection phase, the cells are smoothly remapped through an integral operation over the basis functions.

On the lower order extreme, the simplest parametrization of the profile is an n-dimensional Dirac delta function representation for each of the n dimensions of the considered phase space (zero size particle, ZSP), where the appropriate remapping rule of an MC at its postpoint to the Eulerian grid is the *nearest grid point* (NGP) assignment (the ZSP-NGP scheme). Such a trivialization of the distribution function of each MC leads to significant numerical error and unphysical artifacts such as amplified staircasing (as compared to higher order modeling) in the approximation of smooth curves (e.g. consider a Coulomb potential of a single charged particle) [4].

Here, we take the density of any MC to be uniform in configurational space, which is convected according to a cell-centered velocity assignment so that we can regard the velocity space profile to be made up of a collection of Dirac delta functions. As for the shape of the moving cell itself, it is natural to take the Cartesian product of the sets of values that correspond to those subdomains in both configuration and velocity spaces that span the volume of each cell. That is, we have the direct product of two convex polyhedra [26], which more specifically produces a set of moving cells (Lagrangian mesh) represented as a collection of n-dimensional hypercubes (n-cube, or "measure polytope"), whose union is equal to the problem domain $\mathbb{R}_x^{d_x} \times \mathbb{R}_v^{d_v}$. In graph theory, an n-cube skeleton is denoted by Q_n, where n denotes both the number of vertices in the object and the corresponding dimension. In geometry, we refer to the solid n-cube by a label γ_n. For the constant velocity, one-dimensional case (advection equation), the MC is a line segment (1-cube, γ_1). For the 1D-1V case, a 2-cube (γ_2, rectangle) describes the MC. In n-dimensions, the MC is an n-cube (γ_n).

The remapping rule to the Eulerian grid is executed at the end of a ballistic move (2). To make the remap assignment specific, we take the problem domain in phase space $(\vec{x}, \vec{v})$ to be partitioned into cells C_{ij} labelled by their centroids $(\vec{x}_i, \vec{v}_j)$ such that $\cup C_{ij} = \Omega$ (the formal discrete problem in all detail is presented in section 3.1). The particle number N_{MC} of the moving cells originating from a centroid $(\vec{x}_0, \vec{v}_0)$ are distributed among all such destination cells $\{C_{ij}\}$ in proportion to the phase space volume overlap of the fixed cells with that of the final (exact) location of the MC at the end of a time step, which is generally not an integral shift on the grid (figure 2.11). If we consider the set all of all MCs $C_{i'j'}$ originating from centroids $(\vec{x}_{i'}, \vec{v}_{j'})$, each with a particle number $N_{i'j'}$, we have the update in number N_{ij} for every Eulerian cell C_{ij}:

$$N_{ij} \mathrel{+}= \zeta_{i'j' \to ij} N_{i'j'} \quad : \quad \forall (i,j), (i',j') \in \boldsymbol{N} \oplus \boldsymbol{N}' \tag{2.25}$$

where $0 \le \zeta_{i'j' \to ij} \le 1$ and the grid index domain is given by $\boldsymbol{N} = \boldsymbol{N}' = \mathbb{N}_x \times \mathbb{N}_v$
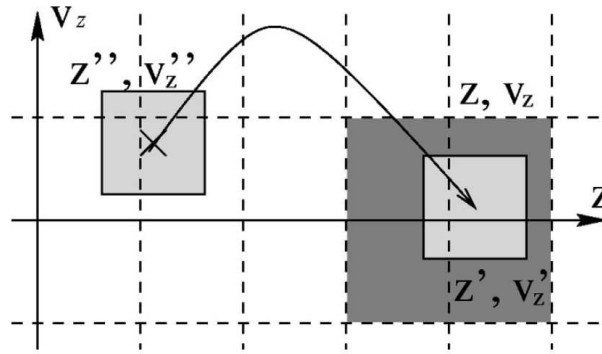
Figure 2.11: A moving cell with (z'', v_z'') is propogated to a final location marked by (z, v_z), which overlaps four cells (dotted outlines) in this 2D phase space. The CS remapping rule assigns the proportion of the MC to each overlapped cell according to its physical area overlap with the grid cells [20, p.3162].

The index sets $\mathbb{N}_x$ and $\mathbb{N}_v$ correspond to the indices enumerating cell-centers $(\vec{x}_i, \vec{v}_j)$ in the phase space $\mathbb{R}_x \times \mathbb{R}_v$ (e.g. $\mathbb{N}_x = \{i \colon \vec{x}_i \in \mathbb{R}_x \text{ for all } i \in \mathbb{N}_0\}$). The object $\mathrel{+}=$ is the increment operator, defined such that $a \mathrel{+}= b$

is interpreted as $a = a + b$. The quantity $\zeta_{i'j' \to ij}$ is the fraction of a moving cell $C_{i'j'}$ appropriated to a fixed cell C_{ij}. In general, this fraction changes from one time step to the next. This fraction $\zeta_{i'j' \to ij}$ is shown just below to correspond to the overlap fraction of each MC with phase space volume $\Gamma_{MC} \equiv \Gamma_{i'j'}$ relative to the fixed cells of volume Γ_{ij}. [4, 26, 29]. This may be seen by writing the mapping statement from one MC, $C_{MC} \equiv C_{i'j'}$, to a single overlapped cell C_{ij} on the Eulerian grid. This rule can then be looped over all MCs to remap the entire Lagrangian mesh to the Eulerian grid.

$$
\begin{aligned}
N_{ij} \quad &+= \quad \int_{C_{ij}} d^3\vec{x}\, d^3\vec{v}\, f_{MC}(t, \vec{x}, \vec{v}) \\
&= \quad f_{MC}(t, \vec{x}_{i'}, \vec{v}_{j'}) \int_{C_{ij} \cap C_{MC}} d^3\vec{x}\, d^3\vec{v} \\
&= \quad f_{MC}(t, \vec{x}_{i'}, \vec{v}_{j'})(\Gamma_{ij} \cap \Gamma_{MC}) \\
N_{ij} \quad &+= \quad \underbrace{f_{MC}(t, \vec{x}_{i'}, \vec{v}_{j'})\Gamma_{MC}}_{N_{i'j'}} \underbrace{\frac{(\Gamma_{ij} \cap \Gamma_{MC})}{\Gamma_{MC}}}_{\zeta_{i'j' \to ij}}, \qquad (i', j') \text{ denotes the MC} \\
N_{ij} \quad &+= \quad \zeta_{i'j' \to ij} N_{i'j'}
\end{aligned}
$$

Which is the same as equation (2.25). The second equality follows from the choice that f_{MC} is constant, and thus the integral results in only the total volume in phase space that overlaps. It is obvious from the definition of $\zeta_{i'j' \to ij}$

$$
\zeta_{i'j' \to ij} = \frac{(\Gamma_{i'j'} \cap \Gamma_{ij})}{\Gamma_{i'j'}}
$$

that each fraction $\zeta_{i'j' \to ij}$ for a given MC $C_{i'j'}$ sum to unity.

$$
\sum_{i \in \mathbb{N}_x} \sum_{j \in \mathbb{N}_v} \zeta_{i'j' \to ij} = \sum_{i \in \mathbb{N}_x} \sum_{j \in \mathbb{N}_v} \frac{(\Gamma_{i'j'} \cap \Gamma_{ij})}{\Gamma_{i'j'}} = \frac{\Gamma_{i'j'}}{\Gamma_{i'j'}} = 1
$$

Thus, the the mass is ensured to be conserved through the time stepping procedure. However, this remap assignment has the consequence of producing significant numerical diffusion as is now clear in view of the form the rule takes above (figure 2.12).

This is a statement that every particle from the moving cell $C_{MC} \equiv C_{i'j'}$ is deposited to a cell C_{ij} on the Eulerian grid, i.e. conservation of particle number. Note, that this rule introduces an allocation in phase space that is not automatically consistent with, for example, energy conservation. If desired, energy conservation can be implemented in the scheme above, e.g. Feng et. al [20], by tying the velocity volume overlap fraction to an energy conservation statement. This is conveniently implemented by transforming to an energy mesh in favor of the velocity mesh for the remapping step so that factors pertaining to the kinetic energy v^2 replace the velocity factors v in the fraction prescribed above. Notwithstanding, the approach in this research will be to reduce the remap error in velocity and configurational space so that the targeted consequence of this accuracy approximately conserves energy, or at least possess no secular increase in energy with respect to time stepping. An implementation of the Classic CS is given in algorithm 1, which is presented after the discrete problem is introduced (section 3.1).
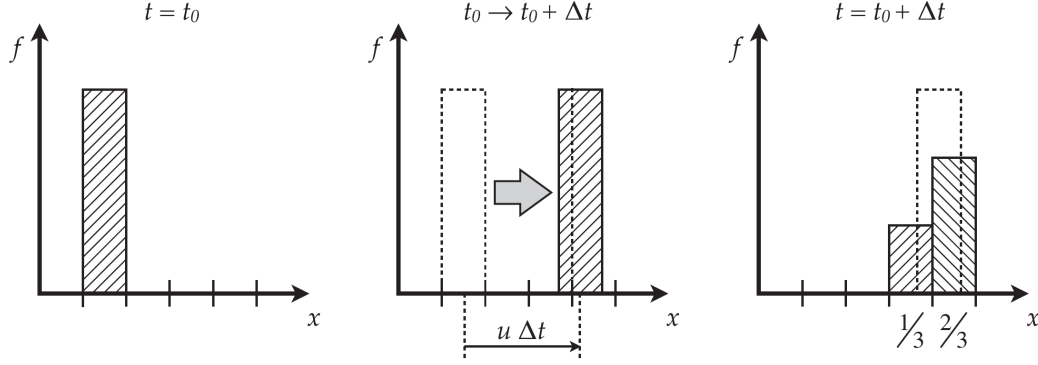
Figure 2.12: A one-dimensional illustration is shown of the remap assignment where the MC starting at a cell center has been pushed with a speed u towards the right that does not coincide exactly with only one cell, thus the remapping rule is employed as shown in frame 3. The result is an artificial spreading of the density across two contiguous cells. The effect compounds as time marching continues for all points in space, so that the overall effect is diffusion [26, p.3293].

2.5 Higher order convected scheme

The classic convected scheme is first order accurate in space. We follow the workup presented by Güçlu [27] to develop a higher order method of the convected scheme (CS) for a uniform velocity v of the one-dimensional advection equation (3.4):

$$\frac{\partial f}{\partial t} + v\frac{\partial f}{\partial x} = 0, \qquad x \in \mathcal{D}, \quad t \in (0,T]$$

And, subsequently inform how to use these high order solutions as building blocks to solve more complicated equations (e.g. with an acceleration term, and higher dimensions). Thus, the stepthrough is to advect density packets f at each location x along characteristics:

$$f(x, t+\Delta t) = f(x - v\Delta t, t)$$

which must then be remapped to the Eulerian grid according to the fractional overlap in cells. The displacement of each MC can equivalently be described by the number of grid points $\mathcal{C}$ the cell-center of an MC traverses in a time step. This normalized displacement is known as the Courant-Friedrichs-Lewy (CFL) parameter:

$$\mathcal{C} := \frac{v\Delta t}{\Delta x} = S + \alpha, \qquad S \in \mathbb{Z}, \alpha \in (-1, 1) \tag{2.26}$$

where Δx is the spacing between grid points. Here, we make a point to emphasize the quantity α is a relative (fractional) distance between the MC postpoint and a designated (left or right) nearest neighbor grid point. This fraction is related to the aforementioned area overlap fraction ζ ($\alpha = 1 - \zeta$), and is used for its ease of implementation as compared to the fractional area ζ which was used to motivate the remap rule as it is most physical (section 3.1). According to this decomposition, it is shown in section 3.1 that after pushing a density packet by S grid points, the CS remap update for the advection equation is given by:

$$(f_{i+S}^{n+1})_{\text{CS}} = \begin{cases} \alpha_{i-1}^n f_{i-1}^n + (1 - \alpha_i^n)f_i^n & \text{if } v \geq 0 \\ (1 + \alpha_i^n)f_i^n - \alpha_{i+1}^n f_{i+1}^n & \text{else} \end{cases} \tag{2.27}$$

where we subscript the fractions α according to the cell they originated from. The high order CS seeks to discern how to modify this term α so that the method performs at higher accuracy.

37

In the following, we provide the derivation of this higher order prescription for the case of $v \geq 0$. At the conclusion, we discuss differences for the $v < 0$ case and quote its corresponding result. The procedure amounts to matching Taylor series expansions of the exact solution with that of the CS update statement up to an order $\mathcal{O}(\Delta x^N)$, so that the local truncation error (LTE) is at most of order $\mathcal{O}(\Delta x^{N+1})$. Thus, we adopt the definition that an Nth order method is accurate up to order $\mathcal{O}(\Delta x^N)$. After a proper ansatz is made, the problem reduces to seeking higher order corrections to only the fractional part α of the CFL number, as the integer shift S presents no source of error for the remapping.

Thus, we analyze how to increase the accuracy of the remap assignment after shifting an MC by S cells in general. For specificity, we choose α to be measured such that it is the fractional distance measured between the cell-center of the MC with respect to the nearest Eulerian grid point so that S and α are of the same sign. To clearly distinguish between corrected and uncorrected terms, we swap labels so that α indicates the uncorrected term and U denotes the corresponding corrected version. The determination of the form of U is the objective of this section. In this way, eq. (2.26) becomes,

$$\mathcal{C} := S + U, \qquad S \in \mathbb{Z},\, U \in (-1, 1)$$

The Courant parameter $\mathcal{C}$ is a dimensionless displacement $\mathcal{C} := v\Delta t/\Delta x$, and thus so are S and U; however, we can equivalently view these terms as normalized velocities, e.g.

$$U := \frac{v\Delta t}{\Delta x} - S = [v + \tilde{v}(t, x)]\frac{\Delta t}{\Delta x} - S = \alpha + \tilde{\alpha}(t, x)$$

where we assert an additional perturbation $\tilde{v}$ is present at higher orders to correct the velocity v so that the final position of an MC is more optimal for the convected scheme. These velocities are nondimensionalized by the grid velocity $\Delta x/\Delta t$. Thus, it makes sense to speak of corrected normalized fluxes of the form $U(t, x)f(t, x)$. In the absence of higher order corrections, the $U(t, x) = \alpha$ for all time. Güçlü refers to the term $\tilde{\alpha}(t, x)$ above as an *anti-diffusive correction* with the admission that strictly the term encompasses higher order effects should the method be extended to higher order N. No matter the form these correction terms take, they have no bearing on particle number conservation. From the update (2.27), we see that

$$\sum_i f_i^{n+1} = \sum_i f_{i+S}^{n+1} = \sum_i [U_{i-1}^n f_{i-1}^n + (1 - U_i^n)f_i^k] = \underbrace{\sum_i U_{i-1}^n f_{i-1}^n - \sum_i U_i^n f_i^n}_{=\,0} + \sum_i f_i^k = \sum_i f_i^n$$

The anti-diffusive correction adjusts the postpoint of the trajectory within its cell after a push along its characteristics so that a more *optimal* fraction U is used in the remapping step with respect to the convected scheme. That is, each MC is exactly convected by $\mathcal{C} = S + \alpha$, however the process of dispersing the density among neighboring grid points using the fraction α caps the accuracy at $\mathcal{O}(\Delta x^2)$ given the discreteness of the mesh [2]. Thus, the "corrected" value U is not a correction to the position of the MC inasmuch as it is a modification to the fraction α to produce the optimal fraction U so that the CS algorithm *performs* at a higher level than its first order stencil permits. This notion is emphasized by recognizing the convected scheme update for a uniform velocity can be written as an upwind finite difference scheme (eq. (2.28)).

For the case of a single uniform speed v, it is easily seen that only two loaded prepoints i and $i \pm 1$ are mapped onto the grid point $i + S$, with the proportion $(1 \mp U_i)$ and $(\pm U_{i \mp 1})$, respectively. Choosing the top sign corresponds to $v \geq 0$ whereas the bottom sign pertains to the case $v < 0$. Thus, (2.27) amounts to:

$$(f_{i+S}^{n+1})_{\text{CS}} = \begin{cases} U_{i-1}^n f_{i-1}^n + (1 - U_i^n)f_i^n & \text{if } v \geq 0 \\ (1 + U_i^n)f_i^n - U_{i+1}^n f_{i+1}^n & \text{else} \end{cases} \tag{2.28}$$

Where, as noted, there is zero remapping error associated with translating an MC by an integer number of cells, thus the remapping assignment above is reported at a general shifted location $i + S$, which is subject to boundary conditions.

To find the form of the corrections U_i above, we begin by expanding all terms in (2.28) for $v \geq 0$ about the point (t, x) up to order $\mathcal{O}(\Delta x^N)$. The left-hand term becomes:

$$U_{i-1}^n f_{i-1}^n = U(t, x - \Delta x) f(t, x - \Delta x) = U(t, x) f(t, x) + \sum_{p=1}^{N} \frac{(-\Delta x)^p}{p!} \frac{\partial^p (Uf)}{\partial x^p} \bigg|_{(t,x)} + \mathcal{O}(\Delta x^{N+1})$$

so that the CS update reads

$$
\begin{aligned}
(f_{i+S}^{n+1})_{\mathrm{CS}} &= U_{i-1}^n f_{i-1}^n + (1 - U_i^n) f_i^n \\
&= \left(U(t, x) f(t, x) + \sum_{p=1}^{N} \frac{(-\Delta x)^p}{p!} \frac{\partial^p (Uf)}{\partial x^p} \bigg|_{(t,x)} + \mathcal{O}(\Delta x^{N+1}) \right) + f(t, x) - U(t, x) f(t, x)
\end{aligned}
$$

$$(f_{i+S}^{n+1})_{\mathrm{CS}} = f(t, x) + \sum_{p=1}^{N} \frac{(-\Delta x)^p}{p!} \frac{\partial^p (Uf)}{\partial x^p} \bigg|_{(t,x)} + \mathcal{O}(\Delta x^{N+1}) \tag{2.29}$$

The exact solution is advected by $S + \alpha$ so that the following solution holds $f(t + \Delta t, x + S\Delta x) = f(t, x - \alpha \Delta x)$. Taylor expanding this solution, we see that

$$(f_{i+S}^{n+1})_{\mathrm{exact}} = f(t + \Delta t, x + S\Delta x) = f(t, x) + \sum_{p=1}^{N} \alpha^p \frac{(-\Delta x)^p}{p!} \frac{\partial^p f}{\partial x^p} \bigg|_{(t,x)} + \mathcal{O}(\Delta x^{N+1}) \tag{2.30}$$

An Nth order method ensures the local truncation error (LTE) is at greatest $\mathcal{O}(\Delta x^{N+1})$:

$$\mathrm{LTE}\,(t, x, \Delta x) := f_{\mathrm{exact}}(t + \Delta t, x + S\Delta x) - f_{\mathrm{CS}}(t + \Delta t, x + S\Delta x) = \mathcal{O}(\Delta x^{N+1})$$

comparing eqs. (2.29) and (2.30), we see the above equality implies

$$
\begin{aligned}
\mathrm{LTE}\,(t, x, \Delta x) &= \left(f(t, x) + \sum_{p=1}^{N} \alpha^p \frac{(-\Delta x)^p}{p!} \frac{\partial^p f}{\partial x^p} \bigg|_{(t,x)} \right) - \left(f(t, x) + \sum_{p=1}^{N} \frac{(-\Delta x)^p}{p!} \frac{\partial^p (Uf)}{\partial x^p} \bigg|_{(t,x)} \right) + \mathcal{O}(\Delta x^{N+1}) \\
&= \sum_{p=1}^{N} \frac{(-\Delta x)^p}{p!} \alpha^p \frac{\partial^p f}{\partial x^p} \bigg|_{(t,x)} - \sum_{p=1}^{N} \frac{(-\Delta x)^p}{p!} \frac{\partial^p (Uf)}{\partial x^p} \bigg|_{(t,x)} + \mathcal{O}(\Delta x^{N+1}) \\
\mathrm{LTE}\,(t, x, \Delta x) &= \mathcal{O}(\Delta x^{N+1})
\end{aligned}
$$

provided that the following condition is met,

$$\sum_{p=1}^{N} \frac{(-\Delta x)^p}{p!} \alpha^p \frac{\partial^p f}{\partial x^p} \bigg|_{(t,x)} - \sum_{p=1}^{N} \frac{(-\Delta x)^p}{p!} \frac{\partial^p (Uf)}{\partial x^p} \bigg|_{(t,x)} = 0 \tag{2.31}$$

To make further progress, [27] proposes the following power series ansatz for the product $U(t,x)f(t,x)$, which we introduce as correct to order $\mathcal{O}(\Delta x^N)$ to ensure Nth order accuracy is maintained:

$$U(t,x)f(t,x) := \sum_{q=0}^{N-1} \beta_q(\alpha)(-\Delta x)^q \frac{\partial^q f}{\partial x^q}\bigg|_{(t,x)} + \mathcal{O}(\Delta x^{N+1}) \tag{2.32}$$

Inserting this series into (2.31), the problem reduces to seeking coefficients $\beta_q(\alpha)$ that define the correction U. The right-hand term becomes:

$$\sum_{p=1}^{N} \frac{(-\Delta x)^p}{p!} \frac{\partial^p(Uf)}{\partial x^p}\bigg|_{(t,x)} = \sum_{p=1}^{N} \frac{(-\Delta x)^p}{p!} \left(\sum_{q=0}^{N-1} \beta_q(\alpha)(-\Delta x)^q \frac{\partial^q f}{\partial x^q}\bigg|_{(t,x)} \right) + \mathcal{O}(\Delta x^{N+1})$$

$$= \sum_{p=1}^{N} \sum_{q=0}^{N-1} (-\Delta x)^{p+q} \frac{\beta_q(\alpha)}{p!} \frac{\partial^{p+q} f}{\partial x^{p+q}}\bigg|_{(t,x)} + \mathcal{O}(\Delta x^{N+1}) \tag{2.33}$$

$$\sum_{p=1}^{N} \frac{(-\Delta x)^p}{p!} \frac{\partial^p(Uf)}{\partial x^p}\bigg|_{(t,x)} = \sum_{r=1}^{N} \sum_{q=0}^{r-1} (-\Delta x)^r \frac{\beta_q(\alpha)}{(r-q)!} \frac{\partial^r f}{\partial x^r}\bigg|_{(t,x)} + \mathcal{O}(\Delta x^{N+1}), \qquad r = p+q \tag{2.34}$$

The $\mathcal{O}(\Delta x^{N+1})$ term collects higher order terms that result from the combined terms of both sums. To prove the index change in the final equality, note for $p = 1, 2, \ldots, N$, $q = 0, 1, \ldots, N-1$, we have $r = p+q \Rightarrow p = r-q$, such that

$$\sum_{p=1}^{N} \sum_{q=0}^{N-1} \rightarrow \sum_{r-q=1}^{N} \sum_{q=0}^{N-1} = \sum_{r=q+1}^{N} \sum_{q=0}^{N-1}$$

The sum interchange is easily visualized in tabular form, where the elements are represented as entries and the double summation amounts to adding all elements together.

$r = q+1$ \ q	0	1	2	...	$N-1$
1	$(-\Delta x)^1 \dfrac{\beta_0(\alpha)}{(1-0)!}\dfrac{\partial^1 f}{\partial x^1}$	...	...	...	...
2	$(-\Delta x)^2 \dfrac{\beta_0(\alpha)}{(2-0)!}\dfrac{\partial^2 f}{\partial x^2}$	$(-\Delta x)^2 \dfrac{\beta_1(\alpha)}{(2-1)!}\dfrac{\partial^2 f}{\partial x^2}$	...	...	...
3	$(-\Delta x)^3 \dfrac{\beta_0(\alpha)}{(3-0)!}\dfrac{\partial^3 f}{\partial x^3}$	$(-\Delta x)^3 \dfrac{\beta_1(\alpha)}{(3-1)!}\dfrac{\partial^3 f}{\partial x^3}$	$(-\Delta x)^3 \dfrac{\beta_2(\alpha)}{(3-2)!}\dfrac{\partial^3 f}{\partial x^3}$	...	...
...	...	...	...	...	...
N	$(-\Delta x)^N \dfrac{\beta_0(\alpha)}{(N-0)!}\dfrac{\partial^N f}{\partial x^N}$	$(-\Delta x)^N \dfrac{\beta_1(\alpha)}{(N-1)!}\dfrac{\partial^N f}{\partial x^N}$	$(-\Delta x)^N \dfrac{\beta_2(\alpha)}{(N-2)!}\dfrac{\partial^N f}{\partial x^N}$	...	$(-\Delta x)^N \dfrac{\beta_{N-1}(\alpha)}{[N-(N-1)]!}\dfrac{\partial^N f}{\partial x^N}$

Table 2.3: Visualizing the double sum of eq. (2.33) as entries in a table. The top row is over the $q = 0, 1, \ldots, N-1$, whereas the left-most column enumerates $p = 1, 2, \ldots, N$, which has been put in terms of $r = p + q$ in order to discern the limits of a double sum in terms of q and r alone. The colored cells indicate entries where $r < q + 1$, which are not terms that appear in the summation. Derivatives are evaluated at the point (t, x).

Thus, we can see by adding up all entries in the table, the following set of limits are equivalent:

$$\sum_{p=1}^{N}\sum_{q=0}^{N-1} = \sum_{r=1}^{N}\sum_{q=0}^{r-1}$$

which is the sum we have written in eq. (2.34). For transparency, relabel the dummy index r as p, so that the same equation can be written as:

$$\sum_{p=1}^{N}\frac{(-\Delta x)^p}{p!}\frac{\partial^p (Uf)}{\partial x^p}\bigg|_{(t,x)} = \sum_{p=1}^{N}\sum_{q=0}^{p-1}(-\Delta x)^p\frac{\beta_q(\alpha)}{(p-q)!}\frac{\partial^p f}{\partial x^p}\bigg|_{(t,x)} + \mathcal{O}(\Delta x^{N+1})$$

.

Thus, the order condition (2.31)

$$\sum_{p=1}^{N}\frac{(-\Delta x)^p}{p!}\alpha^p\frac{\partial^p f}{\partial x^p}\bigg|_{(t,x)} - \sum_{p=1}^{N}\frac{(-\Delta x)^p}{p!}\frac{\partial^p (Uf)}{\partial x^p}\bigg|_{(t,x)} = 0$$

$$\sum_{p=1}^{N}\frac{(-\Delta x)^p}{p!}\alpha^p\frac{\partial^p f}{\partial x^p}\bigg|_{(t,x)} - \sum_{p=1}^{N}\sum_{q=0}^{p-1}(-\Delta x)^p\frac{\beta_q(\alpha)}{(p-q)!}\frac{\partial^p f}{\partial x^p}\bigg|_{(t,x)} = 0$$

$$\sum_{p=1}^{N}(-\Delta x)^p\left(\frac{\alpha^p}{p!} - \sum_{q=0}^{p-1}\frac{\beta_q(\alpha)}{(p-q)!}\right)\frac{\partial^p f}{\partial x^p}\bigg|_{(t,x)} = 0$$

So, that the coefficient functions $\beta_q(\alpha)$ are defined by the relationship

$$\sum_{q=0}^{p-1}\frac{\beta_q(\alpha)}{(p-q)!} = \frac{\alpha^p}{p!} \tag{2.35}$$

The set of equations produced from (2.35) constitute a matrix system with a lower triangular coefficient matrix, thus it is easy to verify through forward substitution that the following recursive definitions hold [27], where we also show the result for $v < 0$ for completeness:

$$\beta_0(\alpha) = \alpha, \qquad\qquad \forall\, v \in \mathbb{R} \tag{2.36}$$

$$\beta_p(\alpha) = \begin{cases} \dfrac{\alpha^{p+1}}{(p+1)!} - \displaystyle\sum_{q=0}^{p-1}\frac{\beta_q(\alpha)}{(p+1-q)!} & \text{for } v \geq 0,\ p = 1,2,\ldots, \\[3ex] \dfrac{\alpha^{p+1}}{(p+1)!} - \displaystyle\sum_{q=0}^{p-1}(-1)^{p+q}\frac{\beta_q(\alpha)}{(p+1-q)!} & \text{for } v < 0,\ p = 1,2,\ldots, \end{cases} \tag{2.37}$$

The result for $v < 0$ follows similarly, where we begin from the latter form of the CS update (2.28), and assert an identical power series (2.32) for the product $U(t,x)f(t,x)$. Güçlu presents a series of equivalent formulations for these coefficients. Namely, it is shown that these coefficient functions $\beta_p(\alpha)$ are scaled combinations of

Bernoulli numbers, and may be computed either by consult of a table of Bernoulli numbers, or extracted from specific values of the Bernoulli polynomials [27].

The optimized fractional displacement U_N is most gently introduced by computing its Nth order accurate flux $[U_N(t,x)f(t,x)] \equiv [Uf]_i^n$ from the ansatz (2.32)

$$U_N(t,x)f(t,x) := \sum_{q=0}^{N-1} \beta_q(\alpha)(-\Delta x)^q \frac{\partial^q f}{\partial x^q}\bigg|_{(t,x)} = \alpha f(t,x) + \sum_{q=1}^{N-1} \beta_q(\alpha)(-\Delta x)^q \frac{\partial^q f}{\partial x^q}\bigg|_{(t,x)} \qquad (2.38)$$

defined such that $U(t,x)f(t,x) = U_N(t,x)f(t,x) + \mathcal{O}(\Delta x^{N+1})$. Thus, the above equation provides the form of the corrected terms that appear in (2.28). For brevity, the subscript N will be suppressed given it is understood an exact value is beyond the means of a numerical calculation. By working with the normalized flux, we evade a direct source for numerical overflow. *The implementation then requires $N-1$ derivatives of the distribution function $f(t,x)$ to create an Nth order method*. In order to maintain the accuracy, the means of obtaining these derivatives must be at least of order N. For example, these derivatives can be estimated through finite differences with sufficient sized stencils relative to scheme (see the *FD5* scheme in section 3.1.3). Alternatively, a direct method applicable to periodic domains (as is the case considered) is to compute the derivatives in the Fourier domain (section 3.1.4) which are equivalent to $\mathcal{O}(N^2)$ algebraic operations on the distribution function whereafter an inverse can be taken to record the value of the derivative. The use of a discrete fast Fourier transforms (FFT) and its corresponding inverse operation (IFFT) reduces the computational cost to $\mathcal{O}(N \log_2 N)$. In this way, the CS achieves spectral convergence as the truncation error rapidly approaches machine precision for orders of $N \gtrsim 20$ in double precision [27], hence this scheme is sometimes referred to as *spectral CS* (*FN* methods). However, for orders larger than approximately $N \approx 25$, high frequency white noise from the discrete transform process shows significant effect. Since the artificial noise is especially significant for low-level amplitudes, a low-pass filter can be incorporated in the implementation to reduce the pronounced numerical error for low amplitude densities. Using such a filter provides optimal orders between $20 \lesssim N \lesssim 25$ [27]. We delay the detailed description of this implementation for a moment to discuss issues associated with applying the correction U_N as a direct calculation of (2.38).

Design of a positivity and mass preserving limiter for the correction terms

Applying the correction term (2.38) requires a careful approach. As motivated above, one safeguard that can be taken is to forego the explicit calculation of U_i^n in favor of keeping the product $[Uf]_i^n$ intact. This measure evades one means of numerical overflow given that the computation of U_i^n according to eq. (2.38) invariably involves division by small values of the density f_i^n in sparse regions of phase space. However, there are still means by which the normalized flux $[Uf]_i^n$ can become either too large or otherwise erroneous. Notwithstanding, it is more direct to describe these remaining sources of error in terms of the problems they cause with the correction U_i^n rather than the upwind flux $[Uf]_i^n$. Thus, we choose to introduce the issues in terms of their effect on U_i^n below, and thereafter carryover their consequences to the flux in order to design a filter (or limiter) for the flux that ameliorates these problems [27].

The remaining sources of error pertain to (1) the actual values of the $N-1$ derivatives themselves, and/or (2) the accuracy of their numerical estimation. As concerns (1), should the actual values of the derivatives be large relative to f_i^n, it can be the case that the computed correction $|U_i^n| > 1$ (cf. (2.38)) so that it no longer presents a correction to the final position of an MC within the cell, but pushes it past the nearest cell-center. This violates the definition of the corrective term given that U_i^n is a correction to a fraction $|\alpha_{i' \to i}^n| < 1$ which must demand that $|U_i^n| < 1$. The second point (2) regards the fidelity of the derivative calculations. The numerical estimation of the derivatives will contain errors from undersampling if the mesh is not sufficiently resolved in the vicinity of regions of sharp changes due to not satisfying the Nyquist criterion. In this way, an

overall erroneous U_i^n value can result due to compounded error from unreliable derivative calculations from aliasing.

Finally, we mention that the sign is also not restricted in (2.38), so that given the previous discussion it is possible to create largely negative corrections U_i^n. Thus, an unrestrained application of the correction U_i^n can conspire with the CS update (2.28) to cause negative densities to evolve in time from an initially nonnegative distribution. Thus, a limiter needs to be designed to preserve two properties: (1) The distribution function must maintain its positivity, and (2) the term U_i^n must not be so large that it pushes the MC past its nearest grid point. To motivate the stepthrough of the selection of an appropriate correction, we refer to the calculation of eq. (2.38) as the *nominal* correction $\tilde{U}_i^n$. Further, we choose to label the corresponding nominal normalized flux as $\Gamma_i^n = \tilde{U}_i^n f_{i'}^n$, as opposed to $[\tilde{U}f]_i^n$ to keep consistent track of the the density parcel f_i^n that is first convected by the integer part of the Courant parameter to a location marked by i' before it is allocated to cell centers according to the correction $\tilde{U}_i^n$. Tagging the density parcel in this way proves most direct with respect to implementing the algorithm.

According to the above two concerns, for $v \geq 0$ we must have:

1. *Sign preservation*: $U_i^n \geq 0$ implies need to select a correction $U_i'^n = \max(0, \tilde{U}_i^n)$

2. *Numerical limiting*: The correction $U_i^n \leq 1$ implies $U_i^n = \min(U_i'^n, 1)$

Thus, the above two considerations design the following simple limiter, which are given in terms of both U_i^n and the normalized flux $U_i^n f_{i'}^n$:

$$U_i^n = \begin{cases} \min\left[\max\left(0, \tilde{U}_i^n\right), 1\right] & \text{if } v \geq 0 \\ \max\left[\min(-1, \tilde{U}_i^n), 0\right] & \text{else} \end{cases} \quad \text{or} \quad U_i^n f_{i'}^n = \begin{cases} \min\left[\max\left(0, \tilde{U}_i^n f_{i'}^n\right), f_i^n\right] & \text{if } v \geq 0 \\ \max\left[\min\left(-f_i^n, \tilde{U}_i^n f_{i'}^n\right), 0\right] & \text{else} \end{cases} \tag{2.39}$$

The result for negative velocities ($v < 0$) has also been provided, which is arrived at through analogous considerations. Thus, an algorithm employing the above limiter ensures positivity preservation of the distribution function, as well as unphysical overcorrections. The general implementation for the higher order CS is provided in algorithm 2. These algorithms are given after the discrete problem is introduced in full (section 3.1).

Windowed Fourier methods

If the derivatives required for the correction (2.38) are computed in the Fourier domain (FN methods, section 3.1.4, algorithm 3), we note a clear concern for any Fourier-based method is the appearance of unphysical oscillations. Hence, the Fourier-based convected scheme methods are challenged in the presence of sharp boundaries given artifacts introduced by the Gibbs phenomenon. If the unphysical oscillations are left unchecked, they introduce additional error nonlocally and contaminate the global solution. A straightforward means to limit their extent in pervading the solution is to window the Fourier transform by use of an adequately designed low-pass filter [62, 59, 27]. Following the motivation used by G.W. Wei and Sun et. al, a simple filter can be arrived at by considering the transform of the same function written in an equivalent manner. That is, realizing the Dirac delta function is the identity element for the convolution operation permits the following to be written:

$$f(x) = (\delta * f)(x) = \int_{-\infty}^{\infty} \delta(x - X) f(X) dX \tag{2.40}$$

where the Dirac delta function is the kernel in this convolution statement. To realize the Dirac delta function on a computer, the class of wave packets $\delta_{\sigma,\Delta x}$ is chosen as an approximation, which analytically approaches the delta function in the limit,

$$\lim_{\Delta x \to 0} \delta_{\sigma,\Delta x}(x) = \delta(x), \qquad \delta_{\sigma,\Delta x} = \frac{\sin \frac{\pi}{\Delta x} x}{\frac{\pi}{\Delta x} x} \exp\left(-\frac{x^2}{2\sigma^2}\right) \tag{2.41}$$

Here, $\sigma > 0$ is a parameter that characterizes the width of the envelope that acts in part to regularize the sinc function over the domain which is otherwise known as the regularized Shannon kernel (RSK) in signal processing theory. This is related to the Nyquist-Shannon sampling theorem with a windowing function (Gaussian envelope) to regularize the usual sinc function interpolant over the domain. In general, this result is a solution to the *interpolation problem*, which was mentioned in section 3.1.4.

The interpolation problem requires finding the minimum oscillation function that both fits the sampled points and which suitably approximates values in-between. The notion of minimum oscillation means to only retain as many high frequency components as needed to fit the data. Thus, the interpolation problem is most aptly solved in the frequency (Fourier) domain, whereafter we take the inverse to recover the interpolation $f(x)$ in real space. The exercise of eliminating all unneeded high frequency components amounts to seeking the cutoff frequency $p/2$ in the Fourier domain for the minimum oscillation interpolation. For real-valued signals, the Fourier transform is symmetric so that the resulting transformed interpolation function $\mathcal{F}[f]$ is *bandlimited* with bandwidth p in Fourier space so that its domain is given by $\mathcal{D}_{\mathcal{F}} = [-p/2, p/2]$.

The interpolation is arrived at indirectly by considering the equivalent representation of the windowed periodized extension of its Fourier transform over the frequency domain

$$\mathcal{F}[f] = \text{rect}_p \left(\mathcal{F}[f] * \text{Ⅲ}_p\right)$$

where $\text{rect}_p(z)$ is the natural "window," or rectangle function, that is only nonzero within the bandlimited domain of width p. The Shah function $\text{Ⅲ}_p(x) = \sum_{\ell=-\infty}^{\infty} \delta(x - \ell p)$, has been employed and its so-called *replication* property is exploited by convolving it with transformed test function $\mathcal{F}[f]$ in order to repeat it over the frequency domain. It is clear that periodizing and windowing with a frame the size of the original domain in Fourier space $(\mathcal{D}_{\mathcal{F}} = [-p/2, p/2])$ recovers the original transformed function so the above equality holds. Essentially, we have extended the function outside of the domain and then applied a wrapper that covers this extension. This trivial equivalence remarkably produces the required interpolation when the inverse Fourier transform is applied, a fact that can be attributed to the transform's action of swapping convolution and multiplication operations and by understanding the motivation to include the Shah function was to involve a delta function to execute the sampling:

$$f(x) = \mathcal{F}^{-1}[\mathcal{F}[f(x)] = \mathcal{F}^{-1}[\text{rect}_p \left(\mathcal{F}[f] * \text{Ⅲ}_p\right)]$$

In more words, we have designed an equivalent representation of $\mathcal{F}[f]$ with the goal in mind that the inverse of the term $\mathcal{F}[f] * \text{Ⅲ}_p$ samples the function due to the delta function, and the inverse of the rectangle function endows each of these samples with a smooth extension which are combined as a superposition from the involved sum. The details are provided below:

$$
\begin{aligned}
f(x) &= \mathcal{F}^{-1}[\text{rect}_p\,(\mathcal{F}[f] * \text{III}_p)] \\[2mm]
&= \mathcal{F}^{-1}[\text{rect}_p] * \mathcal{F}^{-1}[[\mathcal{F}[f] * \text{III}_p]] \\[2mm]
&= \mathcal{F}^{-1}[\text{rect}_p] * \mathcal{F}^{-1}\mathcal{F}[f] \cdot \mathcal{F}^{-1}\text{III}_p]] \\[2mm]
&= \mathcal{F}^{-1}[\text{rect}_p] * f(x) \cdot \mathcal{F}^{-1}[\text{III}_p] \\[2mm]
&= \mathcal{F}^{-1}[\text{rect}_p] * f(x) \cdot \tfrac{1}{p}\text{III}_{1/p} \\[2mm]
&= p\,\text{sinc}(px) * f(x) \cdot \tfrac{1}{p}\text{III}_{1/p}(x) \\[2mm]
&= \text{sinc}(px) * f(x) \cdot \text{III}_{1/p}(x) \\[2mm]
&= \text{sinc}(px) * f(x) \cdot \sum_{\ell=-\infty}^{\infty} \delta(x - \tfrac{\ell}{p}) \\[2mm]
&= \sum_{\ell=-\infty}^{\infty} f\left(\tfrac{\ell}{p}\right) \text{sinc}(px) * \delta(x - \tfrac{\ell}{p}) \\[2mm]
&= \sum_{\ell=-\infty}^{\infty} f\left(\tfrac{\ell}{p}\right) \text{sinc}\left(p\left[x - \tfrac{\ell}{p}\right]\right) \\[2mm]
f(x) &= \sum_{\ell=-\infty}^{\infty} f(x_\ell)\,\text{sinc}(p[x - x_\ell])), \quad x_\ell = \ell/p
\end{aligned}
$$

The frequency and real domains are inversely related, i.e. the linear frequency domain has a bandlimit $f_p = 1/(2N_x)$, whereas the circular frequency components $\xi_k \Delta x$ are spaced equally on the interval $\xi_k \Delta x \in [-\pi, \pi] = [-p/2, p/2]$. Thus, the bandwidth is $p = \pi/\Delta x$. Writing the sinc function in terms of sines renders the above representation as:

$$
\boxed{f(x) = \sum_{\ell=-\infty}^{\infty} f(x_\ell)\,\frac{\sin \frac{\pi}{\Delta x}x}{\frac{\pi}{\Delta x}x}} \qquad \underline{\text{Whittaker-Shannon interpolation formula}} \qquad (2.42)
$$

This is the statement of the *Whittaker-Shannon interpolation formula*, which is also known by several other names including credits attributed to Nyquist. Because it is a necessary bridge in completing the *Nyquist-Shannon Sampling theorem*, sometimes this interpolation formula is also called the sampling theorem. The statement informs what the proper interpolation function $f(x)$ should be given sample points located at each $x_\ell = \ell/p$. The sampling theorem takes one step further in identifying the frequency p as the Nyquist, or sampling, rate. If the sampling rate is greater than p then a perfect reconstruction is given by eq. (2.42).

The connection to the kernel (2.41) is clear, approximating the above by a finite number of terms produces an approximation and we choose to regularize the sinc function over the domain by applying a Gaussian envelope per G.W. Wei [62] where the kernel then becomes known as the *regularized Shannon kernel* (RSK):

$$f(x) \simeq \sum_{\ell=[x]-W}^{[x]+W} f(x_\ell) \frac{\sin \frac{\pi}{\Delta x} x}{\frac{\pi}{\Delta x} x} \quad \rightarrow \quad \boxed{f(x) \simeq \sum_{\ell=[x]-W}^{[x]+W} f(x_\ell) \frac{\sin \frac{\pi}{\Delta x} x}{\frac{\pi}{\Delta x} x} \exp\left(-\frac{x^2}{2\sigma^2}\right)} \quad \underline{\text{DSC-RSK filter}} \quad (2.43)$$

where the term $[x]$ denotes the grid point ℓ that is nearest to the point x, and the kernel support W can be chosen to exploit the localization accomplished by the Gaussian envelope. That is, the kernel bandwidth $2W+1$ can be taken to be less than the computational domain, though it is obvious that better approximations are achieved by considering larger half-widths W. Because these represent a special case of discrete singular convolutions (DSC), the regularized Shannon kernel (RSK) is referred to in literature as the DSC-RSK filter, and now the role of the parameter σ becomes clear: the lower its value the stronger the filter.

The parameter $r = \frac{\sigma}{\Delta x}$ is called the *regularizer*. Sun et. al pursued an adaptive filter with time that was able to accurately capture shocks. Here, it suffices to defer to Güçlü [27] who chooses to use a static kernel $K_r(x) \equiv K(x)$ of this same RSK $\delta_{\sigma,\Delta x}(x)$ $(K(x) \equiv \delta_{\sigma,\Delta x}(x))$. Thus, the filter parameters $(W, r = \sigma/\Delta x)$ are chosen to best fit a particular problem and are unchanged during the simulation. In general, we remark the stronger the filter the more localized the spurious errors from discontinuities are; however, this comes at the cost of losing the required information necessary to correct for numerical diffusion. Thus, the stronger the filter the higher the numerical diffusion. The convolution (2.40) is then approximated by the discrete version

$$f(x) \simeq \sum_{\ell=[x]-W}^{[x]+W} K(x - x_\ell) f(x_\ell) = (K * f)(x) \tag{2.44}$$

Hence, as with every Dirichlet type kernel, $K(x)$ is an approximation to the identity element for the convolution operation.

The filter can be implemented in real or Fourier space. Since the foundation of the filter is an expensive convolution operation (in real space), it is most efficient to apply the corresponding transformed filter in Fourier space where convolution operations are traded for complex multiplications. Thus, referring the discrete transforms for a function g as $\hat{g}$ which approximates the continuous transform $\mathcal{F}[g] \simeq \hat{g}$, recall the definition of the discrete transform pair:

$$\mathcal{F}[g](\xi_k) \simeq \hat{g}(\xi_k) \equiv \text{DFT}[g]_k := \frac{1}{N_x} \sum_{\ell=0}^{N_x-1} g(x_\ell) e^{-\mathrm{j}\xi_k x_\ell} \tag{3.14a revisited}$$

$$g(x_\ell) \simeq \text{IDFT}[\hat{g}](x_\ell) := \sum_{k=0}^{N_x-1} \hat{g}_k e^{\mathrm{j}\xi_k x_\ell} \tag{3.14b revisited}$$

Where, as before, $g_k = g(\xi_k) = g(\frac{2\pi k}{L})$ is the kth Fourier coefficient. Taking $g(x)$ to be $f(x)$ as given by the approximate identity convolution in terms of the kernel $K(x)$ (equation (2.44)) we write the Fourier transform according to the above definition and take its inverse to recover the windowed Fourier transform result,

$$\begin{aligned}
\mathcal{F}[f]_k \simeq \hat{f}_k &= \frac{1}{N_x} \sum_{\ell=0}^{N_x-1} f(x_\ell) e^{-\mathrm{j}\xi_k x_\ell} \\
&= \frac{1}{N_x} \sum_{\ell=0}^{N_x-1} \sum_{\ell'=[x_\ell]-W}^{[x_\ell]+W} K(x_\ell - x_{\ell'}) f(x_{\ell'}) e^{-\mathrm{j}\xi_k x_\ell} \\
\hat{f}_k &\simeq \hat{K}_k \cdot \hat{f}_k
\end{aligned}$$

47

Thus, we obtain a simple, and uncrowded, statement that shows the carryover of the kth component of the kernel $\hat{K}$ also acts to approximate identity, just as K does in real space. Taking the inverse transform,

$$f(x_\ell) \simeq \text{IDFT}\,[\hat{f}](x_\ell) \quad = \quad \sum_{k=0}^{N_x-1} \hat{K}_k \cdot \hat{f}_k e^{\mathrm{j}\xi_k x_\ell}$$

So that it is seen in practice we can filter the function by backward transforming the weighted Fourier coefficients $\hat{K}(\xi_k) \cdot \hat{f}(\xi_k)$ which correspond to a localized version of the original function. *Since the transform now acts on coefficients of the localized function, the Gibbs phenomenon errors associated with discontinuities must also be localized* [59]. The overall procedure is referred to as a *windowed Fourier transform*, and Fourier-based CS methods that use windowing are referred to as WFN schemes. For example, a $WF15$ scheme is a 15th order accurate windowed spectral CS method.

Put in more computational terms, the discrete transforms can be computed using a DFT/IDFT or through an optimized FFT/IFFT stepthrough:

$$f_{k,K}(x_m) = \text{IDFT}\,[\text{DFT}[K]_k \cdot \text{DFT}[f]_k], \quad \text{or} \quad f_{k,K}(x_m) = \text{IFFT}\,[\text{FFT}[K]_k \cdot \text{FFT}[f]_k] \qquad (2.45)$$

and the resulting windowed result $f_{k,K} = f_K(\xi_k)$ is labelled with the signature of the kernel K to distinguish it from its unprocessed counterpart f_k used in algorithm 3. The only change to the spectral CS algorithm is $\hat{f}_k \to \hat{f}_{k,K}$ in the computation of the derivative coefficients $\hat{d}_q^n$ in the changeover from $FN \to WFN$, i.e.

$$FN : \hat{d}_q^n = (\mathrm{j}\xi_k)^q \hat{f}_k \quad \text{becomes} \quad WFN : \hat{d}_q^n = (\mathrm{j}\xi_k)^q \hat{f}_{k,K}$$

in algorithm 3 to generate the WFN schemes. The future work will implement this filter via a switch that is activated whenever the monitored local error is seen to exceed a specified threshold.

2.6 The semi-Lagrangian approach to the Vlasov-Poisson system

Recall the system of equations that govern a collisionless electrostatic plasma ($\vec{E}(t,\vec{x}) = -\vec{\nabla}\phi(t,\vec{x})$):

$$\frac{\partial f_\alpha}{\partial t} + \vec{v}\cdot\frac{\partial f_\alpha}{\partial\vec{x}} + \frac{q_\alpha}{m_\alpha}\vec{\nabla}\phi\cdot\frac{\partial f_\alpha}{\partial\vec{v}} = 0 \qquad \text{(2.10a revisited)}$$

$$-\nabla^2\phi = \sum_\alpha \frac{q_\alpha}{\epsilon_0}\int d^3\vec{v}f_\alpha(t,\vec{x},\vec{v}), \qquad \alpha = 1,2,\ldots \qquad \text{(2.10b revisited)}$$

where the symbols have their usual meanings as defined on page 25. Given that the generalization to higher dimensions is straightforward, the 1D-1V case is discussed here for clarity in presentation and because it is the next step to be taken in this research. We omit any subscripting that would indicate any particular coordinate (resp. direction) for $\vec{x}$ (resp. $\vec{v}$) with the understanding the following equation models any such component. This lower dimensional case then implies that the scalar potential is related to the electric field by ($E = -\partial_x\phi$). The 1D-1V scenario is described by

$$\frac{\partial f_\alpha}{\partial t} + v\frac{\partial f_\alpha}{\partial x} - \left(\frac{q_\alpha}{m_\alpha}\frac{\partial\phi}{\partial x}\right)\frac{\partial f_\alpha}{\partial v} = 0 \qquad (2.46)$$

$$-\frac{\partial^2\phi}{\partial x^2} = \sum_\alpha \frac{q_\alpha}{\epsilon_0}\int dv f_\alpha(t,x,v), \qquad \alpha = 1,2,\ldots \qquad (2.47)$$

the characteristics that describe the trajectories take the form:

$$\frac{dx}{dt} = v \tag{2.48a}$$

$$\frac{dv}{dt} = -\frac{q_\alpha}{\epsilon_0}\frac{d\phi}{dx} \tag{2.48b}$$

The integration of the characteristics can be done as one (convecting the "full" phase space fluid), or the integration of each characteristic can be staggered in so-called *split methods*. The full phase space ballistic move was demonstrated by Hitchon et. al using a Runge-Kutta scheme of desired order accuracy, where the intermediate values used in the integrator calculate the necessary averages along the particle trajectory needed in the potential field calculations for accurate velocity updates [29, p.84]. A decade later, Feng and Hitchon integrated the characteristics using a classic Euler scheme whose field quantities were evaluated at the cell-centers of the prepoints of the MC in a long-lived MC (LLMC) approach [22] and whose velocity remap was tied to an energy grid in order to actively enforce conservation. While the approach by Feng and Hitchon preserves the energy in the autonomous system, a consideration that is not obvious from only the prior discussion that afflicts both aforementioned ideas is that these methods cannot retain the inherent symplecticity of phase space flow. Splitting methods can be built that preserve this property, and it has been demonstrated that high stability over long time simulations is readily possible with geometric integrators that preserve the phase space volume of each trajectory. The idea of split methods is developed after a brief mathematical exposition of the key concepts.

2.6.1   Operator splitting theory

While general numerical integrators such as Runge-Kutta or linear multi-step methods can produce accurate solutions, these integrators are not designed to take into account the unique algebraic structure of Hamilton's equations, failing to retain special symmetries such as symplecticity (differential phase volume preservation). To this end, significant attention in the past two decades has turned to *geometric* integrators, where geometric properties of the exact solution are studied, and numerical integrators are designed with attention to preserve these. A special case of geometric integrators are *symplectic* integrators. It has been seen that creating schemes that preserve the symplectic nature of the phase space increases stability as compared to non-symplectic versions of the same scheme (e.g. compare the Euler method with its symplectic counterpart applied to a model problem such as the 1D harmonic oscillator). Another example of an algebraic structure is in the Hamiltonian itself in that it is a sum of two distinct terms. A Hamiltonian with partitioned structure is said to be *explicit*. Integrators minding this additive property can create significantly more efficient schemes that can be carefully set up so that high order accuracy is still obtained all the while inheriting the benefit of ease of implementation. These last considerations are some of the prime motivators for the idea of *splitting methods* applied to the Maxwell-Boltzmann systems. It will be proposed, though not rigorously proven, that ostensibly *all* symplectic integrators are splitting algorithms.

Vlasov-Poisson splitting theory

In the electrostatic case, the canonical momentum is the same as the physical momentum ($\vec{p} = m_\alpha \vec{v}$), and its time derivative $\dot{\vec{p}} = q_\alpha \vec{E}$. Further specializing to the 1D-1V case, the Vlasov equation takes the scalar form:

$$\frac{\partial f_\alpha}{\partial t} + v\frac{\partial f_\alpha}{\partial x} + \frac{q_\alpha E}{m}\frac{\partial f}{\partial v} = 0, \qquad \text{where } f_\alpha = f_\alpha(t, x, v) \tag{2.46 revisited}$$

The electric field is related, as usual, to its scalar potential $E = -\partial_x \phi$. Since Hamiltonian formulations are so commonly presented in terms of canonical variables, for clarity we introduce the Hamiltonian H of this electrostatic system and the corresponding set of Hamilton's equations in terms of both of the canonical variables (q, p) alongside the form they take in phase space (x, v) that is tracked most usually in plasma physics, whereafter we continue with only the latter:

$$\underline{\text{Canonical variables } (q, p)} \qquad\qquad \underline{\text{Phase space variables } (x, v)}$$

Hamiltonian:
$$H \;=\; \frac{p^2}{2m_\alpha} + q_\alpha \phi(q) \qquad\qquad H = \frac{m_\alpha v^2}{2} + q_\alpha \phi(x) \qquad (2.49\text{a})$$

Hamilton's equations:
$$\begin{aligned}
\dot{q} &= +\partial_p H = p/m_\alpha & \dot{x} &= m_\alpha^{-1}\partial_v H = v & (2.49\text{b})\\
\dot{p} &= -\partial_q H = q_\alpha(-\partial_q \phi) = q_\alpha E(q) & m_\alpha \dot{v} &= -\partial_x H = q_\alpha E(x) & (2.49\text{c})
\end{aligned}$$

The differential operators are mapped according to $(\partial_q, \partial_p) \mapsto (\partial_x, m_\alpha^{-1}\partial_v)$. The utility of taking on a Hamiltonian perspective is its transparency seeing the exact solution to the Vlasov equation, which is written in terms of the Lie algebraic language [49]. We match the coefficients of the derivatives in the Vlasov equation (2.46) with those of the Hamiltonian shown just above (eqs. (2.49b) and (2.49c)). Defining the Poisson bracket $\{A, B\} = (\partial_x A)(\partial_v B) - (\partial_v A)(\partial_x B)$ then permits a compact form of the Vlasov equation to be written down:

$$\begin{aligned}
\frac{\partial f_\alpha}{\partial t} + \frac{1}{m_\alpha}\frac{\partial H}{\partial v}\frac{\partial f_\alpha}{\partial x} - \frac{1}{m_\alpha}\frac{\partial H}{\partial x}\frac{\partial f_\alpha}{\partial v} &= 0 \\
\frac{\partial f_\alpha}{\partial t} - \frac{1}{m_\alpha}\{H, f_\alpha\} &= 0
\end{aligned}$$

Since the Poisson bracket satisfies an alternating property and obeys a Jacobi identity, it is in general a Lie product whose group elements must sit on a differentiable manifold $\mathcal{M}$ given the product relates group elements in part through a derivative operator, i.e. the Poisson bracket is a bilinear mapping $\{\cdot, \cdot\} \colon C^\infty(\mathcal{M}) \times C^\infty(\mathcal{M}) \to C^\infty(\mathcal{M})$ which has the structure of a Lie algebra. Equivalently, the above equation can be written as

$$\frac{\partial f_\alpha}{\partial t} = \frac{1}{m_\alpha}\{H, f_\alpha\} \equiv \Lambda f_\alpha \tag{2.50}$$

where the operator $\Lambda := m_\alpha^{-1}\{H, \cdot\}$ is the *Liouvillian operator*, or in general a Lie derivative. Since the Hamiltonian does not explicitly depend on time, the *exact* solution of eq. (2.50) is readily obtained over an interval $t \in [0, \tau]$:

$$f_\alpha(\tau, x, v) = e^{\tau\Lambda} f_\alpha(0, x_0, v_0) = T^\tau f_\alpha(0, x_0, v_0) \tag{2.51}$$

where the distribution function at time $t = 0$ represents the values at the start of a time step. The exponential mapping is exactly the mechanism for passing information from the Lie algebra to the Lie group. So, we see naturally the embedding of Lie groups takes root in the study of these Hamiltonian systems, and thus this exponentiation is a well-defined object. This object can also be viewed in terms of the Hamiltonian vector field X_H and is also known as the *Hamiltonian flow* $\varphi_\tau(x, v) \colon \mathbb{R}^+ \times \mathbb{R}_x \times \mathbb{R}_v \mapsto \mathbb{R}_x \times \mathbb{R}_v$, defined with the initial flow corresponding to the initial data $\varphi_0(x, v) = (x_0, v_0)$ (see section 2.3.3). In this section, we borrow the notation of Mangeney et. al [41] for its transparency. Thus, the Lie operator is labeled as $T^\tau \equiv \exp(\tau\Lambda)$. Its action is clear from previous discussion, it is a *time evolution operator* for the distribution function along its characteristics. Notwithstanding, its action is convincingly obtained in the following in order to remove any need for suggestion.

The specific form of this Hamiltonian permits the operator to be separated and written down explicitly. Consider the Hamiltonian (eq. (2.49a)):

$$H(x,v) = \frac{m_\alpha v^2}{2} + q_\alpha \phi(x) = H_T(v) + H_V(x)$$

where the labels refer to kinetic and potential energies, T and V, respectively. Working with the Poisson bracket in the definition of the Liouvillian operator, we see

$$\Lambda = \{H, \cdot\} = \{H_T + H_V, \cdot\} = \{H_T, \cdot\} + \{H_V, \cdot\} = \Lambda_x + \Lambda_v \tag{2.52}$$

The additive property above can be readily verified by unfolding the terms involved in each bracket and collecting terms. The subscripts x and v are so labeled as these operators will soon be shown to be responsible for advection in configurational and velocity spaces, respectively. The time evolution operator takes the form

$$T^\tau = e^{\tau \Lambda} = e^{\tau \Lambda_x + \tau \Lambda_v}$$

To split an operator means to parse a compound operator and isolate its (decoupled) constituents. Algebraically this amounts to seeking a partition in the exponential. Applying this idea directly to the form just above is called *Lie-Trotter splitting*:

$$T^\tau = e^{\tau \Lambda_x + \tau \Lambda_v} = e^{\tau \Lambda_x} e^{\tau \Lambda_v} + \mathcal{O}(\tau^2) \qquad \underline{\text{Lie-Trotter splitting}}$$

However, in the effort to split the operator we incur a second order error in time τ as the two operators Λ_x and Λ_v do not commute in general. This can be proven by Taylor expanding both sides of the above equality and matching coefficients [64]. In general, [5, 15, 65] provides the appropriate prescription to obtain nth order integrators, which amounts to seeking coefficients $\{c_i, d_i\}$ ($i \in \mathbb{N}$), such that

$$\exp(\tau(\Lambda_x + \Lambda_v)) = \prod_{i=1}^{n} \exp(c_i \tau \Lambda_x) \exp(d_i \tau \Lambda_v) + \mathcal{O}(\tau^{n+1}), \qquad \sum_i c_i = \sum_i d_i = 1 \tag{2.53}$$

That is, so that the Taylor series expansions on both sides of the above equation match up to the desired order n.

The most general case of the scheme just above are termed *Runge-Kutta-Nyström (RKN) methods* where the two vector fields Λ_x and Λ_v are qualitatively distinct; such a case is common among physical systems. This is in contrast to the special case where the two operators are interchangeable (i.e. the operators commute), which are known as *Partitioned Runge-Kutta (PRK) methods*. Note, the operators as shown in (2.53) act right-handedly in succession on their operands, such that the right-most operator in the product sequence acts directly on the initial data $f(0, x_0, v_0)$, the next operator acts on the result of this first step's action, the operator after that acts on the previous step's result, and so on. In this way, it is seen that this product represents a formal composition.

The composition (2.53) is a trivial consequence of the *Baker-Campbell-Hausdorff* (BCH) formula. Its corollary, the *Lie-Trotter product formula* indicates that in the limit of an operator decomposition that takes an infinitesimal time step we recover the exact operator:

$$\exp(\tau(\Lambda_x + \Lambda_v)) = \lim_{N \to \infty} \left(\prod_{i=1}^{n} e^{c_i \tau \Lambda_x / N} e^{d_i \tau \Lambda_v / N} \right)^N, \qquad \sum_i c_i = \sum_i d_i = 1$$

A statement which informs it is always possible to find coefficients for a given order n so that the design of arbitrary order integrators is possible at least in principle. With the understanding that using a split operator to analytically solve the Vlasov equation incurs a splitting error, we now turn our attention to investigating

which equations these operators ($\exp(c_i\tau\Lambda_x)$ and $\exp(d_i\tau\Lambda_v)$) exactly solve. These equations turn out to have known solutions which allows us to directly match each operator with its action. In fact, we will show in section 2.6.2, that the prescription for the nth order scheme (2.53) has an equivalent form in the integrated characteristics that for lower order integrators is more amenable to use in extracting the required coefficients c_i and d_i.

Begin by considering the Vlasov equation in Poisson bracket representation given in terms of the explicit Hamiltonian $H(x,v) = H_T(v) + H_V(x)$:

$$\frac{\partial f_\alpha}{\partial t} + v\frac{\partial f_\alpha}{\partial x} + \frac{q_\alpha E}{m_\alpha}\frac{\partial f}{\partial v} = 0$$

$$\frac{\partial f_\alpha}{\partial t} - \frac{1}{m_\alpha}\{H_T(v), f_\alpha\} - \frac{1}{m_\alpha}\{H_V(x), f_\alpha\} = 0$$

The exponentiated Liouvillian operators, Λ_x and Λ_v (eq. (2.52)), do not appear in the solution of this full equation, but rather the two *split* equations:

$$\frac{\partial f_\alpha}{\partial t} - \frac{1}{m_\alpha}\{H_T(v), f_\alpha\} = 0 \Rightarrow f_\alpha(\tau, x, v) = e^{\tau\Lambda_x} f_\alpha(0, x, v) \tag{2.54a}$$

$$\frac{\partial f_\alpha}{\partial t} - \frac{1}{m_\alpha}\{H_V(x), f_\alpha\} = 0 \Rightarrow f_\alpha(\tau, x, v) = e^{\tau\Lambda_v} f_\alpha(0, x, v) \tag{2.54b}$$

But, these equations are the same as standard advection equations whose exact solutions are obtainable through the method of characteristics, that is:

$$\frac{\partial f_\alpha}{\partial t} + v\frac{\partial f_\alpha}{\partial x} = 0 \Rightarrow f_\alpha(\tau, x, v) = f_\alpha(0, x - v\tau, v) \tag{2.55a}$$

$$\frac{\partial f_\alpha}{\partial t} + \frac{q_\alpha E}{m_\alpha}\frac{\partial f}{\partial v} = 0 \Rightarrow f_\alpha(\tau, x, v) = f_\alpha(0, x, v - \tau\frac{q_\alpha E(x)}{m_\alpha}) \tag{2.55b}$$

Thus, comparing the right-hand sides of eqs. (2.54a) with (2.55a) as well as eq. (2.54b) with (2.55b) allows the connection to be made between the operators and their action. To convey this compactly, define

$$\mathcal{X}^\tau := e^{\tau\Lambda_x}|_{v=const} = e^{\tau\{H_T,\cdot\}} \qquad \text{Configuration advection operator at constant } v \tag{2.56}$$

$$\mathcal{V}^\tau := e^{\tau\Lambda_v}|_{x=const} = e^{\tau\{H_V,\cdot\}} \qquad \text{Velocity advection operator at constant } x \tag{2.57}$$

Then, we understand that

$$f_\alpha(\tau, x, v) = \mathcal{X}^\tau f(0, x, v) = e^{\tau\Lambda_x} f(0, x, v) = f_\alpha(0, x - v\tau, v) \tag{2.58}$$

$$f_\alpha(\tau, x, v) = \mathcal{V}^\tau f(0, x, v) = e^{\tau\Lambda_v} f(0, x, v) = f_\alpha(0, x, v - (q_\alpha E/m_\alpha)\tau) \tag{2.59}$$

where the superscripts indicate the time step, and $\mathcal{X}$ (resp. $\mathcal{V}$) are operators that advect only the phase space variable x (resp. v). It should not be interpreted that the above time stepping must start at time $t = 0$, here

we are communicating a time step over $\Delta t \equiv \tau$ where we choose to label the start of a time step as $t = 0$ and ending at $t = \tau$.

The prescription for higher order schemes (2.53), can then be written as,

$$T^\tau = \prod_{i=1}^{n} \mathcal{X}^{c_i \tau} \mathcal{V}^{d_i \tau} + \mathcal{O}(\tau^{n+1}), \qquad \sum_i c_i = \sum_i d_i = 1 \tag{2.60}$$

As mentioned previously, when applied to an operand $f_\alpha(0, x_0, v_0)$, it is appropriate to represent the action of the operators as an *iterated composition*, for example, the first order scheme $c_1 = d_1 = 1$ can be written as

$$f(t, x, v) = \mathcal{X}^\tau \circ \mathcal{V}^\tau f(0, x_0, v_0) + \mathcal{O}(\tau^2) \tag{2.61}$$

Thus, the crux of designing operator splitting methods is to seek coefficients c_i and d_i per any equivalent form of eq. (2.60) that exactly matches the Taylor series of the exact solution up to a desired order n.

In closing, we make mention of an additional signature of the Lie algebra that naturally shows up in our development. This formal operator takes the form of the exponential map of the Liouvillian operator (see eq. (2.51)), which is exactly the Hamiltonian vector field X_H (i.e. the tangent space). It is well known that the exponential map of the tangent space taken at the identity of a Lie group $G \in \mathcal{M}$, where $\mathcal{M}$ is an n-manifold, has the structure of a Lie algebra which formally permits an inheritance of the conventional Lie bracket product which is known to obey several well-defined properties. Specifically, the differentiability allows the use of a special case of the Lie bracket known as the *Poisson bracket*, which was used above. Appropriately, products of this form are sometimes referred to as constituting a *Lie-Poisson* algebra.

2.6.2 Strang splitting

The general procedure put forth by Neri [49] and Yoshida [64] involves expanding both sides of eq. (2.53) and seeking coefficients c_i and d_i in order to match the Taylor series of the exponentials up to the desired order of accuracy. For the purposes of lowest order integrators in the specific case at hand, a more basic stepthrough is possible. First, consider the Hamiltonian as defined in terms of phase space variables (x, v) in the previous section

$$H(x, v) = \frac{1}{2} m_\alpha v^2 + q_\alpha \phi(x) \tag{2.49a revisited}$$

Hamilton's equations describe the phase space coordinate trajectories:

$$\dot{x} = \frac{1}{m_\alpha} \frac{\partial H}{\partial v} = v \tag{2.49b revisited}$$

$$\dot{v} = -\frac{1}{m_\alpha} \frac{\partial H}{\partial x} = -q_\alpha \partial_x \phi(x) = q_\alpha E(x) \tag{2.49c revisited}$$

Integrating the above two forms over duration τ with the functions evaluated at time t produces the Euler method:

Standard Euler method

$$x(t + \tau) = x(t) + \tau v(t) + \mathcal{O}(\tau^2) = x(t) + \tau \left(\frac{1}{m_\alpha} \frac{\partial H}{\partial v} \right) \Big|_{v=v(t)} + \mathcal{O}(\tau^2) \tag{2.62}$$

$$v(t + \tau) = v(t) + \tau \left(\frac{q_\alpha E(x)}{m_\alpha} \right)_{x=x(t)} + \mathcal{O}(\tau^2) = v(t) - \tau \left(\frac{1}{m_\alpha} \frac{\partial H}{\partial x} \right) \Big|_{x=x(t)} + \mathcal{O}(\tau^2) \tag{2.63}$$

which agree with the Taylor series expansions of the exact solutions up to first order. However, this scheme is not symplectic. In order to see this, consider the differential volume generated by the *basic* two-form $\omega = dv \wedge dx = dvdx$; symplecticity of the Hamiltonian phase space indicates this differential volume is preserved for all time. To see this clearly, the notation is adopted that $v' = v(t + \tau)$, and $x' = x(t + \tau)$, while the starting function values are denoted as $v = v(t)$ and $x = x(t)$. Then we require

$$dv' \wedge dx' = dv \wedge dx, \qquad \underline{\text{symplecticity condition}} \qquad (2.64)$$

In other words, the Jacobian determinant must be shown to be equal to unity in the following

$$dv' \wedge dx' = \left| \frac{\partial(v', x')}{\partial(v, x)} \right| dv \wedge dx$$

The required derivatives are calculated from the forms of v' and x' from eqs. (2.63) and (2.62), respectively.

$$dv' \wedge dx' = \begin{vmatrix} \frac{\partial v'}{\partial v} & \frac{\partial v'}{\partial x} \\ \frac{\partial x'}{\partial v} & \frac{\partial x'}{\partial x} \end{vmatrix} dv \wedge dx$$

$$= \begin{vmatrix} 1 & \tau\left(\frac{q_\alpha}{m_\alpha}\frac{\partial E(x)}{\partial x}\right) \\ \tau & 1 \end{vmatrix} dv \wedge dx \qquad (2.65)$$

$$dv' \wedge dx' = \left[1 - \tau^2 \left(\frac{q_\alpha}{m_\alpha} \frac{\partial E(x)}{\partial x} \right) \right] dv \wedge dx \qquad (2.66)$$

So, it is seen that in general $dv(t+\tau) \wedge dx(t+\tau) \neq dv(t) \wedge dx(t)$ for the Euler method, showing the numerical integration does not preserve this property. However, eqs. (2.65) and (2.66) invite a simple modification to render the method symplectic. It is obvious that if the off-diagonal product were to vanish in the determinant calculation, only the term of unity would remain on the right-hand side of (2.66), so that the differential volume would be preserved. A basic way to accomplish this is to ensure one of the elements on the off-diagonal is zero in eq. (2.65). This amounts to staggering the time stepping. For example, the following interleaving of time steps produces the well-known symplectic Euler method.

Symplectic Euler method

$$\underline{\text{Step 1}}: \quad x' = x + \tau v \quad \text{where} \quad v' = v \qquad (2.67)$$

$$\underline{\text{Step 2}}: \quad x'' = x' \qquad \text{where} \quad v'' = v' + \tau\left(\frac{q_\alpha E(x'')}{m_\alpha} \right) \qquad (2.68)$$

so that now x'' and v'' correspond to the function values at the conclusion of the time step $t + \tau$, whereas x' and v' are only included to make the bookkeeping clear between the two distinct steps. It can be seen that each step is a symplectic mapping.

Step 1

$$dv' \wedge dx' = \left| \frac{\partial(v', x')}{\partial(v, x)} \right| dv \wedge dx$$

$$= \begin{vmatrix} \frac{\partial v'}{\partial v} & \frac{\partial v'}{\partial x} \\[2mm] \frac{\partial x'}{\partial v} & \frac{\partial x'}{\partial x} \end{vmatrix} dv \wedge dx$$

$$= \begin{vmatrix} 1 & 0 \\[2mm] \tau & 1 \end{vmatrix} dv \wedge dx$$

$$dv' \wedge dx' = dv \wedge dx$$

Step 2

$$dv'' \wedge dx'' = \left| \frac{\partial(v'', x'')}{\partial(v', x')} \right| dv' \wedge dx'$$

$$= \left| \frac{\partial(v'', x'')}{\partial(v', x')} \right| dv' \wedge dx'$$

$$= \begin{vmatrix} \frac{\partial v''}{\partial v'} & \frac{\partial v''}{\partial x'} \\[2mm] \frac{\partial x''}{\partial v'} & \frac{\partial x''}{\partial x'} \end{vmatrix} dv' \wedge dx'$$

$$= \begin{vmatrix} 1 & \tau \left(\frac{q_\alpha}{m_\alpha} \frac{\partial E(x'')}{\partial x'} \right) \\[2mm] 0 & 1 \end{vmatrix} dv \wedge dx$$

$$dv'' \wedge dx'' = dv' \wedge dx'$$

Noting that $dv' \wedge dx' = dv \wedge dx$ from *step 1*, it is seen that $dv'' \wedge dx'' = dv \wedge dx$ so that the phase volume is conserved as required. This stepthrough is a formal composition, so that we see composing two symplectic integrators produces another symplectic integrator. As can be seen a posteriori, there was no restriction on the number of steps taken just above. The stepthrough could have been taken at several fraction time steps each characterized by the same symplectic forms above. Thus, it is clear that compositions of any number of symplectic mappings preserves symplecticity. The above example brings to mind the suggestion mentioned at the closing of section 2.6.1. That is, it appears the only way to construct symplectic integrators is to use a split method so that an off-diagonal zero element appears in the Jacobian matrix; a conclusion emphasized by Güçlu. Thus, we claim *all* explicit Runge-Kutta-Nyström symplectic integrators must be split methods [27, p.716].

The basic prescription for seeking the higher order methods has been described previously in terms of exponential operators, but comparing the arguments of (2.58) and (2.59), we see that

$$f(t+\tau,x,v) = \mathcal{X}^\tau f(t,x,v) = f(t,x-\tau v,v) \quad \Rightarrow \quad x(t) = x(t+\tau) - \tau v(t)$$

$$f(t+\tau,x,v) = \mathcal{V}^\tau f(t,x,v) = f(t,x,v-\tau\frac{q_\alpha E(x)}{m_\alpha}) \quad \Rightarrow \quad v(t) = v(t+\tau) - \tau\frac{q_\alpha E(x)}{m_\alpha}\Big|_{x=x(t)}$$

We take a moment here to apprehend language that is commonplace in literature. It is often stated in works that these solutions are exact. However, the tacit perspective that accompanies this statement is that this is so when each problem is considered separately. Indeed, the time evolution of x above is exact only in the case of constant velocity, whereas the velocity evolution v would be exact if the positions were unchanged. Thus, while the advection operators do solve the individual split problems exactly as it is usually framed in publishings, their tandem application can only furnish approximate solutions to the coupled system as the time evolution of each state variable is tied to the continuous change of the other (cf. Fig. 2.13).
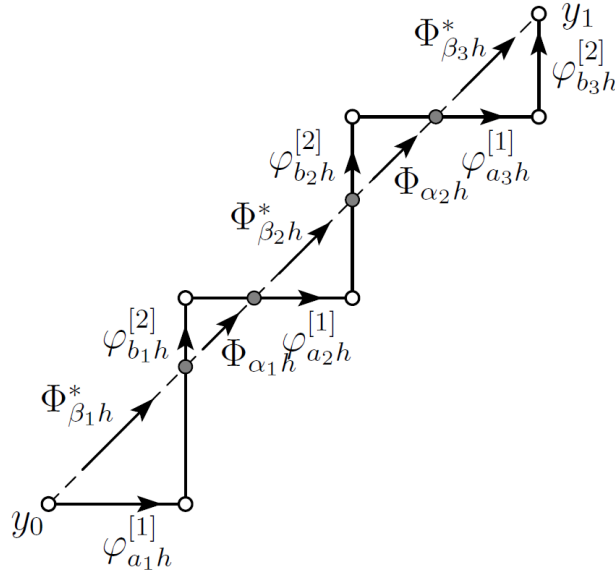
Figure 2.13: For a 2D problem, the propagation of a solution y from an initial value y_0 to a postpoint solution y_1 is shown over three substeps that amount to a full time step h ($\sum_i \beta_i = \sum_i a_i = 1$). The exact solution is shown generically as the directed diagonal line segment, which records the path of the true solution as accomplished by the exact integrator Φ_h^*. The composition of split integrators φ_h, each of which individually only convect the solution along one phase space variable while holding the other constant, are shown to approximate the final solution y_1. In general, the actual postpoint $\tilde{y}_1$ approximated by the split method is not exactly y_1, an error which depends on time. For an Nth order method for time step h, the error is $\mathcal{O}(h^{N+1})$ by our definition.

In fact, we notice these are first order accurate as the above are identical to the Euler method time marching scheme (eqs. (2.62) and (2.63)). In the interest of designing a splitting algorithm that preserves symplecticity, the time evaluation can be staggered such that we interpose symplectic Euler schemes (eqs. (2.67) and (2.68)). Thus, the important realization is made that the Lie (advection) operators $\mathcal{X}^\tau$ and $\mathcal{V}^\tau$ in a split scheme constitute a first order symplectic Euler scheme over a time step τ. An equivalent statement to (2.53) can then be assembled as the general problem of interleaving symplectic Euler method substeps of fractional time $c_i\tau$ and $d_i\tau$ that sum to a full time step τ in k stages ($\sum_i c_i\tau = \sum_i d_i\tau = \tau$). Taking $\tau \to c_i\tau, d_i\tau$, we see from above that the coefficients c_i and d_i enter the integrated characteristics equations:

$$x_i = x_{i-1} + c_i \tau v_{i-1}, \qquad\qquad x_k = x(t + \tau) \tag{2.69}$$

$$v_i = v_{i-1} + d_i \tau \left(\frac{q_\alpha E(x_i)}{m_\alpha} \right), \qquad v_k = v(t + \tau) \tag{2.70}$$

for each stage $i = 1, 2, \ldots, k$, which is a symplectic Euler stepthrough. Thus, rather than expanding exponentials (eq. (2.53)) or otherwise obtaining the coefficients by careful manipulation of the exponential arguments themselves per Yoshida [64] by using the Baker-Campbell-Hausdorff formula, the above two equations are straightforwardly matched up to the desired order of accuracy in Taylor series expansion:

$$x(t + \tau) = x(t) + \tau \dot{x} + \frac{1}{2}\tau^2 \ddot{x} + \ldots$$

$$v(t + \tau) = v(t) + \tau \dot{v} + \frac{1}{2}\tau^2 \ddot{v} + \ldots$$

whereupon physically identifying derivatives, we see that the order conditions take the following forms

$$x(t + \tau) = x(t) + \tau v + \frac{1}{2}\tau^2 \left(\frac{q_\alpha E(x)}{m_\alpha} \right)\Bigg|_{x=x(t)} + \ldots \tag{2.71}$$

$$v(t + \tau) = v(t) + \tau \left(\frac{q_\alpha E(x)}{m_\alpha} \right)\Bigg|_{x=x(t)} + \frac{1}{2}\tau^2 \left(\frac{q_\alpha}{m_\alpha} \frac{dE}{dt} \right)\Bigg|_{x=x(t)} + \ldots \tag{2.72}$$

To derive the coefficients required for a second order accurate method in time ($n = 2$) two stages of the above scheme are the minimum required ($k \geq 2$). However, it should be noted that split methods are not unique in that multiple schemes with varying stages k can satisfy the same order requirements. This idea is revisited in a moment. Let ① and ② denote stages evaluated at times t_1 and t_2 ($t < t_1 < t_2$ and $t_2 = t + \tau$) for corresponding quantities (x_1, v_1) and (x_2, v_2). Applying the advections (2.69) and (2.70) iteratively, we have the stepthrough from a prepoint (x, v) at time t in phase space to a postpoint (x_2, v_2) at time $t + \tau$

Stage ①

$$x_1 = x + c_1 \tau v$$

$$v_1 = v + d_1 \tau \left(\frac{q_\alpha E(x_1)}{m_\alpha} \right)$$

$$= v + d_1 \tau \frac{q_\alpha}{m_\alpha} E(x + c_1 \tau v)$$

$$= v + d_1 \tau \frac{q_\alpha}{m_\alpha} \left[E(x) + (c_1 \tau v)\frac{\partial E}{\partial x} + \ldots \right]$$

$$v_1 = v + d_1 \tau \frac{q_\alpha E(x)}{m_\alpha} + (c_1 d_1 \tau^2)\frac{q_\alpha}{m_\alpha} \frac{dE}{dt}\Bigg|_{x=x(t)} + \ldots$$

Where we have identified $v(\partial E/\partial x) = (dx/dt)(\partial E/\partial x) = dE/dt$.

Stage ②

$$
\begin{aligned}
x_2 &= x_1 + c_2 \tau v_1 \\
&= (x + c_1 \tau v) + c_2 \tau \left[v + d_1 \tau \frac{q_\alpha E(x)}{m_\alpha} + (c_1 d_1 \tau^2) \frac{q_\alpha}{m_\alpha} \frac{dE}{dt} \Big|_{x=x(t)} + \dots \right]
\end{aligned}
$$

$$
x(t+\tau) \equiv x_2 = x + (c_1 + c_2)\tau v + c_2 d_1 \tau^2 \left(\frac{q_\alpha E(x)}{m_\alpha} \right) + \mathcal{O}(\tau^3) \tag{2.73}
$$

and,

$$
\begin{aligned}
v_2 &= v_1 + d_2 \tau \left(\frac{q_\alpha E(x_2)}{m_\alpha} \right) \\
&= \left[v + d_1 \tau \frac{q_\alpha E(x)}{m_\alpha} + (c_1 d_1 \tau^2) \frac{q_\alpha}{m_\alpha} \frac{dE}{dt} \Big|_{x=x(t)} + \dots \right] + d_2 \tau \frac{q_\alpha}{m_\alpha} E\left(x + (c_1 + c_2)\tau v + \dots\right) \\
&= \left[v + d_1 \tau \frac{q_\alpha E(x)}{m_\alpha} + (c_1 d_1 \tau^2) \frac{q_\alpha}{m_\alpha} \frac{dE}{dt} \Big|_{x=x(t)} + \dots \right] + d_2 \tau \frac{q_\alpha}{m_\alpha} \left[E(x) + (c_1 + c_2)\tau v \frac{\partial E}{\partial x} \Big|_{x=x(t)} + \dots \right] \\
&= \left[v + d_1 \tau \frac{q_\alpha E(x)}{m_\alpha} + (c_1 d_1 \tau^2) \frac{q_\alpha}{m_\alpha} \frac{dE}{dt} \Big|_{x=x(t)} + \dots \right] + d_2 \tau \frac{q_\alpha}{m_\alpha} \left[E(x) + (c_1 + c_2)\tau \frac{dE}{dt} \Big|_{x=x(t)} + \dots \right]
\end{aligned}
$$

$$
v(t+\tau) \equiv v_2 = v + (d_1 + d_2)\tau \frac{q_\alpha E(x)}{m_\alpha} + [c_1(d_1 + d_2) + c_2 d_2]\tau^2 \frac{q_\alpha}{m_\alpha} \frac{dE}{dt} \Big|_{x=x(t)} + \mathcal{O}(\tau^3) \tag{2.74}
$$

Matching the coefficients in each order of τ of the compositions and the Taylor expansions gives equations that determine the constants c_1, c_2, d_1, d_2. Beginning with the x advection, we compare (2.73) with (2.71):

$$
\begin{aligned}
\mathcal{O}(\tau) &: \quad c_1 + c_2 = 1 \\
\mathcal{O}(\tau^2) &: \quad c_2 d_1 = \frac{1}{2}
\end{aligned}
$$

Two additional equations are provided by comparing composition (2.74) with Taylor series (2.72):

$$
\begin{aligned}
\mathcal{O}(\tau) &: \quad d_1 + d_2 = 1 \\
\mathcal{O}(\tau^2) &: \quad c_1(d_1 + d_2) + c_2 d_2 = \frac{1}{2}
\end{aligned}
$$

These order constraints are consistent with the general result for $k \geq n$ stages for $n = 2$ accuracy in time as quoted in [64, p.263]:

$$
\begin{aligned}
c_1 + c_2 + \dots + c_k &= 1 \\
d_1 + d_2 + \dots + d_k &= 1 \\
c_1(d_1 + d_2 + \dots + d_k) + c_2(d_1 + d_2 + \dots + d_k) + c_k d_k &= \frac{1}{2}
\end{aligned}
$$

A simple solution to the set of simultaneous equations above yield $c_1 = c_2 = 1/2, d_1 = 1, d_2 = 0$. The derivation above only required expanding the electric field to first order accuracy, which is ensured by evaluating the electric field after the first x-step, i.e. at $x + v\tau/2$. Thus, the interleaving in time of two symplectic Euler schemes over a time t to $t + \tau$ amounts to

(a) $x(t + \frac{\tau}{2}) = x(t) + \frac{1}{2}\tau v(t)$

(b) $v(t + \tau) = v(t) + \tau \frac{q_\alpha E(x)}{m_\alpha}\Big|_{x=x(t+\tau/2)}$

(c) $x(t + \tau) = x(t + \frac{\tau}{2}) + \frac{1}{2}\tau v(t + \tau)$

Or, equivalently in terms of operator notation, the solution to the numerical solution of the Vlasov-Poisson system can be recorded as

$$\boxed{f(t + \tau, x, v) = \mathcal{X}_{t+\tau/2}^{\tau/2} \circ \mathcal{V}_t^\tau \circ \mathcal{X}_t^{\tau/2} f(t, x, v) + \mathcal{O}(\tau^3)}$$
<div align="right">Strang (LF2) splitting</div>

Here, a redundant subscript has been introduced to transparently communicate at what time each advection operator begins. Later, we refer to this scheme as LF2 ("Leapfrog 2nd order") to be consistent with the notation of Blanes et. al, As before, the superscript indicates the time step increment. Substituting successively the steps of the scheme just above into the arguments of the distribution function, we see that

$$f(t + \tau, x, v) = f(t, x - \tau v - \frac{1}{2}\frac{q_\alpha E(\bar{x})}{m_\alpha}\tau^2), v - \frac{q_\alpha E(\bar{x})}{m_\alpha}\tau)$$

So that,

$$x(t) = x(t + \tau) - \tau v(t + \tau) - \frac{1}{2}\frac{q_\alpha E(t + \tau/2, \bar{x})}{m_\alpha}\tau^2$$

$$v(t) = v(t + \tau) - \frac{q_\alpha E(t + \tau/2, \bar{x})}{m_\alpha}\tau$$

Where the position at half-time step $\bar{x} \equiv x_1 = x + v\tau/2$. This is a statement of the *Leapfrog*, or *Störmer-Verlet*, algorithm. Finally, this composition follows a fortiori from the stronger result that Blanes [5] reminds us of. That is, the composition of a lower order integrator φ_τ with its adjoint $\varphi_\tau^\dagger$ yields a higher order integrator. For example, recall the first order integrator (2.61), $\varphi_{\tau,1st} := \mathcal{X}^\tau \circ \mathcal{V}^\tau$, that approximates the exact time evolution operator $T^\tau = \varphi_{\tau,1st} + \mathcal{O}(\tau^2)$. The Strang splitting scheme is reproduced as follows

$$\varphi_{2\tau,2nd} \equiv \varphi_{\tau,1st} \circ \varphi_{\tau,1st}^\dagger = \mathcal{X}^\tau \circ \mathcal{V}^\tau \circ \mathcal{V}^\tau \circ \mathcal{X}^\tau = \mathcal{X}^\tau \circ \mathcal{V}^{2\tau} \circ \mathcal{X}^\tau$$

over a time step 2τ. Clearly, this is equivalent to

$$T^\tau = \varphi_{\tau,2nd} + \mathcal{O}(\tau^3) = \mathcal{X}_{t+\tau/2}^{\tau/2} \circ \mathcal{V}_t^\tau \circ \mathcal{X}_t^{\tau/2} + \mathcal{O}(\tau^3)$$

In turn, this second order integrator can be used as a building block to construct higher order schemes.

2.6.3   Higher order integrators

It is clear from the previous section that while the approach to obtain higher order methods is straightforward, the algebraic burden to obtain even modest orders of accuracy for a minimum number of stages quickly becomes nontrivial. As suggested at the closing of the previous section, a popular method is to compose lower order schemes in order to build higher order integrators. Specifically, [64] later put forth that eq. (2.53) is equivalent to

$$\prod_{i=1}^{n} \exp(c_i \tau \Lambda_x) \exp(d_i \tau \Lambda_v) = \exp(\tau(\Lambda_x + \Lambda_v) + \mathcal{O}(\tau^{n+1})), \qquad \sum_i c_i = \sum_i d_i = 1 \qquad (2.75)$$

In this way, the Baker-Campbell-Hausdorff (BCH) formula can be applied iteratively, which states for two operators $X = c_1 \Lambda_x$, and $Y = d_1 \Lambda_v$ for the lowest order method, the exponential product can be written as a single exponential as given by

$$\exp X \exp Y = \exp Z$$

where Z is found to be

$$Z = X + Y + \frac{1}{2}[X,Y] + \frac{1}{12}([X,[X,Y]] + [Y,[Y,X]])$$
$$- \frac{1}{24}[Y,[X,[X,Y]]] - \frac{1}{720}([[[[X,Y],Y],Y],Y] + [[[[Y,X],X],X],X]) + \ldots$$

and the commutator $[X,Y] = XY - YX$. To see how higher order methods can be obtained, we consider proving the Strang splitting scheme. Assembling the Strang splitting scheme by composing $\exp X \exp Y \exp X = \exp(\frac{1}{2}\tau\Lambda_x)\exp(\tau\Lambda_v)\exp(\frac{1}{2}\tau\Lambda_x)$ by iterating on the BCH formula to find $(\exp X \exp Y)\exp X = \exp W$, The argument W is found to be [64]:

$$W = \tau\alpha_1 + \tau^3\alpha_3 + \tau^5\alpha_5 + \mathcal{O}(\tau^7)$$

where $\alpha_1 = \Lambda_x + \Lambda_v$, $\alpha_3 = \frac{1}{12}[\Lambda_v,[\Lambda_v,\Lambda_x]]$, $\alpha_5 = \frac{7}{5760}[\Lambda_x,[\Lambda_x,[\Lambda_x,[\Lambda_x,[\Lambda_v]]]]]$. Thus, truncating the series after the first term assures we are accurate to $\mathcal{O}(\tau^2)$, and the factor of unity in front of the α_1 term confirms our splitting coefficients were correct. Alternatively, if we did not know the coefficients a priori, we could have modelled the coefficients as c_1, c_2 and d_1, d_2 in order expand as above to obtain the necessary order conditions. Higher orders can be obtained by composing additional exponentials and ascertaining the exponential argument by iteration of the BCH formula so that we are able to directly manipulate the arguments of the exponentials alone without the need to expand the exponentials to find and match order conditions. Thus, careful handling of the terms that develop permits extracting the necessary coefficients for a given scheme expediently, where the coefficients are easily computed numerically. Blanes et. al optimized the search for coefficients with respect to a defined error [5], giving rise to popular methods used later (O6-4, O11-6).

Chapter 3

Preliminary work

This chapter presents the research approach and the results accomplished thus far. The aim of this thesis is to develop a computational foundation that furnishes efficient, high order numerical kinetic simulations of edge plasma with self-consistent field calculations in the context of plasmas in the magnetic confinement devices. First, we formally define the discrete advection problem and present the algorithms for the schemes presented in chapter 2. We then present results for three schemes: (1) the classic convected scheme (CS), (2) a fifth order accurate method $FD5$ using finite differences to compute the correction terms, and (3) a class of Nth order accurate CS methods where Fourier transforms are used to calculate the correction terms (FN methods). As a first pass, we confirm they have been implemented properly by verifying their numerical order of accuracy through customary convergence analysis. After, we showcase several test cases including 1D advection with variable velocity, a 2D rotating system to investigate four different splitting schemes (LF2, Y4, O6-4, and O11-6), and finally provide an intermediate test case in handling the 1D-1V Vlasov-Poisson system. That is, the solution to a Vlasov equation is given with a prescribed electric field.

3.1 The discrete advection problem

The continuous problem is to seek the solution $f = f(t, x)$ of the hyperbolic partial differential (advection) equation (2.9) in one dimension subject to a periodic boundary condition:

Continuous advection problem

$$\frac{\partial f}{\partial t} + v\frac{\partial f}{\partial x} = 0, \qquad (x, v) \in \mathcal{M}, \quad t \in \mathbb{R}^+ \tag{3.1}$$

$$\text{initial condition: } f(0, x) = f_0(x) \qquad x \in [0, L] \tag{3.2}$$

$$\text{boundary condition: } f(t, x) = f(t, x + L) \qquad x \in \mathbb{R} \tag{3.3}$$

where the velocity $v \in \mathbb{R}$ in this test problem is a specified constant so that the continuous problem space is the manifold $\mathcal{M} = \mathbb{R} \times \mathbb{R}$, whereas a particular solution space for a given problem coincides with a line in the two-dimensional tangent bundle $x - v$ at a constant v over the entire real line. The initial distribution f_0 is defined over the domain $\mathbb{R}_x = [0, L]$, and is completed over all real numbers $\mathbb{R}$ by the periodicity requirement. Thus, the plasma is both infinite in extent and periodic.

For the discrete problem, at each time $t^n = n\Delta t$ we find a solution $f_h(t^n, x_i, v_j)$ that is an approximation to the exact solution $f(t^n, x_i, v_j)$ for all $(x_i, v_j) \in \mathcal{M}_h$, where the mesh $\mathcal{M}_h$ is the discretization of the manifold $\mathcal{M}$. For this model problem, the mesh (Fig. 3.1) is defined in terms of N_x one-dimensional cells $\{C_i\}_{i=0}^{N_x-1}$ at a constant value v in phase space,
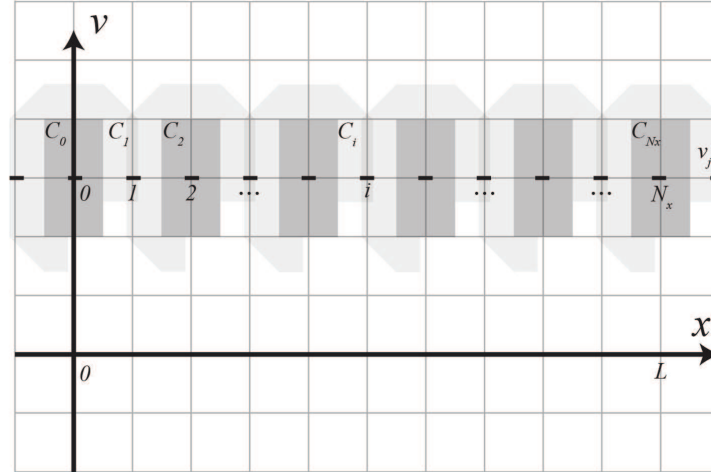
Figure 3.1: The mesh $\mathcal{M}_h$ is shown. For the 1D, one-speed, advection case for a periodic plasma, the mesh is a one-dimensional array of cells C_i, centered at a constant velocity $v = v_j$. The cells are illustrated as alternating shaded and unshaded objects with a nonzero height for clarity, but it should be noted that in actuality do not have any extent vertically as the velocity is constant.

$$\mathcal{M}_h = \cup_i C_i, \qquad C_j \cap C_k = \varnothing \quad \text{for } j \neq k, \qquad i, j, k = 0, 1, \dots, N_x - 1$$

Where the cell boundaries are defined at half-distances $x_{i\pm 1/2} = x_i \pm \frac{\Delta x}{2}$ whose cell-centers are located at each $\bar{C}_i = x_i$.

$$C_i = ([x_{i-1/2}, x_{i+1/2}], v_j), \qquad x_0 = 0, \ x_{N_x - 1} = L$$

Hence, we always refer to grid points as simple $\{x_i\}$ since their positions do not change with time, whereas convected cell-centers at a time t^n are denoted as $\{x_i^n\}$, or sometimes as $\{x_{i'}^n\}$. The widths $\Delta x_i = \Delta x = L/(N_x - 1)$ are chosen to be uniform such that $x_i = x_0 + i\Delta x$ produces the increasing sequence $(x_i)_{i=0}^{N_x - 1}$ that spans a full period $[0, L]$. We complete the sequence over all integers $i \in \mathbb{Z}$ by periodicity, which is most efficiently accomplished using modular arithmetic. In this way, although the system is infinite, all the unique information about its dynamics are contained within a single period L, so that numerical simulation is only required over the finite interval $[0, L]$. Thus, the discrete problem is summarized as follows:

Discrete advection problem

$$\frac{\partial f}{\partial t} + v\frac{\partial f}{\partial x} = 0, \qquad (x, v) \in \mathcal{M}, \quad t^n \in (0, T] \tag{3.4}$$

$$\text{initial condition:} \quad f(0, x_i) = f_0(x_i), \qquad \forall x_i \in [0, L] \tag{3.5}$$

$$\text{boundary condition:} \ f(t^n, x_i) = f(t^n, x_{i \bmod N_x}), \qquad \forall x_i \in \mathbb{R} \tag{3.6}$$

$$i = 0, 1, \ldots, N_x - 1, \quad \Delta x = \frac{L}{N_x}, \qquad x_i = x_0 + i\Delta x, \ x_0 = 0, \ x_{N_x} = L \tag{3.7}$$

$$n = 0, 1, \ldots, N_t, \quad \Delta t = \frac{T}{N_t}, \qquad t^n = n\Delta t, \qquad t^0 = 0, \ t^{N_t} = T \tag{3.8}$$

The solution to the discrete problem gives the approximation $f_h(t, x_i) \approx f(t, x_i)$ at the constant $v = v_j$ for all i that exists on the mesh $\mathcal{M}_h$. Henceforth the subscript h is omitted whenever the meaning is clearly understood. Here, we take the centroid values $f_h(t^n, x_i) \equiv f_i^n$.

Defining the rectangle basis function $\mathrm{rect}(z)$, each cell can be used to assemble the approximate solution for the total distribution:

$$f_h(t^n, x) = \sum_{i \in \mathbb{Z}} f_i^n \, \mathrm{rect}\left(\frac{x - x_i^n}{\Delta x}\right), \qquad \text{where } \mathrm{rect}(z) = \begin{cases} 0 & \text{if } |z| > \frac{1}{2} \\ \frac{1}{2} & \text{if } |z| = \frac{1}{2} \\ 1 & \text{if } |z| < \frac{1}{2} \end{cases}$$
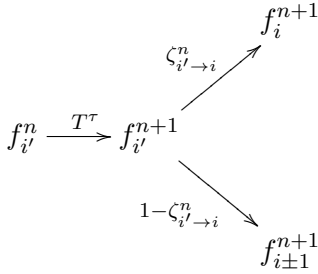
so that each loaded cell has the following form, which we have chosen to have compact support,

$$f_{h,i}(t^n, x) = f_i^n \, \mathrm{rect}\left(\frac{x - x_i^n}{\Delta x}\right), \qquad x \in \mathbb{R}$$

The solution is obtained by convecting densities f_i^n at time t^n located at prepoints $\{x_{i'}^n\} = \{x_i\}$ on the Eulerian grid along characteristics to final positions $\{x_{i'}^{n+1}\}$ over a single time step. The final position of the centroid of a convected cell is generally not coincident with any mesh point x_i. Density parcels are thus appropriated to cell-centers according to the CS remapping rule (section 2.4.2), which assigns the fraction of density in each MC, $C_{i'}^{n+1}$, to all overlapped cells $\{C_i\}$ in proportion the phase space volume overlap $\zeta_{i' \to i}^n$ assigned for the moving cell whose convection began at time t^n. Here, we have simplified the notation of

the fraction $\zeta_{i'j' \to ij}$ from section 2.4.2 given the velocity v_j is constant in this model problem $(j = j')$, and appended a time label n that marks the start time the convected MC is associated with. In general, this remap fraction is not constant in time nor is it uniform among all cells.

In the 1D case, each MC is remapped to exactly two adjacent cells, C_i and $C_{i\pm1}$. While the fraction $\zeta_{i' \to i}^n$ can be calculated for each MC relative to every fixed cell $C_i \in \mathcal{M}_h$, we save significant computational cost by calculating a single overlap fraction for a cell C_i for each $C_{i'}$, and use its stored value to deposit the remaining density $1 - \zeta_{i' \to i}^n$ to the other cell C_{i+1}. This also proves most natural for hard coding a particle conservation check at each remapping for careful error handling. In this way, we update each cell according to:

$$
\begin{aligned}
f_i^{n+1} &\mathrel{+}= \zeta_{i' \to i}^n f_{i'}^n \\
f_{i\pm1}^{n+1} &\mathrel{+}= (1 - \zeta_{i' \to i}^n) f_{i'}^n
\end{aligned}
$$

where the time evolution operator T^τ is shown on the right figure to evolve the solution over a time step (cf. section 2.6.1). To formalize the computation of the overlap fraction $\zeta_{i' \to i}^n$, we first note that each cell has constant density in configurational space, so that a moving cell with loaded density $f_{i'}^n$ at time t^n that was convected from $x_{i'}^n$ to a final location $x_{i'}^{n+1}$ at a time t^{n+1} is represented as

$$
f(t^{n+1}, x_{i'}) = f_{i'}^n \operatorname{rect}\left(\frac{x - x_{i'}^{n+1}}{\Delta x} \right)
$$

Then, the fraction of overlap between a cell C_i centered at x_i on the fixed Eulerian grid and the moving cell $C_{i'}$ is given by

$$
\zeta_{i' \to i}^n = \int \operatorname{rect}\left(\frac{x - x_i}{\Delta x} \right) \operatorname{rect}\left(\frac{x - x_{i'}^{n+1}}{\Delta x} \right) dx = \operatorname{tri}\left(\frac{x_i - x_{i'}^{n+1}}{\Delta x} \right); \qquad \operatorname{tri}(z) = \begin{cases} 1 - |z| & \text{if } |z| < 1 \\ 0 & \text{else} \end{cases} \tag{3.9}
$$

where the defined antiderivative of the rectangle function $\operatorname{tri}(z) = \int dz \operatorname{rect}(z)$ is the triangle function.

While the interpretation of the cell overlap is most physical, in practice it is more direct to work with a fraction $\alpha_{i' \to i}^n$ that measures the distance between cell-centers rather than cell boundaries which is the measure that defines $\zeta_{i' \to i}^n$. Thus, it is convenient to not focus on the trajectories themselves, but the overall grid shift that amounts from the convection. In general, the moving cells may be convected by more than one cell spacing Δx from their prepoints. By introducing the CFL parameter $\mathcal{C}_{i'}^n$ for the cell i' at a time step beginning at t^n, we can decompose the shift into two parts:

$$
\mathcal{C}_{i'}^n := S_{i'}^n + \alpha_{i' \to i}^n, \qquad S_{i'}^n \in \mathbb{Z}, \; \alpha_{i' \to i}^n \in (-1, 1) \tag{3.10}
$$

such that $x_{i'}^n \mapsto x_{i'}^{n+1}$, where $x_{i'}^{n+1} = [x_{(i' + S_{i'}^n) \bmod N_x} + \alpha_{i' \to i}^n \Delta x] \bmod L$

The parameter $S_{i'}^n$ is the integer part of the shift of the moving cell $C_{i'}$ in a single time step that began at time t^n with position $x_{i'}^n \equiv x_{i'}$ on the fixed grid and convected to a location $x_{i'}^{n+1} = x_{(i' + \mathcal{C}_{i'}^n)}$ as defined above on the interval $[0, L]$ per periodic boundary conditions. The fraction $\alpha_{i' \to i}^n$ is the normalized distance between the centroid of the final position of the MC and a nearest neighbor grid point on the fixed mesh $x_{(i' + S_{i'}^n) \bmod N_x}$,

which depends on how one chooses to define $S_{i'}^n$ (see below). Again, we have used the convention where the omission of the time value superscript indicates grid points on the fixed mesh. Introducing the fraction $\alpha_{i' \to i}^n$ removes the need to invoke a triangle function as is used for the calculation of $\zeta_{i' \to i}^n$ (eq. (3.9)).

There is flexibility in the definition of the two terms in (3.10); namely, we can choose the integral shift $S_{i'}^n$ to undershoot (resp. overshoot) the exact final position $(i + \mathcal{C}_{i'}^n) \bmod N_x$, which then fixes the sign of the fraction $\alpha_{i' \to i}^n$ as either positive (resp. negative) for $v \geq 0$ or negative (resp. positive) for $v < 0$. The choice of how these terms are defined, in turn, determines the proportion remapped to each grid point. We choose the general rule that $S_{i'}^n$ and $\alpha_{i' \to i}^n$ have the same sign. Thus, the velocity v and the fraction $\alpha_{i' \to i}^n$ also have the same sign. In physical terms, this amounts to defining the integer shift $S_{i'}^n$ to always undershoot the final location in the direction of convection, so that $\alpha_{i' \to i}^n \geq 0$ (resp. $\alpha_{i' \to i}^n < 0$) when $v \geq 0$ (resp. $v < 0$). Thus,

$$\alpha_{i' \to i}^n = \mathcal{C}_{i'}^n - S_{i'}^n, \qquad \text{where } S_{i'}^n = \lfloor \mathcal{C}_{i'}^n \rfloor \qquad\qquad (v \geq 0)$$

$$\alpha_{i' \to i}^n = \mathcal{C}_{i'}^n - S_{i'}^n, \qquad \text{where } S_{i'}^n = \lceil \mathcal{C}_{i'}^n \rceil \qquad\qquad (v < 0)$$

Since the CS is a semi-Lagrangian method, its stability is not restricted by the value of the Courant parameter; however, it naturally appears in the stepthrough of the convection process.

Thus, the CS update statement is written as:

<u>Non-uniform $v \geq 0$</u>

$$f_{(i+S_{i'}^n) \bmod N_x}^n \mathrel{+}= (1 - \alpha_{i' \to i}^n)\, f_{i'}^n \tag{3.11a}$$

$$f_{(i+S_{i'}^n+1) \bmod N_x}^n \mathrel{+}= \alpha_{i' \to i}^n f_{i'}^n \tag{3.11b}$$

<u>Non-uniform $v < 0$</u>

$$f_{(i+S_{i'}^n) \bmod N_x}^n \mathrel{+}= \quad (1 + \alpha_{i' \to i}^n)\, f_{i'}^n \tag{3.12a}$$

$$f_{(i+S_{i'}^n+1) \bmod N_x}^n \mathrel{+}= -\alpha_{i' \to i}^n f_{i'}^n \tag{3.12b}$$

where a complete mesh update amounts to looping over all cells moving cells $C_{i'}$. We note in closing that the unfactored form of the above equations represent an update in terms of normalized "fluxes" $\alpha_{i' \to i}^n f_{i'}^n$; a form that will be favored in the following section when higher order corrections are included.

Finally, we specialize the above to the specific case of a uniform constant speed ($v \geq 0$) for the model problem (3.4), which is the case at hand. All MCs are convected the same amount $v\Delta t$. Thus, the implementation can be significantly optimized by noting the fractions $\alpha_{i' \to i}^n$ (and, hence $\zeta_{i' \to i}^n$) are constant for all MCs at each time step. It is opportune to compute this single value $\alpha \equiv \alpha_{i' \to i}^n$ at the start, and employ its stored value as needed. The constant speed case will be considered in all schemes with the understanding that no generality is lost given that each MC presents a CS problem with constant speed at each time step.

3.1.1   Classic convected scheme algorithm

Using the framework developed in section 2.4.1, the implementation for the uniform case is summarized below *algorithm 1*. The

---

Algorithm 1: Classic convected scheme solution to the one-speed 1D advection equation

---

1. Load cell-centers with densities $\{f_i^0\}$ from initial condition for all $i = 0, 1, 2, \ldots, N_x$.

2. Compute normalized cell displacements $C = \frac{v\Delta t}{\Delta x}$ and decompose into integral and fractional parts, S and ζ, respectively

$$C = S + \alpha, \qquad S \in \mathbb{Z}, \alpha \in [0, 1) \subset \mathbb{R}$$

   where $S = \lfloor C \rfloor, \quad \alpha = C - S$

3. Convect density parcels at each $x_i^n \equiv x_i$ on the fixed grid by an integral shift S to an intermediate grid point $x_{i'}^n$

$$f(t^n, x_i) \mapsto f(t^n, x_{i'}) \qquad i = 0, 1, 2, \ldots, N_x$$

   where $x_{i'}^n = x_{(i+S) \bmod N_x}$

4. Assign integer-shifted densities $f_{i'}^n$ to cell-centers $\{x_i\}$ according to the fraction α. For nonnegative velocities, eqs. (3.11) provide the update

$$f_{i' \bmod N_x}^{n+1} \qquad += (1 - \alpha) f_{i'}^n$$

$$f_{(i'+1) \bmod N_x}^{n+1} \quad += \alpha f_{i'}^n$$

   where $i' = (i + S) \bmod N_x$. A similar update is used for negative velocities (eqs. (3.12)).

3.1.2   High order convected scheme algorithm

Regarding the theory developed in section 2.5, the algorithm with respect to the discrete problem is presented below (section 3.1).

Algorithm 2: (General) high order convected scheme solution to the one-speed advection equation

---

1. Load cell-centers with densities $\{f_i^0\}$ from initial condition for all $i = 0, 1, 2, \ldots, N_x$.

2. Compute normalized cell displacements $C = \frac{v \Delta t}{\Delta x}$ and decompose into integral and fractional parts, S and ζ, respectively

$$C = S + \alpha, \qquad S \in \mathbb{Z}, \alpha \in [0, 1) \subset \mathbb{R}$$

   where $S = \lfloor C \rfloor, \quad \alpha = C - S$

3. Convect density parcels at each $x_i^n \equiv x_i$ on the fixed grid by an integral shift S to an intermediate grid point $x_{i'}^n$

$$f(t^n, x_i) \mapsto f(t^n, x_{i'}) \qquad i = 0, 1, 2, \ldots, N_x$$

   where $x_{i'}^n = x_{(i+S) \bmod N_x}$

4. Calculate $N - 1$ weighted derivatives coefficients d_q^n required for Nth order accuracy (e.g. according to (3.13) for $N = 5$):

$$d_q^n = (\Delta x)^q \frac{\partial^q f}{\partial x^q}\bigg|_i^n$$

5. Calculate $N - 1$ correction coefficients c_q:

$$c_q = (-1)^q \beta_q(\alpha)$$

   where the functions $\beta_q(\alpha)$ are computed according to (2.37).

6. Calculate the nominal (normalized) fluxes Γ_i^n from eq. (2.38), and select corrected flux $[Uf]_i^n$ according to the limiter (2.39):

$$\Gamma_i^n = \sum_{q=0}^{N-1} c_q d_q^n, \qquad U_i^n f_{i'}^n = \max[\min(0, \Gamma_i^n), f_i^n]$$

7. Assign integer-shifted densities $f_{i'}^n$ to cell-centers $\{x_i\}$ according CS remapping rule in flux form:

$$f_{i' \bmod N_x}^{n+1} \quad += f_{i'}^n - U_i^n f_{i'}^n$$

$$f_{(i'+1) \bmod N_x}^{n+1} \quad += U_i^n f_{i'}^n$$

   Where $i' = (i + S) \bmod N_x$. A similiar update is used for negative velocities (eqs. (3.12)).

To compute the terms d_q^n in algorithm 2, any means for numerically calculating the contained derivatives may be employed so long as it matches the desired order conditions of the overall method. The coefficients d_q^n involving derivatives have the form:

$$d_q^n = (\Delta x)^q \frac{\partial f}{\partial x}\bigg|_i^n, \qquad q = 1, 2, \dots N - 1$$

That is, the estimation for an order N method must satisfy:

$$(\Delta x)^q \frac{\partial^q f}{\partial x^q}\bigg|_i^n = d_q^n + \mathcal{O}(\Delta x^{N+1})$$

Since the derivatives are multiplied by a $(\Delta x)^q$ ($q \geq 1$), the numerical approximation of the derivatives has the relaxed constraint on the local truncation error (LTE) for each term:

$$\text{LTE}\left(\frac{\partial^q f}{\partial x^q}\bigg|_i^n\right) = \mathcal{O}(\Delta x^{N+1-q}), \qquad \underline{\text{Correction criterion on derivative estimates}}$$

For example, a 5th order method requires a first derivative approximation so that its greatest LTE is of order $(5 + 1 - 1) = 5$, a second derivative estimate requires the error to be no larger than order $(5 + 1 - 2) = 4$, and so on. Thus, the strictest requirement is always on the estimation of the first derivative. Two methods for calculating these derivatives are reviewed below.

3.1.3   Finite difference corrections

A central finite difference approximation with a stencil $\{i + r\}$ where $r = \{-2, -1, 0, 1, 2\}$ supplies the following estimates accurate up to the required order for each derivative for an overall 5th order [24]:

$$(\Delta x)\frac{\partial f}{\partial x}\bigg|_i^n = \frac{1}{12}f_{i-2}^n - \frac{2}{3}f_{i-1}^n + \frac{2}{3}f_{i+1}^n - \frac{1}{12}f_{i+2}^n + \mathcal{O}(\Delta x^5) \tag{3.13a}$$

$$(\Delta x)^2\frac{\partial^2 f}{\partial x^2}\bigg|_i^n = -\frac{1}{12}f_{i-2}^n + \frac{4}{3}f_{i-1}^n + -\frac{5}{2}f_i^n + \frac{4}{3}f_{i+1}^n - \frac{1}{12}f_{i+2}^n + \mathcal{O}(\Delta x^6) \tag{3.13b}$$

$$(\Delta x)^3\frac{\partial^3 f}{\partial x^3}\bigg|_i^n = -\frac{1}{2}f_{i-2}^n + f_{i-1}^n - f_{i+1}^n + \frac{1}{2}f_{i+2}^n + \mathcal{O}(\Delta x^5) \tag{3.13c}$$

$$(\Delta x)^4\frac{\partial^4 f}{\partial x^4}\bigg|_i^n = f_{i-2}^n - 4f_{i-1}^n + 6f_i^n - 4f_{i+1}^n + f_{i+2}^n + \mathcal{O}(\Delta x^6) \tag{3.13d}$$

So that the overall method calculates a density f with an local truncation error $\mathcal{O}(\Delta x^{5+1})$. In this work, we refer to a method that employs these finite difference (FD) approximations for the derivatives required in the anti-diffusive correction accurate to fifth order as *FD5*. Note, this is equivalent to what Güçlu refers to as "P6" [27]. We choose to label in this alternative manner due to the personal preference of the author; this convention is also used in numerical methods literature elsewhere. That is, we refer to an Nth order method as being correct up to order $\mathcal{O}(\Delta x^N)$ with respect to local truncation error which is the usual definition that appears in (for example) the literature of operating splitting methods, whereas others prefer to define an Nth order method such that the largest error term is of order $\mathcal{O}(\Delta x^N)$ (e.g. [27]).

It is noted by Güçlu [27] and proven in [24] that a general recipe for obtaining finite difference coefficients is provided by taking derivatives of the appropriate order Lagrange interpolating polynomial (which is the namesake for Güçlu's P6 scheme). Thus, it is at least straightforward to derive higher order versions as needed; however, it quickly becomes computationally expensive (cf. [57]).

3.1.4   Spectral corrections

The derivative calculation cost can be reduced as compared to a finite difference estimate by computing the derivatives in Fourier space whereafter an inverse can be taken to recover approximations to the same derivatives in physical space. Using a fast Fourier transform (FFT) algorithm, the number of operations is reduced to $\mathcal{O}(N_x \log_2 N_x)$ per time step.

Conceptually, we expand a function $f(t,x) \equiv f(x)$ in space at a given time t according to the series

$$f(x) = \sum_{k=-\infty}^{\infty} \mathcal{F}[f](k) e^{\mathrm{j}\xi_k x}$$

where the wave number $\xi_k = 2\pi k/L$, and the time dependence has been suppresed for brevity given we transform the configurational variable x. The imaginary unit $\mathrm{j} := \sqrt{-1}$ is formatted in plain text to avoid any confusion with the velocity index j previously employed. The choice of this *orthogonal* basis fixes the form of the Fourier coefficients $\mathcal{F}[f](k)$ by consequence; these coefficients are the projections of the density $f(x)$ onto the subspace spanned by each complex exponential basis characterized by the wave number ξ_k. The projection is given by the inner product, which is known as the Fourier transform:

$$\mathcal{F}[f](\xi_k) = \frac{1}{L} \int_{-\infty}^{\infty} f(x) e^{-\frac{2\pi \mathrm{j}}{L} kx} dx$$

To shift to a discrete domain, we choose to center the grid so that the locations in configurational space are represented as $x_m = m\Delta x = m\frac{L}{N_x}$, $m = 0, 1, \ldots, N_x - 1$, as is the usual enumeration adopted when invoking Fourier methods given the notational convenience inherited by preserving symmetry. In order to define derivatives consistently, an interpolation is needed between each sample point x_m. Minimizing the mean-square slope determines the unique minimium oscillation *trigonometric interpolation* needed which requires the coefficient $\mathcal{F}[f](N_x/2)$ (N_x even), i.e. the Nyquist term, to be equally split between the positive and negative terms in the series [34]. The details of this stepthrough in Fourier space are visited in more depth in section 2.5. The transform pair is approximated in a discrete domain by the discrete Fourier transform (DFT) and inverse discrete Fourier transform (IDFT):

$$\mathcal{F}[f](\xi_k) \simeq \mathrm{DFT}[f]_k := \frac{1}{N_x} \sum_{m=0}^{N_x - 1} f(x_m) e^{-\mathrm{j}\xi_k x_m} \tag{3.14a}$$

$$f(x_m) \simeq \mathrm{IDFT}[\mathrm{DFT}[f]]_m := \sum_{k=0}^{N_x - 1} \mathrm{DFT}[f]_k e^{\mathrm{j}\xi_k x_m} \tag{3.14b}$$

Here, the subscripting k and m on the transforms indicate the resulting dependence of the transform (e.g. $\mathrm{DFT}[f]_k$ is a function of $\xi_k = \frac{2\pi k}{L}$). In algorithm 3 below, the transformed terms are referred to as $\{\hat{f}_k\}$. Differentiating the function $f(x_m)$ in the definition (3.14b) above, we see that differentiation is equivalent to complex multiplication operations in Fourier space,

$$\left. \frac{d^q f}{dx^q} \right|_{x=x_m} \simeq \frac{1}{N_x} \sum_{k=0}^{N_x - 1} (\mathrm{j}\xi_k)^q \mathrm{DFT}[f]_k e^{\mathrm{j}\xi_k x_m} = \mathrm{IDFT}\{(\mathrm{j}\xi_k)^q \mathrm{DFT}[f]_k\}$$

Each transform requires $\mathcal{O}(N_x^2)$ operations. Choosing to use a fast Fourier transform (FFT) and its corresponding inverse (IFFT), the transforms can be reduced to $\mathcal{O}(N_x \log_2 N_x)$ complexity. Thus, an FFT/IFFT prodedure can be substituted above in order to calculate each derivative as needed according to:

$$\boxed{\left.\frac{d^q f}{dx^q}\right|_{x=x_m} \simeq \text{IFFT}\{(\text{j}\xi_k)^q \text{FFT}[f]_k\}} \qquad \text{where } \xi_k = \frac{2\pi k}{L}, k = 0, 1, \ldots, N_x - 1 \qquad (3.15)$$

Because of the distinct procedure involved in the spectral calculation, a dedicated algorithm is presented below so that the particulars are clear. We refer to this method as *spectral-CS*, and identify an Nth order method that calculates derivatives in Fourier space as above, as an FN algorithm. For example, a 15th order method is called *F15*. Note, that a low-pass filter is used to zero out any Fourier coefficients that contribute insignificantly on their own, but whose usually high frequency (especially amplified at higher powers) presents a source of white noise. The algorithm 3 is given below in full:

Algorithm 3: (Spectral derivatives) high order convected scheme solution to the one-speed advection equation

---

1. Load cell-centers with densities $\{f_i^0\}$ from initial condition for all $i = 0, 1, 2, \ldots, N_x$.

2. Compute normalized cell displacements $C = \frac{v\Delta t}{\Delta x}$ and decompose into integral and fractional parts, S and ζ, respectively

    $$C = S + \alpha, \qquad S \in \mathbb{Z}, \alpha \in [0, 1) \subset \mathbb{R}$$

    where $S = \lfloor C \rfloor, \quad \alpha = C - S$

3. Convect density parcels at each $x_i^n \equiv x_i$ on the fixed grid by an integral shift S to an intermediate grid point $x_{i'}^n$

    $$f(t^n, x_i) \mapsto f(t^n, x_{i'}) \qquad i = 0, 1, 2, \ldots, N_x$$

    where $x_{i'}^n = x_{(i+S) \bmod N_x}$

4. Calculate the Fourier transform $\hat{f}_k^n$ of the density f_i^n via a fast Fourier transform (FFT) algorithm:

    $$\hat{f}_k = \text{FFT}[f], \qquad \text{if } |\hat{f}_k| \leq A\varepsilon, \text{then set} \hat{f}_k^n = 0$$

    where $\varepsilon = 2 \times 10^{-15}$ and $A = \max_k |\hat{f}_k^n|$).

5. Calculate the associated wave numbers ξ_k:

    $$\xi_k = \begin{cases} 2\pi k/L & \text{if } k \leq N_x/2 \\ 2\pi(k - N_x)/L & \text{else} \end{cases}$$

6. Calculate the $N - 1$ derivative coefficients $\hat{d}_q^n$ in Fourier space required for the desired order of accuracy N.

    $$\hat{d}_q^n = (\text{j}\xi_k)^q \hat{f}_k, \qquad q = 0, 1, \ldots, N - 1$$

7. Calculate $N - 1$ correction coefficients c_q:

    $$c_q = (-1)^q \beta_q(\alpha)$$

    where the functions $\beta_q(\alpha)$ are computed according to (2.37).

8. Calculate the nominal (normalized) fluxes $\hat{\Gamma}_i^n$ by assembling the equivalent of eq. (2.38)

    $$\hat{\Gamma}_i^n = \sum_{q=0}^{N-1} c_q \hat{d}_q^n$$

9. Calculate the flux in configurational space by applying an inverse fast Fourier transforom (IFFT), and select corrected flux $[Uf]_i^n$ according to the limiter (2.39):

    $$\Gamma_i^n = \text{Re}\left[\text{IFFT}\{\hat{\Gamma}_i^n\}\right], \qquad U_i^n f_{i'}^n = \max[\min(0, \Gamma_i^n), f_i^n]$$

    where the real part is applied to remove any residual (negligibly small) imaginary component artifacts that can result from the IFFT process.

10. Assign integer-shifted densities $f_{i'}^n$ to cell-centers $\{x_i\}$ according CS remapping rule in flux form:

    $$f_{i' \bmod N_x}^{n+1} \quad += f_{i'}^n - U_i^n f_{i'}^n$$

    $$f_{(i'+1) \bmod N_x}^{n+1} \quad += U_i^n f_{i'}^n$$

    Where $i' = (i + S) \bmod N_x$. A similiar update is used for negative velocities (eqs. (3.12)).

71

3.2   Results

For all simulations, we solve an advection equation with unit velocity $(v = 1)$

$$\frac{\partial f}{\partial t} + \frac{\partial f}{\partial x} = 0, \qquad x \in \mathcal{D}, t \in [0, T] \tag{3.16a}$$

$$f(0, x) = f_0(x), \quad f(t, x + L) = f(t, x) \tag{3.16b}$$

where the domain $\mathcal{D}$ and time duration T depends on the test case at hand. The initial distributions $f_0(x)$ are chosen to prove numerical order of accuracy, and thereafter we investigate test cases that challenge the fidelity of each scheme's calculations in telling situations. The boundary condition is, as usual, periodicity (cf. section 3.1). As described in the previous section, a mesh refinement exercise is conducted by holding the Courant-Friedrichs-Lewy (CFL) parameter $\mathcal{C}$ constant while increasing the resolution of the grid consistently. The problem can be set up so that this amounts to maintaining a simple relationship between the grid points N_x and N_t for a given simulation. Here, the time T and domain length L are decided so that a full simulation covers the advection of a density over a full domain length L in time T (i.e. $L/T = 1$). The CFL parameter for this unit velocity case is then given by: $\mathcal{C} = \frac{v \Delta t}{\Delta x} = \frac{1 \cdot N_t/T}{N_x/L} = \frac{N_x}{N_t}$. The value $\mathcal{C} = 0.32$ is selected so that any combination of $N_x = 8r, N_t = 25r$ for $r \in \mathbb{N}^+$ is permissible for each simulation. However, we note that N_x should be chosen as a power of 2 for spectral CS cases in order to take full advantage of the treecode algorithm on which all FFT procedures are based.

3.2.1   Verifying numerical order of accuracy

We choose to define an Nth order method when the local truncation error (LTE) is $\mathcal{O}(\Delta x^{N+1})$ (i.e. the numerical approximation is said to be Nth order accurate). To ensure the method has been implemented correctly, an error calculation of the simulation results for test cases where the exact solution is known can be straightforwardly computed. For the one-speed advection equations, the exact solution f is given by the method of characteristics as $f(t^n, x_i) = f_0(x_i - n\Delta t)$, where f_0 is the initial distribution so that the initial condition can be used to directly calculate values of the exact solution for any position and time. A standard measure of the error is the *global error* (GE), which is the error accumulated over a full simulation time $T = N_t \Delta t$, where N_t is the total number of time steps and Δt is the associated time step. Formal treatments of the error involved in semi-Lagrangian schemes and the conditions for convergence are provided by a series of papers by Besse et. al (e.g. [2]). The same order result can be obtained less rigorously by understanding the error over N_t time steps for an Nth order method whose LTE is by definition $\mathcal{O}(\Delta x^{N+1})$ is given by the accumulated error $N_t \cdot \mathcal{O}(\Delta x^{N+1}) = \frac{T}{\Delta t} \mathcal{O}(\Delta x^{N+1}) = \mathcal{O}(\Delta x^{N+1}/\Delta t)$, where the asymptotic behavior of the spatial and time error depend on how one is related to the other.

It is common practice to hold, say, Δt constant and to refine the mesh Δx in successive simulations in order to map the converged error on a logarithmic plot or otherwise provide a direct order calculation. For several numerical methods, the stability of the method asserts an upper limit on the Courant-Friedrichs-Lewy (CFL) parameter $\mathcal{C} := \frac{v \Delta t}{\Delta x}$, where v is a specified constant that characterizes the problem. While semi-Lagrangian methods have no CFL restriction; however, we note that they are unique in that there is zero error associated with convecting cells an integral number of cells, i.e. whenever $\mathcal{C} \in \mathbb{Z}$. Thus, following the prescription in a semi-Lagrangian scheme produces errors that vary nonmonotonically with the grid spacing Δx since the error dips to zero when $\mathcal{C}$ is an integer. Notwithstanding, the error of the scheme is still present as the enveloping curve that traces the upper bound of the error. Here, we choose instead to fix the CFL parameter between successive simulations so that by its definition $\mathcal{C} \Delta x = v \Delta t$, and the orders are simply related: $\mathcal{O}(\Delta t) = \mathcal{O}(\Delta x)$. Thus, the global error result above is $\mathcal{O}(\Delta x^{N+1}/\Delta t) = \mathcal{O}(\Delta x^{N+1}/\Delta x) = \mathcal{O}(\Delta x^N)$.

For an order N method, we thus will observe a global error (GE) of $\mathcal{O}(\Delta x^N)$ which is associated with a local truncation error (LTE) of $\mathcal{O}(\Delta x^{N+1})$.

To gauge a measure of the global error, we choose to work with the standard deviation of the numerical solution to the convected scheme $f_{\mathrm{CS}}(t^n, x_i)$ relative to the exact solution $f_{\mathrm{exact}}(t^n, x_i)$, whereafter an average value is recorded by dividing over the length of the spatial domain. Such an error measure is otherwise known as the *standard error* in statistics. In estimation theory, we regard the estimator $\hat{\theta}$ as the numerical solution $f_{\mathrm{CS}}(t^n, x_i)$ ($\hat{\theta} \equiv f_{\mathrm{CS}}(t^n, x_i)$) and the observable $\theta \equiv f_{\mathrm{exact}}(t^n, x_i)$, so that *normalized root-mean-square error* (NRMSE) is given by $\mathrm{NRMSE} = \sqrt{\mathbb{E}[(\theta - \hat{\theta})^2]}/L$ where L is the domain length. That is, we have elected to use the L^2 norm as an appropriate measure of the distance between solution and estimator given the solution exists in the Sobolev space $W^{k,p}$, where $W^{k,p} \supset W^{k,2}$ contains the Banach space that is Lebesgue integrable under the 2-norm so that our calculations are guaranteed to be finite and bounded.

Thus, the NRMS of the local truncation error ($\overline{\mathrm{LTE}}_{\Delta x}^n$) of a simulation with mesh spacing Δx can be calculated at a particular time step t^n as:

$$\overline{\mathrm{LTE}}_{\Delta x}^n = \left[\frac{\sum_{i=0}^{N_x-1} [f_{\mathrm{CS}}(t^n, x_i) - f_{\mathrm{exact}}(t^n, x_i)]^2 \Delta x}{\sum_{i=0}^{N_x-1} \Delta x} \right]^{1/2} = \frac{1}{\sqrt{L}} \left[\sum_{i=0}^{N_x-1} [f_{\mathrm{CS}}(t^n, x_i) - f_{\mathrm{exact}}(t^n, x_i)]^2 \Delta x \right]^{1/2}$$

where it is recognized that the sum in the denominator is equal to the length of the domain L. For an Nth order method, this measure will produce an error of $\mathcal{O}(\Delta x^{N+1})$. Similarly, the NRMS of the global error ($\overline{\mathrm{GE}}_{\Delta x}$) is found by evaluating the above at the simulation completion time $t^{N_t} = T$

$$\overline{\mathrm{GE}}_{\Delta x} = \frac{1}{\sqrt{L}} \left[\sum_{i=0}^{N_x-1} [f_{\mathrm{CS}}(T, x_i) - f_{\mathrm{exact}}(T, x_i)]^2 \Delta x \right]^{1/2} \tag{3.17}$$

For an Nth order method, this error should be on the order $\mathcal{O}(\Delta x^N)$. In order to calculate the order directly, we consider two successive simulations over the same time interval $[0, T]$ where one mesh size Δx (and N_x grid points) is twice as large as the other $\Delta x/2$ (and $2N_x$ grid points). The numerical order observed can be seen as extractable of the ratio of these two global errors:

$$\frac{\overline{\mathrm{GE}}_{\Delta x}}{\overline{\mathrm{GE}}_{\Delta x/2}} = \frac{\frac{1}{\sqrt{L}} \left[\sum_{i=0}^{N_x-1} [f_{\mathrm{CS}}(T, x_i) - f_{\mathrm{exact}}(T, x_i)]^2 \Delta x \right]^{1/2}}{\frac{1}{\sqrt{L}} \left[\sum_{i=0}^{N_x-1} [f_{\mathrm{CS}}(T, x_i) - f_{\mathrm{exact}}(T, x_i)]^2 \frac{\Delta x}{2} \right]^{1/2}} = \frac{\left[\sum_{i=0}^{N_x-1} [f_{\mathrm{CS}}(T, x_i) - f_{\mathrm{exact}}(T, x_i)]^2 \right]^{1/2}}{\left[\sum_{i=0}^{N_x-1} [f_{\mathrm{CS}}(T, x_i) - f_{\mathrm{exact}}(T, x_i)]^2 \left(\frac{1}{2} \right) \right]^{1/2}}$$

Noting that we are evaluating the final time $t^{N_t} = T$, we understand that the error terms are of order $\mathcal{O}(\Delta x^N)$ from the discussion just above. That is, there exists a constant C, independent of mesh size Δx, such that $|f_{\mathrm{CS}}(T, x_i) - f_{\mathrm{exact}}(T, x_i)| \leq C\Delta x^N$. Or, an equality can be asserted such that $|f_{\mathrm{CS}}(T, x_i) - f_{\mathrm{exact}}(T, x_i)| = C\Delta x^N + \mathcal{O}(\Delta x^{N+1})$. Lastly, noting that the effect of each sum amounts to multiplying the bounded squared error term by the number of grid points (N_x or $2N_x$), we can write:

$$\frac{\overline{\text{GE}}_{\Delta x}}{\overline{\text{GE}}_{\Delta x/2}} = \left\{\frac{N_x[C\Delta x^N + \mathcal{O}(\Delta x^{N+1})]^2}{(2N_x)[C\left(\frac{\Delta x}{2}\right)^N + \mathcal{O}\left(\frac{\Delta x}{2}\right)^{N+1}]^2 \left(\frac{1}{2}\right)}\right\}^{1/2}$$

$$= \left\{\frac{[C\Delta x^N + \mathcal{O}(\Delta x^{N+1})]^2}{[C\left(\frac{\Delta x}{2}\right)^N + \mathcal{O}\left(\frac{\Delta x}{2}\right)^{N+1}]^2}\right\}^{1/2}$$

$$= \left|\frac{C\Delta x^N + \mathcal{O}(\Delta x^{N+1})}{C\left(\frac{\Delta x}{2}\right)^N + \mathcal{O}\left(\frac{\Delta x}{2}\right)^{N+1}}\right|$$

$$= 2^N \frac{C\Delta x^N + \mathcal{O}(\Delta x^{N+1})}{C\Delta x^N + \frac{1}{2}\mathcal{O}(\Delta x^{N+1})}, \qquad (C > 0)$$

$$= 2^N \left(\frac{C\Delta x^N + \frac{1}{2}\mathcal{O}(\Delta x^{N+1})}{C\Delta x^N + \frac{1}{2}\mathcal{O}(\Delta x^{N+1})} + \frac{\frac{1}{2}\mathcal{O}(\Delta x^{N+1})}{C\Delta x^N + \frac{1}{2}\mathcal{O}(\Delta x^{N+1})}\right)$$

$$= 2^N \left(1 + \frac{\frac{1}{2}\mathcal{O}(\Delta x^{N+1})}{C\Delta x^N + \frac{1}{2}\mathcal{O}(\Delta x^{N+1})}\right)$$

$$\frac{\overline{\text{GE}}_{\Delta x}}{\overline{\text{GE}}_{\Delta x/2}} = 2^N \left(1 + \frac{\mathcal{O}(\Delta x)}{2C + \mathcal{O}(\Delta x)}\right)$$

In practice, we can refine the mesh so that Δx gets smaller and smaller ($\Delta x \to 0$). For $\mathcal{O}(\Delta x) \ll 1$, we have $\frac{\mathcal{O}(\Delta x)}{2C} \ll 1$, ($C > 0$) so that the above fraction can be properly expanded:

$$\frac{\mathcal{O}(\Delta x)}{2C + \mathcal{O}(\Delta x)} = \frac{\mathcal{O}(\Delta x)}{2C}\left[1 + \frac{\mathcal{O}(\Delta x)}{2C} + \frac{1}{2}\left(\frac{\mathcal{O}(\Delta x)}{2C}\right)^2 + \ldots\right] \simeq \frac{\mathcal{O}(\Delta x)}{2C} = \mathcal{O}(\Delta x)$$

So the ratio of NRMS global errors give the estimate in the limit of decreasing mesh spacing Δx:

$$\frac{\overline{\text{GE}}_{\Delta x}}{\overline{\text{GE}}_{\Delta x/2}} = 2^N(1 + \mathcal{O}(\Delta x))$$

Thus, it can be seen that operating with the base-2 logarithm recovers the numerical order of accuracy:

$$\log_2 \frac{\overline{\text{GE}}_{\Delta x}}{\overline{\text{GE}}_{\Delta x/2}} = \log_2 2^N(1 + \mathcal{O}(\Delta x))$$

$$= \log_2 2^N + \log_2(1 + \mathcal{O}(\Delta x))$$

$$= N + (\mathcal{O}(\Delta x) - \frac{1}{2}(\mathcal{O}(\Delta x))^2 + \frac{1}{3}(\mathcal{O}(\Delta x))^3 + \ldots)$$

$$\log_2 \frac{\overline{\text{GE}}_{\Delta x}}{\overline{\text{GE}}_{\Delta x/2}} = N + \mathcal{O}(\Delta x)$$

where the smallness of the parameter $\mathcal{O}(\Delta x) \ll 1$ has been exploited again to expand the logarithm. Thus, we see that in the limit of decreasing mesh spacing $\Delta x \to 0$, this computation exactly recovers the observed order of accuracy in the numerical simulation:

	Classic CS	
	NRMS(GE$_{\Delta x}$)	Order
N_x		
32	1.4448	–
64	1.0073	0.5204
128	6.0739×10^{-1}	0.7298
256	3.3537×10^{-1}	0.8569
512	1.7646×10^{-1}	0.9264
1024	9.0539×10^{-2}	0.9626
2048	4.5863×10^{-2}	0.9812

Table 3.1: Mesh refinement results for the classic convected scheme applied to the density (3.19). The normalized root mean square (NRMS) of the global error (GE) for each mesh with spacing $\Delta x = L/N_x$ is given by eq. (3.17), and the observed numerical order of accuracy is computed per (3.18). The CFL number $\mathcal{C} = 0.32$ for all simulations. The numerical order is converging towards $\mathcal{O}(\Delta x^1)$.
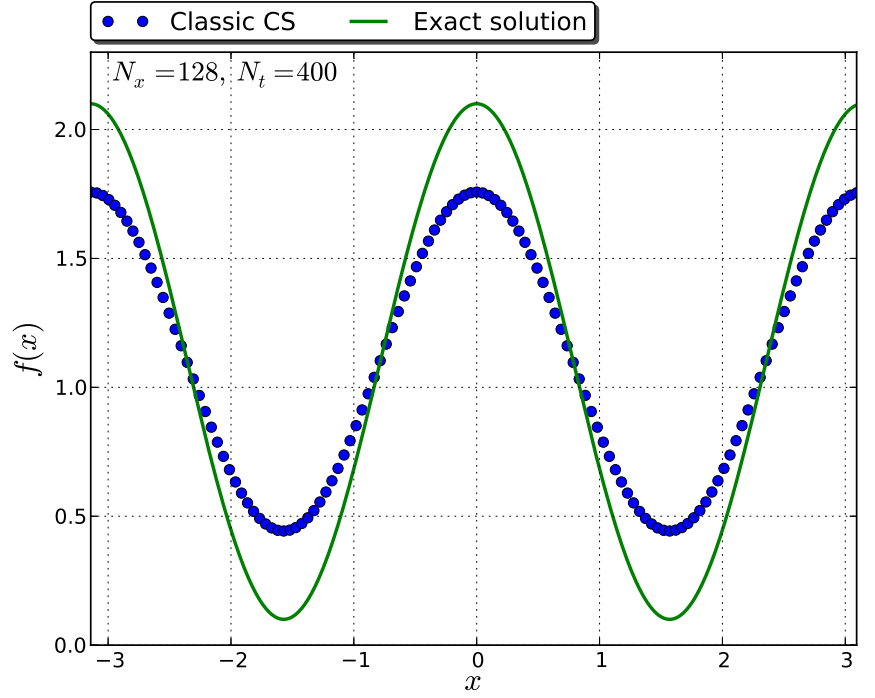

Figure 3.2: A classic CS solution after $N_t = 400$ time steps of eq. (3.23) for f_0 given by (3.19).

$$N := \log_2\left(\frac{\overline{\text{GE}}_{\Delta x}}{\overline{\text{GE}}_{\Delta x/2}}\right), \quad \Delta x \to 0 \qquad \underline{\text{Observed order of accuracy}} \qquad (3.18)$$

In practice, we perform a mesh refinement series where the same simulation is run for spacings Δx halved from one simulation to the next. This order computation approaches a value when the spacing Δx reaches sufficiently small values.

Convergence results

To demonstrate that the classic CS has been implemented properly, we consider a smoothly varying initial distributionand prescribe a domain of two full periods over a time interval $0 \leq t \leq 2\pi$:

$$f_0(x) = 1.1 + \cos(2x), \qquad x \in \mathcal{D} = [-\pi, \pi], \quad t \in [0, 2\pi], \quad \text{domain length } L = 2\pi \qquad (3.19)$$

The results are summarized in table 3.1, and one case of a numerical solution taken at the end of the simulation time is shown in figure 3.2.

It is seen that numerical diffusion contaminates the results and acts to artificially flatten the distribution over time. The effect is most pronounced in coarse grids where the remapping rule appropriates density over larger spatial extents (see figure 3.3). In this present case of a slowly varying density, this consequence is modest given the lack of fine structure that needs to be retained along many remap stages. Thus, this cosine distribution (3.19) permits the early suggestion ($N_x \sim 1024$, table 3.1) that the order of accuracy is converging

towards $\mathcal{O}(\Delta x^1)$, as is known from theory about semi-Lagrangian methods such as the CS [2] (i.e. the error is $\mathcal{O}(\Delta x^2)$). calculation at relatively coarse mesh sizes.
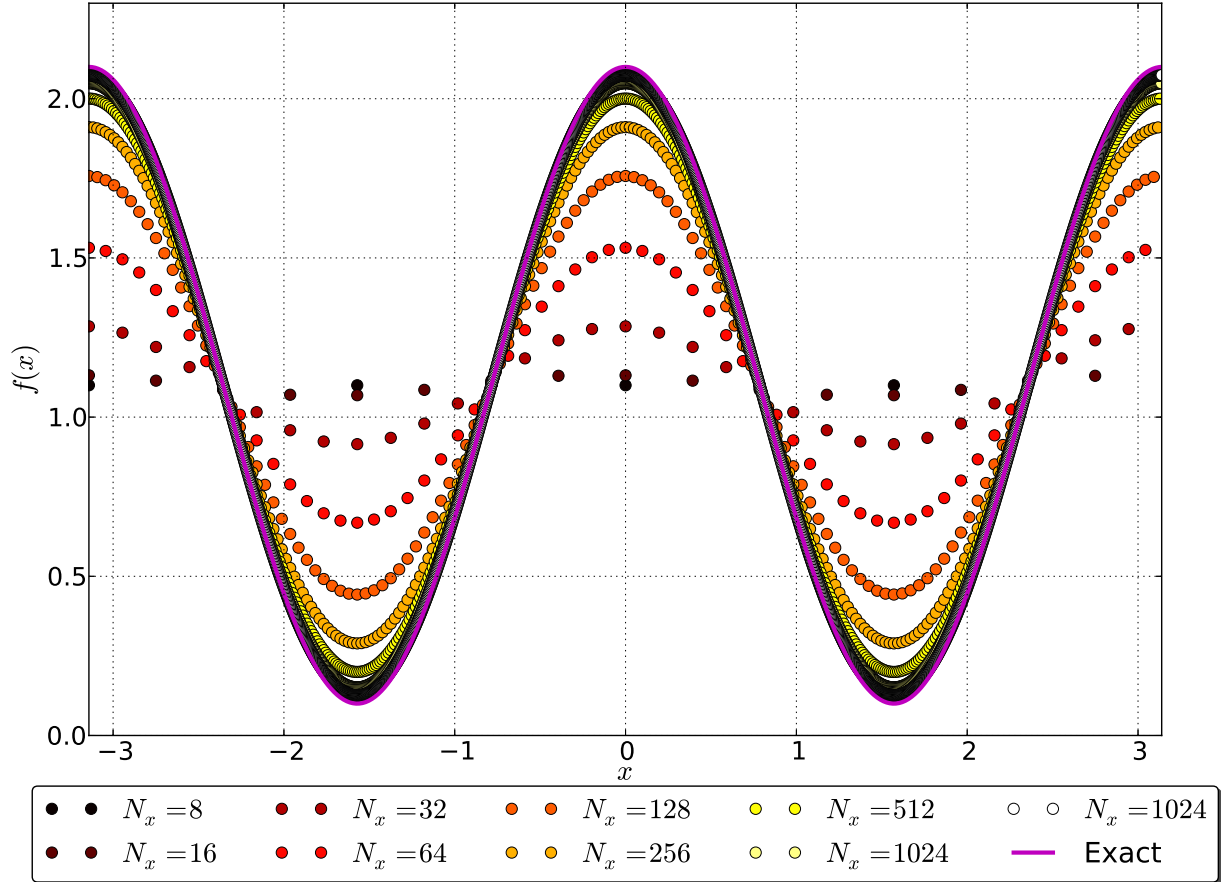
Figure 3.3: Several cases of the final solution obtained from the classic convected scheme algorithm are shown. For all simulations, the CFL number $\mathcal{C} = 0.32$. The figure records the number of spatial grid points N_x ($N_t = N_x/0.32$ for $\mathcal{C} = 0.32 =$ constant). For coarse grids, even with their comparatively fewer remappings over a simulation time, the numerical diffusion spreads out the density significantly.

To verify the higher order schemes have been implemented correctly, an order of accuracy exercise is pursued as before. However, for the corrected schemes, the above slowly varying initial distribution did not allow the order to be measured as the error too quickly approaches machine precision ($\epsilon \sim 10^{-16}$) even for coarse grids. Thus, a more irregular and challenging distribution is needed so that the effect of mesh refinement can be observed during the gradation. First, we examine numerical order of convergence of the *FD5* method, which computes derivatives using finite differences. After which we verify spectral CS (*FN* schemes) for various orders N.

Thus, we choose a non-periodic distribution that contains fine structure that must be adequately resolved relative to the order of accuracy of the scheme. The symmetric initial distribution (eq. (3.20)) selected is a superposition of three Gaussian bells [27]:

Figure 3.4: Numerical solution for FD5 after $N_t = 400$ time steps of eq. (3.23) for f_0 given by (3.20).

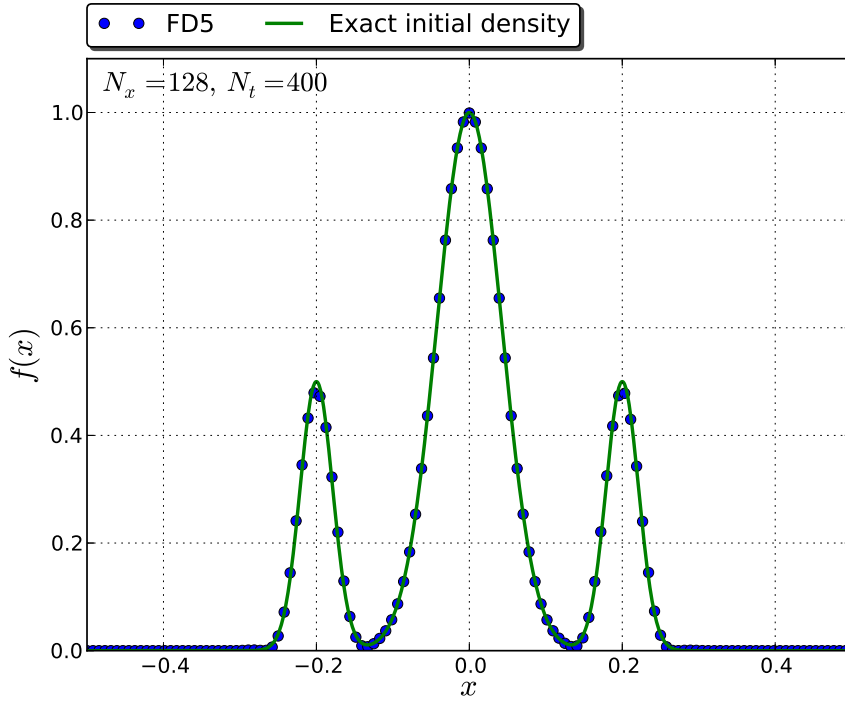|	FD5	
	NRMS(GE$_{\Delta x}$)	Order
N_x		
32	7.2912×10^{-2}	–
64	2.5443×10^{-2}	1.5189
128	4.4472×10^{-3}	2.5163
256	2.1447×10^{-4}	4.3741
512	7.0343×10^{-6}	4.9302
1024	2.2142×10^{-7}	4.9895
2048	6.9307×10^{-9}	4.9976

Table 3.2: Mesh refinement results for the FD5 scheme applied to the density (3.20). The normalized root mean square (NRMS) of the global error (GE) for each mesh with spacing $\Delta x = L/N_x$ is given by eq. (3.17), and the observed numerical order of accuracy is computed per (3.18). The CFL number $\mathcal{C} = 0.32$ for all simulations. The numerical order is converging towards $\mathcal{O}(\Delta x^5)$.

$$f_0(x) = \frac{1}{2}e^{-\left(\frac{x+0.2}{0.03}\right)^2} + e^{-\left(\frac{x}{0.06}\right)^2} + \frac{1}{2}e^{-\left(\frac{x-0.02}{0.03}\right)^2} \tag{3.20a}$$

$$x \in \mathcal{D} = [-0.5, 0.5], \qquad t \in [0, 1] \tag{3.20b}$$

the results are given in table 3.2 whereas a representative solution is shown figure 3.18 alongside the exact solution. Additionally, several cases are shown in figure 3.5.

Even for meshes containing only $N_x = 32$, the corrected advected density maintains the shape of the three Gaussian bells, though the error is still visibly too significant. As few as $N_x = 128$ grid points gives a solution that has a NRMS global error of 4.4472×10^{-3} (table 3.2) and is seen to match the solution by inspection well (figure 3.18). The results clearly suggest a convergence towards $\mathcal{O}(\Delta x^5)$ as needed for the FD5 method.

To emphasize the previous motivation for using the cosine distribution (3.19) to verify the order for classic CS, the results for the classic convected scheme applied to the superposed Gaussian bell case (3.20) above is summarized in table 3.3, which shows the difficulty in proving numerical order of accuracy for the classic CS when considering more rapidly varying densities. Convergence was evidence as early as $N_x = 1024$ for the cosine distribution (cf. table 3.1), whereas for the following distribution, even at $N_x = 2048$ (order ~ 0.7208) the convergence is still not suggestive and the error is significant (figure 3.6).

With the classic CS and FD5 method verified, we move onto verifying Nth order Fourier based (spectral) CS schemes (FN). For orders $N \gtrsim 10$, the numerical order of accuracy is not directly measurable as the error approaches machine precision too soon in the mesh refinement exercise. In order to prove the implementation,

Classic CS		
	NRMS(GE$_{\Delta x}$)	Order
N_x		
32	2.0352×10^{-1}	–
64	1.9009×10^{-1}	0.0985
128	1.6814×10^{-1}	0.1770
256	1.3216×10^{-1}	0.3474
512	9.2754×10^{-2}	0.5108
1024	6.0219×10^{-2}	0.6232
2048	3.6539×10^{-2}	0.7208

Table 3.3: Mesh refinement results for the classic convected scheme applied to the density (3.20). The normalized root mean square (NRMS) of the global error (GE) for each mesh with spacing $\Delta x = L/N_x$ is given by eq. (3.17), and the observed numerical order of accuracy is computed per (3.18). The CFL number $\mathcal{C} = 0.32$ for all simulations. The convergence of the numerical order of accuracy is not clear for this rapidly varying distribution. A much more resolved grid would be needed.
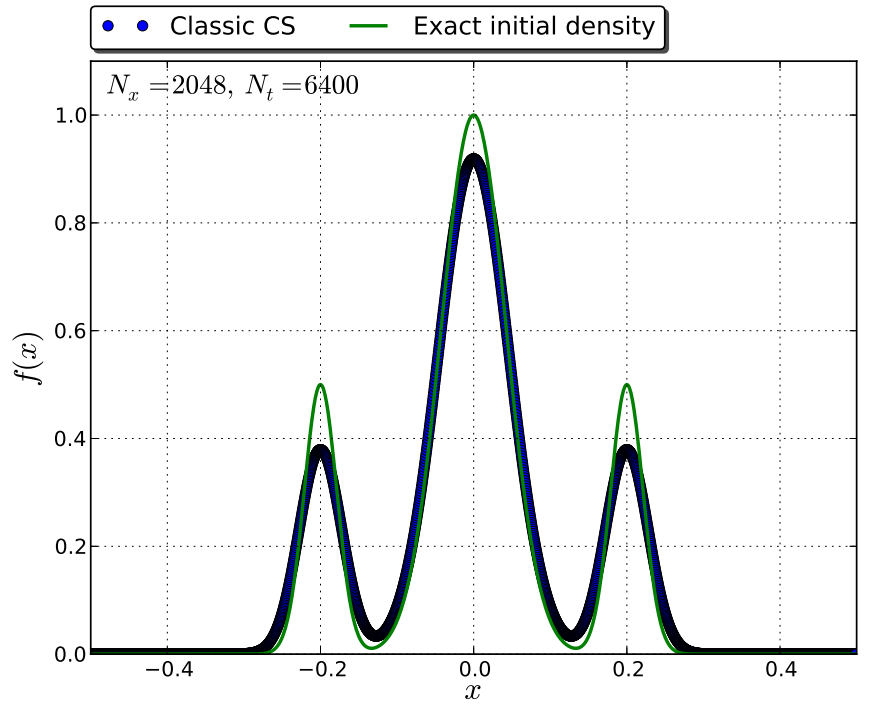
Figure 3.6: A classic CS solution after $N_t = 6400$ time steps of eq. (3.23) for f_0 given by (3.20).
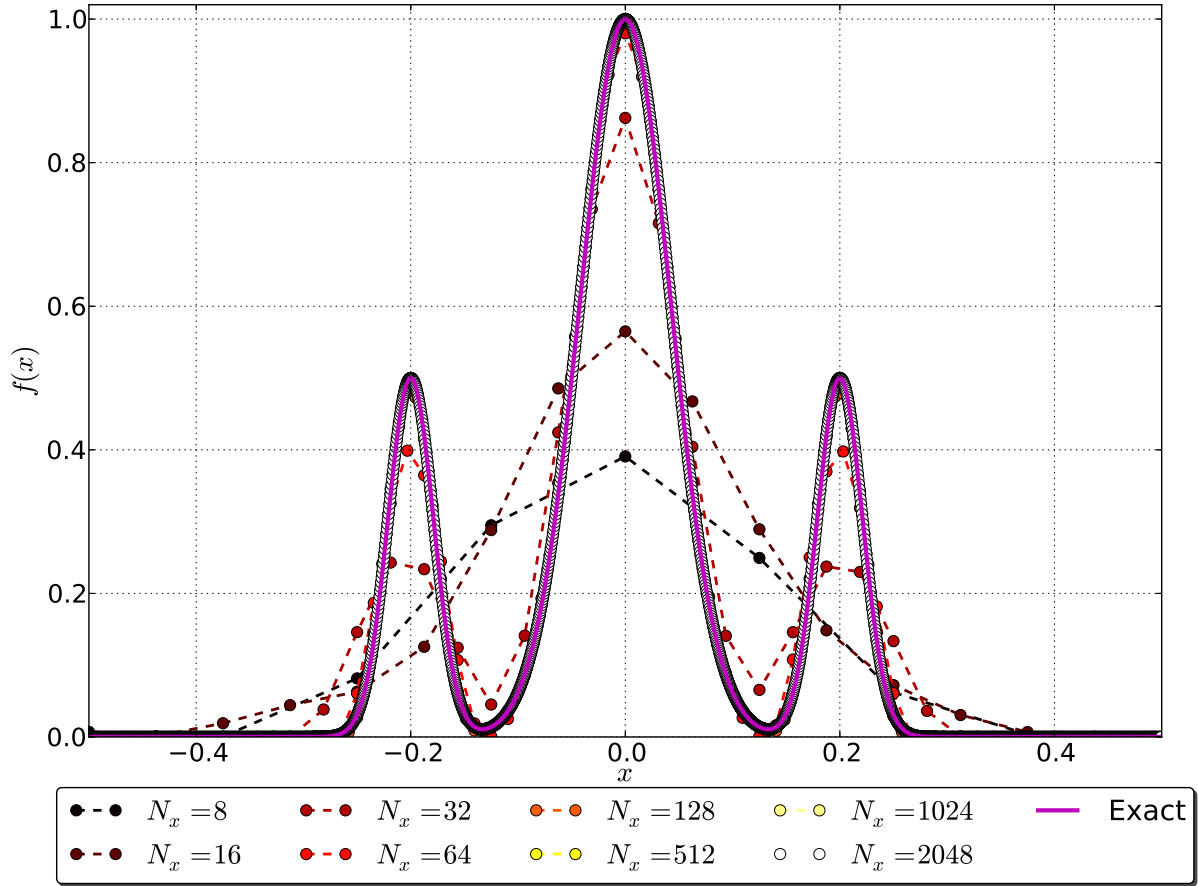
Figure 3.5: Several cases of the final solution obtained from the $N = 5$ order accurate *FD5* scheme are shown. For all simulations, the CFL number $\mathcal{C} = 0.32$. The figure records the number of spatial grid points N_x ($N_t = N_x/0.32$ for $\mathcal{C} = 0.32 = $ constant). With as coarse a mesh as $N_x = 32$ grid points the numerical solution is seen to retain the key features of the distribution (the three bells), and at $N_x = 128$, the solution is seen to reasonably approximate the density.

we choose to instead verify each order N as high as possible, and will take this as sufficient proof that higher orders follow suit. These results are summarized in table 3.4.

	F2		F3		F4		F5	
	NRMS(GE$_{\Delta x}$)	Order	NRMS(GE$_{\Delta x}$)	Order	NRMS(GE$_{\Delta x}$)	Order	NRMS(GE$_{\Delta x}$)	Order
N_x								
32	9.3763×10^{-2}	–	5.3484×10^{-2}	–	2.1076×10^{-2}	–	1.8251×10^{-2}	–
64	4.7483×10^{-2}	0.9816	1.9892×10^{-2}	1.4269	4.9206×10^{-3}	2.0987	2.0776×10^{-3}	3.1350
128	2.06645×10^{-2}	1.2003	4.4207×10^{-2}	2.1698	3.6497×10^{-4}	3.7529	8.0741×10^{-5}	4.6855
256	6.170×10^{-3}	1.7436	6.2759×10^{-4}	2.8163	2.2817×10^{-5}	3.9996	2.5658×10^{-6}	4.9757
512	1.5708×10^{-3}	1.9739	8.0046×10^{-5}	2.9709	1.4228×10^{-6}	4.0032	8.0364×10^{-8}	4.9967

	F6		F7		F8		F9	
	NRMS(GE$_{\Delta x}$)	Order	NRMS(GE$_{\Delta x}$)	Order	NRMS(GE$_{\Delta x}$)	Order	NRMS(GE$_{\Delta x}$)	Order
N_x								
32	1.4003×10^{-2}	–	1.3743×10^{-2}	–	1.3757×10^{-2}	–	1.3711×10^{-2}	–
64	4.7357×10^{-4}	4.8860	2.2053×10^{-4}	5.9615	5.6229×10^{-5}	7.934635	3.0591×10^{-5}	8.808078
128	7.7235×10^{-6}	5.9381	1.9207×10^{-6}	6.8432	2.1371×10^{-7}	8.039495	5.8963×10^{-8}	9.019098
256	1.1945×10^{-7}	6.0147	1.5166×10^{-8}	6.9847	8.2396×10^{-10}	8.018886	1.1657×10^{-10}	8.982388
512	1.8608×10^{-9}	6.0042	–	–	–	–	–	–

	F10		F11		F12		F13	
	NRMS(GE$_{\Delta x}$)	Order	NRMS(GE$_{\Delta x}$)	Order	NRMS(GE$_{\Delta x}$)	Order	NRMS(GE$_{\Delta x}$)	Order
N_x								
32	1.3742×10^{-2}	–	1.3733×10^{-2}	–	1.3736×10^{-2}	–	1.3734×10^{-2}	–
64	1.4822×10^{-5}	9.8566	1.3397×10^{-5}	10.0014	1.2761×10^{-5}	10.0721	1.2718×10^{-5}	10.0766
128	7.3664×10^{-9}	10.9745	2.2175×10^{-9}	12.5607	3.0450×10^{-10}	15.3548	9.8644×10^{-11}	16.9763
256	7.0799×10^{-12}	10.0230	1.1011×10^{-12}	10.9758	9.7125×10^{-14}	(m.p.)	6.7652×10^{-14}	(m.p.)
512	1.0166×10^{-13}	(m.p.)	1.0180×10^{-13}	(m.p.)	1.0187×10^{-13}	(m.p.)	1.0182×10^{-13}	(m.p.)

Table 3.4: Order calculations (eq. (3.17)) for various orders of spectral CS. For $N \gtrsim 10$, the order of convergence cannot be observed as machine precision (m.p.) is achieved too soon. The calculations highlighted in red indicate when convergence is sufficiently suggestive.

	$F15$	
	NRMS($GE_{\Delta x}$)	Order
N_x		
32	1.3733×10^{-2}	–
64	1.2690×10^{-5}	10.079
128	5.0649×10^{-12}	21.257
256	6.6473×10^{-14}	$(m.p.)$

Table 3.5: Mesh refinement results for the spectral CS $F15$ scheme applied to the density (3.20). The normalized root mean square (NRMS) of the global error (GE) for each mesh with spacing $\Delta x = L/N_x$ is given by eq. (3.17), and the observed numerical order of accuracy is computed per (3.18). The CFL number $\mathcal{C} = 0.32$ for all simulations. The term $(m.p.)$ indicates the solution has reached machine precision, and the order of convergence is not directly observable.
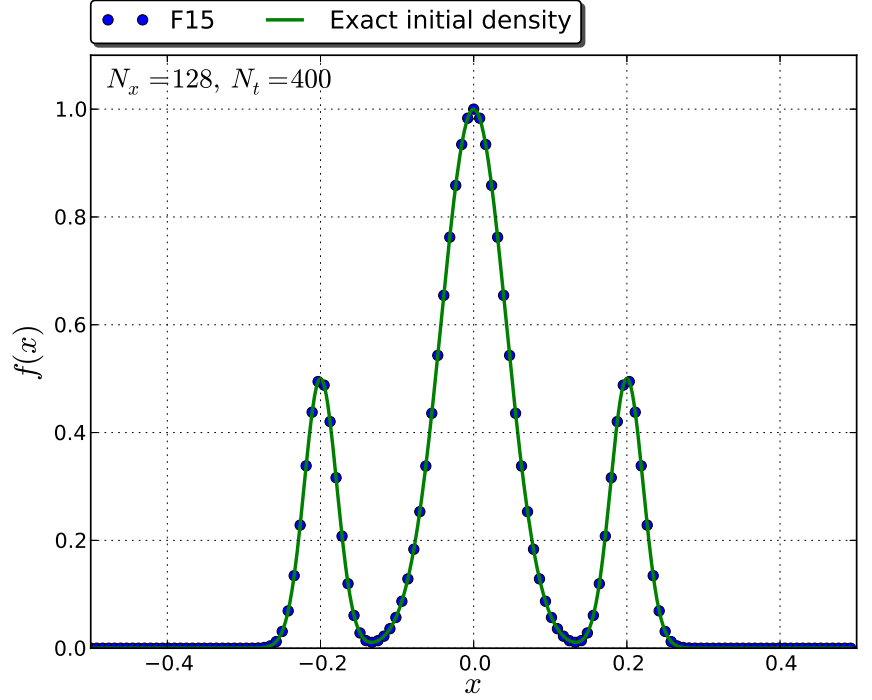

Figure 3.7: An F15 solution after $N_t = 400$ time steps of eq. (3.23) for f_0 given by (3.20).

Thus, since the numerical order of accuracy is convincingly observed for orders $N = 2$ to $N = 10$ in the FN schemes, it can only be expected that the higher orders must follow suit. We regard this as sufficient proof of implementation for the FN methods. For orders $N \geq 18$, machine precision is obtained for grids with as few points as $N_x = 128$. While the Fourier-based CS scheme is very efficient, the computational cost increases significantly with each order of accuracy. The simulation times for several orders of accuracy N for FN schemes are compared in table 3.6 for serial processing of a Python implemented CS on a 3.4 GHz CPU with 8 GB of RAM. Regarding the spectral CS algorithm 3, we note that computing corrections required for the next higher order requires computing N_x additional derivatives per time step, and $N_x N_t$ derivatives over a full simulation. Each derivative amounts to one complex multiplication operation in Fourier space to compute the derivative coefficients $\hat{d}_q^n$, which still requires recursively calculating the next $\beta_{N-1}(\alpha)$ coefficient (eq. (2.37)), and so on. Thus, it is clear that the processor time quickly increases as the order N becomes large. With computational expense and accuracy in mind, we consult table 3.6 and the errors found from test cases to come to the decision of selecting the 15th order accurate method ($F15$) to use henceforth. The F15 method achieves machine precision for the Gaussian bell test case at $N_x = 256$ (table 3.5), and requires only minutes to complete a full simulation under this model case. A plot of the numerical solution for this scheme is shown in figure 3.7.

3.2.2 1D advection: variable velocity

The corrections are modeled to the one-speed advection equation. However, the implementation on a computer is such that the equation is solved for one moving cell (MC) at a time. Thus, the correction and methods develop apply to non-constant speeds, since each problem itself is a one-speed problem. To this end, we extend the code to permit the scenario variable velocity fields. In particular, we examine a

	$N_x = 64$	$N_x = 128$		$N_x = 64$	$N_x = 128$
	Processor time [sec]	Processor time [sec]		Processor time [sec]	Processor time [sec]
FN			FN		
$F2$	5.4000×10^{-1}	2.1900	$F12$	3.4920×10^1	7.1450×10^1
$F3$	7.2000×10^{-1}	2.7900	$F13$	6.7950×10^1	1.3566×10^2
$F4$	8.8000×10^{-1}	3.4000	$F14$	1.3281×10^2	2.6560×10^2
$F5$	1.0500	3.7700	$F15$	2.6576×10^2	5.3762×10^2
$F6$	1.4700	4.8100	$F16$	5.3070×10^2	1.0863×10^3
$F7$	2.0600	6.0900	$F17$	1.0794×10^3	2.1287×10^3
$F8$	3.1700	8.5700	$F18$	2.1190×10^3	4.1185×10^3
$F9$	5.3800	1.3250×10^1	$F19$	4.2417×10^3	4.1786×10^3
$F10$	9.8700	2.2260×10^1	$F20$	9.3585×10^3	1.8601×10^4
$F11$	1.7850×10^1	3.8110×10^1	$F21$	2.4623×10^4	4.6311×10^4

Table 3.6: Processor times required to compute a numerical solution to advection equation (3.23) with the initial distribution of the superposed Gaussian bell density (3.20) for various order N for spectral CS (FN). The processor time picks up significantly for order $N \gtrsim 15$. For a 20th order accurate method on a grid of $N_x = 128$ points (at machine precision, the global L^2 error $= 6.7205 \times 10^{-15}$), simulations for this Python-implemented CS algorithm requires around 5.01 hours on serial processing with a 3.4 GHz CPU and 8 GB RAM. A 15th order method at the same resolution requires only 9 minutes (L^2 error $= 5.0649 \times 10^{-12}$) The errors for orders up to $N = 13$ are provided in 3.4.

spatially-dependent velocity field in one dimension.

$$\frac{\partial f}{\partial t} + v(x)\frac{\partial f}{\partial x} = 0, \qquad x \in \mathcal{D}, t \in [0, T] \tag{3.21a}$$

$$f(0, x) = f_0(x) \tag{3.21b}$$

$$f(t, x + L) = f(t, x) \tag{3.21c}$$

Such circumstances are familiar in fluid mechanics where the local speed of a fluid parcel can be tracked by an independent variable that parameterizes the pathlines of the fluid flow (e.g. a physical stream where the parcel speed increases along pathlines that suffer elevation drops). The numerical solution over time is computed for an initial condition given by an (unnormalized) Gaussian with a narrow peak $f_0 \sim \mathcal{N}(0, 0.04)$, and a sinusoidal velocity field $v_S(x)$, i.e.

$$f_0(x) = e^{-x^2/(2(0.04)^2)}, \qquad \text{and} \qquad v_S(x) = \sin 2\pi x, \qquad x \in [-0.5, 0.5] \tag{3.22}$$

Thus, the domain covers one full period of the velocity function and the Gaussian distribution has compact support close to the origin.
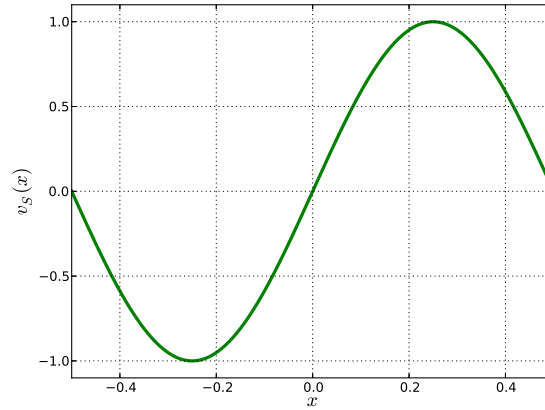

Figure 3.8: the velocity field varies as a function of position.

The solution is stepped through in the figures below as computed with the $F15$ scheme on a mesh with $N_x = 256$ cells over $N_t = 400$ time steps until a final time $T = 1.0$ (CFL number $\mathcal{C} = 0.32$). The initial distribution (shown in green) is plotted alongside the instantaneous solution (blue) for comparison. Note, from the figures it is seen that the density at $x = 0$ stays put throughout the simulation as $v_S(x = 0) = 0$.

Discussing the snapshots shown in the series of figures, we see that soon after the simulation start time $(t^0 = 0)$, the oppositely directed velocities near the origin convect densities left and right to spread out the profile (seen in at time t^{44}). As more density is advected left (resp. right) the density encounters regions where the velocities become increasingly negative (resp. positive) which accelerates the spreading of the distribution as evident at time t^{118}. The velocity magnitudes increase in both directions in x until the turning points at $x = \pm 0.25$ (see figure 3.8) whereafter the magnitudes decrease in both directions towards the edges of the domain. Density travelling beyond these points encounter slower speeds so that density ahead of the velocity extrema ($x = \pm 0.25$) is advected slower than that behind it (within $|x| < 0.25$). The pushed density from behind accumulates in slower regions so that the effect at time t^{230} results. The trend continues until the final
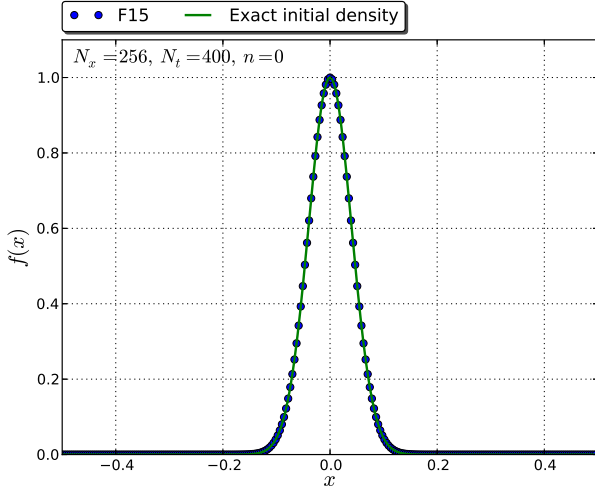
Figure 3.9: Variable density case: $t^0 = 0$
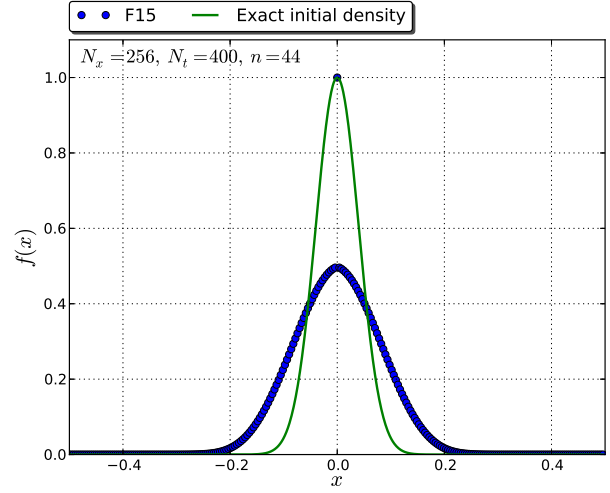
Figure 3.10: Variable density case: time $t^{44} = 0.11$
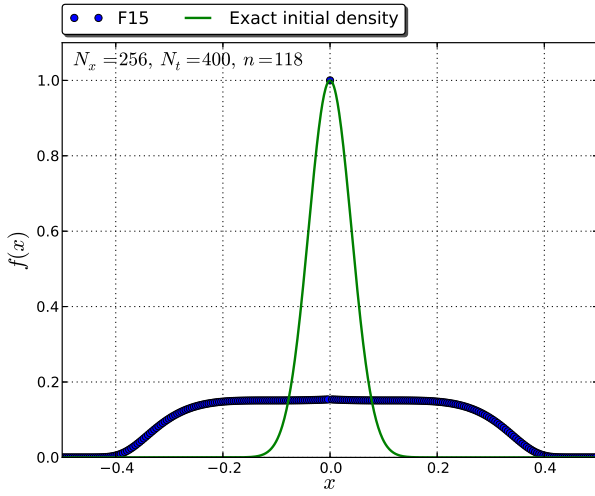
Figure 3.11: Variable density case: time $t^{118} = 0.295$
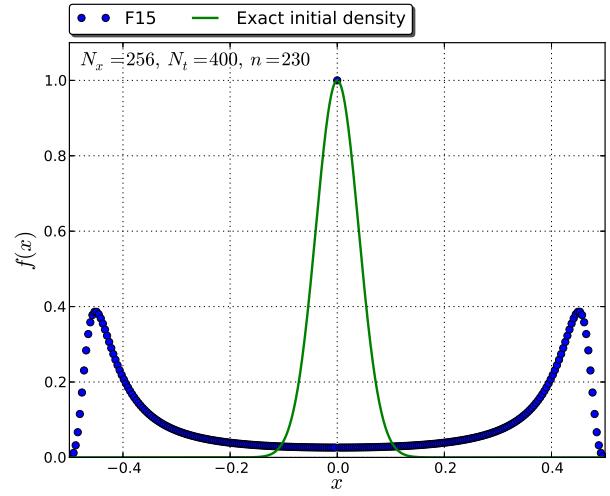
Figure 3.12: Variable density case: time $t^{230} = 0.575$

time step until density is pushed against the walls of the domain, leaving mostly zero density inside by the end of simulation time $t^{400} = 1$.

3.2.3   2D rotating advection system: time splitting schemes analysis

Splitting is necessary to move onto the next order of complication in the immediate goal to apprehend the Vlasov-Poisson system (cf. section 2.6). As a next step, we consider a two-dimensional advection equation with velocity field $\vec{v}(x,y) = \langle v_x, v_y \rangle$:

$$\frac{\partial f}{\partial t} + v_x(x,y)\frac{\partial f}{\partial x} + v_y(x,y)\frac{\partial f}{\partial y} = 0, \qquad (x,y) \in \mathcal{D}, t \in [0,T] \tag{3.23a}$$

$$f(0,x,y) = f_0(x,y) \tag{3.23b}$$

$$f(t,x+L,y) = f(t,x,y), \quad f(t,x,y+L) = f(t,x,y) \tag{3.23c}$$

The model case is the system investigated by [27] which convects a 22nd order "cosine cross" density packet in a rotating velocity field about the origin, which is characterized by a frequency $\omega = 2\pi/P$ where P is the period for a full revolution. For clockwise rotation we have

$$v_x(y) = \omega y, \qquad \text{and} \qquad v_y(x) = -\omega x \tag{3.24}$$

We choose a period $P = 1$ so that one full revolution is executed in unit time $T = 1$ (i.e. $\omega = 2\pi$). We also select a domain $\mathcal{D} = [-1,1] \times [-1,1]$, along with the aforementioned cosine cross initial density. This particular function was chosen by Güçlu with the intention to use the $F21$ corrected scheme (N.B. this scheme is labeled as $F22$ in [27] due a difference in what they define as an Nth order method). Choosing a 22nd order cosine bell ($C^{21}(\mathbb{R}^2)$) ensures we have the required $N-1$ derivatives needed to correct the CS up to N = 21, or $\mathcal{O}(\Delta x^{21}, \Delta y^{21})$ in space (cf. page 43 and eq. (2.38)). Using a corrected scheme on a sufficiently resolved grid so that the accuracy in space is at machine precision permits the splitting error in time to be observed (section 2.6.1). To this end, we elect to use an $F12$ method with $N_x = N_y = 256$ grid points, as previous convergence analysis has shown that we are well below machine precision and save some computational cost by not using a higher order method.

Thus, the rotating system to be analyzed is a solution to the following:

$$\frac{\partial f}{\partial t} + 2\pi y\frac{\partial f}{\partial x} - 2\pi x\frac{\partial f}{\partial y} = 0, \qquad (x,y) \in [-1,1] \times [-1,1], t \in [0,T] \tag{3.25}$$

The initial density eq. (3.26) is a superposition of two mutually transverse cosine bells whose cross-section has a major radius $2a = 0.5$, both of which are centered at a position $(x_c, y_c) = (0, 0.5)$ at time $t = 0$. A contour plot is given alongside the function definition in figure 3.13.

We defer the convergence analysis for future work, but report preliminary results using four splitting schemes where the names are kept identical to the seminal paper by Blanes et. al [5], which was also done by [27]. Table 3.7 summarizes the splitting coefficients, which is reproduced from [5] using the notation of section 2.6.1 for transport operators in space $\mathcal{X}^{c_i\tau} := \exp(c_i\tau\Lambda_x)$ along the x direction and $\mathcal{Y}^{d_i\tau} := \exp(d_i\tau\Lambda_y)$ along the y direction. In the terminology of Blanes et. al, we consider two orderings for symmetric Runge-Kutta-Nyström (SRKN) compositions. These orderings are labeled as type c and d (SRKN$_s^c$ and SRKN$_s^d$, respectively) for as many substages s as required. Here, the superscripting labels the outermost terms in the compositions, where c labels coefficients corresponding to the fractional time steps taken of the full step τ by the first operator ($\mathcal{X}$) and d plays the analogous role for the second ($\mathcal{Y}$).

$$f_0(x, y) = 0.5B(r_1(x, y)) + 0.5B(r_2(x, y)) \quad (3.26)$$

$$B(r, a) = \begin{cases} \cos^{22}\left(\frac{\pi r}{2a}\right) & \text{for } r \le a \\ 0 & \text{else} \end{cases} \quad (3.27a)$$

$$r_1(x, y) = \sqrt{(x - x_c)^2 + 8(y - y_c)^2} \quad (3.27b)$$

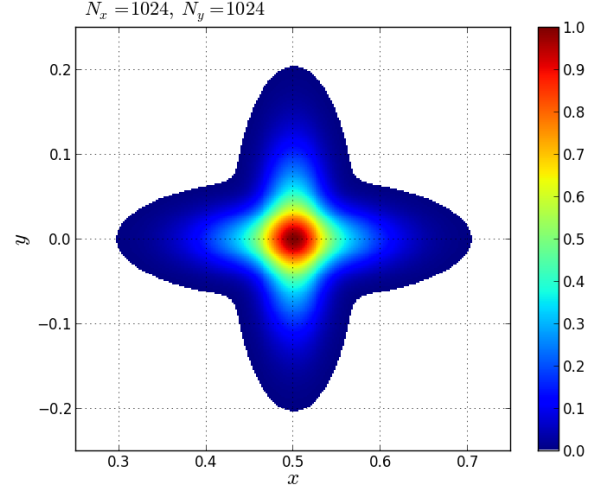$$r_2(x, y) = \sqrt{8(x - x_c)^2 + (y - y_c)^2} \quad (3.27c)$$

Figure 3.13: The cosine cross initial density $N_x = N_y = 1024$ spatial cells spanning (x, y).

The result quoted earlier for the Nth order scheme (2.75) from section 2.6.1 corresponded to the first ordering, which can be equivalently represented as:

$$\text{SRKN}_s^c := \mathcal{X}^{c_1\tau} \circ \mathcal{Y}^{d_1\tau} \circ \ldots \circ \mathcal{X}^{c_s\tau} \circ \mathcal{Y}^{d_s\tau} \mathcal{X}^{c_{s+1}\tau} \quad (3.28)$$

where $c_{s+2-i} = c_i$ and $d_{s+1-i} = d_i$. These types of compositions contain schemes such as the second order accurate LF2 (leapfrog or Strang) method, and the fourth order Yoshida Y4 method. The second ordering is then given by

$$\text{SRKN}_s^d := \mathcal{Y}^{d_1\tau} \circ \mathcal{X}^{c_1\tau} \circ \ldots \circ \mathcal{Y}^{d_s\tau} \circ \mathcal{X}^{c_s\tau} \circ \mathcal{Y}^{d_{s+1}\tau} \quad (3.29)$$

where $c_{s+1-i} = c_i$ and $d_{s+2-i} = d_i$. The two methods used in this work O6-4 and O11-6 follow this prescription.

$\mathcal{X}$ coefficients	$\mathcal{Y}$ coefficients	$\mathcal{X}$ coefficients	$\mathcal{Y}$ coefficients
LF2 : SRKN_1^c, $N = 2$		**O11-6**: SRKN_{11}^d, $N = 6$	
$c_1 = 1/2$	$d_1 = 1$	$c_1 = 0.123229775946271$	$d_1 = 0.0414649985182624$
$c_2 = 1/2$		$c_2 = 0.290553797799558$	$d_2 = 0.198128671918067$
		$c_3 = -0.127049212625417$	$d_3 = -0.0400061921041533$
Y4 : SRKN_3^c, $N = 4$		$c_4 = -0.246331761062075$	$d_4 = 0.0752539843015807$
$c_1 = \frac{1}{2(2-2^{1/3})}$	$d_1 = \frac{1}{2-2^{1/3}}$	$c_5 = 0.357208872795928$	$d_5 = -0.0115113874206879$
$c_2 = \frac{1-2^{1/3}}{2(2-2^{1/3})}$	$d_2 = -\frac{2^{1/3}}{2-2^{1/3}}$	$c_6 = 1 - 2(c_1 + \ldots + c_5)$	$d_6 = 1/2 - (d_1 + \ldots + d_5)$
$c_3 = c_2$	$d_3 = d_1$		
$c_4 = c_1$			
O6-4: SRKN_6^d, $N = 4$			
$c_1 = 0.245298957184271$	$d_1 = 0.0829844064174052$		
$c_2 = 0.604872665711080$	$d_2 = 0.396309801498368$		
$c_3 = 1/2 - (c_1 + c_2)$	$d_3 = -0.0390563049223486$		
	$d_4 = 1 - 2(d_1 + d_2 + d_3)$		

Table 3.7: Splitting coefficients are given for various schemes. Two SRKN_s^c methods are listed (LF2 and Y4 [64]), as well as two optimized SRKN_s^d methods presented by Blanes et. al [5] (O6-4 and O11-6). The order of accuracy N is also recorded.

The various schemes were used to solve the 2D rotating advection system for the initial cosine cross distribution. An example of a set of plots produced by the LF2 scheme is given in the following series.
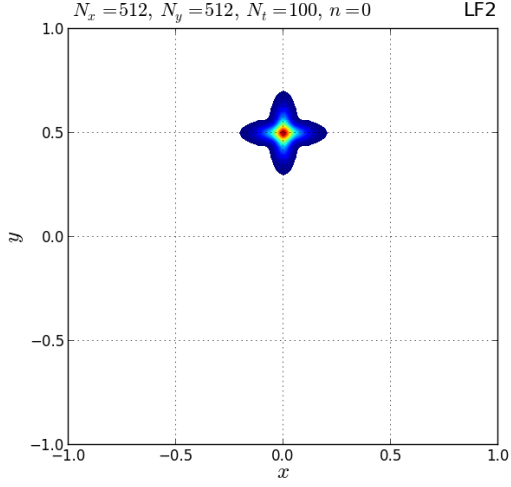
Figure 3.14: 2D rotating case: time $t^0 = 0$
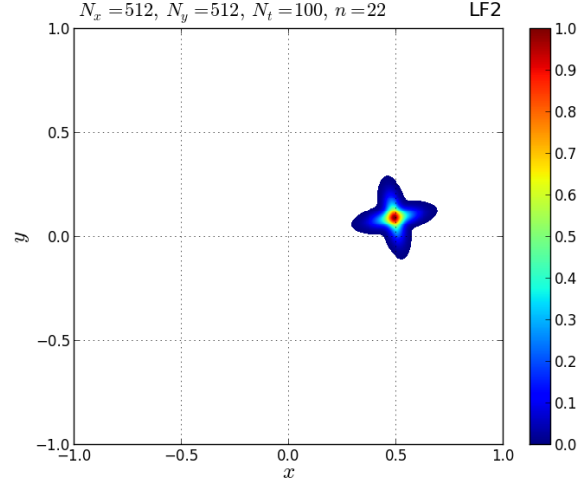
Figure 3.15: 2D rotating case: time $t^{22} = 0.22$
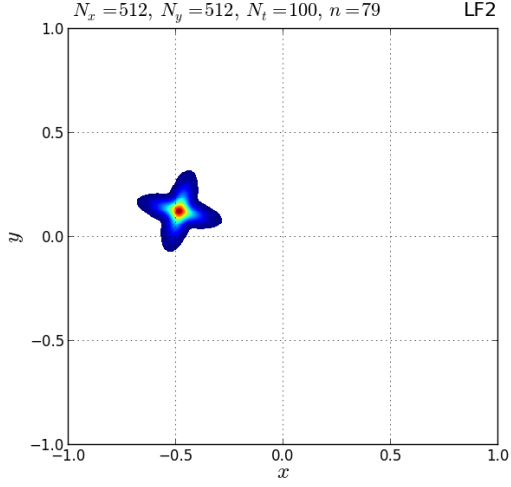
Figure 3.16: 2D rotating case: time $t^{79} = 0.79$
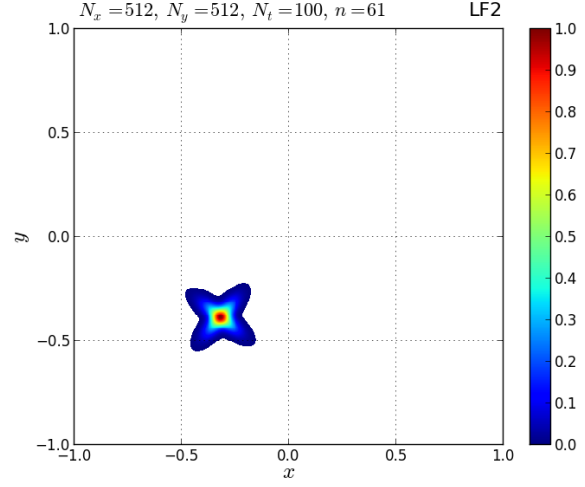
Figure 3.17: 2D rotating case: time $t^{54} = 0.61$

To assess the error, we seek a relevant extension of the NRMSE error for the 1D case to obtain the global error. The straightforward generalization is the 2D measure of the averaged global error:

$$\overline{\mathrm{GE}}_h = \frac{1}{\sqrt{L_x L_y}} \left[\sum_{i=0}^{N_x-1} \sum_{k=0}^{N_y-1} [f_{\mathrm{CS}}(T, x_i, y_k) - f_{\mathrm{exact}}(T, x_i, y_k)]^2 \Delta x \Delta y \right]^{1/2} \tag{3.30}$$

Here, h is short for the finite grid space $h = (\Delta x, \Delta y)$. In other words, we elect to use a scaled Frobenius norm $|| \cdot ||_F$, where the definitions for the domain lengths (L_x, L_y) are spanned by N_x and N_y cells with spacings Δx and Δy in their respective directions. A time mesh refinement analogous to the spatial mesh
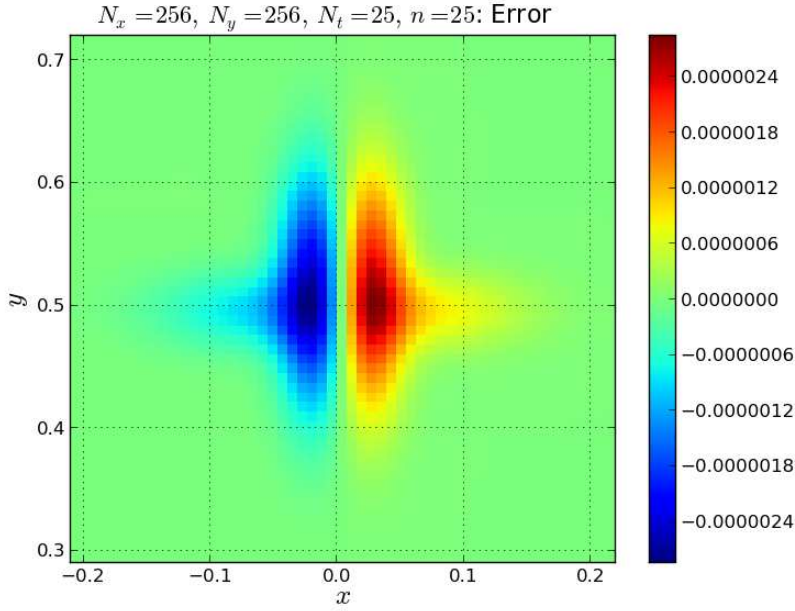
Figure 3.18: The error is plotted at the end of simulation time. The pixelation is due to the coarseness of the grid.

	NRMS(GE$_h$)	Sim. time [s]
Scheme		
$LF2$	4.5338×10^{-3}	5.2938×10^{3}
$Y4$	1.4368×10^{-3}	1.0561×10^{4}
O6-4	2.1483×10^{-5}	1.6298×10^{4}
O11-6	3.9634×10^{-7}	2.9528×10^{4}

Table 3.8: Error and simulation times required for various splitting schemes applied to the solution of 3.25 for the initial density (3.26). The normalized root mean square (NRMS) of the global error (GE$_h$) for the mesh $h = (\Delta x, \Delta y)$ is given by eq. (3.30). For all simulations, $N_x = N_y = 256$, $N_t = 25$.

refinement exercise performed earlier in this section for the 1D cases allows one to verify the theoretical order of convergence through successive simulations. Again, this exercise will be part of future work. The aim here is to only showcase the implementation of four popular split schemes, and to compare their global errors, which is summarized in table 3.8. A plot of the error for the case of LF2 at the end of a full simulation is also given for comparison.

The largest errors at the end of the simulation are localized around the center $(x_c, y_c) = (0, 0.5)$ of the density with a clear offset towards the right-hand side (in the direction of rotation [clockwise]). In other words, the principal error is ostensibly the overshoot in the rotation, by an overestimation of $\vec{v} = \vec{\omega} \times \vec{r}$. At this moment, the linear velocity $\vec{v} = \vec{v}_x = \vec{\omega} \times \vec{r} = -\omega \hat{k} \times y_c \hat{j} = +y_c \omega \hat{i}$.

3.2.4 1D-1V Vlasov test case: external electric field

With the split methods implemented, the natural intermediate test case before handling the Vlasov-Poisson system with self-consistent electric field calculations is the 1D-1V Vlasov equation coupled to a time-independent acceleration term as influenced by a prescribed spatially-varying electric field. Recall that the Vlasov equation is given by

$$\frac{\partial f_\alpha}{\partial t} + v \frac{\partial f_\alpha}{\partial x} - \left(\frac{q_\alpha}{m_\alpha} \frac{\partial \phi}{\partial x} \right) \frac{\partial f_\alpha}{\partial v} = 0 \qquad\qquad (2.46, \text{revisited})$$

For convenience, we elect to examine only electrons (negative charge), and to normalize the equation so that the Vlasov equation reads as:

$$\frac{\partial f}{\partial t} + v \frac{\partial f}{\partial x} + \frac{\partial \phi}{\partial x} \frac{\partial f}{\partial v} = 0 \qquad\qquad (3.31)$$

89

where the subscript has been dropped given only one species is considered. The normalization renders the measurement of the x coordinate as multiples of the Debye length $(x \rightarrow x/\lambda_D)$, $v \rightarrow v/\sqrt{kT_e/m}$, and $t \rightarrow t \cdot \frac{2\pi}{\omega_{pe}}$.

In particular, an electric field is chosen to emblemize an ordinary event in plasma systems: trapped electrons in electrostatic wells. Accordingly, we investigate an electric field whose electrostatic potential is given by [27]:

$$\phi(x) = 0.2 + 0.2\cos(\pi x^4) + 0.1\sin(\pi x) \tag{3.32}$$

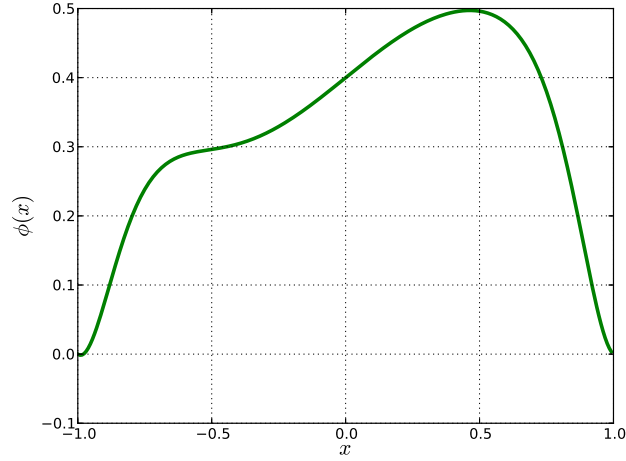This potential and electric field are plotted in figures 3.19 and 3.20.

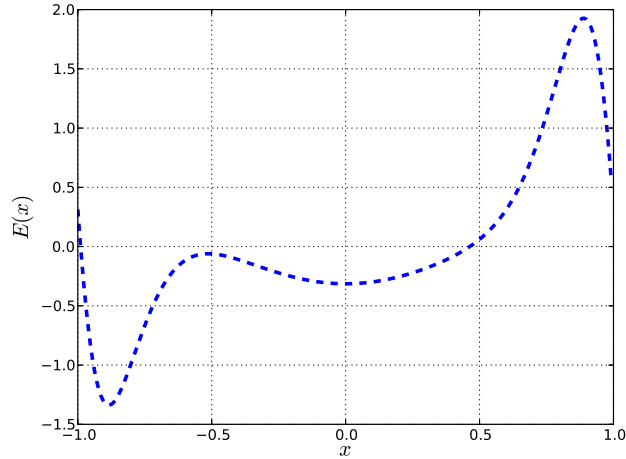Figure 3.19: The time-independent scalar potential of the 1D-1V Vlasov test case of section 3.2.4

Figure 3.20: The time-independent Electric field of the 1D-1V Vlasov test case of section 3.2.4

Since the force is conservative, the Hamiltonian does not explicitly depend on time and hence the system is autonomous. Hence, the Hamiltonian (total energy) is conserved:

$$H = \frac{1}{2}v_0^2 - \phi(x_0) = \frac{1}{2}v(t)^2 - \phi(x(t))$$

Thus, the solution should show the propagation of the electron density along constant energy contours (figure 3.21).
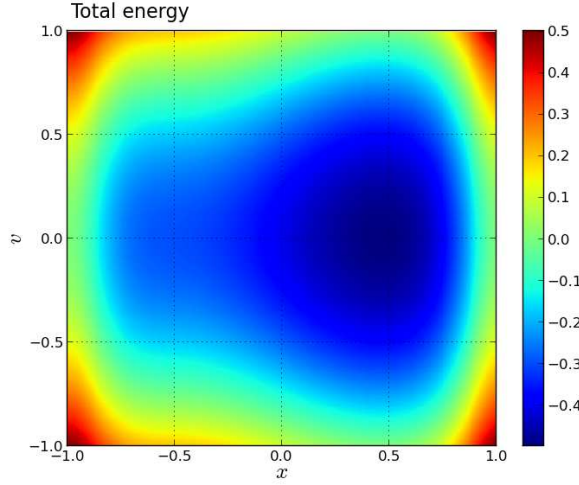

Figure 3.21: The time-independent Electric field of the 1D-1V Vlasov test case of section 3.2.4

The 22nd order cosine bell 3.27a is reused from section 3.2.3:

$$B(r, a) = \begin{cases} \cos^{22}\left(\frac{\pi r}{2a}\right) & \text{for } r \leq a \\ 0 & \text{else} \end{cases} \tag{3.27a, revisited}$$

where $r = \sqrt{(x - x_c)^2 + (v - v_c)^2}$. Following the setup by Güçlu , we choose $(x_c, v_c) = (-0.2, 0), a = 0.75$, and perform a simulation with $N_x = N_v = 256$ over $N_t = 128$ time steps for a simulation time of $t = 0.32$ using the leapfrog (LF2) splitting scheme and the 12th order accurate Fourier based CS method (F12). The results are shown in the four figures below at various times.

On serial processing at 3.4 GHz and 8 GB of RAM, the entire simulation required around 11.34 hours. Thus, despite the efficiency of the convected scheme and the higher order corrective approaches, the simulation times for more involved physics is already becoming enough. In future work, we aim to parallelize the code so that simulating more complicted systems at better resolution and accuracy can be made even more accessible.

Discussing the evolution of the distribution function above, we first regard the plots of the electric potential and field at the centroid of the distribution $(x_c, v_c) = (-0.2, 0)$. The initial location is chosen to sit near the left edge of the electrostatic well in figure 3.20 so that only a small portion of the electrons have a chance of leaking out the left-edge. The v centroid on axis for this symmetric distribution indicates we begin with an equal number of positive and negative velocities. The electrons are accelerated in the direction of decreasing (resp. increasing) electric field strength (resp. scalar potential). Thus, consulting plots 3.19 and 3.20 indicate a shallow electrostatic well within which electrons will be confined to $(-0.5 \lesssim x \lesssim 0.7)$. Also, the trajectories in phase space must be consistent with conservation of the Hamiltonian, so that the initial density should follow contours of constant energy in figure 3.21. Thus, initially the larger proportion of electrons $(x \gtrsim -0.4$, see $t^0 = 0$ plot) are accelerated rightward in the direction of decreasing electric field where the effect is most pronounced for the velocities above the $v = 0$ axis (i.e. positive velocities) as this acts to increase their already directed speed. The velocities below the this axis are negative and head leftward all the while stagnating to some extent as they are being decelerated given the direction of the electric field gradient. The populations near the edge of the well in the E field $(-0.5 \lesssim x \lesssim -0.4)$ are just touching the peak in the lefthand side of this well so that any electrons with negative or very small velocities can leak out this edge;
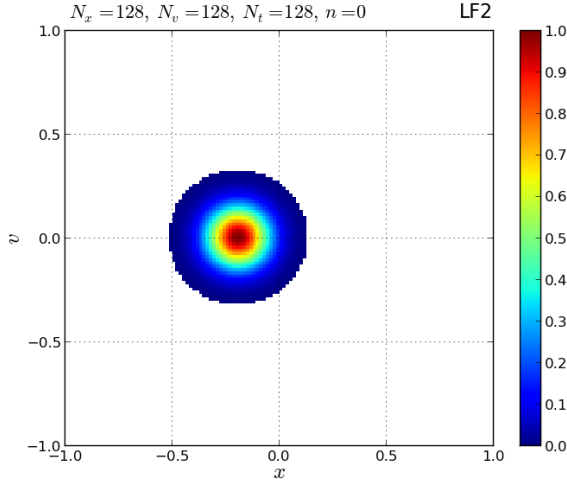
Figure 3.22: 1D-1V Vlasov case: time $t^0 = 0$
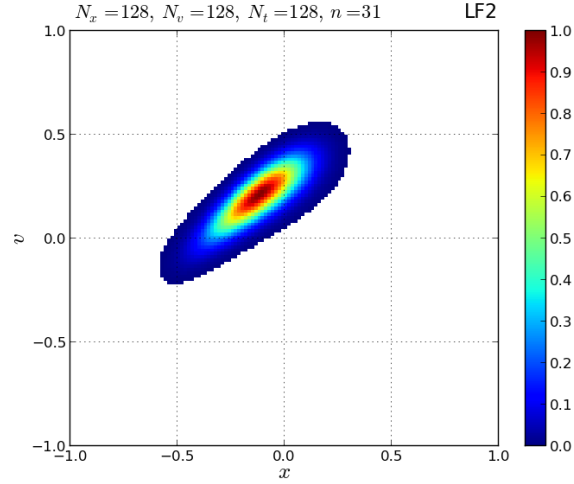
Figure 3.23: 1D-1V Vlasov case: time $t^{31} = 0.775$
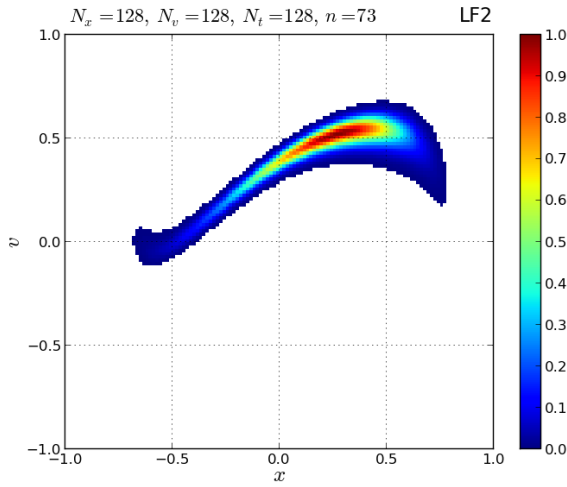
Figure 3.24: 1D-1V Vlasov case: time $t^{73} = 1.825$
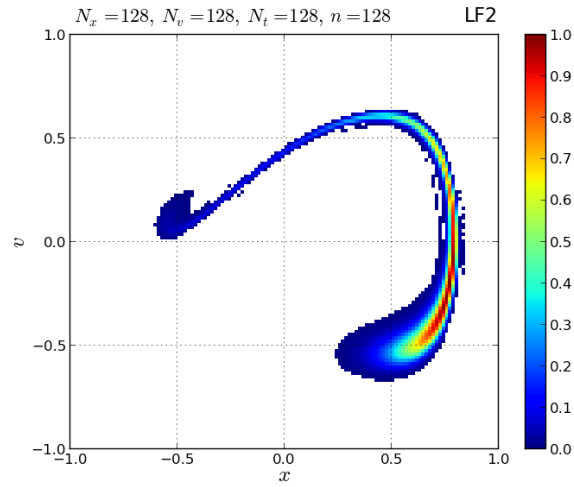
Figure 3.25: 1D-1V Vlasov case: time $t^{128} = 3.20$

however, this population is not the majority of the distribution. Since the majority of the population is inside this well, we initially have the bottom half of the population moving left but slowing down (i.e. moving up in velocity space). The other (top) half is speeding up as they head right, which does so at a faster rate than the negative portion as there is no deceleration stage. The skew produces the elongation of the distribution shown at time $t^{31} = 0.775$. As the simulation progresses, the effect continues as more initially left-moving electrons are decelerated past their velocity turning point so that they begin moving rightward and start to increase in speed thereafter. In doing so, the distribution follows constant energy curves which causes the accentuated stretching of the density (time $t^{73} = 1.825$ as the electrons drift past a neighborhood of the origin where they encounter their maximum speed and encounter regions of deceleration given the electric field begins to increase in regions right of the origin. The inertia they gained from the electric field in passing the origin causes them to overshoot the well beyond $x \sim 0.5$ to a value $x \sim 0.7$, whereafter they reach a turning point and must

head back into the well. They do so while on constant energy curves in phase space so that we witness the expected behavior (time $t^{128} = 3.2$ s). We note as that a fraction of the population did leak out the left edge (see time t^{73}), but by the end of simulation time $t^{128} = 3.2$ all such electrons have gained positive velocity so will continue to execute the path along the constant energy curve following the trail of the others. Since there are no external forces in the system or means to inject energy, the electron population will remain *trapped* indefinitely.

Chapter 4

Proposed research

This chapter enumerates subsequent research goals based on both the preliminary work developed thus far, and the overarching context of numerical solutions to plasma systems in the edge region of magnetic fusion devices. We begin with the proposition of several feasible goals that are seen to be reachable, and conclude with some larger objectives that will be addressed if progress permits.

4.1   1D-1V Vlasov-Poisson system

The immediate next step will be to develop an efficient Poisson solver, so that the high order convected scheme with time splitting methods can be extended to allow for the full solution of the Vlasov-Poisson system with self-consistent electric field calculations. Classic test cases including the bump-on-tail instability as well as Landau damping will be modeled in order verify and validate the implementation.

4.2   Boundaries: Harten filter

The edge of fusion devices presents a region where hot plasmas and materials coexist. As discussed in chapter 2.1.2, there exist sharp gradients, vast scales, discontinuities at the wall contact, and significant sparsity in phase space that make this region challenging to model. The transition layer from the edge to the material boundary is particularly nettlesome for computational solution especially given the amount of detail required to be captured within a comparatively minute region in configurational space and the discontinuity at the wall. We plan to investigate employing a *Harten filter* which prescribes a two-step processing of the numerical solution. First, multiple scales of *resolutions* are prescribed so that local mesh refinement exists on the grid in the vicinity of the wall. This allows for a locally accurate solution on a sufficiently resolved grid. Next, the Harten filter acts as an artificial compression method (ACM) [28] to sharpen this obtained numerical solution at a discontinuity to properly simulate the edge cutoff. Computational expense can also be saved in the sense that the Harten filter is designed to avoid superfluous computations and to filter out insignificant information. It differs from adaptive mesh refinement in that instead of indicating where a mesh should be refined, it informs where to not. In this sense, the grid is always refined enough and no key features will be missed since the approach is fundamentally opposite. If necessary, we will investigate how to make the filter higher order.

4.3   Collisions: defect corrections

Including collisions for electrostatic plasmas renders the system to be solved as the Boltzmann-Poisson set of equations. Accurately modeling collisions as well as a relevant boundary are the main goals of the proposed research. Collisions are ubiquitous in plasma physics, and their calculation lies in the inhomogeneity in the Boltzmann equation which captures the point-wise change in the distribution function due to these effects. Many realistic collision operators relevant for fusion plasmas must be cast as integrals containing the involved distribution functions of the interacting species with a kernel that defines the associated physics. Thus, a means to calculate this integral as part of this integro-differential equation to high order is required. We will investigate so-called *defect* (or deferred) *correction methods*, which take a predictor-corrector approach so that the solution can be refined in order to achieve higher accuracy. With each loop, an approximation to the error is obtained similar to Richardson extrapolation [12]. The error is modified according to the deferred correction prescription in such a way that the order of accuracy can be assessed and is known to increase in a predictable way. This robust method will be looked into and applied to the collisional problem. In particular, we will investigate the integral deferred correction form developed by Christlieb [12]. Representative test cases can include a demonstration of angular spreading from a beam distribution for a Coulomb collision operator. Other forms of the collision operator including the Landau form will also be considered.

4.4   Electrostatic sheath physics

Developing the above methods and implementing will allow for accurate modelling of the sheath, which properly must be modelled in this kinetic framework. By considering two charge species in the presence of

a boundary subject to the Vlasov-Poisson or Boltzmann-Poisson, we should be able to investigate sheath physics. In particular, there are well-known benchmark values that can be calculated (e.g. the wall potential for a hydrogenic plasma $V_{wall} \sim -0.7kT_e/e$), and by exploring the effects of collisionality different regimes can be explored and compared with various analytical models as well as experimental data.

4.5 Higher dimensions

If possible, higher dimensions will be considered. First, an additional velocity dimension can be introduced. This opens the path towards considering the effect of collisions as well as including magnetic fields. In this way, we would aim to solve a Boltzmann-Maxwell system. Progressing further to include another spatial dimension invites investigation of the electrostatics in the edge with the cross-field B transport which would approach the important problem of transport in the scrape-off layer.

Chapter 5

Summary

We have presented a preliminary framework demonstrating the design of arbitrarily high order accurate solutions to reduced cases of the Boltzmann-Maxwell system for periodic plasmas with the goal of moving forward to handling the Boltzmann-Poisson system. In particular, two varieties of the higher order convected scheme have been implemented. Both methods can be made arbitrarily accurate in space. The first version computes correction terms using finite differences, which was used to form the fifth order accurate $FD5$ method. The second computed derivatives efficiently in Fourier space using fast Fourier transform algorithms (FN methods). Their convergence was demonstrated for representative test cases, and in particular the FN methods were able to quickly approach machine precision over long time integrations. Next, we employed four split operator methods (LF2, Y4, O6-4, O11-6) and demonstrated their application in two test problems: a 2D advection problem, and a 1D-1V Vlasov equation with a prescribed time-independent electric field that acted to accelerate the velocities.

We propose as an immediate objective to verify the split methods converge to their expected order of accuracy in time. After which, we aim to work towards extending these high order accurate solutions to the Vlasov-Poisson system with self-consistent field calculations using the corrected convected scheme implemented in this document in tandem with high order splitting techniques (e.g. O11-6). As this is a well studied problem, there exist a number of benchmark cases that will allow for not only numerical verification and validation, but also to permit comparison with experiment.

The next goal is to move towards developing a framework to handle boundaries and collisions (a Boltzmann equation) to high order accuracy. To this end, a Harten filter will be looked into for handling the boundary properly, and the possibility of employing so-called deferred correction techniques will be investigated for collisions. The presence of the boundary will provide access to the interesting problem of sheath physics and the deterministic nature of the convected scheme kinetic solutions will allow appropriate tracking of the sheath thickness. On the other hand, the addition of collisions will allow us to investigate a more recent challenge in plasma physics collisional theory. That is, some studies suggest that large angle collision events make up a larger proportion of large angle deflections in comparison with cumulative small angle scatters of comparable magnitude. Accurate modeling of collisions will provide the means to investigate this firsthand.

We aim to extend the number of dimensions in our model, beginning with velocity in order to capture more of the physics involved in the edge. If progress permits, we will work towards extending a second spatial dimension. Including an additional velocity component gives a scenario where the effect of collisions and magnetic fields can be explored so that our utmost goal could be extending high order methods to the case of magnetized plasmas (Boltzmann-Maxwell systems). Adding the extra spatial dimension on top of velocity will allow for the interaction between electrostatics in the edge with the cross B-field transport to be studied. Additionally, other researchers have observed that the onset of beam-plasma instabilities in edge plasma is sensitive to the beam injection velocity. Alternatively, the kinetic model could be used to

inform the design of accurate fluid models for 2D edge problems. The computational expense of including the aforementioned physics will likely preclude serial processing. Thus, on the computational side, we will work towards parallelizing the implementation to make these goals achievable.

Bibliography

[1] BADER, A., ET AL. Modeling of HSX plasmas with EMC3-EIRENE. Presented at the US-Japan JIFT Workshop http://www.cptc.wisc.edu/conf/usjapan2013/Bader_US_Japan_JIFT_2013.pdf, June 2013. Accessed June 28, 2014.

[2] BESSE, N., AND MEHRENBERGER, M. Convergence of classes of high-order semi-Lagrangian schemes for the Vlasov-Poisson system. *Mathematics of Computation 77*, 267 (January 2008), 92–123.

[3] BIRDSALL, C., AND LANGDON, A. *Plasma Physics via Computer Simulation*. Taylor & Francis Group, 1985.

[4] BIRDSALL, C. K., AND FUSS, D. Clouds-in-clouds, clounds-in-cells physics for many-body plasma simulation. *Journal of Computational Physics 135* (1997). reprint.

[5] BLANES, S., AND MOAN, P. C. Practical symplectic partitioned Runge-Kutta and Runge-Kutta-Nyström methods. *Journal of Computational and Applied Mathematics 143* (2002), 313–330.

[6] BOHM, D. *The Characteristics of Electrical Discharges in Magnetic Fields*. McGraw-Hill, New York, 1949. chapter 3.

[7] BORCHARDT, M., RIEMANN, J., SCHNEIDER, R., AND BONNIN, X. W7âĂŞX edge modelling with the 3D SOL fluid code BoRiS. *Journal of Nuclear Materials 3*, 290-293 (2001), 546–550.

[8] BURRELL, K., AUSTIN, M., BRENNAN, D., ET AL. Quiescent H-mode plasmas in the DIII-D tokamak. *Plasma Physics and Controlled Fusion 44*, A253 (2002).

[9] BURRELL, K., ET AL. Quiescent H-mode plasmas in the DIII-D tokamak. Tech. rep., General Atomics, November 2001. preprint.

[10] BURRELL, K., OSBORNE, T., SNYDER, P., ET AL. Quiescent H-mode plasmas with strong edge rotatoin in the cocurrent direction. *Physical Review Letters 102*, 155003 (2009).

[11] CHODURA, R. *Physics of Plasma Wall Interaction in Controlled Fusion*. Plenum Press, New York, 1984. pages 99-134.

[12] CHRISTLIEB, A., ET AL. Integral deferred correction methods constructed with high-order Runge-Kutta integrators. *Mathematics of Computation* (March 2009).

[13] CHRISTLIEB, A., HITCHON, W. N. G., AND KEITER, E. A computational investigation of the effects of varying discharge geometry for an inductively coupled plasma. *IEEE T. Plasma Sci. 28*, 6 (2000), 2214–2231.

[14] CONNOR, J. W., ET AL. Edge localised modes (ELMs): experiments and theory. Presented at the First ITER international summer school http://www.ccfe.ac.uk/assets/Documents/AIPCONFPROC103p174.pdf. Accessed December 4, 2014.

[15] CROUSEILLES, N., AND MEHRENBERGER, E. F. M. High order Runge-Kutta-Nyström splitting methods for the Vlasov-Poisson equation. http://hal.inria.fr/inria-00633934. Accessed September 16, 2014.

[16] EUROfusion. http://www.euro-fusion.org. Accessed October 9, 2014.

[17] EUROPEAN FUSION DEVELOPMENT AGREEMENT (EFDA). Divertor operation in fusion reactors. http://www.efda.org/fusion/focus-on/limiters-and-divertors/divertor-operation-in-fusion-reactors/. Accessed June 25, 2014.

[18] EUROPEAN FUSION DEVELOPMENT AGREEMENT (EFDA). Lawsons three criteria. http://www.efda.org/2013/02/triple-product/. Accessed June 25, 2014.

[19] EVANS, T., ET AL. RMP ELM suppression in DIII-D plasmas with ITER similar shapes and collisionalities. *Nuclear Fusion 48*, 2 (2008).

[20] FENG, J., AND HITCHON, W. N. G. Self-consistent kinetic simulations of plasmas. *Physical Review E 61*, 3 (2000), 1292–1297.

[21] FENG, Y. *Contributions to Plasma Physics 57-59*, 1-3 (2004).

[22] FENG, Y., SARDEI, F., AND KISSLINGER, J. 3d fluid modelling of the edge plasma by means of a monte carlo technique. *Journal of Nuclear Materials*, 266-269 (2003), 812–818.

[23] FINKEN, K., ET AL. The structure of magnetic field in the TEXTOR-DED. *Schriften des Forschungszentrums Jülich Reihe Energietechnik / Energy TechnologyPlasma Physics and Controlled Fusion 45* (2005).

[24] FORNBERG, B. Generation of finite difference formulas on arbitrarily spaced grids. *Mathematics of Computation 51* (1988), 699–706.

[25] FRIEDBERG, J. P. *Plasma Physics and Fusion Energy*, first ed. Cambridge University Press, New York, United States, August 2008.

[26] GÜÇLU, Y., AND HITCHON, W. N. G. A high order cell-centered semi-lagrangian scheme for multidimensional kinetic simulations of neutral gas flows. *Journal of Computational Physics 231* (2012).

[27] GÜÇLU, Y., AND HITCHON, W. N. G. Arbitrarily high order convected scheme solution of the Vlasov-Poisson system. *Journal of Computational Physics 270*, 0 (2014), 711 – 752.

[28] HARTEN, A. The artificial compression method for computation of shocks and contact discontinuities: Iii. Self adjusting hybrid schemes. *Mathematics of Computation 32*, 142 (April 1978), 363–389.

[29] HITCHON, W. N. G., KOCH, D. J., AND ADAMS, J. B. An efficient scheme for convection-dominated transport. *J. Comput. Phys. 83*, 1 (July 1989), 79–95.

[30] HOCKNEY, R., AND EASTWOOD, J. *Computer Simulation Using Particles*. Taylor & Francis Group, 1988.

[31] INTERNATIONAL ATOMIC ENERGY AGENCY (IAEA). http://www.iaea.org. Accessed October 9, 2014.

[32] JACKSON, G., ET AL. *Physical Review Letters 67*, 3098 (1991).

[33] JACKSON, J. D. *Classical Electrodynamics*, third ed. Wiley, August 1998.

[34] JOHNSON, S. G. Notes on FFT-based differentiation. http://math.mit.edu/~stevenj/notes.html. Accessed November 22, 2014.

[35] KWON, EUNHEE. KSTAR announces successful ELM suppression. National Fusion Research Institute via ITER Newsline http://www.iter.org/newsline/198/950. Accessed June 28, 2014.

[36] LAPENTA, G. Particle in Cell Method: A brief description of the PIC method. https://perswww.kuleuven.be/~u0052182/weather/pic.pdf. Accessed June 14, 2014.

[37] LAWRENCE LIVERMORE NATIONAL LABORATORY. Scientific breakeven for fusion energy. Fusion Ignition Research Experiment (FIRE) memorandum at PPPL: http://fire.pppl.gov/ICF_Scientific_Breakeven_LLNL2.pdf. Accessed June 25, 2014.

[38] LAWSON, J. Some criteria for a power producing thermonuclear reactor. Tech. Rep. GP/R 1807, Atomic Energy Research Establishment, Harwell, Berkshire, U.K., December 1955.

[39] LEWIS, H. Energy-conserving numerical approximations for vlasov plasmas. *Journal of Computational Physics 1*, 6 (2003), 136–141.

[40] LOARTE, A., SAIBENE, G., ET AL. Characteristics of type I ELM energy and particle losses in existing devices and their extrapolation to ITER. *Plasma Physics and Controlled Fusion 43* (August 2003), 1549–1569.

[41] MANGENEY, A., CALIFANO, F., CAVAZZONI, C., AND TRAVNICEK, P. A numerical scheme for the integration of the vlasov-maxwell system of equations. *Journal of computational physics 179* (2002), 495–538.

[42] MARANDET, Y. (Some) challenges in MCF edge plasma physics. Presented at CNRS, Aix-Marseille Université, PIIM http://www.edu.upmc.fr/physique/master/S_fusion/fichiers/documents/seminaire_mf_paris_20_11_2013_ym.pdf, November 2013. Accessed June 28, 2014.

[43] MARKIDIS, S., AND LAPENTA, G. The energy conserving particle-in-cell method. *Journal of Computational Physics 18*, 230 (2003), 7037–7052.

[44] MEADE, D. M. q, break-even and the $n\tau_e$ diagram for transient fusion plasmas. Tech. Rep. GP/R 1807, Princeton Plasma Physics Laboratory, Princeton, NJ, United States, April 1998.

[45] MEISS, J. Hamiltonian systems. *Scholarpedia 2*, 8 (2007), 1943. revision #129925.

[46] MIYAMOTO, K. *Controlled Fusion and Plasma Physics*. CRC Press, 2006.

[47] MOSSESSIAN, D. A., SNYDER, P. B., GREENWALD, M., HUGHES, J. W., LIN, Y., MAZURENKO, A., MEDVEDEV, S., WILSON, H. R., AND WOLFE, S. H-mode pedestal characteristics and MHD stability of the edge plasma in Alcator C-Mod. *Plasma Physics and Controlled Fusion 44*, 4 (2002), 423.

[48] NARDON, E., ET AL. Elm control by resonant magnetic perturbations on jet and mast. *Journal of Nuclear Materials 390-391* (2009).

[49] NERI, F. Lie algebras and canonical integration. preprint.

[50] PARK, J.-K., ET AL. Observation of edge harmonic oscillation in NSTX and theoretical study of its active control using HHFW antenna at audio frequencies. Tech. rep., Princeton Plasma Physics Laboratory, Princeton, NJ, United States.

[51] R. Baker, H. N., et al. Jet elm control coil feasibility study. Tech. Rep. GP/R 1807, JET ELM coil study team, Harwell, Berkshire, U.K., December 1955.

[52] Rossmanith, J., and Seal, D. A positivity-preserving high-order semi-Lagrangian discontinuous Galerkin scheme for the Vlasov-Poisson equations. *J. Comput. Phys. 230*, 16 (July 2011), 6203–6232.

[53] Runov, A., Kasilov, S., Reiter, D., McTaggart, N., Bonnin, X., and Schneider, R. Transport in complex magnetic geometries: 3D modelling of ergodic edge plasmas in fusion experiments. *Journal of Nuclear Materials 3*, 313 (2003), 1292–1297.

[54] Runov, A., Kasilov, S., Riemann, J., Borchardt, M., Reiter, D., and Schneider, R. Benchmark of the 3-dimensional plasma transport codes E3D and BoRiS. *Journal of Computational Physics 2-4*, 42 (2002), 169–174.

[55] Schneider, R., Borchardt, M., Riemann, J., Mutzke, A., and Weber, S. Concept and status of a 3D SOL fluid code. *Contributions to Plasma Physics 3-4*, 40 (2000), 340–345.

[56] Science, P., and of Technology, F. C. P. M. I. Tokamak parameter comparison. http://www.psfc.mit.edu/~marmar/5year_2008/06_tokamak_parameter_comparisons.xls. Accessed December 3, 2014.

[57] Shukla, R., and Zhong, X. Derivation of high-order compact finite difference schemes for non-uniform grid using polynomial interpolation. *Journal of Computational Physics 204* (2005), 404–429.

[58] Stangeby, P. C. *The Plasma Boundary of Magnetic Fusion Devices.* CRC Publishing, Bristol, England, United Kingdom, January 2000.

[59] Sun, Y., Zhou, Y. C., Li, S.-G., and Wei, G. W. A windows Fourier pseudospectral method for hyperbolic conservation laws. *Journal of Computational Physics 214* (2005), 466–490.

[60] Suttrop, W., Conway, G., Fattorini, L., et al. Study of quiescent h-mode plasmas in asdex upgrade. *Plasma Physics and Controlled Fusion 46*, A15 (2004).

[61] Tuckerman, M. Preservation of phase space volume and Liouville's theorem. http://www.nyu.edu/classes/tuckerman/stat.mech/lectures/lecture_2/node2.html, January 2002. Accessed September 18, 2014.

[62] Wei, G. W. Discrete singular convolution for the solution of the Fokker-Planck equations. *Journal of Chemical Physics 110* (1999), 8930–8942.

[63] Wesson, J., et al. *Tokamaks*, third ed. Oxford Press Inc, New York, United States, 2004.

[64] Yoshida, H. Construction of higher order integrators. *Physics letters 150*, 5,6,7 (1990).

[65] Yoshida, H. Recent progress in the theory and application of symplectic integrators. *Celestial Mechanics and Dynamical Astronomy 56*, 1-2 (1993), 27–43.

[66] Zohm, H. Edge localized modes (ELMs). *Plasma Physics and Controlled Fusion 38* (1996).