Adaptive multiscale methods for 3D streamer discharges in air

Robert Marskar

SINTEF Energy Research, Sem Sælands vei 11, 7034 Trondheim, Norway.

E-mail: `robert.marskar@sintef.no`

Abstract. We discuss spatially and temporally adaptive implicit-explicit (IMEX) methods for parallel simulations of three-dimensional fluid streamer discharges in atmospheric air. We examine strategies for advancing the fluid equations and elliptic transport equations (e.g. Poisson) with different time steps, synchronizing them on a global physical time scale which is taken to be proportional to the dielectric relaxation time. The use of a longer time step for the electric field leads to numerical errors that can be diagnosed, and we quantify the conditions where this simplification is valid. Likewise, using a three-term Helmholtz model for radiative transport, the same error diagnostics show that the radiative transport equations do not need to be resolved on time scales finer than the dielectric relaxation time. Elliptic equations are bottlenecks for most streamer simulation codes, and the results presented here potentially provide computational savings. Finally, a computational example of 3D branching streamers in a needle-plane geometry that uses up to 700 million grid cells is presented.

1. Introduction

Streamer discharges are fast filamentary transients that evolve due to self-enhanced electric fields at their tips. Streamers are the natural precursors of sparks, lightning, and sprites, and have found use in sterilization of polluted gases and breakup of molecules, plasma assisted combustion, and control of airflow in the boundary layer of airplane wings [1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11]. The spatial scales for streamer discharges span several orders of magnitude. Firstly, a streamer is essentially a non-thermal plasma filament surrounded by a space charge layer with a thickness one to two orders of magnitude thinner than the filament diameter. Secondly, the length of the filament can be much longer than its thickness, leading to a numerical problem with widely different spatial scales. On the one hand, a fine numerical resolution is required for resolving the space charge layer. On the other, a large computational domain is required in order to facilitate the propagation of the streamer. Due to the nonlinearity of streamers, numerical solutions can usually not be obtained on coarse grids.

Streamers can be described by using either fluid or kinetic approaches, or a combination of them [12, 13]. In a fluid approach, electrons and ions are evolved according to their fluid moments (usually truncated to first order) by using tabulated or analytic transport data. Kinetic approaches approximate the phase space distribution function by evolving computational particles, using cross-sectional collision data as input parameters. The kinetic approach is computationally far more exhaustive, and includes far more physics. Fluid approximations are more common on larger scales, and this is the description that we use in this paper.

It is generally believed that spatially adaptive methods are ideally suited - and maybe even necessary - for large scale simulations of 3D streamers. The reason for this is that a streamer represents a dynamically evolving structure with possible stochastic behavior, restricting the use static grids to cases where the streamer path is a priori known, or small-scale cases where the entire domain can be resolved at the finest spatial resolution. Because of these difficulties, 3D simulations of streamers are rare [14, 15, 16, 17, 18, 19, 19, 20, 21]. In addition to multiple spatial scales, various characteristic time scales also exist for streamer discharges. For example, the avalanche-to-streamer time describes the time it takes for seed electrons to reach a critical size through impact ionization, whereas the dielectric relaxation time describes the local rate-of-change of the electric field due to motion of charge carriers. The relaxation time can be expressed in terms of the electric field $\mathbf{E}$ and the electric current density $\mathbf{J}$ as

$$\Delta t_{\mathbf{E}} = \frac{\epsilon_0 \, |\mathbf{E}|}{|\mathbf{J}|}, \tag{1}$$

which follows from a first-order trunction of Ampere's law. This time scale is an important one for streamer simulations. Note that $\Delta t_{\mathbf{E}}$ is a *physical* time scale that does not reference the grid resolution.

Computer codes have restrictions on which time step sizes can be used to evolve the fluid equations. For example, explicit methods for advection obey a Courant-Friedrichs-

Lewy (CFL) time step restriction $\Delta t_{\text{cfl}} = \Delta x/|\mathbf{v}|$ where Δx is the spatial resolution and $\mathbf{v}$ is the advective velocity. For parabolic equations, such as diffusion equations, the time step restriction on explicit methods is $\Delta t_D \sim \Delta x^2$, which quickly becomes even stricter than Δt_{cfl} when diffusion is relevant. Furthermore, although not required for stability, one may derived a fourth time step constraint $\Delta t_S = \frac{S}{n}$, where S is a source term and n is a species density, which describes a characteristic time scale for growth due to ionization. For multiple components, the above time step sizes also minimized over species according to the worst offender, which is usually the electrons. On the other hand, kinetic models like Particle-In-Cell (PIC) choose their time steps differently, being restricted by the inverse plasma frequency and the inverse collision frequencies.

For streamer simulations, the above mentioned time scales are widely varying. Fully implicit methods are likely to be the most robust, but it is not yet clear how these should be implemented in the multiphysics environment of streamers, particularly if high-order shock-capturing methods on adaptive grids are involved for the convection part. For that reason, implicit-explicit (IMEX) methods are more attractive, allowing one to treat restrictive time scales with implicit methods and use explicit methods on others. Previous experience with large scale 3D simulations [21] show that such methods are often CFL bound on the time step, leading us to speculate that run-time performance can be improved through a better segregation between physical and numerical time scales. The rationale behind this idea is that it should be unnecessary to perform electric field updates when $\Delta t_{\text{cfl}} \ll \Delta t_{\mathbf{E}}$, since the correction to the $\mathbf{E}$-field would then be miniscule. Similar ideas have been adopted in particle codes [22, 23]. Likewise, diffusive models approximations of the radiative transport equation reduce to Helmholtz equations, and it is neither not clear how often these need to be updated.

This paper quantifies the prospects of advancing fluid and elliptic equations using different time steps. Potentially, such techniques may lead to speedup (and flexibility) for fluid streamer simulation by elimination of extranous elliptic solves at very fine time scales, which are present for consistent schemes. The outline of this paper is as follows: In Sec. 2 we present a time stepping scheme that segregates the numerical and physical time scales. The numerical error of this scheme is then quantified in Sec. 3 by means of two-dimensional simulation experiments. Thus, we partially answer the question "How often do we need to update the electric field?", which is of interest to both the kinetic and fluid modeling parts of the plasma community. Finally, we present a high-performance computing example in Sec. 4 that uses some of these techniques, and provide some concluding remarks in Sec. 5.

2. Theory

We use a simplified fluid model for gas discharges, based on the following equations

$$\partial_t n_i = \nabla \cdot (D_i \nabla n_i - \mathbf{v}_i n_i) + S_i, \tag{2}$$

$$\nabla^2 \phi = -\frac{\rho}{\epsilon_0}, \tag{3}$$

$$\kappa_j \Psi_j - \nabla \cdot \left(\frac{1}{3\kappa_j} \nabla \Psi_j \right) = \frac{\eta_j}{c}, \tag{4}$$

where n_i is the density of species i, D_i the diffusion coefficient, $\mathbf{v}_i$ the velocity, and S_i the source term. The electric potential is given by ϕ and $\rho = \sum_i q_i n_i$ is the space charge density. The symbols Ψ_j denote the isotropic radiative density of a photon group j, κ_j is the Beer's length and η_j is an isotropic source term.

Equations (2)-(4) are solved with finite volumes over an adaptive mesh based on Cartesian grid patches, and solid boundaries are treated with an embedded boundary formalism. The spatial discretization of Eqs. (2) allows for arbitrarily cut cells (even ones that contain multiple cell fragments), and is as follows:

(i) The convective term $\nabla \cdot (\mathbf{v}_i n_i)$ is discretized with the unsplit Godunov's method. The state at a face center is given by the solution to a Riemann problem with slope-limited left and right states; the Riemann solution is the upwind state at the face. One-sided slopes are used if there are not enough cells available for the left or right slopes, which can occur if a cell face lies completely inside a material. On cut-cells, we require the flux on face centroids in addition to face centers. This is done by first computing fluxes at face centers, and then interpolating these to the respective face centroids. We additionally stabilize the convective derivative by computing a hybrid divergence with charge redistribution that allows us to use a standard CFL condition for cut cells with arbitrarily small volume fractions. Charge injection into the domain is a part of the advective discretization, and thus the injected charge is also redistributed in the neighborhood of the cut cells.

(ii) The diffusion advancement of $\partial_t n_i = \nabla \cdot (D_i \nabla n_i)$ is handled implicitly with the Twizell-Gumel-Arigu (TGA) scheme [24] (see equation (5)). This scheme is very stable in embedded boundary applications. The spatial discretization of the elliptic term $\nabla \cdot (D_i \nabla n_i)$ is done with a second-order accurate cell-centered discretization with Neumann boundary conditions everywhere. The resulting Helmholtz equations that arise from the TGA discretization are solved with a geometric multigrid method with red-black Gauss-Seidel smoothers and a biconjugate gradient stabilized method as the bottom solver.

(iii) The remaining elliptic equations for the Poisson and the radiative transfer equations are discretized with second order accurate cell-centered solvers. The spatial discretization is the same as for the diffusion equation above, with the exception of boundary conditions.

The computer code that we use is compatible with adaptive mesh refinement and runs at high concurrencies (tested for up to 8k cores so far). A full discussion of the underlying software is not possible in the scope of this paper, but can be found elsewhere [21]. We will use this code to explore the numerical error obtained by using different-sized time steps for fluid and elliptic equations, thereby probing the importance of updating the Poisson and radiative transfer equations. We will then apply these findings to a large scale simulation example.

2.1. Regular time stepping

First, we present a consistent second order time stepper that is suitable for embedded boundary applications, which we will refer to as the consistent scheme. This integrator is based on a second order monotone-in-time Runge-Kutta method together with second order implicit diffusion and advances $t^k \rightarrow t^{k+1}$ as follows:

(i) Compute $n^* = n^k + \Delta t \left[S^k - \nabla \cdot \left(\mathbf{v}^k n^k \right) \right]$.

(ii) Compute $\mathbf{E}^*$ by solving the Poisson equation with the space charge density $\rho^* = \rho(n^*)$.

(iii) Compute radiative transfer source terms $\eta^* = \eta \left(n^*, \mathbf{E}^* \right)$ and obtain Ψ_γ^* by solving the RTE equations.

(iv) Advance $n^\dagger = \frac{1}{2} \left(n^k + n^* + \Delta t \left[S^* - \nabla \cdot (\mathbf{v}^* n^*) \right] \right)$ where $S^* = S \left(n^*, \Psi^*, \mathbf{E}^* \right)$ and $\mathbf{v}^* = \mathbf{v} \left(\mathbf{E}^* \right)$

(v) Compute $\mathbf{E}^\dagger$ by solving the Poisson equation with $\rho^\dagger = \rho \left(n^\dagger \right)$.

(vi) Compute diffusion coefficients $D = D \left(\mathbf{E}^\dagger \right)$ and obtain n^{k+1} with an implicit diffusion advance

$$\left(I - \mu_1 L \right) \left(I - \mu_2 L \right) n^{k+1} = \left(I + \mu_3 L \right) n^\dagger, \qquad (5)$$

where L is the diffusion operator. Expressions for μ_1, μ_2, and μ_3 are found in [21]. For non-diffusive species then $n^\dagger \rightarrow n^{k+1}$.

(vii) Obtain the final electric field $\mathbf{E}^{k+1}$ by solving the Poisson equation with $\rho^{k+1} = \rho \left(n^{k+1} \right)$.

(viii) Compute radiative transfer source terms $\eta^{k+1} = \eta \left(n^{k+1}, \mathbf{E}^{k+1} \right)$ and obtain Ψ_γ^{k+1} by solving the RTE equations

In the above, steps (i) through (iv) describe a consistent strongly stability preserving Runge-Kutta method of order two; steps (vi) and (vii) describe an implicit diffusion advance, and step (viii) describe the final radiative transfer update. For the physical model that we will consider, this scheme performs 11 elliptic updates per time step: Three Poisson updates, six radiative transfer equation solves, and two Helmholtz solves for the diffusion step. The computational bottlenecks are due to these elliptic updates.

2.2. Probing elliptic equations - a modified time stepper

The modified scheme segregates the evolution into different-sized time steps. The outline of this method is as follows: We assume the existence of two disparate time scales Δt_f, which is a "fine" numerical time scale, and Δt_c which is a "coarse" numerical time scale. We assume $\Delta t_f < \Delta t_c$ and $\Delta t_c = m \Delta t_f$ where $m > 0$ is an integer such that the time scales synchronize. Typically, Δt_f is is smaller than the CFL time Δt_{cfl}, whereas Δt_c is a coarser numerical time scale, typically taken to be proportional to the dielectric relaxation time. The various time steps are summarized in Table 1. We then evolve the equations of motions in the following way:

(i) Evolve the advective-reactive part of the species equations from t^k to $t^k + \Delta t_c$ by using m non-diffusive steps with individual time step sizes $\Delta t_f < \Delta t_{\text{cfl}}$, corresponding to steps (i)-(iv) in Section 2.1. The advective discretization over a time step Δt_f then occurs as follows:

$$n^* = n^k + \Delta t_f \left[S^k - \nabla \cdot \left(\mathbf{v}^k n^k \right) \right], \tag{6}$$

$$S^* = S \left(n^*, \Psi^k, \mathbf{E}^k \right), \tag{7}$$

$$\mathbf{v}^* = \mathbf{v} \left(\mathbf{E}^k \right), \tag{8}$$

$$n^\dagger = \frac{1}{2} \left(n^k + n^* + \Delta t_f \left[S^* - \nabla \cdot \left(\mathbf{v}^* n^* \right) \right] \right). \tag{9}$$

This process is repeated m times such that the equations are advanced a time Δt_c.

(ii) Perform a diffusion advance $(I - \mu_1 L)(I - \mu_2 L) n^{k+1} = (I + \mu_3 L) n^\dagger$. For non diffusive species, $n^\dagger \rightarrow n^{k+1}$.

(iii) Solve the Poisson equation with $\rho^{k+1} = \rho \left(n^{k+1} \right)$ and obtain the new electric field $\mathbf{E}^{k+1}$.

(iv) Compute radiative transfer source terms $\eta^{k+1} = \eta \left(n^{k+1}, \mathbf{E}^{k+1} \right)$ and obtain Ψ_γ^{k+1} by solving the RTE equations

We will examine three cases in total so that we can estimate which elliptic equations are important for numerical accuracy, and which ones are not:

Case 1: Update all elliptic equations at the coarse time step Δt_c as above.

Case 2: Also update $\mathbf{E}$ consistently. In this case, we additionally update the electric field at each Runge-Kutta stage of (6).

Case 3: Additionally update the RTE equations consistently. In this case, we additionally update the radiative transfer equations at each Runge-Kutta stage of (6).

Thus, the electric field and radiative transport equations are updated at the either the coarse or the fine numerical time step, allowing us to assess the importance of both types of elliptic updates. Note that, at worst, the above scheme is first order accurate in Δt_c due the modified electric field coupling.

Symbol	Formula	Description
Δt_c	-	Coarse numerical time step
Δt_f	-	Fine numerical time step
$\Delta t_{\mathbf{E}}$	$\epsilon_0 \left| \mathbf{E} \right| / \mathbf{J}$	Dielectric relaxation time
Δt_{cfl}	$\Delta x / \left| \mathbf{v} \right|$	CFL time step
Δt_S	S/n	Ionization time step

Table 1. Time step size nomenclature. The time steps Δt_{cfl} and Δt_S are minimized according to the worst offender (usually electrons).

The rationale for the above approach is removal of potentially redundant elliptic solves at every Δt_f, which would be present for a consistent scheme. Indeed, one should

not need to update the Poisson equation if $\Delta t_f \ll \Delta t_{\mathbf{E}}$, but rather resolve its physical time scale $\Delta t_{\mathbf{E}}$ at reasonable accuracy. It is quite clear that an electric field update is redundant when space charge currents are negligible, which is e.g. the case in the avalanche stage of a streamer discharge. Elliptic solves are bottlenecks for most streamer simulation codes, and unnecessary updates can easily become performance killers. This is particularly true when direct solvers are involved, but also for iterative solvers when curved solid boundaries are present since this leads to deteriorated convergence rates for multigrid smoothing. One may possible extend these ideas to kinetic models, although the accuracy of this simplification must then be determined by a kinetic code, which is not a topic in this paper.

The use of longer time steps for the Poisson equation is not new. In the PIC code in [23], the authors update the electric field using a time step that may be as long as $\Delta t_{\mathbf{E}}$, i.e. $\Delta t_c \leq \Delta t_{\mathbf{E}}$. Likewise, the authors in [22] use a fixed time step for the Poisson update and a much finer time step for the particle push. The accuracy of this approximation was not quantified in either paper. Relatedly, the authors of the fluid code in [19] and the authors of the PIC code in [25] use a much stricter condition, updating the Poisson equation consistently at each advective step or particle push. For the conditions in [25], the time steps are on the order of 1 fs whereas the dielectric relaxation time is estimated to be on the order of 100 fs. On the other hand, for most fluid simulations reported to date, the electric field updates are consistent, i.e. they are performed at each advective or diffusive step, regardless of the dielectric relaxation time.

3. Two-dimensional simulation experiments

In this section we estimate how often the $\mathbf{E}$-field should be updated by means of two-dimensional numerical experiments. We consider a Cartesian $(2\,\mathrm{cm})^2$ domain with a rod-plane geometry. The rod is a cylinder with a hemispherical cap with a radius of $500\,\mu\mathrm{m}$ protruding 1 cm from the center of the top domain edge. This edge, and the needle, is live with a voltage of 40 kV whereas the bottom edge is grounded. For simplicity, we use homogeneous Neumann boundary conditions on the left and right side edges.

For chemistry, we consider a three-species model for air at 1 bar that consists of electrons n_{e}, positive ions n_+, and negative ions n_-. The kinetic coefficients for this scheme are given in [26]. All three species are advected, but only electrons are diffusive. Note that for positive streamer discharges that do not touch the cathode, one might possibly simplify the system further by assuming immobile ions, although this is not possible for negative streamer due to the dynamics of the cathode sheath. For radiative transport, we consider the model by [27, 28]. The initial conditions are $n_{\mathrm{e}} = n_+ = 10^{10}\,\mathrm{m}^3$ and the equations are integrated for 2 ns, which is sufficiently long for capturing the three basic simulations phases: electron avalanche, streamer inception, and streamer propagation phase.

The numerical error of the modified scheme is evaluated by comparison with the

consistent scheme at each step. Let n^{k+1} denote the final state after advancing the species equation from t_k to $t_k + \Delta t_c$ by using the modified scheme. Let $\widetilde{n}^{k+1}$ denote the final state after advancing the species equation from t_k to $t_k + \Delta t_c$ by using the consistent scheme. The relative error obtained for n^{k+1} at each grid point $\mathbf{i}$ is then

$$\triangle n_{\mathbf{i}}^{k+1} = n_{\mathbf{i}}^{k+1} - \widetilde{n}_{\mathbf{i}}^{k+1}. \tag{10}$$

We compute the L_2 error norm as

$$L_2 \left(\triangle n^{k+1} \right) = \sqrt{\frac{\sum_{\mathbf{i}} \kappa_{\mathbf{i}} \left(\triangle n_{\mathbf{i}}^{k+1} \right)^2}{\sum_{\mathbf{i}} \kappa_{\mathbf{i}} \left(\widetilde{n}_{\mathbf{i}}^{k+1} \right)^2}}, \tag{11}$$

where $\kappa_{\mathbf{i}}$ is the volume fraction of a cut-cell (see [21] for details). We remark that $\triangle n^k$ is the step-wise error caused by inconsistent elliptic updates and not the accumulated error.

We use a fine spatial grid for all the simulations presented in this section. The grids use a base mesh of $(128)^2$ cells and includes four refinement levels; the refinement factor between each level is 4 except for the last one where it is 2. This yields an effective domain of $(16384)^2$, corresponding to a finest-level resolution of $1.2\,\mu$m.

Figure 1 shows the time-evolution of a simulation that updates the elliptic equations at Δt_c and restricts the time step according to

$$\Delta t_c = \min \left(\Delta t_{\mathbf{E}}, \Delta t_S \right), \tag{12}$$

where the fine step restriction is $\Delta t_f \le 0.5\Delta t_{\text{cfl}}$. The equations are integrated for $2\,$ns and evolve without obvious numerical errors such as spurious oscillations or numerical instabilities. This is true even though the number of advective integration steps per elliptic update ranged up to thirty, with an average of twenty advective steps per elliptic update throughout the simulation.

Next, to answer the question of when we need to update the electric field and radiative transport equations, we perform additional simulations by using smaller values of Δt_c for each of the three cases listed in the preceding section. Specifically, we take $\Delta t_c = \alpha \min \left(\Delta t_{\mathbf{E}}, \Delta t_{S_e} \right)$, where $\alpha = 0.1, 0.25, 0.5, 1.0$, and examine the numerical error for each case. The results of these comparisons are shown in figure 2 for the electrons n_e. To aid in the interpretation of this error diagram, we mention that the error rise around $t = 0.3\,$ns coincides with the inception of the streamer; whereas the initial propagation phase takes place from $0.5\,$ns and beyond. The various figures show the errors for the three different cases above. Figure 2a) shows the error norms when all elliptic equations are updated at Δt_c, corresponding to Case 1. All simulations are stable, although the numerical error can be significant in certain parts of the streamer evolution. For the propagation phase of the streamer, the error decreases to about 10^{-3}, which is (in our opinion) a more acceptable error. For the avalanche phase, which occurs for $t < 0.3\,$ns in figure 2, the obtained errors are due to absence of consistently updated photoionization. However, the computed errors are smaller than 2×10^{-4}. Figure 2b) shows Case 2 where we consistently update the radiative transport equations. Differences between figure 2a)
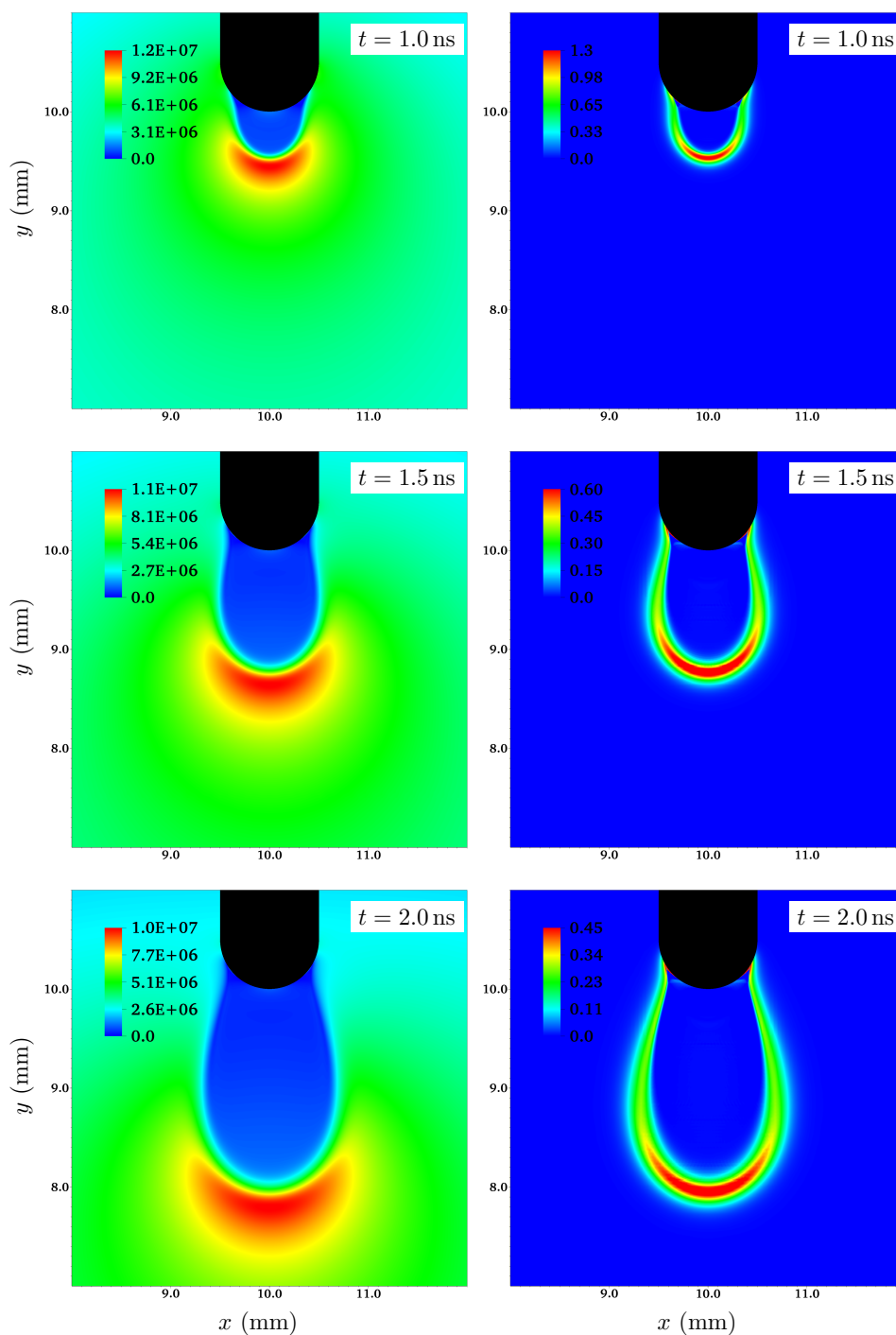
Figure 1. Two-dimensional streamer evolution with $\Delta t_c = \min\left(\Delta t_{\mathbf{E}}, \Delta t_S\right)$ with radiative transport updates at Δt_c. The left hand side column shows the electric field magnitude $|\mathbf{E}|$ and the right hand side column shows the space charge density ρ. All numbers are in SI units and times are indicated in each frame.

and figure 2b) are only seen in the avalanche phase $t < 0.3$ ns. In this evolution regime we have $\Delta t_{\text{cfl}} \ll \Delta t_{\mathbf{E}}$ so that electric field updates are redundant; solution errors occur due to consistent radiative transport updates, but even then the errors are quite small. For
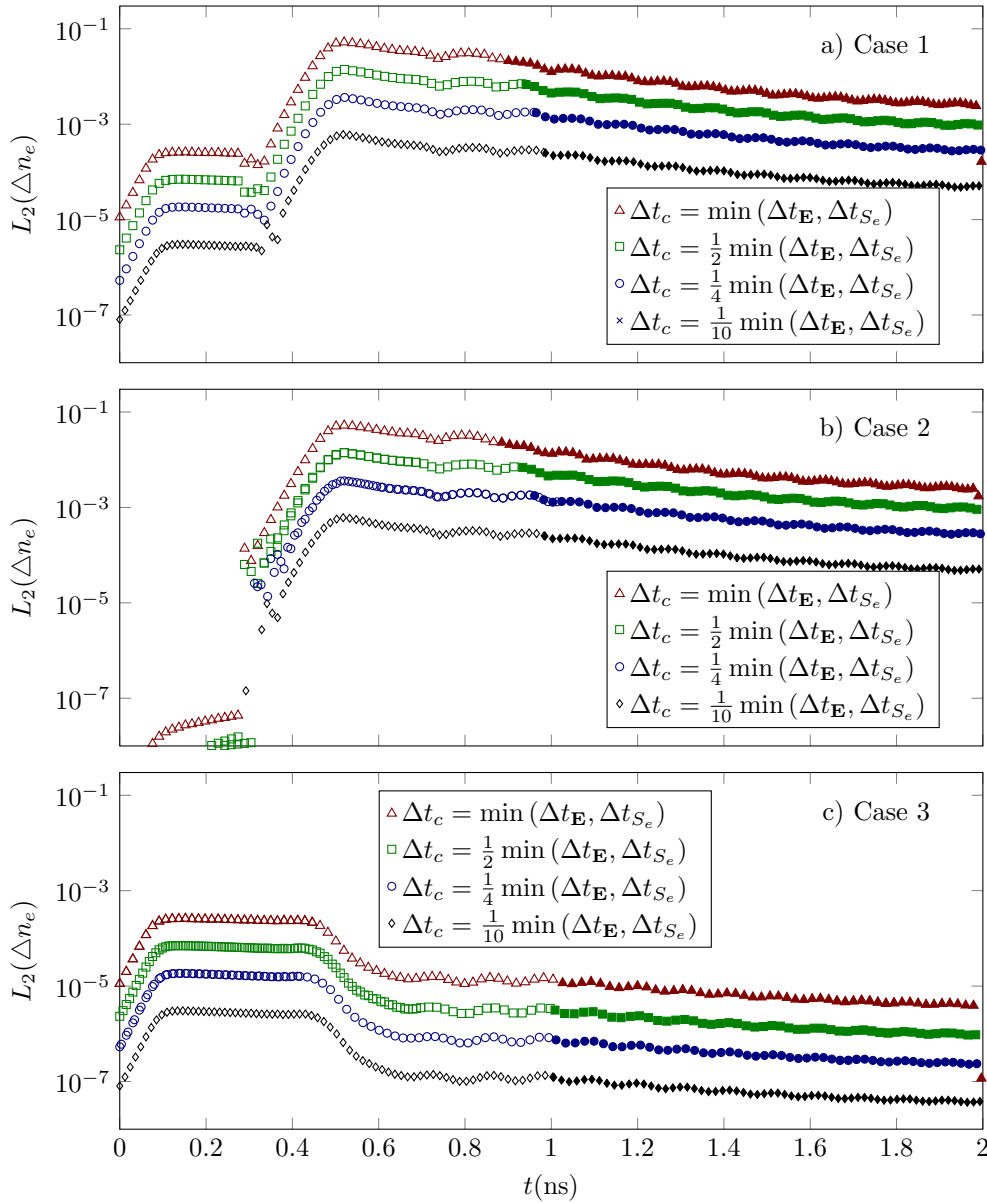
Figure 2. Error estimators for two-dimensional streamer simulation experiments. The various symbols show different time stepping criteria: If a symbol is empty (filled), the time step was restricted by Δt_{S_e} ($\Delta t_{\mathbf{E}}$). The solid line shows the numerical error for a fully consistent scheme. a) Poisson and radiative transfer equations are updated at Δt_c. b) Radiative transport updates at Δt_f. c) Poisson updates at Δt_f.

the streamer inception and propagation phases $t > 0.3\,\text{ns}$, the error increases for both figure 2a) and b), but there is no notable quantitative difference between the computed errors. This error must therefore be due to inconsistent electric field updates. Figure 2c) correponds to Case 3 above, where radiative transport is updated inconsistently and the electric field is updated consistently. The initial errors for the avalanche phase $t < 0.3\,\text{ns}$ correpond to the errors in figure 2a) and are due to inconsistent photoionization updates. For Case 3 the error is small throughout the entire evolution, showing that inconsistent

photoionization updates leads to minor accuracy losses.
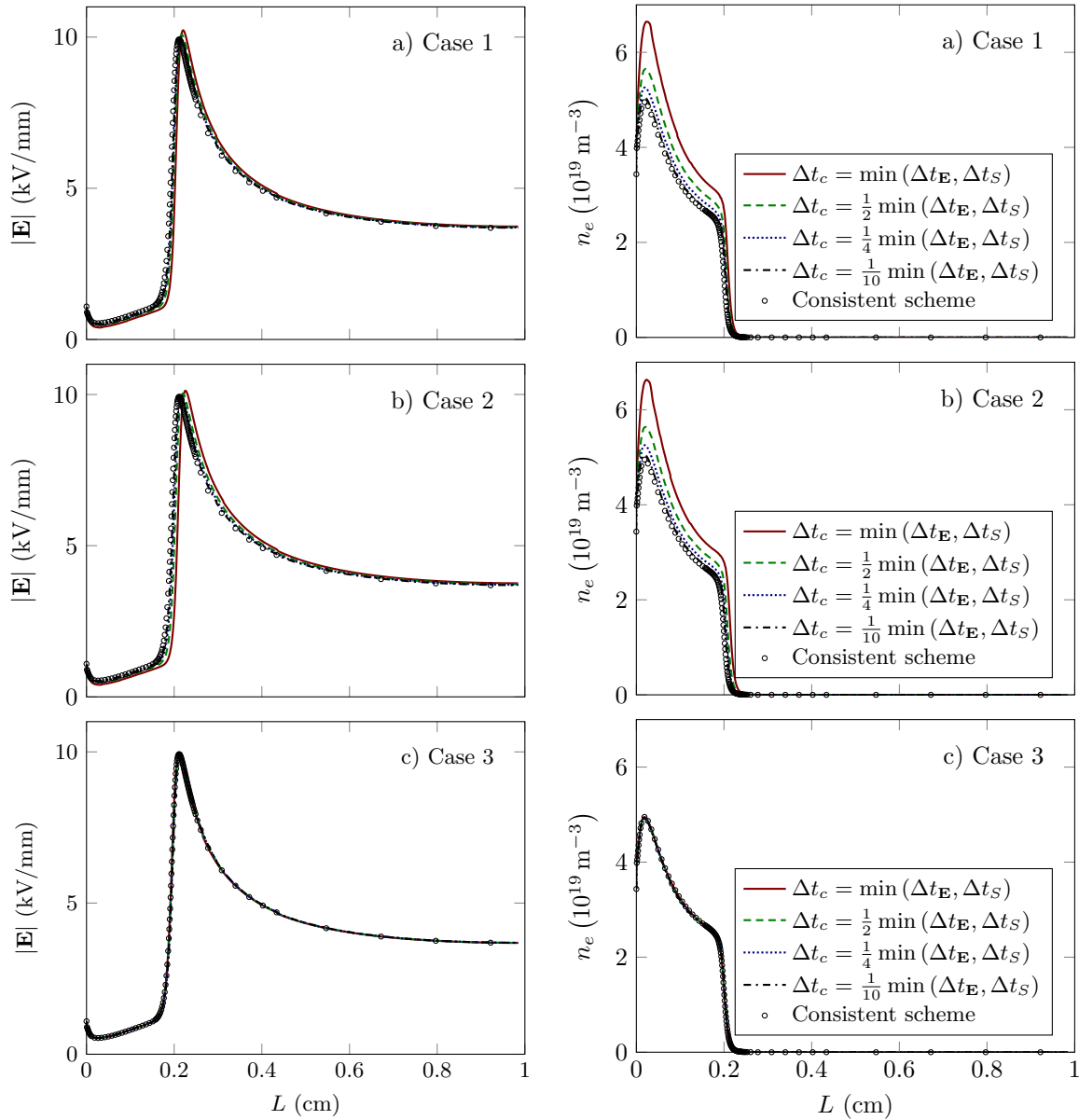
Figure 3. Plots of the electric field magnitude and electron densities on the symmetry axis after integrating the equations for 2 ns - corresponding to figure 2. The various rows show: a) Poisson and radiative transfer equations are updated at Δt_c. b) Radiative transport updates at Δt_f. c) Poisson updates at Δt_f.

To show the effect of the accumulated errors on the streamer for each of the nine simulations above, figure 3 shows the electric field and electron profiles on the symmetry axis at the end of the simulations. As before, Figure 3a) shows Case 1, figure 3b) shows Case 2, and figure 3c) shows Case 3. We only find minor differences with respect to the initial velocities and peak field amplitude of the streamers. For the electron densities, figure 3 shows that omission of electric field updates may lead to significant variation of the predicted electron densities. For the worst resolved case in figure 3a) which updates

the electric field at the relaxation time, the predicted peak electron density is about 25% higher than it should be. Likewise, figure 3c) shows again that consistent updates of the radiative transport model do not lead to improved model accuracy.

| Δt_c	Final $L_2(n_e)$	Poisson updates (modified scheme)	Poisson updates (original scheme)
$\min\left(\Delta t_{\mathbf{E}}, \Delta t_S\right)$	2.43×10^{-1}	133	8745
$\frac{1}{2}\min\left(\Delta t_{\mathbf{E}}, \Delta t_S\right)$	1.10×10^{-1}	237	8862
$\frac{1}{4}\min\left(\Delta t_{\mathbf{E}}, \Delta t_S\right)$	4.09×10^{-2}	450	9216
$\frac{1}{10}\min\left(\Delta t_{\mathbf{E}}, \Delta t_S\right)$	2.72×10^{-2}	1095	9966

Table 2. Total number of Poisson updates for the inconsistent (Case 1) and consistent schemes.

In summary, the preceding discussion shows that for the radiative transfer model considered here one does not need to compute the radiative transfer equations with a fine temporal resolution. Updates of the electric field should be evaluated based on a tradeoff between simulation time and accuracy need. For the model considered here, updating the electric field at $\Delta t_{\mathbf{E}}/10$ yields acceptable accuracy (a maximum step error of 10^{-3}). We have also computed the final L_2 error norm for the electron density by computing error norms between the inconsistent (Case 1) solution and the consistent solution. These accumulated errors are reported in table 2 where we also show the reduction of the number of elliptic solves for the inconsistent scheme. When we update the electric field at $\Delta t_{\mathbf{E}}/10$, the number of Poisson updates is reduced by approximately a factor nine, and the number of radiative transfer updates are reduced by a factor of eighteen. The price to pay for this is a 2.7% increased error in the L_2 norm.

4. A high performance computing example

As a large-scale computational example that uses some of these techniques, we next consider the inception and propagation of streamers in a needle-plane geometry. The domain is a $(2\,\text{cm})^3$ cube with a needle electrode with a $500\,\mu\text{m}$ radius protruding $1\,\text{cm}$ from the center of the top domain face (see figure 4). This face, and the needle, is live with a voltage of $15\,\text{kV}$ whereas the bottom face is grounded. Homogeneous Neumann boundary conditions are used for the Poisson equation on the remaining domain faces. To demonstrate the use of spatially adaptive methods to streamers, we will consider initial conditions that provoke streamer branching and prevent prediction of their paths. Streamers are initiated by considering a stochastic preionization level with peak amplitudes up to $10^{14}\,\text{m}^{-3}$ where the initial density is prescribed by using a landscape function borrowed from computer graphics [29] which guarantees that the resulting noise is C^1 smooth (we use the original hash table in [29]). This function is then exponentiated in order to generate randomly placed plasma spots. The correlation length between these spots is roughly $500\,\mu\text{m}$. The initial electron distribution is shown

in figure 4.
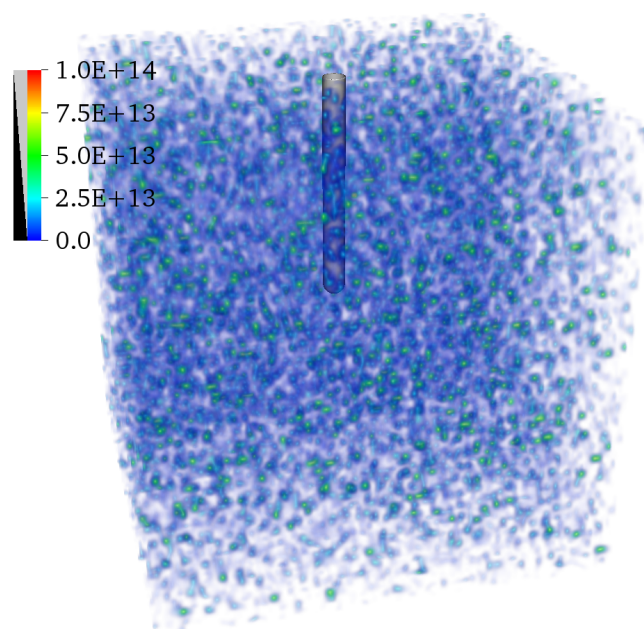
Figure 4. Initial preionization (in SI units) for the example simulation.

The simulation was run with a maximum time step $\Delta t_c = \mathrm{Min}\,(\Delta t_{\mathbf{E}}, \Delta t_S)$ until $t = 15\,\mathrm{ns}$ with a time-to-solution of approximately 36 hours. The electric field was updated at every advective evaluation whereas radiative transfer was updated at Δt_c only (corresponding to Case 3 above). The simulation was run on 128 computing nodes interconnected with Infiniband. Each node contains dual-socket Intel Broadwell (E5-2683v4) chips with 32 cores per node in total. MPI ranks were mapped to cores with a one-to-one ratio for a total concurrency of 4096. For grids, we use a patch-based AMR grid generated by the Berger-Rigoutsos algorithm[30] with a blocking factor of 16 and a maximum patch size of 32. This algorithm takes as input a number of "tags", which specify which cells should be refined, and the output of this algorithm is a properly nested hierarchy of Cartesian grids. The domain is discretized using this algorithm on a base mesh of $(256)^3$ cells with five levels of mesh refinement, which yields an effective domain of $(8192)^3$, corresponding to an effective resolution of $2.44\,\mu\mathrm{m}$. This resolution is sufficient for numerical stability, but we remark that we have not performed grid convergence studies for this example. The refinement criteria that we use is based on resolving streamer heads and space charge layers [21]. The time step restriction on Δt_f is done with $\Delta t_f \leq 0.5\Delta t_{\mathrm{cfl}}$. The number of Δt_f steps per coarse step Δt_c was between 1 and 20, with an average of 2 throughout the simulation. For an even finer spatial resolution, this ratio would be even higher, whereas this ratio would be 1:1 for a fully consistent scheme, which would increase the simulation time by approximately 40%.

Figure 5 shows the final simulation state after 15 ns. The various subplots in the figure show snapshots of the electron density, space charge density, electron source term,
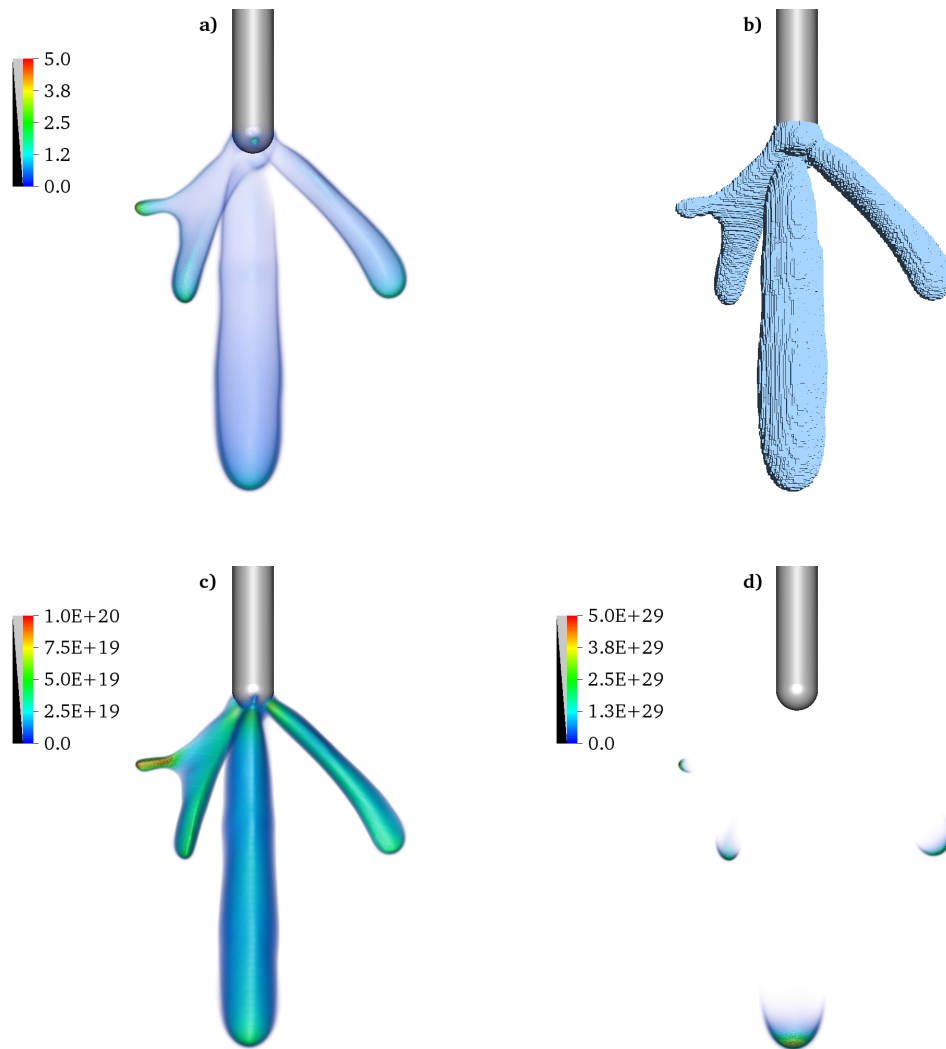
Figure 5. Final simulation state after 15 ns. a) Volume rendered space charge density. The data range has been adjusted to enhance visibility. b) Patch distribution at the finest level. Each brick represents a computational unit of minimum size $(16)^3$ and maximum size $(32)^3$. The "inside" of the brick agglomeration is empty (not shown). c) Volume rendered electron density. d) Volume rendered electron source term. All quantities are presented in SI units.

and the mesh distribution on the finest level. Due to the perturbed initial conditions, we find that multiple streamers start from the electrode. During inception, we do not find uniform field screening over the needle tip (not shown) although this is something that we observe in simulations that do not use perturbed initial conditions. For this simulation, we observe five initial streamers, but two of these start from a position slightly higher up on the electrode and stop after a few nanoseconds. The other three filaments propagate into the gap. Their cross sections are generally not circular; for one of the branches the ratio between the major and minor radii is roughly 2 so that the cross section is comparatively flat, which we believe is due to electrostatic repulsion from the other two nearby branches. Furthermore, this filament branches after approximately

10 ns, as seen in figure 5. Towards the end of the simulation, four individual streamer heads that propagate in different directions can clearly be observed, see e.g. figure 5d). The largest streamer propagates almost parallel with the rod, whereas the smallest streamer (see the branch in Fig. 5) propagates almost perpendicular with it.

5. Conclusion

We have discussed recent advances in multiresolution computer models for streamer discharges. The use of longer time steps for the electric field and radiative transport leads to inherent numerical errors, which we quantify by means of two-dimensional simulations. We show that it is permissible to use a longer time step for the elliptic equations (e.g. Poisson) than for advection and chemistry, and believe that such techniques can be used to speed up simulation cases. The ideas can possibly also be extended to kinetic simulations. A high-performance computing example that uses some of these techniques was then presented.

Acknowledgements

[1] J. P. Boeuf and L. C. Pitchford. Electrohydrodynamic force and aerodynamic flow acceleration in surface dielectric barrier discharge. *Journal of Applied Physics*, 97(10):103307, may 2005.

[2] Eric Moreau. Airflow control by non-thermal plasma actuators. *Journal of Physics D: Applied Physics*, 40(3):605–636, feb 2007.

[3] V R Soloviev and V M Krivtsov. Surface barrier discharge modelling for aerodynamic applications. *Journal of Physics D: Applied Physics*, 42(12):125208, jun 2009.

[4] A Abou-Ghazala, S Katsuki, K H Schoenbach, F C Dobbs, and K R Moreira. Bacterial decontamination of water by means of pulsed-corona discharges. *IEEE Transactions on Plasma Science*, 30(4):1449–1453, aug 2002.

[5] J S Clements, A Mizuno, W C Finney, and R H Davis. Combined removal of SO2, NOx, and fly ash from simulated flue gas using pulsed streamer corona. *IEEE Transactions on Industry Applications*, 25(1):62–69, jan 1989.

[6] J S Clements, M Sato, and R H Davis. Preliminary Investigation of Prebreakdown Phenomena and Chemical Reactions Using a Pulsed High-Voltage Discharge in Water. *IEEE Transactions on Industry Applications*, IA-23(2):224–235, mar 1987.

[7] G Dinelli, L Civitano, and M Rea. Industrial experiments on pulse corona simultaneous removal of NO x and SO2 from flue gas. *IEEE Transactions on Industry Applications*, 26(3):535–541, may 1990.

[8] S.A. Nair, K. Yan, A.J.M. Pemen, G.J.J. Winands, F.M. van Gompel, H.E.M. van Leuken, E.J.M. van Heesch, K.J. Ptasinski, and A.A.H. Drinkenburg. A high-temperature pulsed corona plasma system for fuel gas cleaning. *Journal of Electrostatics*, 61(2):117–127, jun 2004.

[9] David R. Grymonpré, Amit K. Sharma, Wright C. Finney, and Bruce R. Locke. The role of Fenton's reaction in aqueous phase pulsed streamer corona reactors. *Chemical Engineering Journal*, 82(1-3):189–207, mar 2001.

[10] U Ebert, C Montijn, T M P Briels, W Hundsdorfer, B Meulenbroek, A Rocco, and E M van Veldhuizen. The multiscale nature of streamers. *Plasma Sources Science and Technology*, 15(2):S118–S129, may 2006.

[11] Annemie Bogaerts, Erik Neyts, Renaat Gijbels, and Joost van der Mullen. Gas discharge plasmas and their applications. *Spectrochimica Acta Part B: Atomic Spectroscopy*, 57(4):609–658, apr 2002.

[12] Chao Li, Ute Ebert, and Willem Hundsdorfer. Spatially hybrid computations for streamer discharges with generic features of pulled fronts: I. Planar fronts. *Journal of Computational Physics*, 229(1):200–220, jan 2010.

[13] Chao Li and Willem Hundsdorfer. Spatially hybrid computations for streamer discharges : II. Fully 3D simulations. *Journal of Computational Physics*, 231(3):1020–1050, feb 2012.

[14] A Hallac, G E Georghiou, and A C Metaxas. Secondary emission effects on streamer branching in transient non-uniform short-gap discharges. *Journal of Physics D: Applied Physics*, 36(20):2498–2509, oct 2003.

[15] S. Pancheshnyi, P. Ségur, J. Capeillère, and A. Bourdon. Numerical simulation of filamentary discharges with parallel adaptive mesh refinement. *Journal of Computational Physics*, 227(13):6574–6590, jun 2008.

[16] A. Luque, U. Ebert, and W. Hundsdorfer. Interaction of Streamer Discharges in Air and Other Oxygen-Nitrogen Mixtures. *Physical Review Letters*, 101(7):075005, aug 2008.

[17] L Papageorgiou, A C Metaxas, and G E Georghiou. Three-dimensional numerical modelling of gas discharges at atmospheric pressure incorporating photoionization phenomena. *Journal of Physics D: Applied Physics*, 44(4):045203, feb 2011.

[18] S. Nijdam, J. Teunissen, E. Takahashi, and U. Ebert. The role of free electrons in the guiding of positive streamers. *Plasma Sources Science and Technology*, 25(4), 2016.

[19] Jannis Teunissen and Ute Ebert. Simulating streamer discharges in 3D with the parallel adaptive Afivo framework. *Journal of Physics D: Applied Physics*, 50(47):474001, nov 2017.

[20] J-M Plewa, O Eichwald, O Ducasse, P Dessante, C Jacobs, N Renon, and M Yousfi. 3D streamers simulation in a pin to plane configuration using massively parallel computing. *Journal of Physics D: Applied Physics*, 51(9):095206, mar 2018.

[21] Robert Marskar. An adaptive Cartesian embedded boundary method for two- and three-dimensional simulations of low-temperature plasma filaments in complex geometries. *Submitted to Journal of Computational Physics*, 2018.

[22] Andrew Fierro, James Dickens, and Andreas Neuber. Graphics processing unit accelerated three-dimensional model for the simulation of pulsed low-temperature plasmas. *Physics of Plasmas*, 2014.

[23] Jannis Teunissen and Ute Ebert. 3D PIC-MCC simulations of discharge inception around a sharp anode in nitrogen/oxygen mixtures. *Plasma Sources Science and Technology*, 2016.

[24] E. H. Twizell, A. B. Gumel, and M. A. Arigu. Second-order,L 0-stable methods for the heat equation with time-dependent boundary conditions. *Advances in Computational Mathematics*, 6(1):333–352, 1996.

[25] Andrew Fierro, Chris Moore, Ben Yee, and Matthew Hopkins. Three-dimensional kinetic modeling of streamer propagation in a nitrogen/helium gas mixture. *Plasma Sources Science and Technology*, 27(10):105008, oct 2018.

[26] R Morrow and J J Lowke. Streamer propagation in air. *Journal of Physics D: Applied Physics*, 30(4):614–627, feb 1997.

[27] P Ségur, A Bourdon, E Marode, D Bessieres, and J H Paillol. The use of an improved Eddington approximation to facilitate the calculation of photoionization in streamer discharges. *Plasma Sources Science and Technology*, 15(4):648–660, nov 2006.

[28] A Bourdon, V P Pasko, N Y Liu, S Célestin, P Ségur, and E Marode. Efficient models for photoionization produced by non-thermal gas discharges in air based on radiative transfer and the Helmholtz equations. *Plasma Sources Science and Technology*, 16(3):656–678, aug 2007.

[29] Ken Perlin. Improving noise. In *Proceedings of the 29th annual conference on Computer graphics and interactive techniques - SIGGRAPH '02*, page 681, 2002.

[30] Marsha Berger and Isidore Rigoutsos. An Algorithm for Point Clustering and Grid Generation. *IEEE Transactions on Systems, Man and Cybernetics*, 1991.