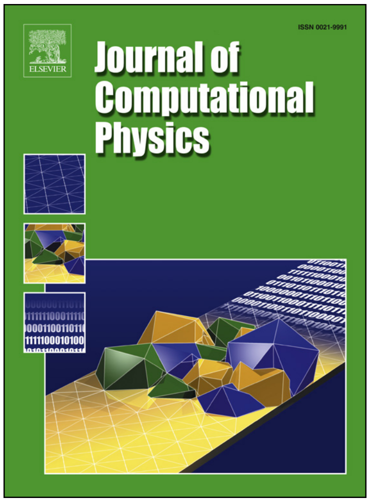


Accurate particle time integration for solving Vlasov-Fokker-Planck equations with specified electromagnetic fields

Patrick Jenny, Hossein Gorji

PII: S0021-9991(19)30157-3
DOI: https://doi.org/10.1016/j.jcp.2019.02.040
Reference: YJCPH 8539

To appear in: *Journal of Computational Physics*

Received date: 29 March 2018
Revised date: 18 January 2019
Accepted date: 14 February 2019

Highlights

- The research highlight presented in this paper is an analytical solution of the coupled Langevin equations as they result from the Vlasov-Fokker-Planck equation for plasma flow with constant electromagnetic fields and constant friction coefficient. This analytical solution involves a transformation into complex space and allows to take extremely large time steps when used in a solution algorithm. This is demonstrated in the paper with numerical studies.

Journal of Computational Physics (2019)

Contents lists available at ScienceDirect

Journal of Computational Physics

journal homepage: www.elsevier.com/locate/jcp

Accurate particle time integration for solving Vlasov-Fokker-Planck equations with specified electromagnetic fields

Patrick Jenny[a,*], Hossein Gorji[b]

[a] Swiss Federal Institute of Technology, Sonneggstrasse 3, 8092 Zurich, Switzerland
[b] École Polytechnique Fédórale de Lausanne, MA C2 642, Station 8, 1015 Lausanne, Switzerland

ARTICLE INFO

Article history:
Received -
Received in final form -
Accepted -
Available online -

Communicated by -

ABSTRACT

The Vlasov-Fokker-Planck equation (together with Maxwell's equations) provides the basis for plasma flow calculations. While the terms accounting for long range forces are established, different drift and diffusion terms are used to describe Coulomb collisions. Here, linear drift and a constant diffusion coefficient are considered and the electromagnetic fields are imposed, i.e., plasma frequency is not addressed. The solution algorithm is based on evolving computational particles of a large ensemble according to a Langevin equation, whereas the time step size is typically limited by plasma frequency, Coulomb collision frequency and cyclotron frequency. To overcome the latter two time step size constraints, a novel time integration scheme for the particle evolution is presented. It only requires that gradients of mean velocity, bath temperature, magnetic field and electric field have to be resolved along the trajectories. In fact, if these gradients are zero, then the new integration scheme is statistically exact; no matter how large the time step is chosen. Obviously, this is a computational advantage compared to classical integration schemes, which is demonstrated with numerical experiments of isolated charged particle trajectories under the influence of constant magnetic- and electric fields. Besides single ion trajectories, also plasma flow in spatially varying electromagnetic fields was investigated, that is, the influence of time step size and grid resolution on the final solution was studied.

1. Introduction

It is well known that flow phenomena far from thermodynamic equilibrium may not be accurately treated by macroscopic equations. Typically, complex manifestations of micro-scale processes in the macroscopic flow behavior is involved here. In other words, the closure assumptions rising to the notion of transport properties are no longer

[*] Corresponding author: Tel.: +41-44-632-6987;
e-mail: jenny@ethz.ch (Patrick Jenny)

adequate. A higher level of closure is obtained through the notion of the probability density functions. Particle Monte-Carlo methods have then been introduced in Direct Simulation Monte-Carlo (DSMC) [1] and Particle-in-Cell (PIC) methods [2] for simulations of gases and plasmas, respectively. While physically accurate results can be obtained from converged DSMC and PIC simulations, two limitations may undermine their applications, i.e., dense operations at high collision rates and statistical errors at low signal-to-noise ratios.

Collisionless plasmas are described by the Vlasov equation, which governs transport of charged particles with some distribution in electromagnetic fields. Similar to neutral gas flow simulations, there exist direct methods, moment methods and particle methods for plasma flow simulations (see [3, 4] for an overview). One of the main approaches belongs to the particle Monte-Carlo PIC methods, which combine the set of Maxwell's equations with particle transport [2, 5].

In many practical situations ranging from plasma assisted material processing to the edge region of tokamak and inertial confinement fusion, short range Coulomb encounters have to be taken into account [6]. An appropriate treatment of this so-called cold plasma regime can be given by the Vlasov-Fokker-Planck (Landau-Fokker-Planck) equation. However, due to the complexity of the resulting potentials (e.g. Rosenbluth potentials), DSMC type algorithms were devised in the seminal works of Takizuka & Abé [7], Nanbu [8] and Bobylev & Nanbu [9]. Yet the so-called PIC-DSMC solvers suffer from the same limitations as neutral gas DSMC solvers, i.e., dense operations at large Debye lengths and large fluctuations close to equilibrium. Though hybrid algorithms have been developed, e.g. by Caflisch et al. [10], in order to tackle these issues, further studies are required for generalizations of these schemes for inhomogeneous and practical settings. Note that while the cumulative collision algorithm derived in [9] and the SDE formulation proposed in [14] do not require resolving the grazing collisions, the resulting schemes still are subject to time integration errors of $O(\Delta t)$ and $O(\Delta t^{3/2})$ in computing collisions, respectively (see [9] and [14]).

Besides dense operations arising from short range Coulomb encounters, the resolution constraint required by the plasma frequency can lead to severly stiff plasma simulations. While simple explicit treatment of the Vlasov-Fokker-Planck system coupled with Maxwell's equations can become significantly demanding, numerous studies are addressing the stiffness issue. The so-called exponential time integration methods rely on asymptotic analysis and homogenization methods, developed mainly by Sonnendrücker and colleagues (see e.g.[20]). Another alternative is implicit time integration either in the context of PIC (developed by Chacón and colleagues [21]) or direct Vlasov-Fokker-Planck solvers e.g. [22]. Note that in this study, since the self induced electromagnetic fields are ignored, we do not address the issues arising from high plasma frequencies. Nevertheless the particle scheme devised here can be in principle combined with the mentioned methods to overcome the stiff coupling issue.

Parallel to DSMC type approaches, simplified Landau-Fokker-Planck equations have been developed by many researchers [11, 13, 14, 12, 15]. The main idea consists in approximating the drift and diffusion coefficients based on physical arguments. A Fokker-Planck model with an *ad-hoc* friction coefficient was considered in [11]. More rigorously, in the collision field method drift and diffusion are derived from fluid equations [13]. Furthermore, Coulomb collisions have been reformulated as stochastic differential equations, that is, as Langevin equations [14]. While nonlinear drift models have been derived, e.g. in [12], they do not ensure convergence to a Maxwellian equilibrium distribution and thus extra caution is required for their use.

Provided accurate time integration schemes, Fokker-Planck models can enjoy much less restrictive spatio-temporal discretization requirements compared to DSMC algorithms. This comes through the fact that the corresponding stochastic processes arising from the Fokker-Planck equation are continuous in time. Motivated by this potential of the Fokker-Planck approach, this work presents a novel integration technique for the Fokker-Planck model with linear drift and constant diffusion coefficient in the presence of elecrtromagnetic forces. The derivation follows closely the Fokker-Planck solution algorithms which have been developed by the authors for neutral gas flows [16, 17]. While the employed Fokker-Planck model can be regarded as one of the simplest in the hierarchy of existing ones, the extension to higher order drift and diffusion terms honoring the H-theorem will be addressed in subsequent works (similar to the approach developed in [17]). Accordingly, the presented numerical scheme will be generalized for more complex Fokker-Planck models.

The paper is structured as follows: Next, in section 2, the Vlasov-Fokker-Planck equation with linear drift and constant diffusion coefficient is reviewed and discussed. Section 3 deals with its numerical solution, that is, the new particle integration scheme is introduced, a scheme to extract statistical moments is explained, the boundary condition treatment is described and the overall solution algorithm is outlined. Numerical studies are presented in section 4. The objective of the first series of test cases, which considers isolated particle trajectories in constant electromagnetic fields, is to demonstrate that the new particle integration scheme is exact. With the second series of test cases it is shown for 2D flow of deuterium ions in a spatially varying electromagnetic field that very accurate solutions can be achieved with extremely coarse grids and large time steps. Finally, the paper closes with conclusions.

2. Vlasov-Fokker-Planck Equation

A plasma with several species s is considered. Magnetic and electric fields $\boldsymbol{B}$ and $\boldsymbol{E}$, respectively, are given, and the species particles interact with each other due to Coulomb forces. The density $\mathcal{F}_s$ of species s in the $\boldsymbol{x}$-$\boldsymbol{v}$-space ($\boldsymbol{x}$ and $\boldsymbol{v}$ are the physical space and velocity coordinates) is governed by

$$\frac{\partial \mathcal{F}_s}{\partial t} + v_i \frac{\partial \mathcal{F}_s}{\partial x_i} + \frac{\partial}{\partial v_i}\left(\left\langle \frac{Du_{s_i}}{Dt}\right\rangle \mathcal{F}_s\right) = 0,$$

where $\langle D\boldsymbol{u}_s/Dt\rangle$ is the mean acceleration of particles at $(\boldsymbol{x},\boldsymbol{v})$. If one represents the density $\mathcal{F}_s$ by a cloud of nominal particles, each of which with a weight w_s^*, a position $\boldsymbol{x}_s^*$ and a velocity $\boldsymbol{u}_s^*$, then the evolution of $\mathcal{F}_s$ is obtained by evolving these nominal particles, e.g. according to the Langevin model

$$d\boldsymbol{x}_s^* = \boldsymbol{u}_s^* dt \quad \text{and} \tag{1}$$

$$d\boldsymbol{u}_s^* = (\boldsymbol{a}_s - \boldsymbol{b}_s \boldsymbol{u}_s^*)dt + \boldsymbol{c}_s\, d\boldsymbol{W}_s^*, \tag{2}$$

where $\boldsymbol{a}_s$ and $\boldsymbol{W}_s^*$ are vectors and $\boldsymbol{b}_s$ and $\boldsymbol{c}_s$ matrices. Note that $W_{s_i}^*(t)$ is a Wiener process with $\langle dW_{s_i}^*\rangle \equiv 0$ and $\langle dW_{s_i}^* dW_{s_j}^*\rangle \equiv dt\delta_{ij}$ ($\langle\cdot\rangle$ denotes expectation). All quantities without superscript $*$ are evaluated at the location $\boldsymbol{x}_s^*(t)$ and time t.

Note that in the limit of infinitely many nominal particles solving system (1)-(2) is equivalent to solving the Fokker-Planck equation

$$\frac{\partial \mathcal{F}_s}{\partial t} + v_i \frac{\partial \mathcal{F}_s}{\partial x_i} + \frac{\partial}{\partial v_i}\left((a_{s_i} - b_{s_{ik}} v_k)\mathcal{F}_s\right) = \frac{\partial^2}{\partial v_i \partial v_j}\left(\frac{c_{sik} c_{s_{jk}}}{2}\mathcal{F}_s\right). \tag{3}$$

Further, with

$$\boldsymbol{a}_s = \frac{q_s}{m_s}\boldsymbol{E} + \eta_s \boldsymbol{U}_s, \tag{4}$$

$$\boldsymbol{b}_s = -\frac{q_s}{m_s}\tilde{\boldsymbol{B}} + \eta_s \boldsymbol{I}, \tag{5}$$

$$\boldsymbol{c}_s = \sqrt{2\eta_s \frac{k_B T_{th}}{m_s}}\boldsymbol{I} \tag{6}$$

and

$$\tilde{\boldsymbol{B}} = \begin{bmatrix} 0 & B_3 & -B_2 \\ -B_3 & 0 & B_1 \\ B_2 & -B_1 & 0 \end{bmatrix} \tag{7}$$

Eq. (3) represents the Vlasov-Fokker-Planck equation with linear drift and constant diffusion coefficient. In this case, Coulomb collisions are described by a drift of the individual particle velocities $\boldsymbol{u}_s^*$ towards the local mean bath velocity $\boldsymbol{U}_s$ at the rate η_s (friction coefficient), superimposed by random walk in velocity space with the constant coefficient $\boldsymbol{c}_s$. Elementary particle mass and charge are denoted by m_s and q_s, respectively, T_{th} is the temperature and $k_B = 1.38064852 \times 10^{-23} m^2 kg/(s^2 K)$ the Boltzmann constant.

3. Numerical Solution

In the following subsection, a statistically exact integration of the system (1)-(2) for constant coefficients $\boldsymbol{a}_s$, $\boldsymbol{b}_s$ and $\boldsymbol{c}_s$ is derived. Such a scheme allows for efficient numerical solutions of Eq. (3) (for a given electromagnetic field) While spatio-temporal variations of the macroscopic fields, including the friction coefficient, bulk velocity and electromagnetic fields still have to be resolved, the time step size can be chosen independent of the Debye length and the strength of external $\boldsymbol{E}$- and $\boldsymbol{B}$-fields. Subsection 3.2 describes how spatially and temporally varying moments like macroscopic spatial density, temperature and velocity, can be estimated, subsection 3.3 explains how to treat wall boundary conditions, in subsection 3.4 a solution algorithm for numerical plasma simulations is presented and in subsection 3.5 the computational cost of the new particle time integration scheme is discussed.

3.1. Time Integration Scheme

To integrate the system (1)-(2) from the time $t = t^n$ to $t = t^{n+1} = t^n + \Delta t$ during one time step, it is transformed such that the individual components get decoupled. For simplicity, the subscript s is omitted for the following derivations. With $\boldsymbol{b} = \boldsymbol{R}\boldsymbol{\beta}\boldsymbol{R}^{-1}$, where $\boldsymbol{R}$ and

$$\boldsymbol{\beta} = \begin{bmatrix} \beta_1 & 0 & 0 \\ 0 & \beta_2 & 0 \\ 0 & 0 & \beta_3 \end{bmatrix}$$

are the eigenvector- and eigenvalue matrices of $\boldsymbol{b}$, one obtains

$$d\boldsymbol{R}^{-1}\boldsymbol{x}^* = \boldsymbol{R}^{-1}\boldsymbol{u}^* dt \quad \text{and} \tag{8}$$

$$d\boldsymbol{R}^{-1}\boldsymbol{u}^* = (\boldsymbol{R}^{-1}\boldsymbol{a} - \boldsymbol{\beta}\boldsymbol{R}^{-1}\boldsymbol{u}^*)dt + \boldsymbol{R}^{-1}\boldsymbol{c}\, d\boldsymbol{W}^*, \tag{9}$$

and with $\hat{\boldsymbol{u}}^* = \boldsymbol{R}^{-1}\boldsymbol{u}^*$, $\hat{\boldsymbol{x}}^* = \boldsymbol{R}^{-1}\boldsymbol{x}^*$, $\boldsymbol{\alpha} = \boldsymbol{\beta}^{-1}\boldsymbol{R}^{-1}\boldsymbol{a}$ and $\boldsymbol{\gamma} = \boldsymbol{R}^{-1}\boldsymbol{c}$ the transformed system reads

$$d\hat{\boldsymbol{x}}^* = \hat{\boldsymbol{u}}^* dt \quad \text{and} \tag{10}$$

$$d\hat{\boldsymbol{u}}^* = \boldsymbol{\beta}\,(\boldsymbol{\alpha} - \hat{\boldsymbol{u}}^*)\, dt + \boldsymbol{\gamma} d\boldsymbol{W}^*, \tag{11}$$

or equivalently with index notation

$$\begin{aligned} d\hat{x}_i^* &= \hat{u}_i^* dt \quad \text{and} \\ d\hat{u}_i^* &= \beta_{(i)}\,(\alpha_i - \hat{u}_i^*)\, dt + \gamma_{ik} dW_k^*. \end{aligned}$$

3.1.1. Integration of velocity along particle trajectories

For frozen values of α_i, β_i, and γ_{ik} exact integration of above equation leads to the conditional expectation

$$\langle \hat{u}_i^*(t^{n+1}) | \hat{u}_i^*(t^n) \rangle = \underbrace{\alpha_i + \left(\hat{u}_i^*(t^n) - \alpha_i\right) e^{-\beta_{(i)}\Delta t}}_{d_i}. \tag{12}$$

Note that any perturbation ϵ added to $\hat{u}_i^*$ at $t' \in [t^n, t^{n+1}]$ decays as

$$\epsilon(t^{n+1}) = \epsilon(t') e^{-\beta_i (t^{n+1} - t')}. \tag{13}$$

Therefore, by interpreting $\gamma_{ik} dW_k^* = \gamma_{ik} \xi_k \sqrt{dt}$ as an infinitesimal perturbation (where ξ_k are independent normal random variables with zero mean and a variance of one), one can write

$$\hat{u}_i^*(t^{n+1}) = d_i + \lim_{N \to \infty} \sum_{l=1}^{N} \left[\gamma_{ik} \xi_k^l \sqrt{\Delta t / N}\; e^{-\beta_{(i)} l \Delta t / N} \right]. \tag{14}$$

Thus one obtains

$$\begin{aligned} \langle \hat{u}_i^*(t^{n+1}) \hat{u}_j^*(t^{n+1}) | \hat{\boldsymbol{u}}^*(t^n) \rangle &= d_i d_j + \gamma_{ik}\gamma_{jk} \lim_{N \to \infty} \sum_{l=1}^{N} \left[\langle \xi_{(k)}^{(l)} \xi_{(k)}^{(l)} \rangle\; e^{-(\beta_{(i)} + \beta_{(j)}) l \Delta t / N} \frac{\Delta t}{N} \right] \\ &= d_i d_j + \gamma_{ik}\gamma_{jk} \int_0^{\Delta t} \left[e^{-(\beta_{(i)} + \beta_{(j)}) t'} dt' \right] \\ &= d_i d_j + \underbrace{\gamma_{ik}\gamma_{jk} \frac{1}{\beta_{(i)} + \beta_{(j)}} \left(1 - e^{-(\beta_{(i)} + \beta_{(j)}) \Delta t} \right)}_{A_{ij}}, \end{aligned} \tag{15}$$

and under the frozen coefficient assumption the scheme

$$\hat{\boldsymbol{u}}^*(t^{n+1}) \quad = \quad \boldsymbol{d} \ + \ \boldsymbol{A}^{1/2}\boldsymbol{\xi}_u^{n+1} \tag{16}$$

is exact. Note that $\boldsymbol{A}^{1/2} = \boldsymbol{R}_A \boldsymbol{\Lambda}_A^{1/2} \boldsymbol{R}_A^{-1}$, where $\boldsymbol{R}_A$ and $\boldsymbol{\Lambda}_A$ are the eigenvector and eigenvalue matrices of the symmetric matrix $\boldsymbol{A}$, respectively. Further, $\xi_{u_k}^{n+1}$ are normal random variables with zero mean and a variance of one.

3.1.2. *Integration of particle positions*

For the conditional expectation of the particle dislocation $\Delta\hat{x}_i^{n+1} = \hat{x}_i^{n+1} - \hat{x}_i^n$ one has to integrate the conditional expectation of the particle velocity (12) and obtains

$$\langle \Delta\hat{x}_i^*(t^{n+1}) | \hat{u}_i^*(t^n) \rangle \quad = \quad \underbrace{\alpha_i \Delta t \ + \ (\hat{u}_i^*(t^n) - \alpha_i)\, \frac{1}{\beta_{(i)}}(1 - e^{-\beta_{(i)}\Delta t})}_{f_i}. \tag{17}$$

Any perturbation ϵ added to $\hat{u}_i^*$ at $t' \in [t^n, t^{n+1}]$ contributes to the dislocation $\Delta\hat{x}_i^*(t^{n+1})$ by the amount

$$\frac{\epsilon(t')}{\beta_{(i)}} \left(1 - e^{-\beta_{(i)}(t^{n+1}-t')}\right).$$

Again, interpreting $\gamma_{ik} dW_k^*$ as an infinitesimal perturbation, one can write

$$\Delta\hat{x}_i^*(t^{n+1}) \quad = \quad f_i \ + \ \frac{\gamma_{ik}}{\beta_{(i)}} \lim_{N\to\infty} \sum_{l=1}^{N} \left[\xi_k^l \sqrt{\Delta t/N} \left(1 - e^{-\beta_{(i)} l \Delta t/N}\right)\right] \tag{18}$$

and obtains

$$\begin{aligned} \langle \Delta\hat{x}_i^*(t^{n+1}) \Delta\hat{x}_j^*(t^{n+1}) | \hat{\boldsymbol{u}}^*(t^n) \rangle \quad &= \quad f_i f_j \ + \ \frac{\gamma_{ik}\gamma_{jk}}{\beta_{(i)}\beta_{(j)}} \lim_{N\to\infty} \sum_{l=1}^{N} \left[\left(1 - e^{-\beta_{(i)} l\Delta t/N}\right)\left(1 - e^{-\beta_{(j)} l\Delta t/N}\right) \frac{\Delta t}{N}\right] \\ &= \quad f_i f_j \ + \ \frac{\gamma_{ik}\gamma_{jk}}{\beta_{(i)}\beta_{(j)}} \int_0^{\Delta t} \left[\left(1 - e^{-\beta_{(i)}t'} - e^{-\beta_{(j)}t'} + e^{-(\beta_{(i)}+\beta_{(j)})t'}\right) dt' \right] \\ &= \quad f_i f_j \ + \ B_{ij} \end{aligned} \tag{19}$$

with

$$B_{ij} \ = \ \frac{\gamma_{ik}\gamma_{jk}}{\beta_{(i)}\beta_{(j)}} \Big(\Delta t \ + \ \frac{1}{\beta_{(i)}}(e^{-\beta_{(i)}\Delta t} - 1) \ + \ \frac{1}{\beta_{(j)}}(e^{-\beta_{(j)}\Delta t} - 1) \ - \ \frac{1}{\beta_{(i)} + \beta_{(j)}}(e^{-(\beta_{(i)}+\beta_{(j)})\Delta t} - 1)\Big). \tag{20}$$

As a result, one obtains the exact scheme

$$\Delta\hat{\boldsymbol{x}}^*(t^{n+1}) = \boldsymbol{f} + \boldsymbol{B}^{1/2}\boldsymbol{\xi}_x^{n+1}, \tag{21}$$

where $\boldsymbol{B}^{1/2} = \boldsymbol{R}_B \boldsymbol{\Lambda}_B^{1/2} \boldsymbol{R}_B^{-1}$ and $\xi_{x_i}^{n+1}$ are normal distributed random variables with zero mean and a variance of one. $\boldsymbol{R}_B$ and $\boldsymbol{\Lambda}_B$ respectively are the eigenvector and eigenvalue matrices of the symmetric matrix $\boldsymbol{B}$. Next it is explained how the two random vectors $\boldsymbol{\xi}_u^{n+1}$ and $\boldsymbol{\xi}_x^{n+1}$ have to be correlated in order to honor the correct conditional covariance $\langle \Delta\hat{x}_i^*(t^{n+1}) \hat{u}_j^*(t^{n+1}) | \hat{\boldsymbol{u}}^*(t^n) \rangle$.

3.1.3. *Conditional covariance of particle velocity and displacement*

The conditional covariance of particle velocity and displacement can be derived by taking the expectation of the product of Eqs. (14) and (18), which leads to

$$\begin{aligned} \langle \Delta\hat{x}_i^*(t^{n+1}) \hat{u}_j^*(t^{n+1}) | \hat{\boldsymbol{u}}^*(t^n) \rangle \quad &= \quad f_i d_j \ + \ \frac{\gamma_{ik}\gamma_{jk}}{\beta_{(j)}} \lim_{N\to\infty} \sum_{l=1}^{N} \left[\langle \xi_{(k)}^{(l)} \xi_{(k)}^{(l)} \rangle \left(e^{-\beta_{(j)} l\Delta t/N} - e^{-(\beta_{(i)}+\beta_{(j)}) l\Delta t/N}\right) \frac{\Delta t}{N}\right] \\ &= \quad f_i d_j \ + \ \frac{\gamma_{ik}\gamma_{jk}}{\beta_{(j)}} \int_0^{\Delta t} \left[\left(e^{-\beta_{(j)}t'} - e^{-(\beta_{(i)}+\beta_{(j)})t'}\right) dt' \right] \\ &= \quad f_i d_j \ + \ C_{ij} \end{aligned} \tag{22}$$

with

$$C_{ij} = \frac{\gamma_{ik}\gamma_{jk}}{\beta_{(j)}}\left(\frac{1}{\beta_{(j)}}\left(1-e^{-\beta_{(j)}\Delta t}\right)+\frac{1}{\beta_{(i)}+\beta_{(j)}}\left(e^{-(\beta_{(i)}+\beta_{(j)})\Delta t}-1\right)\right). \tag{23}$$

In order to achieve consistency with this analytical result, one first choses the independent normal random variables $\xi^{n+1}_{1_k}$ and $\xi^{n+1}_{2_k}$ (with zero mean and variance one) and then sets

$$\begin{aligned} \boldsymbol{\xi}^{n+1}_x &= \boldsymbol{\xi}^{n+1}_1 \quad \text{and} \qquad (24)\\ \boldsymbol{\xi}^{n+1}_u &= \boldsymbol{F}\boldsymbol{\xi}^{n+1}_1 + \boldsymbol{G}\boldsymbol{\xi}^{n+1}_2 \qquad (25) \end{aligned}$$

with $\boldsymbol{F} = \boldsymbol{A}^{-1/2}\boldsymbol{C}^T\boldsymbol{B}^{-1/2}$ and $\boldsymbol{G}\boldsymbol{G}^T = \boldsymbol{I} - \boldsymbol{A}^{-1/2}\boldsymbol{C}^T\boldsymbol{B}^{-1}\boldsymbol{C}\boldsymbol{A}^{-1/2}$. Note that here $(\cdot)^T$ denotes the conventional transpose, although the matrices in general are complex. With these correlated vectors it is straightforward to show that the evolution schemes (16) and (21) are statistically consistent with the exact conditional moments (12), (15), (17), (19) and (22). The new particle positions and velocities in the original space are obtained via the back-transformation

$$\begin{aligned} \boldsymbol{x}^*(t^{n+1}) &= \boldsymbol{R}(\hat{\boldsymbol{x}}^*(t^n) + \Delta\hat{\boldsymbol{x}}^*(t^{n+1})) \quad \text{and} \qquad (26)\\ \boldsymbol{u}^*(t^{n+1}) &= \boldsymbol{R}\hat{\boldsymbol{u}}^*(t^n), \qquad (27) \end{aligned}$$

where $\boldsymbol{R}$ may be different for each particle. A numerical solution algorithm based on this particle evolution scheme allows for large time steps, since only the variation of macroscopic quantities along particle trajectories have to be resolved, and neither the long range nor the collision time scales. This obviously is a computational advantage compared to first or second order time integration schemes.

3.2. Estimation of Moments

Along the particle trajectories spatially varying macroscopic quantities (statistical moments, e.g. macroscopic spatial density, velocity and temperature) have to be estimated. To approximate the average $\overline{Q} \in \{\overline{u}_i, \overline{u_j u_j}\}$ of $Q^* \in \{u^*_i, u^*_j u^*_j\}$ at a location $\boldsymbol{x}$ and time t, ensemble averaging weighted with a local kernel function $\hat{g}(\boldsymbol{x}, \boldsymbol{x}')$ is employed, that is,

$$\overline{Q}(\boldsymbol{x}, t) \approx \frac{\sum_l^{N_p}\left[\hat{g}(\boldsymbol{x}, \boldsymbol{x}^{*^l}(t))w^{*^l}(t)Q^{*^l}(t)\right]}{W(\boldsymbol{x}, t)} \tag{28}$$

with

$$W(\boldsymbol{x}, t) \approx \sum_l^{N_p}\left[\hat{g}(\boldsymbol{x}, \boldsymbol{x}^{*^l}(t))w^{*^l}(t)\right], \tag{29}$$

where w^{*^l} denotes the statistical weight of particle $l \in \{1, \ldots, N_p\}$.

In the solution algorithm described next, all moments are estimated at some specified, fix coordinates $\boldsymbol{x}_j$ (grid nodes with $j \in \{1, \ldots, N_n\}$), and the kernel functions $\hat{g}_j(\boldsymbol{x}') = \hat{g}(\boldsymbol{x}_j, \boldsymbol{x}')$ form a partition of unity, that is,

$$\sum_j^{N_n}\left[\hat{g}_j(\boldsymbol{x})\right] \equiv 1 \tag{30}$$

for all $\boldsymbol{x}$ inside the computational domain Ω. To interpolate the extracted moments from the grid nodes to a particle position $\boldsymbol{x}^*$, the same kernel functions are employed as

$$\overline{Q}(\boldsymbol{x}^*(t), t) \approx \sum_j^{N_n}\left[\hat{g}_j(\boldsymbol{x}^*(t))\overline{Q}(\boldsymbol{x}_j, t)\right]. \tag{31}$$

Further, as a particular choice used here, the grid nodes $\boldsymbol{x}_j$ mark the centers of non-overlapping, space filling sampling volumes Ω_j and

$$\hat{g}_j(\boldsymbol{x}') = \begin{cases} 1 & \text{if} \quad \boldsymbol{x}' \in \Omega_j \\ 0 & \text{else.} \end{cases} \tag{32}$$

Of course, noise in the estimated moments due to a finite number of computational particles within individual sampling volumes is an issue. In steady state calculations the statistical- and bias errors can dramatically be reduced by employing time averaged values of $\boldsymbol{U}$ and T_{th} for the evolution equations (16) and (21). If moving time averaging is applied, $\overline{Q}(\boldsymbol{x}_j, t^{n+1})$ is replaced by

$$\overline{Q}^{\mu}(\boldsymbol{x}_j, t^{n+1}) = (\mu\overline{Q}^{\mu}(\boldsymbol{x}_j, t^n)W^{\mu}(\boldsymbol{x}_j, t^n) + (1-\mu)\overline{Q}(\boldsymbol{x}_j, t^{n+1})W(\boldsymbol{x}_j, t^{n+1})) \;/\; W^{\mu}(\boldsymbol{x}_j, t^{n+1}), \tag{33}$$

where

$$W^{\mu}(\boldsymbol{x}_j, t^{n+1}) = \mu\, W^{\mu}(\boldsymbol{x}_j, t^n) + (1-\mu)\, W(\boldsymbol{x}_j, t^{n+1}) \tag{34}$$

with the time averaging coefficient $\mu \in [0, 1[$. Note that with the choice

$$\mu = \frac{n'}{n'+1} \tag{35}$$

and $n' = n - n^{start}$ one obtains the identities

$$W^{\mu}(\boldsymbol{x}_j, t^{n+1}) = \frac{1}{n'+1}\sum_{k=n^{start}}^{n+1}\left[W(\boldsymbol{x}_j, t^k)\right] \quad \text{and} \tag{36}$$

$$\overline{Q}^{\mu}(\boldsymbol{x}_j, t^{n+1}) = \frac{1}{n'+1}\sum_{k=n^{start}}^{n+1}\left[W(\boldsymbol{x}_j, t^k)\overline{Q}(\boldsymbol{x}_j, t^k)\right]/W^{\mu}(\boldsymbol{x}_j, t^{n+1}). \tag{37}$$

Using time averaging allows to perform simulations with fewer particles, which saves memory and in most cases it also is computationally beneficial.

3.3. Wall Boundary Conditions

Isothermal, diffusive walls are considered in this paper, and the corresponding boundary conditions are implemented as follows: Once a particle crosses a wall boundary, it is reinitialised at the estimated intersection. Its tangential velocity components are drawn independently from a normal distribution with zero mean and a variance of

$$\sigma^2_{wall} = k_B T_{wall}/m_s, \tag{38}$$

where T_{wall} is the wall temperature. The perpendicular velocity component is drawn from the distribution with the probability density function

$$f_{\perp}(v) = v/\sigma^2_{wall}\; e^{-v^2/(2\;\sigma^2_{wall})} \quad (\text{for } v \in \mathbb{R}^+); \tag{39}$$

in particular, the new perpendicular velocity component (pointing into the computational domain) can be set to

$$u^*_{\perp} = \sqrt{-2\sigma^2_{wall}\;\ln(\xi)}, \tag{40}$$

where ξ is an independent uniform random variable between zero and one. Together with the tangential components one then obtains the new particle velocity $\boldsymbol{u}^*_{wall}$.

The remaining challenge is to estimate location $\boldsymbol{x}^*_{coll}$ and time t^*_{coll} of the wall collision and to evolve the particle from there for the remaining time $\Delta t^*_{rem} = \Delta t + t^n - t^*_{coll}$. It is reasonable and simple to estimate $\boldsymbol{x}^*_{coll}$ by linear interpolation between $\boldsymbol{x}^*(t^n) \in \Omega$ and $\boldsymbol{x}^*_{pre} \notin \Omega$ (Ω is the computational domain), where $\boldsymbol{x}^*_{pre}$ refers to the predicted location at the end of time step $n+1$ without applying boundary conditions. Similarly, t^*_{coll} and $\boldsymbol{u}^*_{coll}$ can be estimated as

$$t^*_{coll} = \frac{|\boldsymbol{x}^*_{coll} - \boldsymbol{x}^*(t^n)|}{|\boldsymbol{x}^*_{pre} - \boldsymbol{x}^*(t^n)|}\Delta t + t^n \quad \text{and} \tag{41}$$

$$\boldsymbol{u}^*_{coll} = \frac{|\boldsymbol{x}^*_{coll} - \boldsymbol{x}^*(t^n)|}{|\boldsymbol{x}^*_{pre} - \boldsymbol{x}^*(t^n)|}(\boldsymbol{u}^*_{pre} - \boldsymbol{u}^*(t^n)) + \boldsymbol{u}^*(t^n). \tag{42}$$

Using the matrices

$$\boldsymbol{\Lambda}^*_u = \begin{bmatrix} e^{-\beta_1 \Delta t^*_{rem}} & 0 & 0 \\ 0 & e^{-\beta_2 \Delta t^*_{rem}} & 0 \\ 0 & 0 & e^{-\beta_3 \Delta t^*_{rem}} \end{bmatrix} \quad \text{and} \tag{43}$$

$$\boldsymbol{\Lambda}^*_x = \begin{bmatrix} \frac{1-e^{-\beta_1 \Delta t^*_{rem}}}{\beta_1} & 0 & 0 \\ 0 & \frac{1-e^{-\beta_2 \Delta t^*_{rem}}}{\beta_2} & 0 \\ 0 & 0 & \frac{1-e^{-\beta_3 \Delta t^*_{rem}}}{\beta_3} \end{bmatrix}, \tag{44}$$

the new position and velocity are then computed as

$$\boldsymbol{u}^*(t^{n+1}) = \boldsymbol{u}^*_{pre} + \boldsymbol{R}\boldsymbol{\Lambda}^*_u \boldsymbol{R}^{-1}(\boldsymbol{u}^*_{wall} - \boldsymbol{u}^*_{coll}) \quad \text{and} \tag{45}$$

$$\boldsymbol{x}^*(t^{n+1}) = \boldsymbol{x}^*_{pre} + \boldsymbol{R}\boldsymbol{\Lambda}^*_x \boldsymbol{R}^{-1}(\boldsymbol{u}^*_{wall} - \boldsymbol{u}^*_{coll}). \tag{46}$$

3.4. Solution Algorithm

The solution algorithm is outlined by the pseudo code in table 1. Before entering the time step loop, a grid with N_n nodes $\boldsymbol{x}_j$ and sampling volumes Ω_j with $\bigcup_{j=1}^{N_n} \Omega_j = \Omega$ has to be defined and N_p particles with $\boldsymbol{x}^* \in \Omega$, each having a velocity $\boldsymbol{u}^*$ and a weight w^*, have to be initialized. At the beginning of each time step $n+1$ the timestep size Δt is determined based on a simple CFL criterion, that is, such that no particle evolves further than into one of the neighbouring sampling volumes. If there exist in- and outflow boundaries, particles in ghost cells adjacent to these boundaries are created; with properties consistent with specified boundary conditions. Next, in each sampling volume Ω_j the macroscopic quantities required to solve the particle evolution equations (1) and (2) are estimated using the schemes (28) and (29). If one is only interested in steady state solutions, exponentially weighted time averaging may be applied to reduce statistical and bias errors. In the latter case, the macroscopic estimates as obtained from Eqs. (33) and (34) are employed. To evolve the particles, the macroscopic quantities $\boldsymbol{U} = \boldsymbol{U}(\boldsymbol{x}^*(t^n))$, $T_{th} = T_{th}(\boldsymbol{x}^*(t^n))$ and η together with the electromagnetic field (the latter may be computed by Maxwell's equations; not discussed here) are interpolated from the grid to their positions $\boldsymbol{x}^*(t^n)$, which is achieved with scheme (31), and for each particle two three-dimensional random vectors $\boldsymbol{\xi}_1$ and $\boldsymbol{\xi}_2$ with independent normal distributed components are generated. With this information available for each particle, they can be evolved by half a time step as described in subsection 3.1, which leads to their estimated trajectory segment mid-point positions $\boldsymbol{x}^*(t^{n+1/2})$; if $\boldsymbol{x}^*(t^{n+1/2}) \notin \Omega$, then boundary conditions have to be applied. Using $\boldsymbol{U}(\boldsymbol{x}^*(t^n/2))$, $T_{th}(\boldsymbol{x}^*(t^n/2))$, $\boldsymbol{B}(\boldsymbol{x}^*(t^n/2))$, $\boldsymbol{E}(\boldsymbol{x}^*(t^n/2))$ and $\eta(\boldsymbol{x}^*(t^n/2))$, their new positions $\boldsymbol{x}^*(t^{n+1})$ and velocities $\boldsymbol{u}^*(t^{n+1})$ are computed using the same random vectors as for $\boldsymbol{x}^*_{est}(t^{n+1})$; again boundary conditions have to be applied, if $\boldsymbol{x}^*(t^{n+1}) \notin \Omega$. Once the time step loop has terminated, any statistical moment of interest can be sampled from the resulting particle distribution; in order to obtain smoother time averaged estimates, however, sampling of these quantities has to be performed already during time stepping.

This approach based on a predictor- and a final particle time step is of 2nd order accuracy, which is in particular important to fully exploit the accuracy of the presented particle integration scheme in a domain in which the macroscopic quantities vary.

3.5. Computational Cost

The computational cost of the new particle integration scheme does not scale with the number of particles, since the most expensive calculations have to be done only once per time step for all particles in the same grid cell; in particular $\boldsymbol{\beta}$, $\boldsymbol{R}$, $\boldsymbol{R}^{-1}$, $\boldsymbol{A}^{1/2}$, $\boldsymbol{B}^{1/2}$, $\boldsymbol{C}^{1/2}$, $\boldsymbol{F}$ and $\boldsymbol{G}$ do not have to be recomputed for each particle, as in the presented solution algorithm macroscopic quantities are considered spatially constant in each grid cell. Moreover, the cost per time step is independent of the time step size. For the implementation the Eigen library was employed, which can efficiently handle complex numbers and Eigenvalue decompositions.

4. Numerical Studies

In this section, first the particle integration scheme and then the solution algorithm of table 1 are assessed. The goal was not to validate the employed Vlasov-Fokker-Planck model, but to verify the particle integration and to demonstrate the numerical accuracy and efficiency of the solution algorithm.

```
begin of program

define grid with $N_n$ nodes $\boldsymbol{x}_j$ and sampling volumes $\Omega_j$
initialize $N_p$ particles with $\boldsymbol{x}^*$, $\boldsymbol{u}^*$ and $w^*$ in domain $\Omega$ (init. cond.)
initialise time: $n = 0, t^n = 0$
while ($t^n < t^{end}$) {
    determine $\Delta t$
    $t^{n+1} = t^n + \Delta t$
    initialize particles in ghost cells (in- and outflow boundary conditions)
    $\forall$ grid nodes: {
        estimate $\boldsymbol{U}$ and $T_{th}$
    }
    $\forall$ particles: {
        interpolate $\boldsymbol{U}$, $T_{th}$, $\boldsymbol{B}$, $\boldsymbol{E}$ and $\eta$ from grid nodes to $\boldsymbol{x}^*(t^n)$
        determine $\boldsymbol{\xi}_1^{n+1}$ and $\boldsymbol{\xi}_2^{n+1}$
        estimate new particle locations $\boldsymbol{x}^*(t^{n+1/2})$ after a half time step
        apply boundary conditions to particles with $\boldsymbol{x}^*(t^{n+1/2}) \notin \Omega$
        interpolate $\boldsymbol{U}$, $T_{th}$, $\boldsymbol{B}$, $\boldsymbol{E}$ and $\eta$ from grid nodes to $\boldsymbol{x}^*(t^{n+1/2})$
        determine $\boldsymbol{x}^*(t^{n+1})$ and $\boldsymbol{u}^*(t^{n+1})$
        apply boundary conditions to particles with $\boldsymbol{x}^*(t^{n+1}) \notin \Omega$
    }
    $n + 1 \rightarrow n$
}
$\forall$ grid nodes: {
    estimate $\boldsymbol{U}$, $T_{th}$ and other macroscopic quantities of interest
}

end of program
```

Table 1: Pseudo code of the solution algorithm to solve the Vlasov-Fokker-Planck equation (3) with provided electromagnetic field.

4.1. Time Stepping Scheme Verification

Most commonly Euler-Maryuama and Milstein schemes are used for time integration of stochastic processes [18]. In PIC algorithms they are employed together with the Boris method for integration of the Lorentz force [19, 2]. For verification of our exact time integration scheme, we performed a numerical study and compared our results with those of the first order Euler-Maruyama scheme. Let a Dirac delta be the initial condition $\mathcal{F}(\boldsymbol{v}, \boldsymbol{x}, t_0) = n_{\text{ref}}\delta(\boldsymbol{v})\delta(\boldsymbol{x})$ and the heat bath temperature be T_{th}. Consider $\boldsymbol{u}^n$ as the numerical approximation of $\boldsymbol{u}$ at time t^n, the Euler-Maruyama scheme provides

$$u_i^{n+1} = u_i^n - (a_i^n + b_{ik}^n u_k^n)\Delta t + \sqrt{\Delta t} c_{ij} \xi_j \quad \text{and} \tag{47}$$

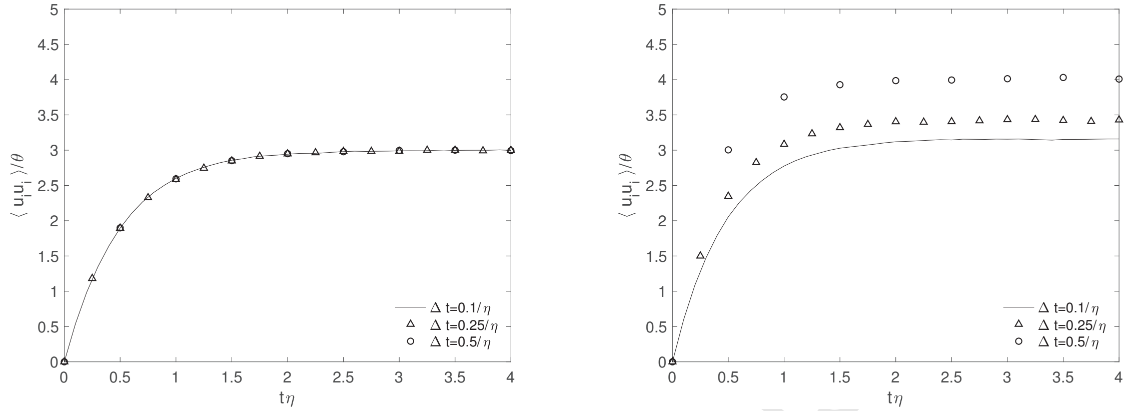
$$x_i^{n+1} = x_i^n + \left(\frac{u_i^n + u_i^{n+1}}{2} \right) \Delta t \tag{48}$$

for time integration of the stochastic processes (2) and (1) during $\Delta t = t^{n+1} - t^n$, where the superscripts n and $n+1$ denote the approximation at t^n and t^{n+1}, respectively. For this convergence study, 10^6 particles have been initialized according to $\mathcal{F}(\boldsymbol{v}, \boldsymbol{x}, t_0)$. The results are computed using different time step sizes, i.e. $\Delta t \in \{1/10, 1/5, 1/2\} 1/\eta$. Note that a constant magnetic field $\boldsymbol{B} = (0\ 0\ 0.1)^T m\eta/q$ in the absence of an electric field is assumed.

The normalized statistics $\langle u_i u_i \rangle/\theta$ (with $\theta = k_B T_{th}/m$) and $\langle u_i x_i \rangle \eta/\theta$ are estimated for $t \in (0 \ \ 4/\eta]$ and are shown in figures 1 and 2, respectively. As expected, the results of the exact scheme do not depend on the time step size, while the first order time integration leads to an artificially large diffusion; even for a time step size as small as $0.1/\eta$.

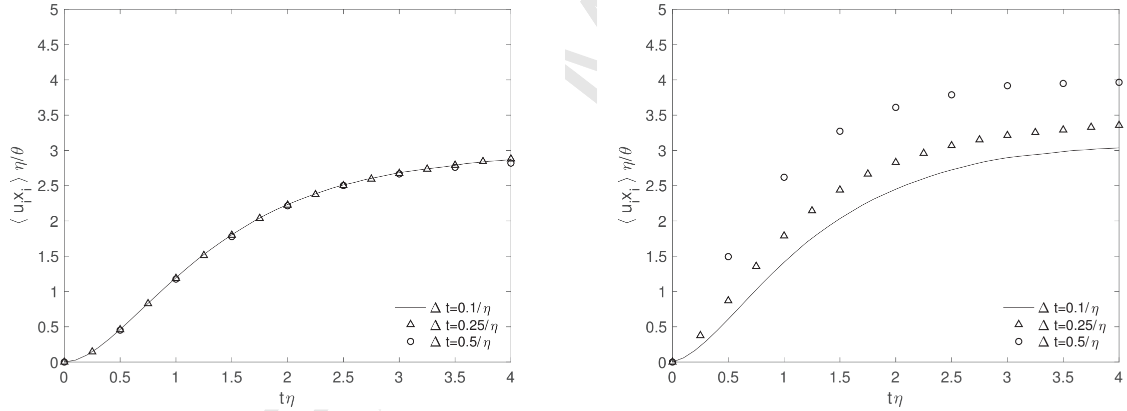
Note that since the SDEs (1)-(2) are subject to the additive noise, here the Euler-Maruyama and Milstein schemes are identical (both result in first order strong and weak convergence) [18]. To further improve the order of a SDE time integration scheme to second order, the resulting schemes become progressively complex. For example, the Runge-Kutta scheme proposed in [23], requires a fourth order Runge-Kutta time integration of the drift, in order to achieve a strong second order convergence. This means, for each time step and each particle, the second-order SDE Runge-

Kutta scheme needs to evaluate the drift term four times. In contrast, the proposed scheme is exact and furthermore it does not require multiple evaluations of the drift functional.



(a) (b)

Fig. 1: Normalized ensemble energy $\langle u_i u_i \rangle/\theta$ computed with the exact scheme of subsection 3.1 and the Euler-Maruyama method (47)-(48) shown on left and right, respectively.



(a) (b)

Fig. 2: Normalized position-velocity correlation $\langle u_i x_i \rangle \eta/\theta$ computed with the exact scheme of subsection 3.1 and the Euler-Maruyama method (47)-(48) shown on left and right, respectively.

4.2. Relaxation from non-equilibrium initial distribution

One of the successful particle Monte-Carlo solution algorithms to account for Coulomb collisions in plasmas is Nanbu's method [8], which once converged it can provide a physically accurate simulation of collision processes in plasmas. In the following, we present a comparison between Nanbu's method and the devised time integration scheme. Yet since the proposed scheme (26)-(27) is built upon the Fokker-Planck model with linear drift and constant diffusion, there are intrinsic simplifying assumptions involved in the devised time integration scheme. This prevents an appropriate comparison between the numerical convergence of the two schemes. However, in the linear relaxation regime one expects that the linear drift model accurately describes the process. Therefore, a comparison between Nanbu's scheme and the proposed time integration in terms of the numerical accuracy becomes relevant.

Consider a plasma relaxation phenomenon due to electron-electron collisions with $\boldsymbol{E} = \boldsymbol{B} = 0$. Let the initial PDF be

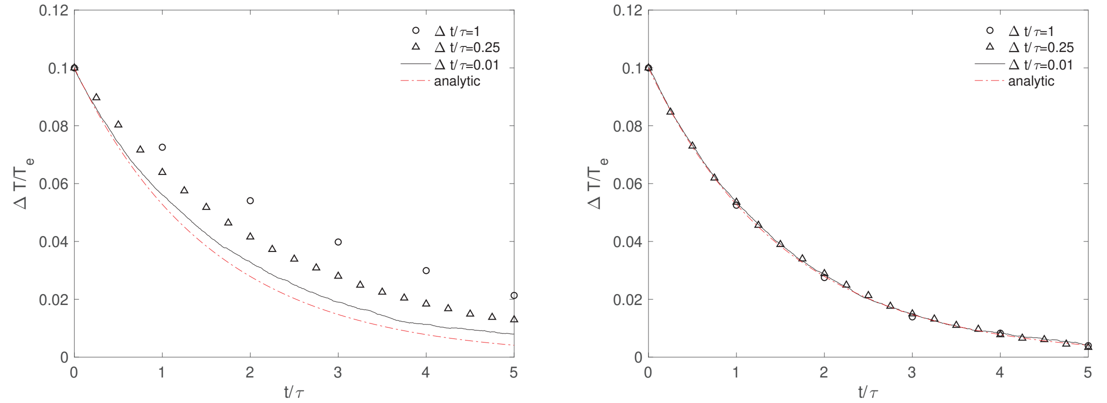


Fig. 3: Normalized temperature anisotropy relaxation computed with Nanbu's method and the proposed scheme (27) shown left and right, respectively. The dashed line denotes the solution of Eq. (50)

an anistropic Maxwellian with directional temperatures $T_x = 1.1T_y$ and $T_y = T_z$. Similar to [8], we introduce a time scale

$$\frac{1}{\tau} \quad = \quad \frac{n_e {q_e}^4 \ln \Lambda}{8\pi \sqrt{2}\epsilon_0^2 \; m_e^{1/2}(kT_e)^{3/2}}, \tag{49}$$

where $T_e = 1/3(T_x + T_y + T_z)$, n_e is the number density, ϵ_0 is the vacuum permittivity and Λ denotes the normalized Debye length. For this setting, an analytical solution exists in the limit of $\Delta T = T_x - T_y \ll T_x$ [8]. In fact ΔT relaxes through

$$\Delta T(\hat{t}) = \Delta T_0 \exp\left(-\frac{8}{5\sqrt{2\pi}}\hat{t}\right), \tag{50}$$

where $\hat{t} = t/\tau$ and ΔT_0 is the initial value of ΔT. In order to recover the correct relaxation behaviour, we adopt $\eta = 4/(5\sqrt{2\pi}\tau)$ and $T_{th} = T_e$ in the drift-diffusion closure of the Fokker-Planck model. At this point we can compare the convergence of the devised scheme with respect to Nanbu's method. We employ $N = 10^6$ particles and different time step sizes $\Delta t \in \{0.01, 0.25, 1\}\tau$. For the baseline we plot the solution given by Eq. (50).

The relaxation of the normalized anisotropy is shown in Fig. 3. The results of the devised integration scheme are independent of Δt, while Nanbu's scheme leads to a significant overshoot at large time step sizes.

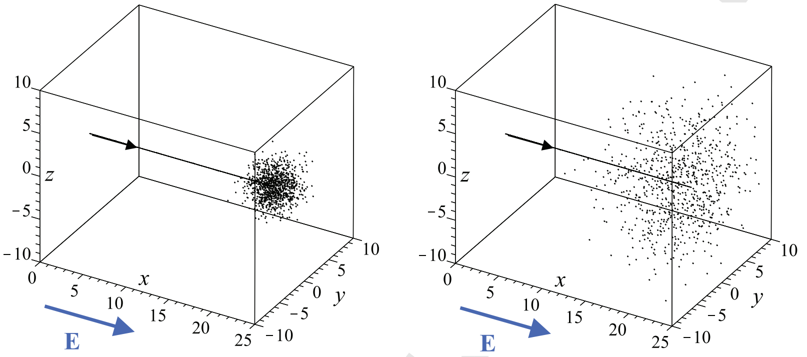
4.3. Particle Trajectories

For the isolated particle trajectories presented in this subsection, the $\boldsymbol{B}$- and $\boldsymbol{E}$-fields as well as the bath temperature T_{th}, the macroscopic velocity $\boldsymbol{U}$ and the friction coefficient η are specified and kept constant. This is representative for one particle time step calculation, during which these macroscopic quantities are kept constant as well, e.g. estimated at the midpoint of the trajectory segment. For all trajectories deuterium ions ($q = 1.6022 \times 10^{-19}C$ and $m = 3.3435 \times 10^{-27}kg$) with zero initial velocity are considered, and in all cases $T_{th} = 500K$, $\boldsymbol{U} = (0\;0\;0)^T m/s$ and $\boldsymbol{E} = (1000\;0\;0)^T kg\; m/(s^2C)$, while different $\boldsymbol{B}$-fields and friction coefficients were chosen; see table 2.

Trajectory 1: . In the first case, a deuterium ion (initially at rest at the origin of the coordinate system) gets accelerated in x-direction by an electric field of 1000 $kg\; m/(s^2C)$, while it interacts with a bath of temperature $T_{th} = 500\; K$ and mean velocity $\boldsymbol{U} = (0\;0\;0)^T m/s$. In figures 4a and 4b, mean trajectories for $t \in [0ns, 30ns]$ and endpoint clouds (each from 1000 samples) computed with $\eta = 4.8 \times 10^4 s^{-1}$ and $\eta = 4.8 \times 10^5 s^{-1}$, respectively, are depicted. Spatial units, like in the following figures, are μm. The results are as expected, that is, the ion experiences a constant acceleration of $4.8 \times 10^{10}\; m/s^2$ due to the electric field, which would lead to a final position of $\boldsymbol{x} = (21.6\;0\;0)^T \mu m$ in the undisturbed case (with $\eta = 0\; s^{-1}$). Note that the expected trajectory endpoint with $\eta > 0\; s^{-1}$ does not exactly coincide with the endpoint of the undisturbed case (see table 3). Also expected is that diffusion increases with η.

		$\boldsymbol{B}\ [kg/(s\ C)]$	$\eta\ [1/s]$
trajectory 1	(a)	$(0\ 0\ 0)^T$	$4.8\ 10^4$
	(b)	$(0\ 0\ 0)^T$	$4.8\ 10^5$
trajectory 2	(a)	$(0\ 0\ 10)^T$	$4.8\ 10^4$
	(b)	$(0\ 0\ 10)^T$	$4.8\ 10^5$
trajectory 3	(a)	$(10\ 0\ 10)^T$	$4.8\ 10^3$
	(b)	$(10\ 0\ 10)^T$	$4.8\ 10^4$

Table 2: Values used to compute deuterium trajectories with zero initial velocity. For all trajectories $q = 1.6022 \times 10^{-19}C$, $m = 3.3435 \times 10^{-27}kg$, $T_{th} = 500K$, $\boldsymbol{E} = (1000\ 0\ 0)^T kg\ m/(s^2C)]$ and $\boldsymbol{U} = (0\ 0\ 0)^T m/s$.



(a) (b)

Fig. 4: Trajectory 1 - mean trajectories for $t \in [0ns, 30ns]$ and endpoint position clouds from 1000 samples; on the left with $\eta = 4.8 \times 10^4 s^{-1}$ and on the right with $\eta = 4.8 \times 10^5 s^{-1}$. The units of the coordinate axes are μm.

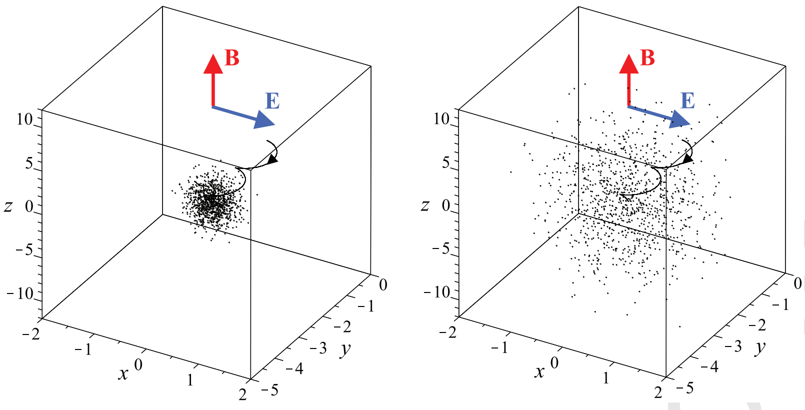
Trajectory 2: . The second case is similar to the first one, but now also with a magnetic field of 10 $kg/(s\ C)$ in z-direction. Figure 5 shows the mean trajectories for $t \in [0ns, 30ns]$ and corresponding endpoint clouds computed with $\eta = 4.8 \times 10^4 s^{-1}$ and $\eta = 4.8 \times 10^5 s^{-1}$, respectively. The $\boldsymbol{B}$-field is responsible for a right turn of the mean trajectories in the x-y-plane after initial acceleration by the $\boldsymbol{E}$-field. Later, while flying in opposite direction to the $\boldsymbol{E}$-field, the ion decelerates and "turns around", from where it gets accelerated again. Note that the net motion is perpendicular to both $\boldsymbol{B}$- and $\boldsymbol{E}$-fields. Again as expected, diffusion increases with η, but it is less intuitive that diffusion is highly anisotropic, which can be observed in figure 7a showing the endpoint cloud in a coordinate system with equally scaled axes.

Trajectory 3: . The third case is a modification of case 2, that is, here $\boldsymbol{B} = (10\ 0\ 10)^T kg/(s\ C)$ is considered instead of $\boldsymbol{B} = (0\ 0\ 10)^T kg/(s\ C)$. Figures 6a and 6b show the mean trajectories for $t \in [0ns, 30ns]$ and corresponding endpoint clouds computed with $\eta = 4.8 \times 10^3 s^{-1}$ and $\eta = 4.8 \times 10^4 s^{-1}$, respectively. The mean trajectories are now much less intuitive. Note that classical numerical integration would require many small time steps, while the analytical integration applied here allows for immediate endpoint calculations independent of the time step size. Again, diffusion increases with η and is highly anisotropic, as observed in figure 7b showing the endpoint cloud in a coordinate system with equally scaled axes.

For the trajectory test cases discussed above, the expected ion end point positions $\langle \boldsymbol{x} \rangle = \boldsymbol{Rf}$ and velocities $\langle \boldsymbol{u} \rangle = \boldsymbol{Rd}$ are shown in table 3 and the corresponding covariance matrices $\langle \boldsymbol{xx}^T \rangle = \boldsymbol{RBR}^T$, $\langle \boldsymbol{uu}^T \rangle = \boldsymbol{RAR}^T$ and $\langle \boldsymbol{xu}^T \rangle = \boldsymbol{RCR}^T$ are found in table 4.

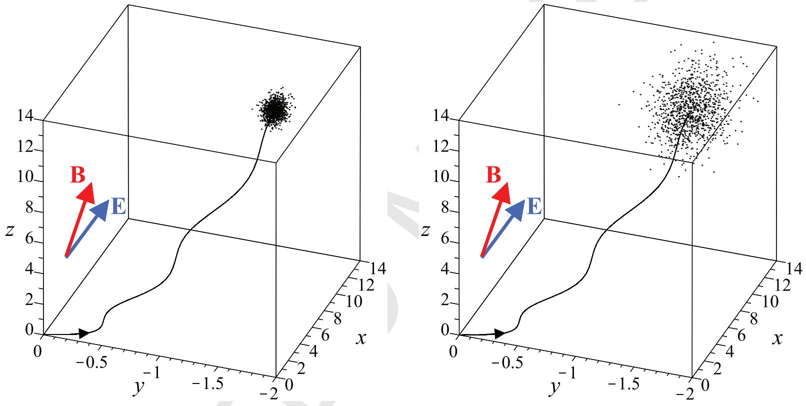
4.4. Convergence Studies

The solution algorithm is tested for flow of charged particles (deuterium ions) through a square computational domain of size $x_{ref} \times x_{ref}$ with a with a rectangular object of size $(0.2\ x_{ref}) \times (0.6\ x_{ref})$ at its center. The geometry of this test case is shown in figure 8, where periodic boundary conditions are applied in x-direction and isothermal,



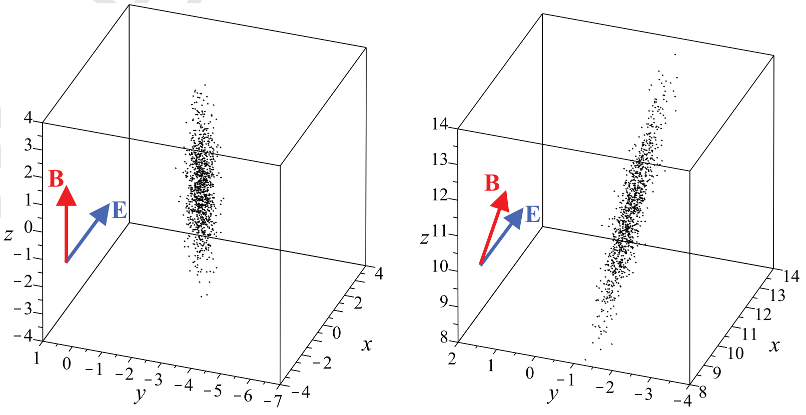
(a) (b)

Fig. 5: Trajectory 2 - mean trajectories for $t \in [0ns, 30ns]$ and endpoint position clouds from 1000 samples; on the left with $\eta = 4.8 \times 10^4 s^{-1}$ and on the right with $\eta = 4.8 \times 10^5 s^{-1}$. The units of the coordinate axes are μm.



(a) (b)

Fig. 6: Trajectory 3 - mean trajectories for $t \in [0ns, 30ns]$ and endpoint position clouds from 1000 samples; on the left with $\eta = 4.8 \times 10^3 s^{-1}$ and on the right with $\eta = 4.8 \times 10^4 s^{-1}$. The units of the coordinate axes are μm.



(a) (b)

Fig. 7: Trajectories 2a and 3b - endpoint position clouds from 1000 samples with $\eta = 4.8 \times 10^4 s^{-1}$ of trajectory 2a on the left and of trajectory 3b on the right. The anisotropy of the endpoint clouds is clearly visible, since here the coordinate axes (units are μm) are equally scaled.

trajectory	$\langle \boldsymbol{x} \rangle$	$\langle \boldsymbol{u} \rangle$
1a:	$\begin{pmatrix} 21.554 \\ 0 \\ 0 \end{pmatrix}$	$\begin{pmatrix} 1436.562 \\ 0 \\ 0 \end{pmatrix}$
1b:	$\begin{pmatrix} 21.461 \\ 0 \\ 0 \end{pmatrix}$	$\begin{pmatrix} 1427.311 \\ 0 \\ 0 \end{pmatrix}$
2a:	$\begin{pmatrix} 0.258 \\ -2.797 \\ 0 \end{pmatrix}$	$\begin{pmatrix} 97.035 \\ -123.609 \\ 0 \end{pmatrix}$
2b:	$\begin{pmatrix} 0.260 \\ -2.800 \\ 0 \end{pmatrix}$	$\begin{pmatrix} 95.899 \\ -123.219 \\ 0 \end{pmatrix}$
3a:	$\begin{pmatrix} 10.829 \\ -1.427 \\ 10.734 \end{pmatrix}$	$\begin{pmatrix} 753.954 \\ -45.522 \\ 683.538 \end{pmatrix}$
3b:	$\begin{pmatrix} 10.824 \\ -1.427 \\ 10.729 \end{pmatrix}$	$\begin{pmatrix} 753.446 \\ -45.525 \\ 683.116 \end{pmatrix}$

Table 3: Expected positions $\langle \boldsymbol{x} \rangle = \boldsymbol{Rf}$ and velocities $\langle \boldsymbol{u} \rangle = \boldsymbol{Rd}$ of the endpoints of trajectories 1-3 of subsection 4.3.

diffusive boundary conditions at the walls with $T_{wall} = T_{th_{ref}}$. The value of x_{ref} is found in table 5, which also shows the other reference values used for normalization of the results.

Note that the objective is to verify the solution algorithm and to present convergence studies. Therefore, in order to better focus on the numerical solution of the Vlasov-Fokker-Planck equation (3), a simplified scenario, which is not representative of realistic systems, is considered. Simplifications are:

- The drift is a linear function of the individual particle velocities, and the diffusion coefficient does not depend on the individual particle velocities; see Eqs. (3)-(6).
- $\boldsymbol{B}$- and $\boldsymbol{E}$-fields are specified and kept constant in time, that is, $\boldsymbol{E} = E_{ref}(1\,0\,0)^T$ and $\boldsymbol{B}(x) = \sin(2\pi x/x_{ref})B_{ref}(0\,0\,5)^T$. The corresponding values are stored at grid points (cell centers) and from there they are interpolated to the particle locations.
- A constant friction coefficient of $\eta = 10\eta_{ref}$ is employed, that is, the Coulomb collision time scale is ten times smaller than the reference time scale.
- Due to the huge mass ratio (ion mass divided by electron mass), the effect of the electrons on the deuterium ions is neglected.

Figures 9(a)-(d) depict steady state mean velocity fields of the deuterium ions computed with 5×5, 10×10, 20×20

trajectory	$\langle \boldsymbol{x}\boldsymbol{x}^T \rangle$ $[\mu m^2]$	$\langle \boldsymbol{u}\boldsymbol{u}^T \rangle$ $[m^2 s^{-2}]$	$\langle \boldsymbol{x}\boldsymbol{u}^T \rangle$ $[\mu m\ ms^{-1}]$
1a:	$\begin{pmatrix} 1.779 & 0 & 0 \\ 0 & 1.779 & 0 \\ 0 & 0 & 1.779 \end{pmatrix}$	$\begin{pmatrix} 5927.810 & 0 & 0 \\ 0 & 5927.810 & 0 \\ 0 & 0 & 5927.810 \end{pmatrix}$	$\begin{pmatrix} 88.917 & 0 & 0 \\ 0 & 88.917 & 0 \\ 0 & 0 & 88.917 \end{pmatrix}$
1b:	$\begin{pmatrix} 17.618 & 0 & 0 \\ 0 & 17.618 & 0 \\ 0 & 0 & 17.618 \end{pmatrix}$	$\begin{pmatrix} 58518.074 & 0 & 0 \\ 0 & 58518.074 & 0 \\ 0 & 0 & 58518.074 \end{pmatrix}$	$\begin{pmatrix} 877.756 & 0 & 0 \\ 0 & 877.756 & 0 \\ 0 & 0 & 877.756 \end{pmatrix}$
2a:	$\begin{pmatrix} 0.048 & 0 & 0 \\ 0 & 0.048 & 0 \\ 0 & 0 & 1.779 \end{pmatrix}$	$\begin{pmatrix} 5927.810 & 0.0 & 0 \\ 0 & 5927.810 & 0 \\ 0 & 0.0 & 5927.810 \end{pmatrix}$	$\begin{pmatrix} -1.066 & 11.534 & 0 \\ -11.534 & -1.066 & 0 \\ 0 & 0 & 88.917 \end{pmatrix}$
2b:	$\begin{pmatrix} 0.479 & 0 & 0 \\ 0 & 0.479 & 0 \\ 0 & 0 & 17.618 \end{pmatrix}$	$\begin{pmatrix} 58518.074 & 0 & 0 \\ 0 & 58518.074 & 0 \\ 0 & 0 & 58518.074 \end{pmatrix}$	$\begin{pmatrix} -10.732 & 113.842 & 0 \\ -113.842 & -10.732 & 0 \\ 0 & 0 & 877.756 \end{pmatrix}$
3a:	$\begin{pmatrix} 0.090 & 0 & 0.088 \\ 0 & 0.002 & 0 \\ 0.088 & 0 & 0.090 \end{pmatrix}$	$\begin{pmatrix} 593.548 & 0 & 0 \\ 0 & 593.548 & 0 \\ 0 & 0 & 593.548 \end{pmatrix}$	$\begin{pmatrix} 4.432 & 0.589 & 4.471 \\ 0.589 & -0.039 & 0.589 \\ 4.471 & -0.589 & 4.432 \end{pmatrix}$
3b:	$\begin{pmatrix} 0.902 & 0 & 0.877 \\ 0 & 0.025 & 0 \\ 0.877 & 0 & 0.902 \end{pmatrix}$	$\begin{pmatrix} 5927.810 & 0 & 0 \\ 0 & 5927.810 & 0 \\ 0 & 0 & 5927.810 \end{pmatrix}$	$\begin{pmatrix} 44.262 & 5.882 & 44.655 \\ -5.882 & -0.393 & 5.882 \\ 44.655 & -5.882 & 44.262 \end{pmatrix}$

Table 4: Covariance matrices $\langle \boldsymbol{x}\boldsymbol{x}^T \rangle = \boldsymbol{RBR}^T$, $\langle \boldsymbol{u}\boldsymbol{u}^T \rangle = \boldsymbol{RAR}^T$ and $\langle \boldsymbol{x}\boldsymbol{u}^T \rangle = \boldsymbol{RCR}^T$ of the endpoints of trajectories 1-3 of subsection 4.3.

and 40×40 grids, respectively, and with respective time step sizes of $t_{ref}/5$, $t_{ref}/10$, $t_{ref}/20$ and $t_{ref}/40$. In all cases the average number of computational particles per cell was 10 and time averaging was applied; for the results in figures 9(a)-(c) a time averaging coefficient of $\mu = 0.999$ was used and for figure 9(d) μ was 0.9999. Note that increasing μ has a similar effect as increasing the number of particles, that is, despite the small number of particles both bias and statistical errors were reduced to a low level. The very good qualitative agreement observed in figures 9 is confirmed by the excellent quantitative agreement of the mass flow rates obtained with the four grids; see table 6. Note that the mass flow rates, which were averaged over 10′000 time steps (starting after 15′000 time steps), are presented in units of total deuterium mass M_{dom} in the domain per reference time t_{ref}.

These results are very encouraging, as they show that very good approximations can be computed quickly on extremely coarse meshes. Obviously this is only possible with the new time integration scheme presented in this paper.

5. Conclusions

The Vlasov-Fokker-Planck kinetic equation, even in the setting of linear coefficients and constant electromagnetic fields, may arise to highly non-linear particle trajectories. Therefore typical time integration schemes (such as Euler-Maruyama or Milstein schemes) employed in PIC, may lead to an artificial diffusion in the phase space and other inaccuracies; unless small time step sizes are employed. Yet the stochastic system resulting from the Vlasov-Fokker-Planck equation with linear coefficients is in fact a six dimensional linear Itô process, for which an analytical solution could be constructed. Therefore, in order to capture the tortuosity of the particle paths besides honoring conserva-

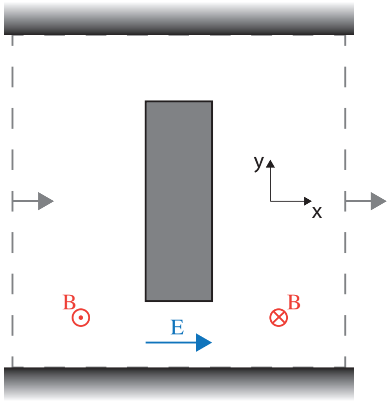


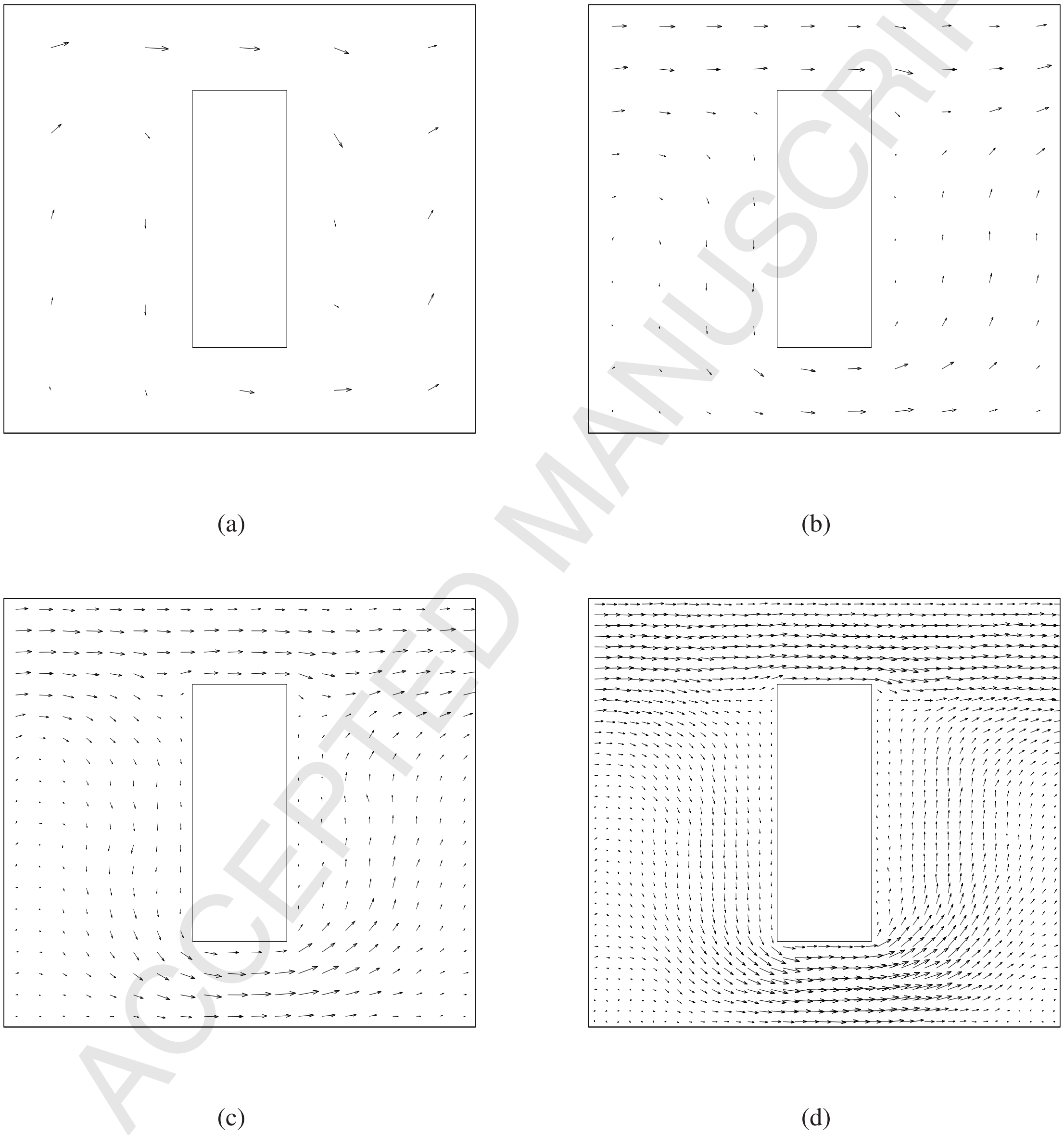
Fig. 8: Geometry of 2D test case with ***E***- and ***B***-fields.

symbol	relation	value
x_{ref}		$0.01m$
$T_{th_{ref}}$		$1000K$
m_{ref}		$3.3435 \times 10^{-27} kg$
q_{ref}		$1.6022 \times 10^{-19} C$
u_{ref}	$\sqrt{k_B T_{th_{ref}} / m_{ref}}$	$2032.1 m/s$
t_{ref}	x_{ref}/u_{ref}	$4.9211\ 10^{-6} s$
B_{ref}	$m_{ref}/(q_{ref} t_{ref})$	$4.241\ 10^{-3} kg/(s\ C)$
E_{ref}	$u_{ref} m_{ref}/(q_{ref} t_{ref})$	$8.1617 kg/(s\ C)$
η_{ref}	$1/t_{ref}$	$1.0321\ 10^{5}$

Table 5: Reference values and their relations.

# grid cells	Δt	flow rate
5×5	$t_{ref}/5$	$0.159\ M_{dom}/t_{ref}$
10×10	$t_{ref}/10$	$0.145\ M_{dom}/t_{ref}$
20×20	$t_{ref}/20$	$0.141\ M_{dom}/t_{ref}$
40×40	$t_{ref}/40$	$0.141\ M_{dom}/t_{ref}$

Table 6: Grid convergence for the 2D test case. The mass flow rates are given in units of total deuterium mass M_{dom} in the domain per reference time t_{ref}.



(a) (b)

(c) (d)

Fig. 9: 2D test case: The plots (a), (b), (c) and (d) show the velocity vectors obtained with the new particle time stepping scheme using 5 × 5, 10 × 10, 20 × 20 and 40 × 40 grids, respectively. The respective time step sizes were $t_{ref}/5$, $t_{ref}/10$, $t_{ref}/20$ and $t_{ref}/40$.

tion laws, an exact time integration scheme was devised in this paper. The spatio-temporal variations of the friction coefficient, bulk velocity and electromagnetic fields still have to be resolved, but the time step size can be chosen independent of the Coulomb collision frequency and cyclotron frequency. Our approach is relevant for cases where external elecromagnetic fields dominate long range Coulomb interactions. The new integration scheme was shown to be highly accurate even for quite coarse spatio-temporal discretizations. Furthermore, a solution algorithm is presented in the context of PIC methods, ensuring efficient sampling of particles statistics along with accurate treatment of wall boundaries. The methodology was assessed in a series of numerical experiments including ensemble trajectories and 2D flow around an obstacle, where the efficiency and accuracy of the devised scheme were demonstrated.

The main limitation of the presented solution algorithm is that it relies on spatially and temporally well resolved macroscopic fields. When coupled with Maxwell's equations especially the temporal rates of change of the electromagnetic fields can impose severe time step size restrictions, which are omitted in this work, but have to be addressed in the future. Another limitation of the new time integration scheme is that it was developed for a Vlasov-Fokker-Planck equation with linear drift and a constant diffusion coefficient. In sequel works, the integration scheme will be generalized for higher order drift and diffusion coefficients, and coupling with Maxwell's equations will be taken into account.

Acknowledgments

Hossein Gorji gratefully acknowledges the funding provided by Swiss National Science Foundation under the grant number 174060.

[1] G.A.Bird, Molecular Gas Dynamics and the Direct Simulation of Gas Flows, Claredon, Oxford (1994).

[2] C. K. Birdsall & A. B. Langdon. Plasma physics via computer simulation. CRC press, 2004.

[3] Garbet, X., Idomura, Y., Villard, L. and Watanabe, T.H., 2010. Gyrokinetic simulations of turbulent transport. Nuclear Fusion, 50(4), p.043002.

[4] Thomas, A. G. R., M. Tzoufras, A. P. L. Robinson, R. J. Kingham, C. P. Ridgers, Mark Sherlock, and A. R. Bell. ”A review of Vlasov-Fokker-Planck numerical modeling of inertial confinement fusion plasma.” Journal of Computational Physics 231, no. 3 (2012): 1051-1079.

[5] Dawson, J. (1962). One-dimensional plasma model. The Physics of Fluids, 5(4), 445-459.

[6] Ricci, Paolo. ”Simulation of the scrape-off layer region of tokamak devices.” Journal of Plasma Physics 81, no. 2 (2015).

[7] Takizuka, T., & Abe, H. (1977). A binary collision model for plasma simulation with a particle code. Journal of computational physics, 25(3), 205-219.

[8] Nanbu, K. (1997). Theory of cumulative small-angle collisions in plasmas. Physical Review E, 55(4), 4642.

[9] Bobylev, A. V., & Nanbu, K. (2000). Theory of collision algorithms for gases and plasmas based on the Boltzmann equation and the Landau-Fokker-Planck equation. Physical Review E, 61(4), 4576.

[10] Caflisch, R., Wang, C., Dimarco, G., Cohen, B., & Dimits, A. (2008). A hybrid method for accelerated simulation of Coulomb collisions in a plasma. Multiscale Modeling & Simulation, 7(2), 865-887.

[11] Lenard, A., & Bernstein, I. B. (1958). Plasma oscillations with diffusion in velocity space. Physical Review, 112(5), 1456.

[12] Rathmann, C. E., & Denavit, J. (1975). Simulation of collisional effects in plasmas. Journal of Computational Physics, 18(2), 165-187.

[13] Jones, M. E., Lemons, D. S., Mason, R. J., Thomas, V. A., & Winske, D. (1996). A grid-based Coulomb collision model for PIC codes. Journal of Computational Physics, 123(1), 169-181.

[14] Lemons, D. S., Winske, D., Daughton, W., & Albright, B. (2009). Small-angle Coulomb collision model for particle-in-cell simulations. Journal of Computational Physics, 228(5), 1391-1403.

[15] Hellinger, P., & Trávníĉek, P. M. (2010). Langevin representation of Coulomb collisions for bi-Maxwellian plasmas. Journal of Computational Physics, 229(14), 5432-5439.

[16] P.Jenny, M.Torrilhon, S.Heinz, A solution algorithm for the fluid dynamics equations based on a stochastic model for molecular motion, J. Comput. Phys. 229 (2010), 1077-1098.

[17] M.H.Gorji, M.Torrilhon & P.Jenny, Fokker-Planck model for computational studies of monatomic rarefied gas flows, J. Fluid Mech. 680 (2011), 574-601.

[18] P. E. Kloeden & E. Platen, Numerical Solution of Stochastic Differential Equations, Springer-Verlag (1992).

[19] Rosin, M. S., L. F. Ricketson, Andris M. Dimits, Russel E. Caflisch, and Bruce I. Cohen. ”Multilevel monte carlo simulation of coulomb collisions.” Journal of Computational Physics 274 (2014): 140-157.

[20] Frenod, Emmanuel and Hirstoaga, Sever A and Lutz, Mathieu and Sonnendrücker, Eric, Long time behaviour of an exponential integrator for a Vlasov-Poisson system with strong magnetic field, Communications in Computational Physics, 18 (2), 263–296, ()2015)

[21] Chen, Guangye and Chacón, Luis and Barnes, Daniel C, An energy-and charge-conserving, implicit, electrostatic particle-in-cell algorithm, Journal of Computational Physics, 230 (18), 7018–7036, ()2011)

[22] Kingham, RJ and Bell, AR, An implicit Vlasov–Fokker–Planck code to model non-local electron transport in 2-D with magnetic fields, Journal of Computational Physics, 194 (1), 1–34, (2004)

[23] Burrage, Kevin and Burrage, Pamela Marion, High strong order explicit Runge-Kutta methods for stochastic ordinary differential equations, Applied Numerical Mathematics, 22 (1), 81–102, (1996)