C. R. Mecanique 342 (2014) 662–670

Contents lists available at ScienceDirect

Comptes Rendus Mecanique

www.sciencedirect.com

Theoretical and numerical approaches for Vlasov–Maxwell equations

Coupled Particle-In-Cell and Direct Simulation Monte Carlo method for simulating reactive plasma flows

Claus-Dieter Munz [a,*], Monika Auweter-Kurtz [b], Stefanos Fasoulas [c], Asim Mirza [c], Philip Ortwein [a], Marcel Pfeiffer [c], Torsten Stindl [c]

[a] Institute of Aerodynamics and Gasdynamics (IAG), Universität Stuttgart, 70550 Stuttgart, Germany
[b] German Aerospace Academy (ASA), Forum 1 am Konrad-Zuse-Platz 1, 71034 Böblingen, Germany
[c] Institute of Space Systems (IRS), Universität Stuttgart, 70550 Stuttgart, Germany

ARTICLE INFO

Article history:
Received 22 March 2014
Accepted 28 June 2014
Available online 8 August 2014

Keywords:
Particle-In-Cell
Direct Simulation Monte Carlo
Discontinuous Galerkin
High-order
Plasma physics
Boltzmann equation

ABSTRACT

Plasma flows with high Knudsen numbers cannot be treated with classic continuum methods, as represented for example by the Navier–Stokes or the magnetohydrodynamic equations. Instead, the more fundamental Boltzmann equation has to be solved, which is done here approximately by particle based methods that also allow for thermal and chemical non-equilibrium. The Particle-In-Cell method is used to treat the collisionless Vlasov–Maxwell system, while neutral reactive flows are treated by the Direct Simulation Monte Carlo method. In this article, a combined approach is presented that allows the simulation of reactive, partially or fully ionized plasma flows. Both particle methods are briefly outlined and the coupling and parallelization strategies are described. As an example, the results of a streamer discharge simulation are presented and discussed in order to demonstrate the capabilities of the coupled method.

1. Introduction

Many plasma applications, including re-entry missions in high altitudes, electric space propulsion or streamer formation, are characterised by high Knudsen numbers and are dominated by strong deviations from thermal and/or chemical equilibrium conditions [1–4]. Therefore, it is physically not justified to apply classic continuum models like the Navier–Stokes or the magnetohydrodynamic equations for their description. Instead, the Boltzmann equation

$$\left(\frac{\partial}{\partial t}+\vec{v}\frac{\partial}{\partial \vec{x}}+\frac{\vec{F}}{m}\frac{\partial}{\partial \vec{v}}\right)f=\left(\frac{\partial f}{\partial t}\right)_{\text{coll}} \tag{1}$$

has to be solved, representing a more fundamental equation based on kinetic gas theory. Here, $f=f(\vec{v},\vec{x},t)$ is the particle distribution function at the six-dimensional phase space point $(\vec{x},\vec{v})$ at the time t. Furthermore, m is the particle mass, $\vec{F}$ an external force and $(\partial f/\partial t)_{\text{coll}}$ the so-called Boltzmann collision integral, which describes intermolecular collisions between the particles in the flow. A well-established way to find an approximate solution to the Boltzmann equation is the use of

* Corresponding author. Tel.: +49 711 685 63401; fax: +49 711 685 53402.
E-mail addresses: munz@iag.uni-stuttgart.de (C.-D. Munz), m.auweter-kurtz@german-asa.de (M. Auweter-Kurtz), fasoulas@irs.uni-stuttgart.de (S. Fasoulas), mirza@irs.uni-stuttgart.de (A. Mirza), ortwein@iag.uni-stuttgart.de (P. Ortwein), mpfeiffer@irs.uni-stuttgart.de (M. Pfeiffer), stindl@irs.uni-stuttgart.de (T. Stindl).

http://dx.doi.org/10.1016/j.crme.2014.07.005
1631-0721/

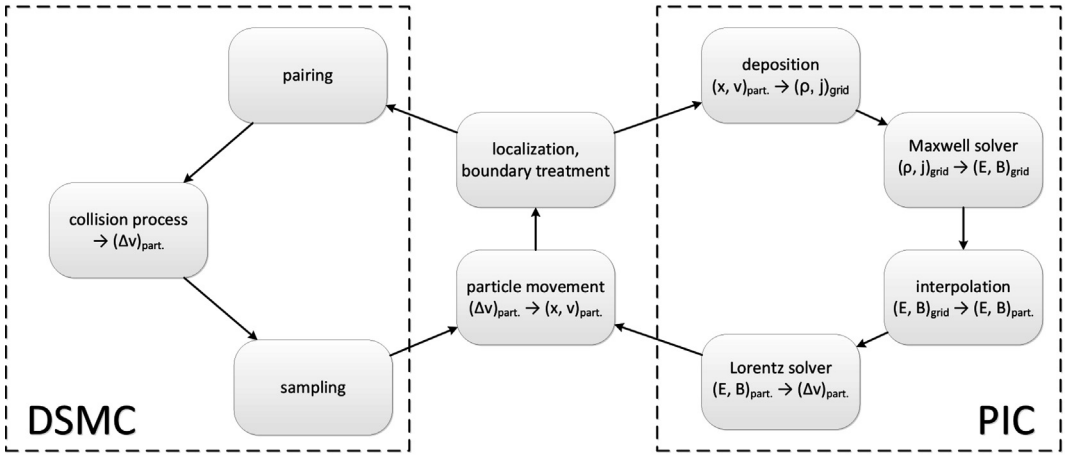

Fig. 1. Flow chart of PIC–DSMC simulation procedures performed during each time step.

numerical particle methods. Two of them, the Particle-In-Cell (PIC) and the Direct Simulation Monte Carlo (DSMC) method have continuously been advanced for decades and successfully applied to simulate a variety of plasma and gas applications and phenomena.

The basic idea of both methods is the approximation of the distribution function by a certain number of particles N at the position $\vec{x}_k$ and the velocity $\vec{v}_k$ at the corresponding time t, leading to

$$f(\vec{x},\vec{v},t)=\sum_{k=1}^{N} w_k \delta\big(\vec{x}-\vec{x}_k(t)\big)\delta\big(\vec{v}-\vec{v}_k(t)\big)$$

In addition, a particle weighting factor w_k is introduced, describing the number of real particles represented by one simulation particle k. This allows the handling of huge particle numbers with a lower number of simulation particles.

The PIC method describes the electromagnetic interaction between charged particles in a Vlasov plasma [5,6]. This means that the flow is collisionless, i.e. $(\partial f/\partial t)_{\text{coll}}=0$, and the only external force F considered in (1) is the Lorentz force, given by

$$\vec{F}_L = Q\,(\vec{E}+\vec{v}\times\vec{B}) \tag{2}$$

where Q, $\vec{E}$ and $\vec{B}$ are the particle charge, the electric and the magnetic field, respectively. The basic concept of the PIC method is the decoupling of the free-particle movement in the computational domain and the calculation of the electromagnetic fields on a fixed simulation grid. Therefore, the PIC method connects the Lagrangian particle handling and the electromagnetic fields that are handled in an Eulerian manner with different interpolation procedures that are described in Section 2. The first methods to calculate the electromagnetic fields were based on Yee's finite difference method [7] using structured grids. Later, finite volume (FV) [8] and finite element (FE) [9] methods were applied. Due to their excellent approximation of wave propagation and the flexible grid requirements, discontinuous Galerkin (DG) schemes have also been investigated in recent times. The successful application of the DG scheme as a field solver is demonstrated in [10], establishing a high-order explicit PIC method based on unstructured grids.

In contrast to the PIC method, the DSMC method includes the collision term $(\partial f/\partial t)_{\text{coll}}$, represented by binary collisions, and assumes that external forces are neglected, i.e. $\vec{F}=0$. As a consequence, the electromagnetic interaction of charged particles cannot be handled. The DSMC method, which was first introduced by Bird [1], performs collisions amongst freely moving particles and between particles and boundaries of the computational domain. Each collision includes an energy and momentum exchange and optionally a relaxation process of internal degrees of freedom as well as chemical reactions. The typical internal degrees of freedom handled in the DSMC method are the rotational and the vibrational energies of diatomic molecules, the latter being also introduced in a quantized model by Bergemann and Boyd [11]. In the last years, the inclusion of the electronic energy of atoms and molecules has become increasingly feasible [12,13]. The simulation of chemical reactions is mainly based on two models, the well-known macroscopic Arrhenius model [1,14] and a new microscopic quantum kinetic model by Bird [15]. The entire collision process, which includes relaxation processes and chemical reactions, is treated in a statistical manner using random numbers. Consequently, the results of the DSMC method, which have to be sampled from the microscopic particle value, contain a statistical noise.

In order to simulate collisional as well as electromagnetic interactions in a plasma flow, the coupled PIC–DSMC code "PICLas" has been developed in the last years and is described in this paper. The scheme uses unstructured, three-dimensional grids allowing one to be able to simulate complex geometries and flows. A schematic flow chart of the coupled code is depicted in Fig. 1. Both PIC and DSMC contribute to the particle movement, which represents the common module of the code together with localization and boundary treatment.

In the following, an overview of the PIC side of the code and its methods is presented in Section 2. Then, an overview of the DSMC side is given in Section 3. The coupling of both methods in the central modules is described in Section 4, together with an overview of the parallelization strategy. Selected simulation results of PICLas for the formation of a streamer discharge are presented in Section 5 in order to demonstrate the capabilities of the new code. Finally, a summary and a brief outlook are given in Section 6.

2. Particle-In-Cell method

A time step of the PIC cycle (as shown in Fig. 1) consists of several modules described in this section. The first step consists in the interpolation of charge and current densities, which are generated by the particles, onto the grid cells in the spatial simulation domain. Using these densities as sources for Maxwell's equations, the electric and magnetic fields are computed on a grid. From these fields, interpolated onto the particle positions, the Lorentz force at the particle position is computed and the particles are then moved and tracked in accordance with the relativistic equations of motion. Initially, the objective of the PIC code development was to get a tool to analyze the physics of high-power gyrotrons as used for heating the plasma flow in fusion reactors. This is a quite challenging problem with respect to the computational effort and storage. Hence, the efficient implementation on parallel high-performance computers was a central topic of this development. Simulations of the resonator, the mode converter, and even unique simulations of the fully coupled resonator and launcher of a 140-GHz gyrotron have already been published [16,17] and [18]. In the following, the main building blocks of this PIC solver are summarized.

2.1. Equation system

In order to be able to simulate electrodynamic phenomena, Maxwell's equations as given by

$$\frac{\partial \vec{E}}{\partial t} = c^2 \nabla \times \vec{B} - \frac{\vec{j}}{\epsilon_0} \tag{3}$$

$$\frac{\partial \vec{B}}{\partial t} = -\nabla \times \vec{E} \tag{4}$$

$$\nabla \cdot \vec{B} = 0 \tag{5}$$

$$\nabla \cdot \vec{E} = \frac{\rho}{\varepsilon_0} \tag{6}$$

have to be solved. Here, ρ and $\vec{j}$ are the charge and current density, respectively. The densities are the source terms of the electromagnetic fields $\vec{E}$ and $\vec{B}$ and are derived from the particles as described in Section 2.4. In addition, ε_0 is the vacuum permittivity. In principle, only the hyperbolic equations (3) and (4) are needed to describe the temporal propagation of the electromagnetic fields. For an initial solution that satisfies the divergence constraints (5) and (6), it is ensured that these equations are fulfilled at all points in time. Due to numerical inaccuracies by solving the evolution equations only and charge conservation errors in the particle treatment, divergence errors may be accumulated and may falsify the solution after a short time. Therefore, the divergence constraints are coupled to the hyperbolic system by introducing two scalar Lagrangian multipliers Ψ and Φ. The assumption that the errors are distributed with finite propagation rate results in a purely hyperbolic coupled Maxwell (PHM) system [19], which allows a robust and efficient simulation of the coupled problem by an extension of the hyperbolic solver to the coupled system. The PHM system reads as

$$\frac{\partial \vec{E}}{\partial t} = c^2 \nabla \times \vec{B} - \chi c^2 \nabla \Psi - \frac{\vec{j}}{\varepsilon_0} \tag{7}$$

$$\frac{\partial \vec{B}}{\partial t} = -\nabla \times \vec{E} - \chi \nabla \Phi \tag{8}$$

$$\frac{\partial \Phi}{\partial t} = -\chi c^2 \nabla \cdot \vec{B} \tag{9}$$

$$\frac{\partial \Psi}{\partial t} = -\chi \nabla \cdot \vec{E} + \chi \frac{\varrho}{\varepsilon_0} \tag{10}$$

Here, χc, a multiple of the speed of light c, is the velocity of the Lagrange multiplier propagation by which divergence errors are transported out of the domain [19].

2.2. The field solver and the calculation of the Lorentz force

The discontinuous Galerkin (DG) method has turned out in the last years to be a powerful tool for the solution of linear and nonlinear wave equations, see, e.g., [20]. High-order accuracy leads to low dispersion and diffusion in the approximation

of wave propagation over long distances or time. For the application of a DG scheme, the PHM equations (7)–(10) are written as a hyperbolic system in the form

$$\frac{\partial \vec{u}}{\partial t} + \sum_{d=1,3} \underline{\underline{\mathcal{K}}}_d \frac{\partial \vec{u}}{\partial \xi_d} = \vec{s}$$

Here, $\vec{u}$ is the state vector, $\vec{s}$ are the source terms, and $\underline{\underline{\mathcal{K}}}_i$ is the resulting flux of the PHM system. The DG scheme is based on the approximation of a weak formulation of this system. In contrast to a finite element method, the multiplication by a test function and the integration is done over every grid cell separately. The transformation to a reference element leads to the following variational formulation

$$\int_{\mathcal{E}} J \frac{\partial \vec{u}}{\partial t} \varphi \, \mathrm{d}\vec{\xi} + \sum_{\mathrm{d}=1,3} \int_{\mathcal{E}} \underline{\underline{\mathcal{K}}}_d \frac{\partial \vec{u}}{\partial \xi_\mathrm{d}} \varphi \, \mathrm{d}\xi_\mathrm{d} = \int_{\mathcal{E}} J \vec{s} \varphi \, \mathrm{d}\vec{\xi}$$

with reference element $\mathcal{E}$, the Jacobian J of the transformation and the test function φ. The approximate solution in the DG method consists then of piecewise polynomials, which may be discontinuous at grid cell interfaces. As in the finite volume approach, numerical fluxes couple then the neighboring grid cells. This establishes a good robustness and allows at the same time a high degree of flexibility, e.g. by choosing the order of accuracy locally. An explicit time approximation requires only to communicate surface data of direct neighbors grid cells between processes. We use hexahedral cells in combination with a tensor product basis and spectral collocation, which is usually called the Discontinuous Galerkin Spectral Element Method (DGSEM), see [21] and [22]. In the DG scheme, the electric and magnetic fields are given at every point by a polynomial, which can be evaluated at each particle position. Hence, an additional interpolation is not necessary.

2.3. Particle motion

After determination of the electric and magnetic field values at the particle positions, the particles are moved in phase space according to the relativistic equations of motion

$$\frac{\mathrm{d}\vec{x}}{\mathrm{d}t} = \vec{v}, \qquad \frac{\mathrm{d}\vec{p}}{\mathrm{d}t} = \vec{F} \tag{11}$$

with the relativistic momentum $\vec{p} = m_0 \gamma \vec{v}$, the particle mass m_0, and the Lorentz factor $\gamma = (1 - |\vec{v}|^2/c^2)^{-1/2}$. The only acting force $\vec{F}$ in this case is the Lorentz force $\vec{F}_\mathrm{L}$ on the particle charge Q generated by the electromagnetic fields $\vec{E}$ and $\vec{B}$ according to Eq. (2). In addition to the fields computed by the DG field solver, external fields can be defined either homogeneously in the computational domain or provided by data files.

2.4. Particle localization and particles-to-grid interpolation

The particles have to be localized in the grid after every movement step due to their free movement. The position is needed to determine the local value of the electromagnetic fields as well as to determine the distribution of the particle charges on the grid as source terms for the field solver. For newly placed particles (either at the beginning of a simulation or at every time step), a Cartesian background mesh is used to reduce the possibilities in the searching algorithm and to handle this efficiently. Once the particles have been localized, a tracking method is used to follow their flight path through the individual grid cells. During the tracking, particle boundary conditions are considered, whenever a particle crosses a grid cell boundary. Currently implemented particle boundary conditions include reflective (perfectly or diffusely), periodic and open boundaries with additional special cases for specific applications, such as mixed boundaries for electrodes.

Charges and currents of the moving particles are deposited onto the grid as sources of the Maxwell solver. Several deposition methods are implemented. The least accurate but fastest approach is the cell mean value method, where the sum of all charges within a grid cell is averaged and homogeneously deposited to all Gauss integration points within the cell. However, charge variations within a cell are not considered. In contrast, the nearest interpolation point method deposits the charge of each particle to the nearest interpolation point in the cell. While this allows for charge variations within a cell, it also leads to significant gradients in the source terms, especially for a particle that contains a high charge. This often artificially fragments the charge, leading to unphysical results, especially for homogeneous plasma flows. The fragmentation is significantly reduced by depositing the charge of a particle onto several grid cells by treating it as a (non-deformable) particle cloud of physical particles. Two methods employing this concept have been implemented. The polynomial shape function method deposits the charge to all integration points within a user-defined radius and weighs the deposited value according to the distance, as proposed in [10]. The background mesh method is similar, but instead of defining a sphere around the particle, in which the charge is deposited, it uses a Cartesian background mesh in conjunction with a B-splines weighting method to deposit the charges onto the background mesh and from there to the integration points. The method is faster but less accurate than the shape function method, which of course depends on the polynomial degree of the B-splines. A more detailed overview of the considered methods and their application to the simulation of a gyrotron can be found in [23].

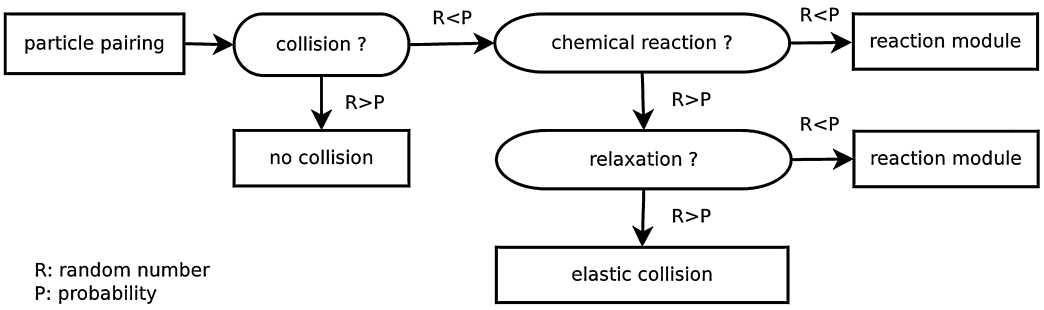

Fig. 2. Flow chart of DSMC collision process.

3. Direct Simulation Monte Carlo method

The collision integral in the Boltzmann equation (1) is neglected in the PIC method and thus collisions between particles are not performed. Therefore, the DSMC method is utilized to consider intermolecular collisions in order to extend the approximation of the Boltzmann equation. Fig. 1 depicts the procedures performed during each time step.

3.1. Particle movement, localization and boundary treatment

Within the DSMC method, the actual particle velocity is computed in each collision directly, otherwise it is kept constant. Therefore, the particles are moved freely within the computational domain according to:

$$\vec{x}_{n+1} = \vec{x}_n + \vec{v}_n \Delta t$$

The particle localization is treated equally to PIC as described in Section 2.4. Implemented particle boundary conditions for DSMC include reflective, periodic and open boundaries. Since reflective boundaries usually denote surfaces of the computational domain like walls, gas surface interactions have to be taken into account. Currently, the most commonly used model, the Maxwell model, is implemented. This gas surface interaction model represents incomplete surface accommodation as a combination of fully diffuse and fully specular reflection [1]. The transfer and exchange of energy and momentum at the wall can be computed on the basis of the particle velocity change due to the reflection.

3.2. Particle pairing

A fraction of particles in each cell is selected to participate in binary collisions during a simulation time step. Currently, an approach called natural sample size, introduced first by Baganoff and McDonald [24], is used for the computation of the collision probability. Here, the number of collision pairs is based on the total number of simulated particles in the current cell only. The simplest case leads to $N/2$ sample pairs out of N particles in a cell. The common approach for actual pairing is to randomly select two particles inside a cell. However, this approach leads to a strong restriction of the DSMC scheme: only particles in the range of the mean free path λ are allowed to collide. Therefore, the typical cell side has to be roughly λ. Recently, a gridless DSMC method has been proposed [25], which substitutes the random pairing by searching the nearest neighbour of each particle. However, a gridless method is not beneficial for a combined PIC–DSMC scheme as the PIC method relies on a computational DG grid. Hence, particle pairing on the basis of searching the nearest neighbor in each computational cell is used. Additionally, this pairing method is combined with an octree scheme in order to reduce the computational costs of searching for the nearest neighbour. Thereby, the restriction of limiting the cell size to λ is removed [26].

3.3. Collision process

Once the particle pairs are defined, the next step is to decide which, if any, collision occurs. In the DSMC method, collisions and possible reactions occur, depending on a random number R that has to be lower than the corresponding probability. Fig. 2 illustrates the implemented collision procedure.

The calculation of the collision probability is based on Bird's NTC scheme [1], modified by the natural sample size method introduced by Baganoff and McDonald [24]. Here, the collision probability of two particles a and b is given by

$$P_{ab} = \frac{N_a N_b}{1+\delta_{ab}} w_k \frac{\Delta t}{V_c S_{ab}} (\sigma_{ab} g_{ab}) \tag{12}$$

where N denotes the number of simulated particles of the species, V_c the cell volume, σ the collision cross section, g the relative velocity of the collision pair, and δ_{ab} the Kronecker delta. It should be noted that the particle weighting w_k is equal for all particles in DSMC. A detailed derivation of this collision probability as it is implemented in this method is given

in [27]. For the calculation of the collision cross section, the variable hard sphere model is implemented. This model was introduced by Bird [1], as a practical approach to the solution to engineering problems. Isotropic scattering is assumed, like the hard sphere model, but its total collision cross section is allowed to vary with the relative velocity g.

The collision model is based on binary collisions, where atomic and diatomic species as well as electrons are possible collision partners. During a collision, momentum and energy of both particles are exchanged with the possibility of a relaxation process. Relaxation means that energy is transferred between internal degrees of freedom (rotation, vibration and electronic excitation). The distribution of the quantum number is brought closer to a Boltzmann distribution due to the assumption that each collision leads towards an equilibrium state. Currently, the vibration and electronic energy is stored in a quantum form. The model of a harmonic oscillator is implemented for the vibration, where the degeneracy of each vibrational level has the constant value of 1 and energy levels are spaced equally. For the electronic energy, however, models allowing to describe the energy of each quantum level are not available. Hence, a database of these levels for each species is required. The characteristic rotational temperature T_{rot} of a molecule is low, typically in the range of $T_{\mathrm{rot,O_2}} = 2.069$ K, $T_{\mathrm{rot,N_2}} = 2.879$ K or $T_{\mathrm{rot,O_2}} = 2.44$ K. Therefore, the level distance of the quantized rotational energy is small for the temperatures in typical plasma flows, making it feasible to use a continuous model of the rotational energy.

Besides relaxation processes, collisions can lead to chemical reactions. Two approaches to model the chemical reactions are implemented in PICLas. The first model is the standard DSMC chemistry model, based on the Arrhenius equation [28]. The reaction probability depends on the Arrhenius coefficients, which are typically gained by measurements under continuum conditions. Thus, the Arrhenius equation needs a macroscopic temperature value, which is not defined in a collision of two particles. Therefore, the microscopic collision energy has to be converted into a macroscopic temperature by making equilibrium assumptions. Bird proposed a Quantum–Kinetic (Q–K) model [15], which depends on statistical mechanics and offers a phenomenological approach. As a result, the Q–K model depends only on the collision energy and fundamental species data.

3.4. Sampling

Macroscopic values like density, bulk velocity or temperature are of interest in most engineering problems rather than particle information like internal energies or microscopic particle motion. The calculation of theses macroscopic values out of microscopic particle information depends on averaging particle values in a computational cell. More specifically, density ρ_{cell}, bulk velocity $\vec{u}_{\mathrm{cell}}$ and translational temperature $T_{\mathrm{tra,cell}}$ are determined by

$$\rho_{\mathrm{cell}} = \frac{w_k m}{V_{\mathrm{cell}}} \langle N \rangle_{\mathrm{cell}}$$
$$\vec{u}_{\mathrm{cell}} = \langle \vec{v} \rangle_{\mathrm{cell}}$$
$$T_{\mathrm{tra,cell}} = \frac{m}{3k_B} \left(\langle \vec{v}^2 \rangle_{\mathrm{cell}} - \langle \vec{v} \rangle_{\mathrm{cell}}^2 \right)$$

where k_{B} is the Boltzmann constant and the operator $\langle\ \rangle_{\mathrm{cell}}$ denotes an unweighted average over all particles in the cell. Therefore, the sampling of particle information is the last step in each DSMC iteration. Here, the sample space can be increased in order to reduce the statistical noise for steady-state problems by sampling over many iteration steps. On the other hand, the sample space for unsteady state problems is restricted to the number of particles in a cell.

4. Combining the PIC and DSMC solver

4.1. Temporal connection

The connection strategy of the combined PIC and DSMC solver is outlined in Fig. 1. The main problem of the coupling is the different outcome of the methods. On the one hand, the result of the PIC solver is the Lorentz force F_{L} acting on the particles during a time step. Therefore, Eqs. (11) have to be integrated over the time step to evaluate the new particle positions $\vec{x}$ and velocities $\vec{v}$. Furthermore, the used time step size in the time integration has to fulfill the Courant Friedrichs Lewy (CFL) condition for the speed of light to guarantee the stability of the Maxwell solver. A time-integration method that leads to a stable Maxwell solver with a $CFL \approx 1$ and a sufficient accuracy to resolve oscillating electromagnetic waves is the used fourth-order low-storage explicit Runge–Kutta (LSERK) scheme [29]. Here, the Maxwell solver as well as the integration of Newton's equations of particle motion are performed in each RK stage. Therefore, the source terms of the Maxwell solver (i.e. the particles) are treated with the same accuracy as the Maxwell solver itself.

On the other hand, the DSMC solver strictly separates the collision process and the particle movement. This results in a change of the velocity Δv of each particle at a discrete point in time instead of a continuous change of the velocity over a time step due to an acting force. Consequently, the most basic explicit method of time integration, the first-order Euler method, is sufficient for the DSMC method. Additionally, the required time step of the DSMC method to resolve the collision frequency is typically much higher than the time step of the Maxwell solver. Thus, using the DSMC solver in each RK stage does not lead to an enhanced accuracy of the time integration and the collision probability (see Eq. (12)) is very small due to the huge disparity in the required DSMC and PIC time steps. Since this means a significantly increased computational effort

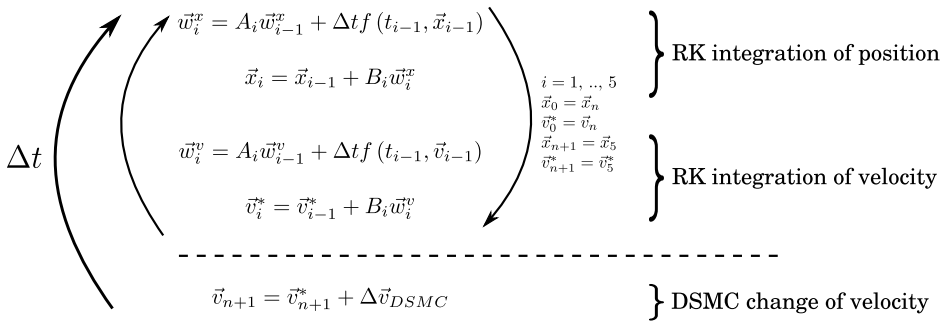

Fig. 3. Coupled RK integration for the PIC method and the velocity change of the DSMC method. The $w_i^{v,x}$ are necessary sub steps of the LSERK integration scheme found in [29]. The A_i, B_i as well as the function $f(t_i, \vec{u}_i)$ are LSERK specific factors and functions. The sub time steps are calculated by $t_i = t_n + C_i \Delta t$ with another LSERK factor C_i. Starting with the states $\vec{x}_n$ and $\vec{v}_n$ at time t_n, the RK integration is performed five times, resulting in the final position $\vec{x}_{n+1}$ and the interim result of the velocity $\vec{v}_{n+1}^*$. In the last step, the velocity change due to the DSMC method is added.

without a considerable gain in accuracy, the DSMC method is performed only once per time step. The resulting changed particle velocity then acts as a new starting condition for the next RK cycle. This strategy is schematically depicted in Fig. 3.

4.2. Parallelization

Parallelization of the coupled code is done via a domain decomposition. The computational domain is split using a space-filling curve so that each process has to compute a distinct part of the whole domain and therefore only needs to have access to the local data of grid cells and particles. This leads to a decrease in required memory for each process and makes large-scale computations with huge numbers of particles and grid cells feasible on high-performance clusters with node based shared memory. Communication between the processes is done via Message Passing Interface (MPI). The data to be communicated include particles passing from one process onto a different one (possibly crossing one or more additional processes) and the values necessary to compute the electromagnetic fields across the process boundaries. Since the particle number and distribution is dynamic and, in most cases, inhomogeneous, a load balancing has to be performed to ensure that the combined computation time of particle processing, including DSMC, and field computation is roughly equal for each process to reach maximum parallelization efficiency. The load balancing is implemented by summing up the number of particles and the number of grid cells times the degree of freedom of each cell, which depends on the order of the DG polynomials. The number of particles is multiplied by a load factor based on the relative computational time for one particle versus that of one degree of freedom. The overall sum of both computational loads is then spread over all processes, ensuring that processes with a high number of particles are assigned a domain with less grid cells and vice versa. The load factor has been derived by a parameter study and needs to be adjusted depending on the methods used for the different modules of the PIC and DSMC schemes. A more detailed description and investigation of the applied parallelization scheme can be found in [30].

5. Simulation results

The formation of a streamer has been chosen as an example to illustrate the capabilities of the coupled code. The so-called streamer is a plasma channel that is formed in a neutral gas, initiated by an externally applied electric field that accelerates free electrons. The accelerated electrons are able to ionize the neutral gas and therefore an electron avalanche is unleashed. A technological example of a streamer formation is the ignition of a spark plug. In the simulation, 100 electrons and 100 argon ions with a particle weight of $w_k = 1$ and a temperature of $T_{e,i} = 1300$ K are initially placed in a space of 0.02 mm $\times$ 0.02 mm $\times$ 0.02 mm at $z = 0.03$ mm. The size of the computational domain is 0.2 mm $\times$ 0.2 mm $\times$ 0.3 mm with $90 \times 90 \times 135$ grid cells. An external field of $E_z = 10^7$ V/m is applied. The neutral atomic gas is argon with a pressure of $p = 1.3 \cdot 10^5$ Pa and a temperature of $T = 1300$ K. It is simulated as a background gas due to the high pressure, which would otherwise result in a huge number of simulated particles. Consequently, only the electrons and ions are actual particles within the simulation, but they can collide with particles that are sampled from the Boltzmann distribution of the neutral background gas in each time step. Finally, the DSMC algorithms described in Section 3 are used for the collisions, but the neutral particles that are newly initialized for each collision are in a state of thermal equilibrium at all times.

Fig. 4 shows the simulation result of the particle induced electric field in the z-direction at different times. It depicts two different effects. First, the magnitude of the electric field is increasing with time because of the ionization process and the resulting increase in charged particles. Second, in consequence of the charge separation induced by the external electric field and the different inertia of electrons and ions, an electric field similar to an electric dipole forms. This is especially visible in the isolines of the electric field E_z. The plot also shows the asymmetric behavior of the electric field in z-direction.

The particles of the simulation are depicted in Fig. 5. The separation of the charged particles is easily seen at the time $t = 35$ ps. The red ions with the greater inertia form a cloud at the bottom, whereas the electrons are accelerated towards

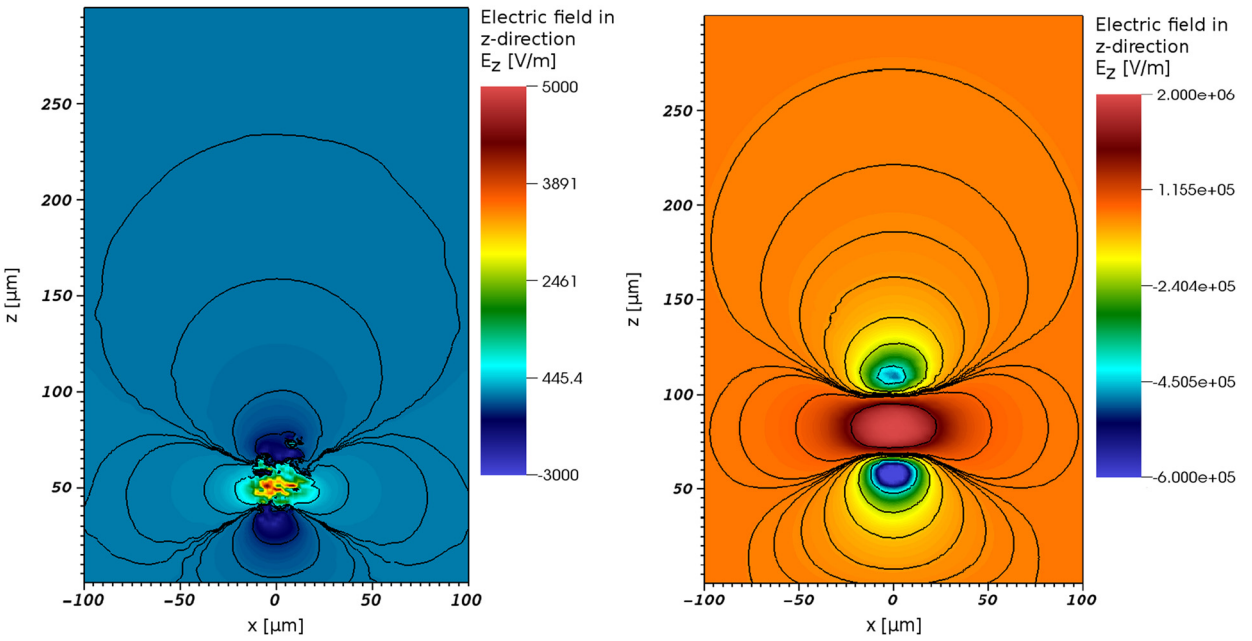

Fig. 4. (Color online.) Particle induced electric field in the z direction E_z with isolines. Left: $t = 10$ ps. Right: $t = 35$ ps.

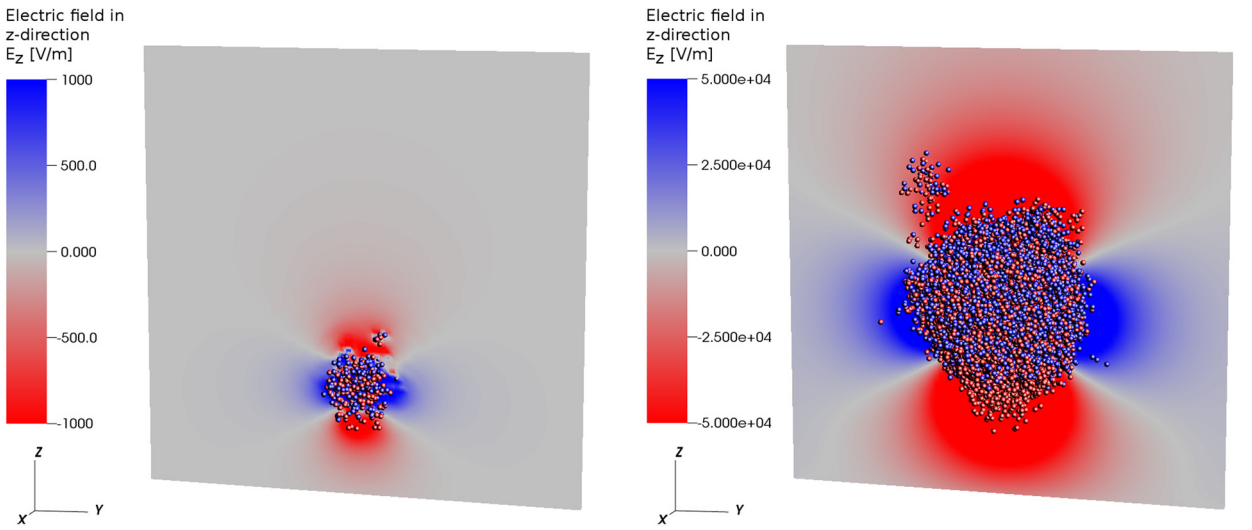

Fig. 5. Electric field in z-direction E_z with the charged particles. The red and blue spheres are the ions and the electrons, respectively. Furthermore, only every 5th particle is shown. Left: $t = 10$ ps. Right: $t = 35$ ps. (For interpretation of references to color in this figure, the reader is referred to the online version of this article.)

the top. Another interesting point is the beginning of a sub-streamer formation in the upper left side at $t = 35$ ps, which was also observed in real ignition processes of spark plugs.

6. Summary

A particle based simulation strategy for rarefied reactive plasma flows in thermal and chemical non-equilibrium is introduced. For this, the combination of a Particle-In-Cell (PIC) and a Direct Simulation Monte Carlo (DSMC) method is used. The PIC solver simulates electromagnetic interactions of charged particles and electromagnetic wave propagation and uses a Discontinuous Galerkin Spectral Element Method (DG-SEM). The DSMC solver treats collisions, including charged and neutral particles by modeling momentum and energy exchange, relaxation processes of internal degrees of freedom and chemical reactions.

A coupling strategy for the PIC and DSMC solver is proposed. Here, it has to be considered that the electromagnetic fields act as a continuous force on the particles, while a collision leads to an instantaneous momentum change. Therefore a high order Runge–Kutta scheme is used for the time integration and particle movement. It is coupled with a first-order Euler method for DSMC, leading to increased accuracy with minimal computational costs. To benefit from modern high performance computers, a parallelization using a domain decomposition is implemented.

The example of a streamer formation demonstrates the combined PIC–DSMC solver with both electromagnetic forces and chemical reactions (ionization) being relevant to the forming process. The simulation results show the increasing electric field due to the generated ions and electrons as well as the emerging charge separation. In addition, the forming of a sub-streamer is reproduced.

Future planned improvements of the coupling include the implementation of more efficient high-order Runge–Kutta schemes to allow for larger time steps as well as relativistic collisions for the DSMC method to enable consistent coupled relativistic simulations. Future applications of the combined scheme include the simulation of electric spacecraft propulsion systems such as the mini magnetospheric plasma propulsion concept [31].

Acknowledgements

We gratefully acknowledge the Deutsche Forschungsgemeinschaft (DFG) for funding within the project “Kinetic Algorithms for the Maxwell–Boltzmann System and the Simulation of Magnetospheric Propulsion Systems”, grant number MU1319/21. T. Stindl thanks the Landesgraduiertenförderung Baden-Württemberg and the Erich-Becker-Stiftung, Germany, for his personal scholarsphips. Computational resources have been provided by the Bundes-Höchstleistungsrechenzentrum Stuttgart (HLRS).

References

[1] G. Bird, Molecular Gas Dynamics and the Direct Simulation of Gas Flows, Oxford University Press, Oxford, 1994.
[2] R. Jahn, Physics of Electric Propulsion, McGraw-Hill, New York, 1968.
[3] O. Chanrion, T. Neubert, A PIC–MCC code for the simulation of streamer propagation in air, J. Comput. Phys. 126 (2008) 434–448.
[4] S. Pancheshnyi, P. Ségur, J. Capeillère, A. Bourdon, Numerical simulation of filamentary discharges with parallel adaptive mesh refinement, J. Comput. Phys. 227 (2008) 6574–6590.
[5] R.W. Hockney, J.W. Eastwood, Computer Simulation Using Particles, McGraw-Hill, Inc., New York, 1988.
[6] C.K. Birdsall, A.B. Langdon, Plasma Physics via Computer Simulation, Hilger, Bristol, 1991.
[7] T. Westermann, Numerical modelling of the stationary Maxwell–Lorentz system in technical devices, Int. J. Numer. Model. 7 (1994) 43–67.
[8] F. Hermeline, S. Layouni, P. Omnes, A finite volume method for the approximation of Maxwell’s equations in two space dimensions on arbitrary meshes, J. Comput. Phys. 227 (2008) 9365–9388.
[9] F. Assous, P. Degond, E. Heintze, P. Raviart, J. Segre, On a finite-element method for solving the three-dimensional Maxwell equations, J. Comput. Phys. 109 (1993) 222–237.
[10] G.B. Jacobs, J. Hesthaven, High-order nodal discontinuous Galerkin particle-in-cell method on unstructured grids, J. Comput. Phys. 214 (2006) 96–121.
[11] F. Bergemann, I. Boyd, New discrete vibrational energy model for the direct simulation Monte Carlo method, in: Progress in Astronautics and Aeronautics, vol. 158, 1994.
[12] D. Liechty, M. Lewis, Treatment of electronic energy level transition and ionization following the particle-based chemistry model, in: AIAA Paper, 8th Aerospace Sciences Meeting, 2010, 2010, pp. 1–12.
[13] G. Bird, The DSMC method, CreateSpace Independent Publishing Platform, 2013.
[14] B.L. Haas, I.D. Boyd, Models for direct Monte Carlo simulation of coupled vibration–dissociation, Phys. Fluids A, Fluid Dyn. 5 (1993).
[15] G.A. Bird, The Q–K model for gas-phase chemical reaction rates, Phys. Fluids 23 (2011) 106101.
[16] A. Stock, J. Neudorfer, M. Riedlinger, G. Pirrung, G. Gassner, R. Schneider, S. Roller, C.-D. Munz, Three-dimensional numerical simulation of a 30 GHz gyrotron resonator with an explicit high-order discontinuous Galerkin based parallel particle-on-cell method, IEEE Trans. Plasma Sci. 40 (2012) 1860–1870.
[17] J. Neudorfer, A. Stock, J. Flamm, F. Hindenlang, G. Gassner, C.-D. Munz, R. Schneider, S. Roller, Numerical investigation of high-order gyrotron mode propagation in launchers at 170 GHz, IEEE Trans. Plasma Sci. 40 (2012) 1512–1521.
[18] A. Stock, A high-order particle-in-cell method for low density plasma flow and the simulation of gyrotron resonator devices, Ph.D. thesis, University of Stuttgart, Germany, 2012.
[19] C.-D. Munz, P. Omnes, R. Schneider, E. Sonnendrücker, U. Voß, Divergence correction techniques for Maxwell solvers based on a hyperbolic model, J. Comput. Phys. 161 (2000) 484–511.
[20] J. Hesthaven, T. Warburton, Nodal Discontinuous Galerkin Methods: Algorithms, Analysis and Applications, Springer-Verlag, New York, 2008.
[21] D.A. Kopriva, Implementing Spectral Methods for Partial Differential Equations, Springer, 2009.
[22] F. Hindenlang, G.J. Gassner, C. Altmann, A. Beck, M. Staudenmaier, C.-D. Munz, Explicit discontinuous Galerkin methods for unsteady problems, Comput. Fluids 61 (2012) 69–93.
[23] T. Stindl, J. Neudorfer, A. Stock, M. Auweter-Kurtz, C.-D. Munz, S. Roller, R. Schneider, Comparison of coupling techniques in a high-order discontinuous Galerkin-based particle-in-cell solver, J. Phys. D, Appl. Phys. 44 (2011) 194004.
[24] D. Baganoff, J.D. McDonald, A collision selection rule for a particle simulation method suited to vector computers, Phys. Fluids A, Fluid Dyn. 2 (1990) 1248–1259.
[25] S.E. Olson, A.J. Christlieb, Gridless DSMC, J. Comput. Phys. 227 (2008) 8035–8064.
[26] M. Pfeiffer, A. Mirza, S. Fasoulas, A grid-independent particle pairing strategy for DSMC, J. Comput. Phys. 246 (2013) 28–36.
[27] M. Laux, Direkte Simulation verduennter, reagierender Stroemungen, Ph.D. thesis, Universitaet Stuttgart, 1995.
[28] I.D. Boyd, Modeling backward chemical rate processes in the direct simulation Monte Carlo method, Phys. Fluids 19 (2007) 126103.
[29] M.H. Carpenter, C.A. Kennedy, Fourth-order 2N-storage Kutta schemes Runge-schemes, in: NASA Technical Memorandum, 109112, 1994, pp. 1–26.
[30] J. Neudorfer, A. Stock, R. Schneider, S. Roller, C.-D. Munz, Efficient parallelization of a three-dimensional high-order particle-in-cell method for the simulation of a 170 GHz gyrotron resonator, IEEE Trans. Plasma Sci. 41 (2013) 87–98.
[31] M. Pfeiffer, D. Petkow, G. Herdrich, S. Fasoulas, Assessment of a numerical approach suitable for the M2P2 problem, Open Plasma Phys. J. 4 (2011) 24–33.