

PAPER • OPEN ACCESS

Collective dynamics of a dense streamer front

To cite this article: M B Teixeira-Gomes *et al* 2023 *Plasma Sources Sci. Technol.* **32** 095010

View the [article online](#) for updates and enhancements.

You may also like

- [Streamer properties and associated x-rays in perturbed air](#)
C Köhn, O Chanrion, L P Babich et al.
- [Surface streamer propagations on an alumina bead: experimental observation and numerical modeling](#)
Woo Seok Kang, Hyun-Ha Kim, Yoshiyuki Teramoto et al.
- [The influence of bremsstrahlung on electric discharge streamers in N₂-O₂ gas mixtures](#)
C Köhn, O Chanrion and T Neubert



Analysis Solutions for your Plasma Research

- Knowledge
- Experience ■ Expertise

[Click to view our product catalogue](#)

Contact Hiden Analytical for further details:
www.HidenAnalytical.com
info@hiden.co.uk



Surface Science

- ▶ Surface Analysis
- ▶ SIMS



Surface Science

- ▶ 3D depth Profiling
- ▶ Nanometre depth resolution



Plasma Diagnostics

- ▶ Plasma characterisation
- ▶ Customised systems to suit plasma Configuration



Plasma Diagnostics

- ▶ Mass and energy analysis of plasma ions
- ▶ Characterisation of neutrals and radicals

Collective dynamics of a dense streamer front

M B Teixeira-Gomes*, F J Gordillo-Vázquez and A Luque* 

Instituto de Astrofísica de Andalucía (IAA), CSIC, Granada, Spain

E-mail: miguel.teixeira.gomes@gmail.com and aluque@iaa.es

Received 25 May 2023, revised 21 August 2023

Accepted for publication 6 September 2023

Published 18 September 2023



Abstract

We explore the dynamics of dense streamer channel fronts. We introduce a novel, fully three-dimensional, adaptive mesh refinement streamer simulation code, which leverages the power of general-purpose graphical processing units to accelerate computations. Our code enables the simulation of systems comprising several parallel-propagating streamers, using appropriate boundary conditions to emulate an infinitely extended front of positive streamers in ambient air. Our findings reveal that denser streamer packings result in slower front propagation and increased electric field screening within the streamers. To interpret these results and progress towards developing a coarse-grained corona model, we present a streamlined model that effectively approximates the behavior of the comprehensive microscopic system.

Keywords: streamer, streamer front, electrical discharge, GPU, adaptive mesh refinement

1. Introduction

A streamer discharge is a weakly ionized filament that propagates by enhancing the electric field in its tip, where electrons gain enough energy to further ionize the embedding medium [1]. Because this mode of propagation is driven by fast electrons and does not require significant heating of the medium, streamers are initiated more readily than other types of gas discharges. They often precede and surround hot discharges such as leaders [2, 3] and, in the presence of a spatially extended electric field, streamers can form complex filamentary structures. One example of these structures where individual streamers can be optically resolved are upper-atmospheric discharges called sprites [4–7]. Fast breakdown occurring within thunderclouds is presumably also composed of streamer channels, possibly numbering around 10^8 [8, 9] per event, although in this case individual streamers have never been independently observed. Fast breakdown [10–12] is the likely source

of narrow bipolar events, which are very low frequency radio pulses with durations of tens of microseconds, associated with high frequency radiation [13–15] and also of Blue Luminous Events (BLUEs) [9, 16, 17] optical emissions emanating from thunderclouds.

In sprites, in fast breakdown and in the corona that surrounds a leader there is a large number of simultaneously propagating streamers. One natural question to ask about these phenomena is whether streamer interaction has to be considered and, in that case, what is its influence on the global dynamics of the streamer system. The collective dynamics of streamer fronts have been investigated previously with models that treated streamers as one-dimensional advancing conductors [18, 19] but a detailed, microscopical model of a streamer front requires expensive, three-dimensional computations that only recently have become within reach [20–22].

Here we apply a new code that combines adaptive mesh refinement (AMR) with graphical processing unit (GPU) computations to simulate the interactions between parallel streamers in a sufficiently dense front. Whereas AMR allows us to reduce the number of computations per time step, running these computations in a GPU allows us to perform many of these computations in parallel, achieving a higher throughput (that is, more operations per second). GPUs perform well for our task of solving the partial differential equations for

* Authors to whom any correspondence should be addressed.



Original Content from this work may be used under the terms of the [Creative Commons Attribution 4.0 licence](https://creativecommons.org/licenses/by/4.0/). Any further distribution of this work must maintain attribution to the author(s) and the title of the work, journal citation and DOI.

streamer evolution because most of the computations consist in applying relatively simple operations to each cell inside a structured grid, which a GPU can process in parallel.

As we describe below, the physical system that we simulate is an infinite, planar front of positive streamers propagating in a common direction. We model this infinite front by a lattice of identical square blocks, each containing a few streamers. In practice only one of these blocks needs to be considered in the simulation, the remaining lattice being taken into account by imposing periodic boundary conditions. The dynamics of the front is partly determined by the surface density of streamers, which we adjust by means of the number of streamers inside the representative lattice cell. We consider cases where the density is high enough that the interaction affects the streamer propagation.

This article is divided as follows: in section 2 we describe our microscopic streamer model and geometrical configuration, section 3 reports simulation results that are discussed with the aid of a macroscopic model in section 4 and finally section 5 provides some concluding remarks.

2. Model and configuration

2.1. Numerical implementation

The simulation code uses tree-based structured AMR (SAMR) and it features a total variation diminishing (TVD) method for the drift-diffusion equation based on the monotone central (MC) flux limiter [23], and full approximation scheme (FAS) full multigrid (FMG) to solve the Poisson and Helmholtz equations [24].

An adaptive mesh was required since streamers are multiscale phenomena that cover a great span of spatial scales. However, as discussed in [25], there are different types of AMR to choose from. The two main classes of AMR are: structured AMR (SAMR), and unstructured AMR (UAMR). From our investigation, we found that UAMR (the common example of which is the finite element method (FEM) meshes) is most suited for a mesh that will change little in time. Streamer behavior is at odds with this requirement because the tip of the streamer (the region that requires the highest resolution) is in constant movement.

Still, within SAMR there are two main approaches: tree-based SAMR, and patch-based SAMR. Tree-based SAMR usually refers to a quadtree or octree data structure where each node corresponds to a cell. However, here we associate each node to a small square or cubic patch of cells. Patch-based SAMR, uses a base grid to cover the entire domain of the simulation, and then uses a collection of rectangular patches to cover the regions that require more resolution. We found the distinction here to be similar to the one between SAMR and UAMR. The more structured of the two, Tree-Based, is better suited for meshes that have to constantly change in time, as is our case.

The final mesh, which we exemplify in figure 1, is also well suited for parallelization, particularly on GPUs. This is

because all the patches have the same structure independently of the resolution.

However, figure 1 only shows a static snapshot of the mesh. During an actual simulation, the mesh will have to adapt to the local need for resolution (or lack thereof). To do this we will define a criterion based on local and global characteristics that will define for each grid cell whether it needs a finer grid or it can have a coarser grid. Then, for each quadrant or octant of a patch we will create a finer patch if a single cell requires higher resolution. If all cells in the patch can work with lower resolution that patch is marked for removal. We provide an example of the adaptive grid in one of our simulations in figure 2.

When designing a GPU-based application there are usually two main options to choose from: CUDA or OpenCL. In this case, we chose CUDA because of its wider range of numerical libraries, because it supports more C++ code features, and because it is more often available in computing servers. We run all our simulations in Nvidia Tesla P100 accelerator cards (released in 2016, 12GB of high-bandwidth memory, theoretical peak double-precision performance of 4.7 TFLOPS, i.e. 4.7×10^{12} floating-point operations per second).

Regarding the programming language, two common choices are to directly develop in C++ or use PyCUDA. Although PyCUDA can reduce the development time by minimizing the code length, it suffers from an outdated interface and misses features from newer CUDA versions. So, we chose to write directly in C++.

Performant GPU-based computations rely on methods called *kernels* than simultaneously apply an operation to a large number of elementary input data (for example, all cells in the domain of a streamer simulation). Launching each kernel as a significant overhead, so ideally one has to minimize the number of launchings.

Most of the compute-intensive calculations needed throughout one of our simulations can be implemented independently on each patch (figure 3) and therefore a single kernel suffices. However, for this to be possible the communication between patches needs to as efficient as possible. To achieve this we keep a data structure where each patch has easy access to the patches it needs to fill its ghost cells. We improve performance further by using a single kernel launch per refinement level of the mesh. The kernel covers all the ghost cell boundaries, including direct copy, interpolation, and boundary conditions. Although this uses branching in the code—usually a problem for GPUs—as long as the boundaries have at least 32 cells, the cost is reasonably small because the kernel operations are divided into small subsets (*warps*) and we ensure that all operations inside a warp are equivalent.

2.2. Physical model and configuration

We use a fluid model for streamers in dry air at atmospheric pressure as previously described in other streamer studies [1, 26–28]. We consider electrons that drift and diffuse at rates that depend on the local electric field. The electron number density n_e follows

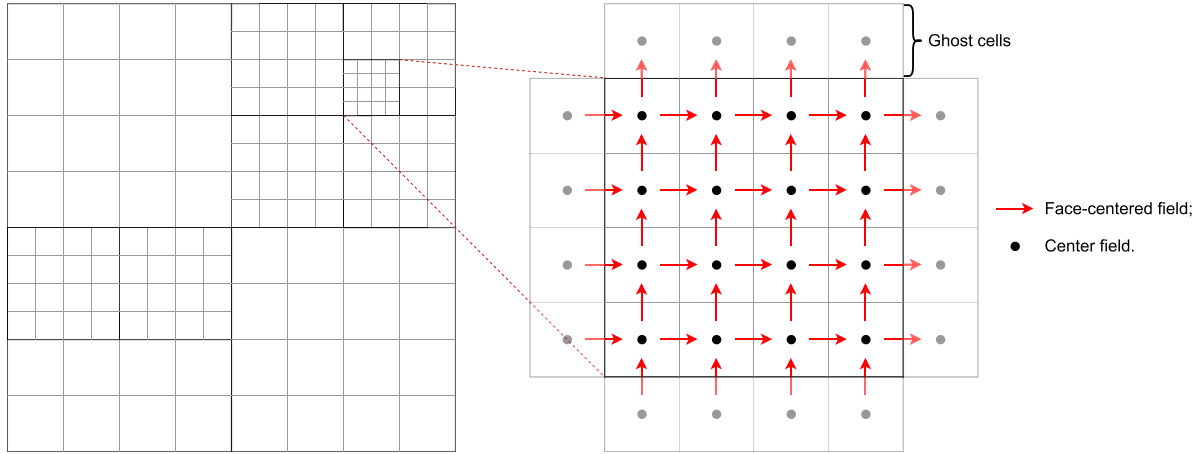


Figure 1. Two-dimensional example of tree-based SAMR grid used in our simulation code. Notice the use of ghost cells which are used to communicate data between the patches.

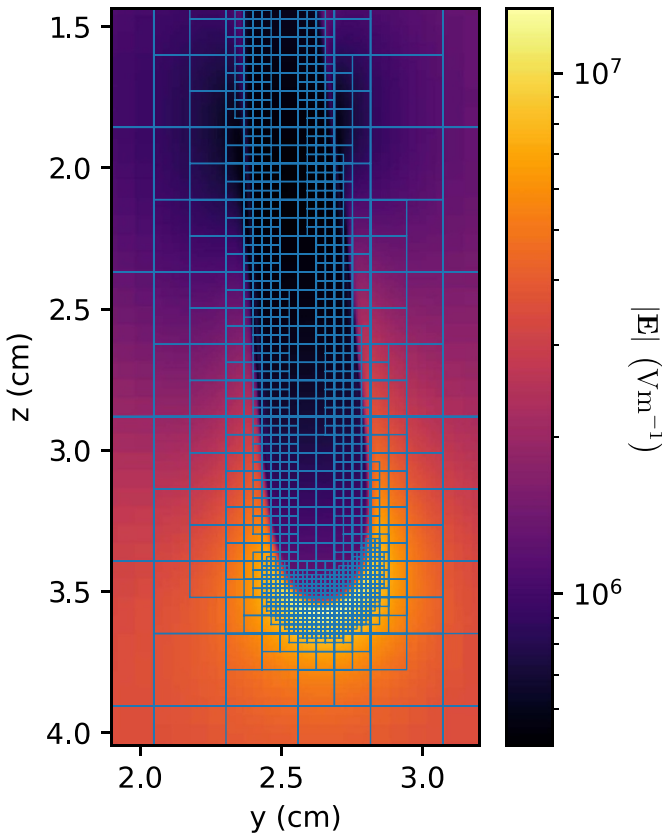


Figure 2. Example of adaptive mesh refinement (AMR) in our code. For one of our simulations with the system and configuration described in section 2.2 (in this case four streamer seeds) we plot the electric field in the plane $x = 1.955$ cm at time $t = 30$ ns, together with the intersections with the oct-tree cube blocks that form the mesh. Each block consists of $8 \times 8 \times 8$ grid cells. The plot represents the same simulation and time instant as figure 5.

$$\frac{\partial n_e}{\partial t} - \nabla \cdot (n_e \mu \mathbf{E} + D \nabla n_e) = S_l + S_{ph}, \quad (1)$$

where μ is the electron mobility, D is a scalar electron diffusion coefficient, S_l is the net local ionization rate and S_{ph} is

the source of free electrons due to photo-ionization. Positive and negative ions, being much heavier than electrons, are considered immobile. We neglect ion-ion or electron-ion recombination so positive and negative ions can be added into a single variable n_i . This is governed by

$$\frac{\partial n_i}{\partial t} = S_l + S_{ph}. \quad (2)$$

The net charge resulting from electrons and ion generates an electric field $E = -\nabla \phi$ where the electrostatic potential ϕ satisfies Poisson's equation

$$\epsilon_0 \nabla^2 \phi = -e(n_i - n_e), \quad (3)$$

where ϵ_0 is the vacuum permittivity and e is the elementary charge.

The local ionization source includes impact ionization and attachment, both dissociative and three-body:

$$S_i = \nu_{\text{eff}} n_e, \quad (4)$$

where ν_{eff} is an effective net ionization rate. The second source of ionization is photo-ionization, which is modeled through a non-local term; for this we follow Zhelezniak's model [29], which we approximate by solving a set of Helmholtz equations [30] using the two-term fit described by [31].

In order to facilitate the validation of our code (see appendix) and the comparison with existing streamer codes, for the electron mobility μ and diffusion coefficient D as well as for the net ionization rate ν_{eff} we use the functional forms described in [28].

The configuration used in our simulations is sketched in figure 4. We model a planar front composed by a large number of streamers propagating approximately in parallel in the z direction. To make this system feasible for numerical simulations we divide the xy plane into an infinite lattice of square cells with a side length L and consider identical dynamics for each cell. With this simplification we reduce the system to the simulation of a single cell on which we impose periodic boundary conditions on each of its four sides. As we see below, each

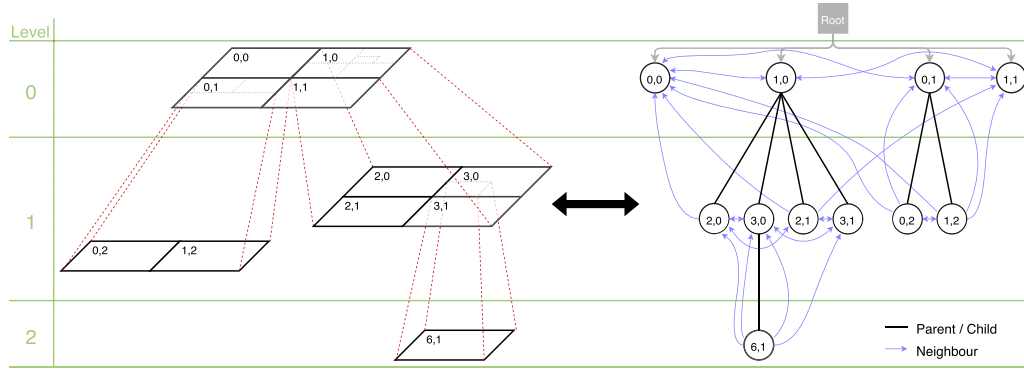


Figure 3. Scheme of the internal representation of the computational grid. At each refinement level grid is divided into blocks with identical size (left) that are then arranged into a tree structure (right). Given a tree structure, we compute the necessary data structures to store the connections between nodes. These structures are used to communicate the boundary conditions for blocks efficiently in a single kernel launch for each computational level.

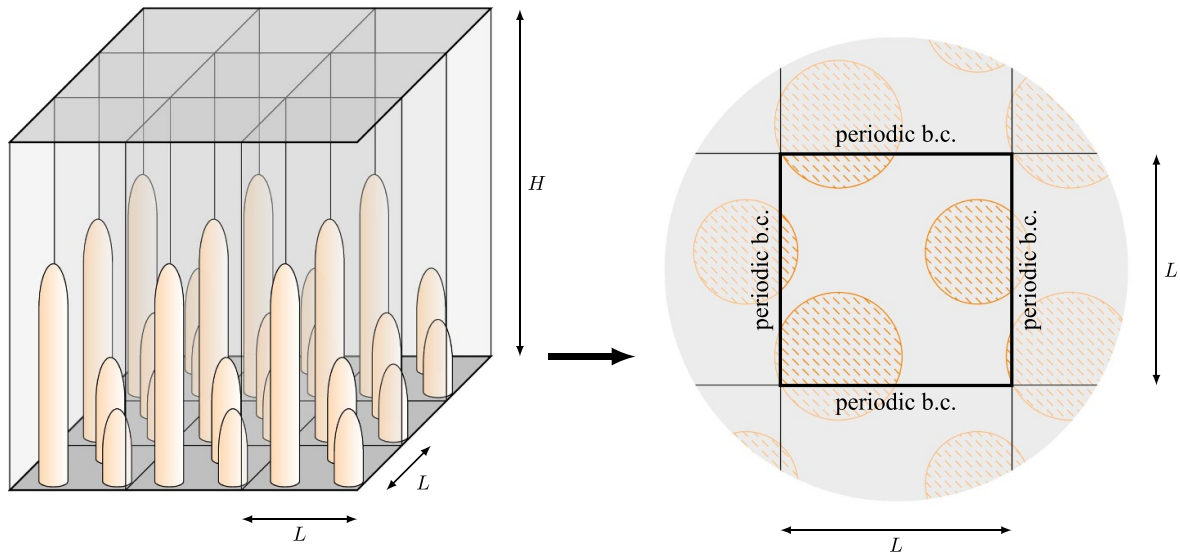


Figure 4. Sketch of the geometry used in our simulations. We consider an infinite, periodically arranged system of positive streamers propagating almost in parallel between two electrodes separated a distance H . The system is composed by a square cell of side L that repeats indefinitely in the two directions perpendicular to the propagation; this is modeled by considering a simulation domain with size $L \times L \times H$ on which we impose periodic boundary conditions in the lateral boundaries (left picture). The sketch represents a configuration of two streamers per cell.

cell contains from one to five streamers. A similar approach to modeling streamer fronts was undertaken by [32], although there the emphasis was on reproducing 2D (planar) analytical results.

Modeling the front as a periodic lattice adds a long-range order to the front which is not present in real streamer coronas. The assumption here is that for long-range interactions the precise locations of companion streamers are not important. However, the interaction between close neighbors must be modeled with some detail, which is why several streamers are included in our computational domain.

All our simulations share similar initial conditions. The simulation box has dimensions $L = 4.096$ cm and $H = 8.192$ cm (respectively 2^{12} and $2^{13} \times 10 \mu\text{m}$). A uniform electric field is established by setting a potential $\phi = 0$ on the bottom plate (where the streamer seeds are placed) and

a potential of $\phi = 204.8$ kV on the top plate, generating a background electric field of $E = 2.5$ MV m^{-1} , which we selected as representative of a high electric field but lower than the breakdown field. To initiate the streamers several seeds, with equal densities of electrons and positive ions, are placed randomly on the bottom plate (anode, to initiate positive streamers). Each seed is a super-Gaussian cylinder capped with a super-Gaussian semi-sphere, resulting in the following initial electron density for N initial seeds:

$$n(\mathbf{x}) = n_{\text{bg}} + \sum_{i=1}^N (n_{\text{max}} - n_{\text{bg}}) \times \begin{cases} \exp\left(-\frac{|\mathbf{x} - \mathbf{x}_i - \mathbf{x} \cdot \hat{\mathbf{z}}|^6}{r^6}\right) & \text{if } |\mathbf{x} \cdot \hat{\mathbf{z}}| < l - r \\ \exp\left(-\frac{|\mathbf{x} - \mathbf{x}_i - (l-r)\hat{\mathbf{z}}|^6}{r^6}\right) & \text{if } |\mathbf{x} \cdot \hat{\mathbf{z}}| \geq l - r \end{cases}, \quad (5)$$

where \mathbf{x}_i is the center of the randomly placed i th seed (always in the $z = 0$ plane), the maximum density is $n_{\max} = 10^{19} \text{ m}^{-3}$, the seed radius is $r = 0.5 \text{ mm}$, and the seed length is $l = 2 \text{ mm}$. With our parameters the probability of two seeds being close enough to produce a single streamer was low and we did not observe such an event. Finally, a background ionization $n_{\text{bg}} = 10^{15} \text{ m}^{-3}$ is added throughout the simulation box.

One of the big challenges was to create a refinement criterium specific for streamer simulations. We start by defining 3 levels of refinement that take into account the time t :

- high: set to $10 \mu\text{m}$ until $t = 14 \text{ ns}$ and to $20 \mu\text{m}$ afterwards;
- medium: set to $20 \mu\text{m}$ until $t = 15 \text{ ns}$ and to $40 \mu\text{m}$ afterwards;
- low: set to $40 \mu\text{m}$ until $t = 16 \text{ ns}$ and to $80 \mu\text{m}$ afterwards;

By adapting the highest resolution we limit the number of patches in late stages of the simulation, which require less resolution due to larger streamer radius. We also run simulations without this reduction and noticed negligible difference in the results despite significantly longer simulation times. The staggered lowering of the resolution for each level prevents the simultaneous collapse of multiple resolution levels. The conditions for each level are as follows:

- high: if $S_{\text{ph}}^1 > 1 \times 10^{20} \text{ m}^{-3} \text{ s}^{-1}$ until 7 ns and if $S_{\text{ph}}^1 > 2 \times 10^{20} \text{ m}^{-3} \text{ s}^{-1}$, afterwards;
- medium: if $5 \times 10^{17} \text{ m}^{-3} < n_e < 2 \times 10^{19} \text{ m}^{-3}$;
- low: if $n_e > 1 \times 10^{16} \text{ m}^{-3}$;

where S_{ph}^1 is the first term in the Helmholtz approximation of the photo-ionization as found in [31]. The specific values listed above were selected as a compromise after testing several simulations. We checked our refinement criterium in a small-scale simulation where we compared with a simulation with a homogeneous high-resolution mesh, seeing no significant difference in the results. The small differences that we observed when we reduced the minimum grid size are discussed below.

3. Results

With the code and configuration described above we run simulations containing from one to five initial seeds, imitating seed surface densities approximately from 0.06 cm^{-2} to 0.3 cm^{-2} . In order to collect meaningful statistics, the initial locations of the seeds were randomly selected 4 times for each of these initial densities, resulting in a total of 20 simulations. Table 1 shows the average time taken per simulation depending on the number of streamers, as well as the number of grid points at the end of the simulation. We appreciate a sub-linear scaling of the computational resources required to simulate increasing numbers of streamers which we attribute to the overlap of the fine meshes around the streamers.

Figure 5 shows a snapshot from one of the simulations with four streamers, obtained at time $t = 30 \text{ ns}$ after the start time. The streamers have similar lengths but in this case one of them

Table 1. Performance of the code for different number of streamers in the simulation box. Second column is the average simulation time in minutes on a Nvidia P100. Third column is the average number of grid points at the end of the simulation.

Number of streamers	Simulation time (min.)	Grid points (10^6)
1	211	18.4
2	354	28.8
3	451	35.5
4	538	40.1
5	626	42.9

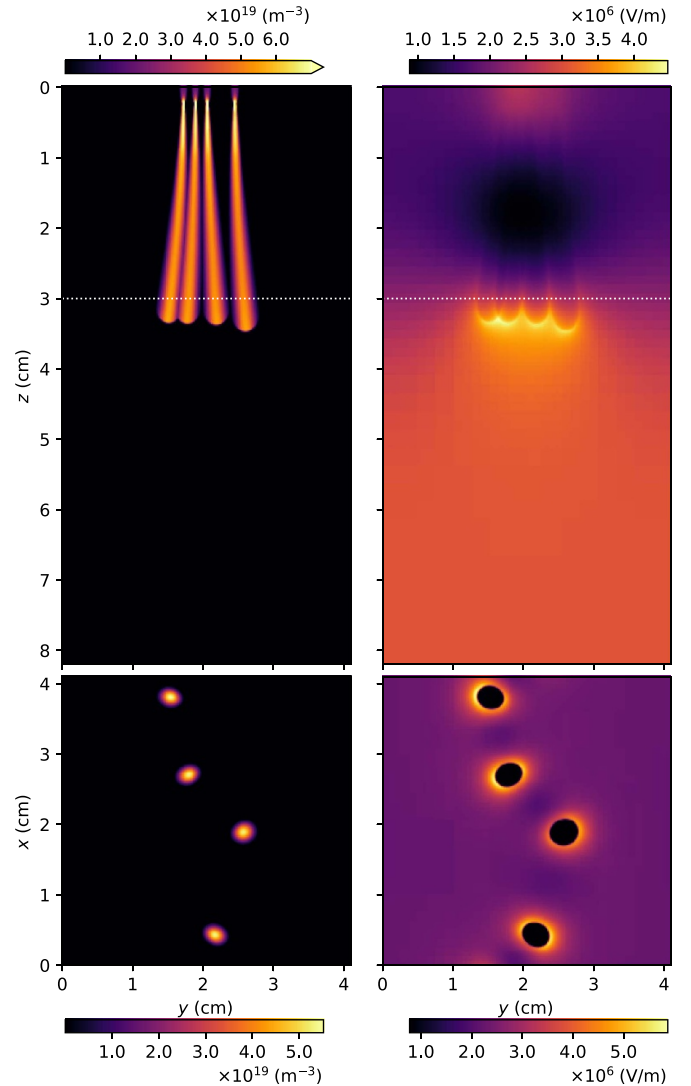


Figure 5. State of a simulation with four streamers (a surface density of $4/L^2 \approx 0.24 \text{ cm}^{-2}$) at time $t = 30 \text{ ns}$. The upper left panel shows the maximum of the electron density along lines aligned with the x axis. The upper right panel shows the average magnitude of the electric fields along these same lines. The lower two panels contain cuts of the electron density (left) and electric field magnitude (right) at horizontal planes indicated by the dashed lines in the upper panels.

(to the right) has advanced slightly more. Each streamer carries a positive charge that accumulates mostly around its head. The effect of this charge on other streamers is to enhance the electric field of those that are ahead and screen the electric field

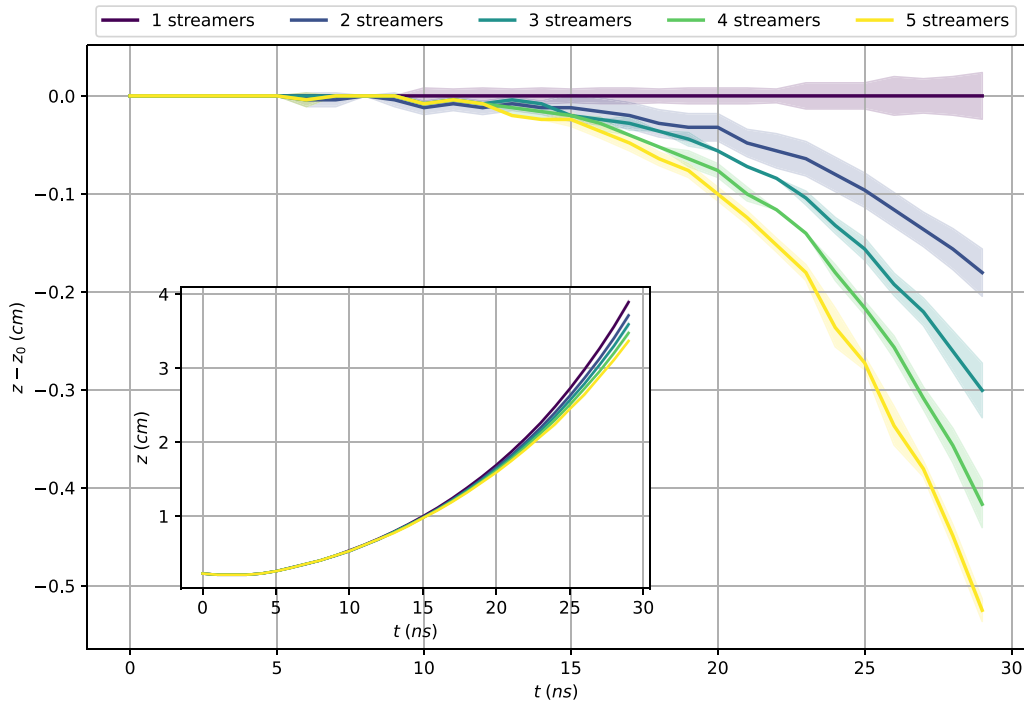


Figure 6. Length of the longest streamer for each configuration. The inset shows the location z of the streamer that has travelled the furthest away from the anode whereas the main plot shows the difference between this distance and the mean distance travelled in simulations with a single streamer. For each configuration we plot the mean within four samples with the same configuration but different streamer starting positions (thick line) as well as the standard deviation of these four samples (shaded areas). Differences between simulations with the same number of streamers are in part due to numerical errors, which depend on the relative location between the seeds and the grid.

of those behind. Therefore the spread in propagation length of all streamers in one simulation increases over time.

Let us now focus on the effect that the presence of neighboring streamers has on the speed of the front. For each configuration we measured the length of the streamer that has traveled the furthest away from the inception electrode. The streamer tip location is defined here as the location of the corresponding local maximum of the electric field.

Figure 6 shows the average of this maximal length for each streamer density together with a range of variation estimated as the standard deviation of the four simulations. We find that a denser packing of streamers leads to slower propagation. After 30 ns the simulations with five seeds have propagated about 8 mm less than the single-streamer simulations, which amounts to about 15% lower average speed. That a higher density of streamers leads to stronger screening and thus slows down the streamer propagation was already observed in a simplified model of a streamer corona around a spherical electrode [19]. Note however that, as we discussed above, the electric field in the longest streamer is enhanced by the presence of surrounding streamers. The fact that this leading streamer is also slower when surrounded by others is therefore not immediately obvious.

Note that in figure 6 substantial part of the variation between simulations with the same number of streamers is due to numerical errors in our simulations. By reducing the minimum grid size a factor 2 in the simulation with a single streamer, the length decreased by 1.4% whereas the spread between simulations decreased by a factor 0.14. The speeds in

our simulations depend also on other, more physical, parameters such as the background ionization density and the length of the gap between the electrodes. We did not perform a parametric study to determine how far our results generalize to different configurations. For example, some of our simulations with a background density of 10^{10} m^{-3} did not finish, presumably because of streamer branching. Avoiding the branching of streamers is one of the motivations that we selected a relatively high background density. Nevertheless, we hypothesize that the underlying physical mechanisms that we discuss below operate also in different scenarios including, for example, streamer branching.

To understand better our observations and to obtain a more detailed view of the dynamics of the streamer front, it is useful to define cross-sectional averages of some quantities of interest. The averaged electron number density and charge density per streamer read

$$\bar{n}_e(z) = \frac{1}{NL^2} \int dx dy n_e(x, y, z), \quad (6a)$$

$$\bar{q}(z) = \frac{1}{NL^2} \int dx dy q(x, y, z), \quad (6b)$$

where N is the number of streamers in a cell of area L^2 . Rather than averaging the electric field across the simulation volume, it is more informative to measure the electric field that acts on the electrons inside each streamer. Thus, we define an internal field E_{int} by weighting the average with the electron density:

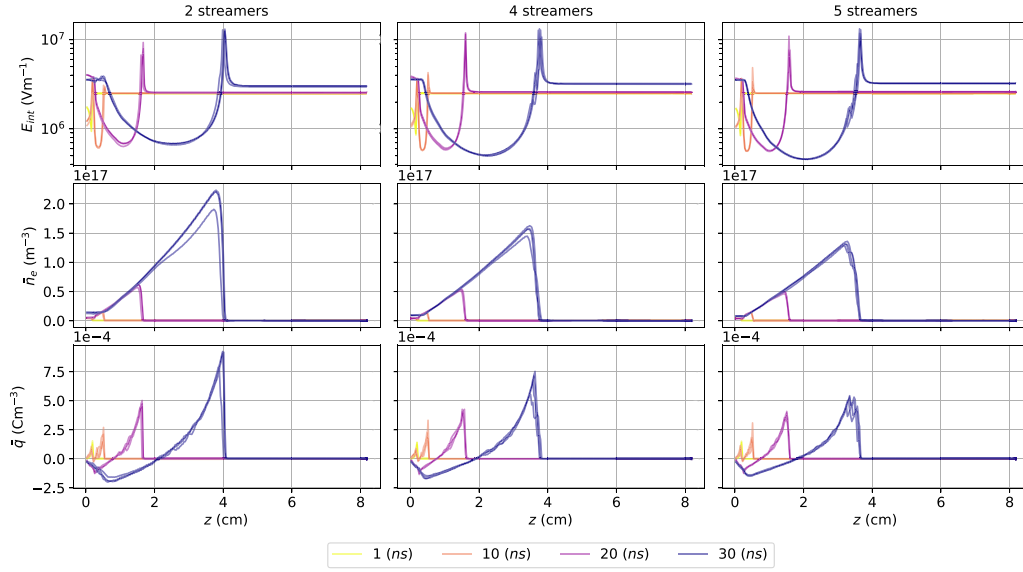


Figure 7. Comparison of simulations with three different streamer surface densities, parametrized by the number of streamers inside a periodic cell (2, 4 and 5 streamers). Each plot shows quantities of interest derived from cross-sections perpendicular to the axis of propagation. See the text for precise definitions of each of these quantities.

$$E_{\text{int}}(z) = \frac{1}{NL^2 \bar{n}_e(z)} \int dx dy E_z(x, y, z) n_e(x, y, z). \quad (7)$$

With these definitions the cross-sectional averaged electrical current can be approximated as

$$\bar{j}_z = \frac{1}{NL^2} \int dx dy e \mu(E) E_z n_e \approx e \mu(E_{\text{int}}) E_{\text{int}} \bar{n}_e, \quad (8)$$

with the latest approximation being exact for a constant electron mobility. The conservation of charge in the direction of propagation reads

$$\frac{\partial \bar{q}}{\partial t} = -\frac{\partial \bar{j}_z}{\partial z}. \quad (9)$$

Figure 7 shows the evolution of E_{int} , $\bar{n}_e(z)$ and \bar{q} for our simulations with 2, 4 and 5 streamers. The following features are worth pointing out:

- (i) The internal electric field is lower in a denser streamer configuration. With two streamers at $t = 30$ ns the lowest internal electric field is about $8 \times 10^5 \text{ V m}^{-1}$ whereas in simulations with five streamers this quantity is about $5 \times 10^5 \text{ V m}^{-1}$. Because the internal electric field depends only on values of the electric field inside the streamers, where the electron density is high, this difference cannot be attributed to the larger volume occupied by five streamers.
- (ii) A higher streamer density leads to lower cross-sectionally averaged electron density.
- (iii) As equations (8) and (9) show, the evolution of the charge density roughly depends on the product of the internal field E_{int} and the averaged electron density \bar{n}_e . Both the electron density and internal fields are lower for simulations with more streamers, leading to lower charge densities, as seen in the bottom row of the figure.

These points are consistent with previous results of macroscopic models such as [19].

4. Discussion

To understand the dynamics described in the previous section it is useful to look at how the presence of other streamers affects the electrostatic interactions between elements. As sketched in figure 8, there are two limiting cases of this interaction, depending on the distance Δz between a given slice of a streamer containing on average a charge $d\bar{Q} = \bar{q} dz$ and the electrons that are affected by this charge.

For small Δz the interaction is dominated by the effect of charges inside the same streamer. The effect on the electric field is proportional to $d\bar{Q}$ and thus independent of other streamers. At long range (larger Δz) neighboring streamers come into play. In this limit the electrostatic interaction may be approximated by that created by an uniformly charged plane with a surface charge density $d\sigma = N d\bar{Q}$ (i.e. it is proportional to the streamer surface density N/L^2).

With these observations in mind, let us return to the behavior of the fastest-propagating streamer discussed above and plotted in figure 6. The interaction of neighboring streamers has two competing effects on the velocity of this leading streamer:

- (i) The electric field at the tip increases due to the contribution of the positive charges of streamers that have been left behind.
- (ii) These same charges have the opposite effect of decreasing the electric field in the interior of the leading streamer. This limits the amount of charge transported to the tip, reducing the field at the streamer tip.

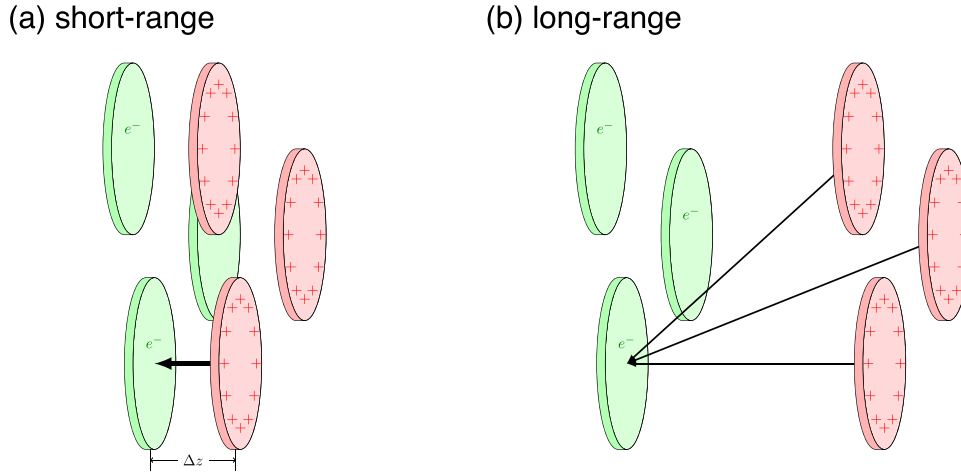


Figure 8. Different limits of the electrostatic interaction as seen by a slab of electrons in a streamer (green disks). At short distances (a) only charges in the same streamer interact significantly with the streamer: in this range the presence of neighboring streamers is irrelevant. At longer distances (b) electrons are affected by interactions from all streamers and the presence of neighboring streamers becomes more relevant.

For point (i) it is mostly the short-range interaction that is important, whereas point (ii) is more dependent on the long-range interaction and therefore it is much more affected by the presence of other streamers. This would explain the slower propagation of denser configurations.

4.1. Macroscopic one-dimensional model

In order to check our understanding of the physics of dense coronas and as a first step in the construction of more rigorous coarse-grained models, we present here a simplified, toy model that can be compared qualitatively with the results of the rigorous, microscopic model. The full source code of our implementation of this 1D model is publicly available [33].

We describe the evolution of the averaged variables defined above, namely $\bar{q}(z, t)$ and $\bar{n}_e(z, t)$ together with the location of the streamer front, $z_{\text{tip}}(t)$. Although in the microscopic model described above the radius of the streamers changes significantly over time, in the simplified model we set a constant radius $R = 1$ mm for all the streamers.

4.1.1. Electrostatic interaction. As we mentioned above, for the long-range part of the electrostatic interaction the contribution of all streamers to the electric field can be approximated by that of a uniformly charged plane. The long-range electric field can thus be written as $E_L = -\partial_z \phi_L$ where the potential ϕ_L satisfies

$$\frac{\partial^2 \phi_L}{\partial z^2} = -\frac{N\bar{q}}{\epsilon_0}. \quad (10)$$

As our goal is to provide a qualitative understanding of the problem, we employ a simplified model for the short-range interaction. We assume that electrons are affected by nearby charges as if the charges are uniformly distributed in disks with radius R . Thus, the integrated charge density $NL^2\bar{q}$ concentrates in an area πNR^2 . Besides, we give this interaction

a limited range R , resulting in the following equation for the short-range potential:

$$\frac{\partial^2 \phi_S}{\partial z^2} - \frac{\phi_S}{R^2} = -\frac{L^2 \bar{q}}{\pi \epsilon_0 R^2}, \quad (11)$$

from where the short-range electric field derives as $E_S = -\partial_z \phi_S$.

We have thus an electric field E_L that accounts for interactions in the long-wavelength limit and another one E_S for the short-wavelength limit. When we combine them however, we must exercise caution as E_L does not vanish in the short-range. We therefore construct a linear combination $E_L + cE_S$ where the constant c is selected to obtain the proper short-range behaviour, namely

$$\frac{cL^2}{\pi R^2} + N = \frac{L^2}{\pi R^2}, \quad (12)$$

which results in

$$c = 1 - \frac{\pi NR^2}{L^2} = 1 - \rho, \quad (13)$$

where $\rho = \pi NR^2/L^2$ is the fraction of a cross-section covered by streamers.

After these considerations, our equation for the internal electric field reads

$$E_{\text{int}} = E_L + (1 - \rho)E_S + E_B, \quad (14)$$

where E_B is an additional, externally imposed electric field.

For the solution of the one-dimensional Poisson equations for ϕ_L and ϕ_S we use homogeneous Dirichlet boundary conditions at both sides of the domain.

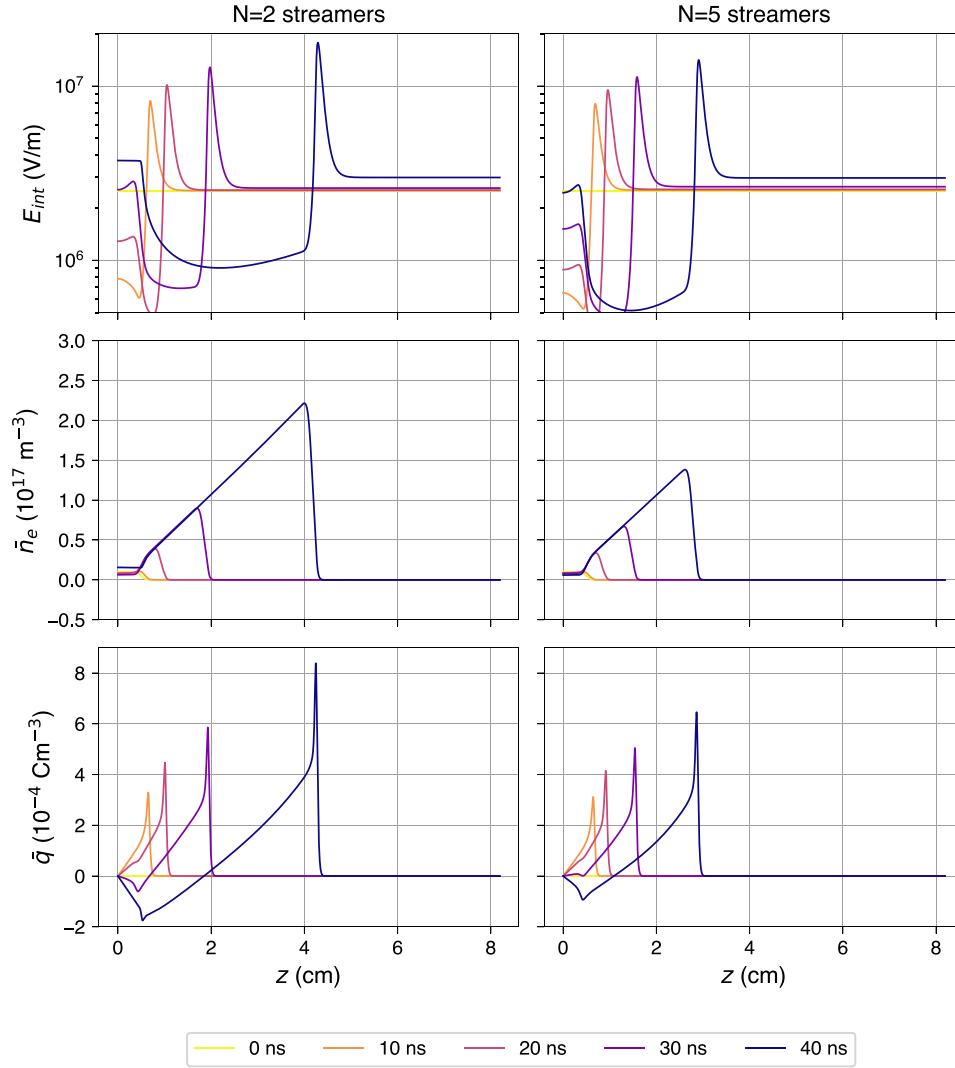


Figure 9. Simulation of a streamer model with a simplified one-dimensional model. We plot results for $N = 2$ and $N = 5$ streamers. These results can be compared with figure 7 but note the slightly different snapshot times.

4.1.2. Front velocity. We take the front velocity to be a function of the peak electric field resulting from (14). Specifically we use the expression proposed by Naidis [34]. This results in

$$\frac{dz_{\text{tip}}}{dt} = v = F_{\text{Naidis}}(\max(E_{\text{int}}), R). \quad (15)$$

4.1.3. Electron density. The evolution of the electron density \bar{n}_e combines two components: in the streamer interior it is determined by electron ionization and attachment whereas close to the tip it is affected by photoionization and impact ionization ahead (two processes that we do not include with any detail in this simplified model). The evolution of electron density and radius of the streamers is outside the scope of this paper so here we fix the average electron density after the passage of the streamer to approximate the results of the microscopic model. We chose a linearly increasing density $\bar{n}_0 = 5 \times 10^{18} \text{ m}^{-4} \times z_{\text{tip}}$ (see figure 7). We represent the transition between head and body by a Gaussian profile with an e -folding length of R . The resulting equation is

$$\frac{\partial \bar{n}_e}{\partial t} = \frac{\bar{n}_0 v}{R\pi^{1/2}} \exp\left(-\frac{(z - z_{\text{tip}})^2}{R^2}\right) + \nu_{\text{eff}}(E_{\text{int}})\bar{n}_e \text{erfc}\left(\frac{z - z_{\text{tip}}}{R}\right). \quad (16)$$

Here ν_{eff} is an effective rate of ionization that includes ionization, dissociative and three-body attachment. The prefactor in the first term is such that its time integration for a uniform propagation with speed v yields \bar{n}_0 . The erfc function in the second term attenuates the effect of ionization close to the head, where this effect is already included in the first term.

4.1.4. Charge conservation. As we mentioned above, our definition of the internal electric field E_{int} leads to a simple expression for the evolution of the average charge density \bar{q} , namely equation (9).

4.1.5. Results. Figure 9 shows simulation results from our simplified model for $N = 2$ and $N = 5$ streamers in the same domain size ($L = 4.096 \text{ cm}$, $H = 8.192 \text{ cm}$) and background electric field $E_B = 2.5 \text{ MV m}^{-1}$ as our detailed, microscopic

simulations. Although crudely, our simplified model captures the essential features of the electrostatic interactions in a streamer front. We reproduce the two most remarkable features of the microscopic simulation: that denser fronts are slower and exhibit stronger screening inside the streamers.

The values of the charge density \bar{q} and the internal electric field E_{int} derived from the simplified model are reasonably close to those of the microscopic simulations (\bar{n}_e is simply selected to match those results). Also the front speed is similar in both cases (for $N=2$ we find $v \approx 1.3 \times 10^6 \text{ m s}^{-1}$ in the microscopic model against $v \approx 10^6 \text{ m s}^{-1}$ in the one-dimensional model). However we have not explored other parameter regimes so we only claim a qualitative similarity between the two models.

From our one-dimensional model we conclude that the dynamics of a dense streamer front can be understood by thinking in terms of a stronger long-range interaction between distant points of the system. We propose that this insight should be applied in future, more accurate models of streamer coronas.

5. Conclusions

In this study, we have made progress in understanding the dynamics of streamer plasmas, particularly in the context of densely packed fronts. We have introduced a novel numerical code that harnesses the power of modern GPUs for massive parallelization of arithmetic operations in grid cells, while simultaneously optimizing performance through the implementation of a moving, adaptive mesh. This approach has enabled us to simulate streamer fronts with up to five nearly parallel streamers.

Our findings reveal that the presence of neighboring streamers within a front slows down the propagation of even the fastest streamer in the group. This phenomenon is accompanied by lower electric fields within each individual streamer. Our one-dimensional model offers insights into these characteristics, suggesting that the long-range effect of charges in a streamer is seemingly enhanced by the presence of other streamers. While this serves as a heuristic explanation, the true underlying cause stems from the combined effects of charges in multiple streamers.

The increased screening inside the streamer channels is consistent with previous explorations of streamer interactions. In [32], for the case of a periodic configuration of planar streamers, it was argued that the front approaches a state of complete field screening in its wake. These arguments should be also applicable to our configuration but our streamers did not propagate long enough to reach this state. On the other hand, a significant mutual screening was also noticed in [18], where the outcome was named *collective streamer front* or, informally, *streamer of streamers*. An important difference between the simulations of [18] and the present work is that in the former the front naturally developed a curved shape, which lead to stronger field enhancement around the center of the corona.

The simultaneous propagation of a multitude of streamers occurs in several contexts and our results point to the

relevance of streamer interactions when streamers are closely packed. We have investigated fronts at atmospheric pressure with densities from about 600 m^{-2} (for 1 streamer in each $4 \text{ cm} \times 4 \text{ cm}$ cell) to about 3000 m^{-2} (for five streamers). Let us provide context to these numbers:

- (i) A sprite consists of on the order of 100 streamers within a radius of about 10 km. This results in a density of about $3 \times 10^{-7} \text{ m}^{-2}$ which, rescaled to atmospheric pressure, gives about 1400 m^{-2} .
- (ii) If fast breakdown [10] consists in a single streamer front with roughly 10^8 streamers [8, 9] distributed in a circular area of radius 100 m their streamer density would be about 3000 m^{-2} .
- (iii) It is difficult to identify single streamers in photographs of laboratory experiments in meter-wide gaps such as those by [35]. To obtain order-of-magnitude estimates we may consider an average distance between branchings at atmospheric pressure of about $\ell = 2 \text{ cm}$ [36] (see also [37, 38] for similar results after scaling pressure) and that the transversal spread of the streamer corona is not very different to its longitudinal span (which is the case, at least, for point-initiated discharges, see e.g. [39] and simulations in [18]). Then after propagating a distance d the front density would be around $\exp(d/\ell)/\pi d^2$. This gives a front density 5000 m^{-2} after propagating a distance $d = 10 \text{ cm}$.

Based on these examples, it is likely that a streamer corona often transforms, through branching, into a front densely populated with streamers. These closely packed streamers interact in a way that should be considered in future models. We believe our work represents a step towards developing coarse-grained models capable of investigating the dynamics of extensive streamer coronas, potentially involving hundreds of millions of streamers. Such investigations are currently beyond the reach of existing computational capabilities, but we hope that our progress lays the foundation for future advancements in this field.

Data availability statement

The data cannot be made publicly available upon publication because they are not available in a format that is sufficiently accessible or reusable by other researchers. The data that support the findings of this study are available upon reasonable request from the authors.

Acknowledgments

This work has received funding from the European Union Horizon 2020 research and innovation program under the Marie Skłodowska-Curie grant agreement SAINT 722337 and from the European Research Council (ERC), Grant Agreement 681257 and by the Spanish Ministry of Science and Innovation, under project PID2019-109269RB-C43. We also acknowledge financial support from the State Agency for Research of the Spanish MCIU through the ‘Center of Excellence Severo Ochoa’ award from the Instituto de Astrofísica de Andalucía (CEX2021-001131-S).

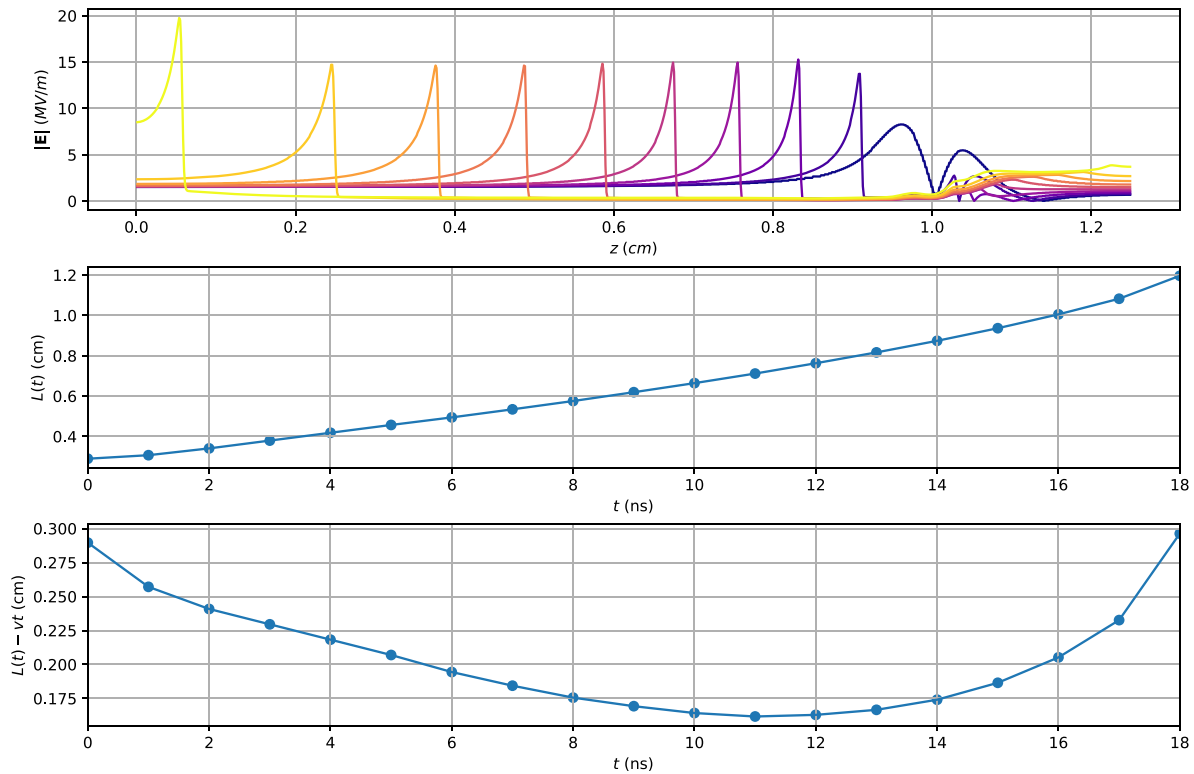


Figure A1. Results of our code when reproducing the setup of the first test case of [28]. The upper panel shows the electric field in the central axis of the simulation at intervals of 1 ns. The lower two panels plot the location and of the streamer as a function of time, with the lower plot showing differences relative to a constant propagation speed $v = 0.05 \text{ cm ns}^{-1}$. These plots should be compared with figure 5 in [28].

Appendix. Code validation

We validated our code by comparing with the streamer simulations described by Bagheri *et al* [28], namely the first case setup in that paper. This is a simulation where a streamer propagates inside a cylindrical domain of height and radius $L_r = L_z = 1.25 \text{ cm}$ in which there is a potential difference of 18.75 kV between the lower and upper boundaries. A positive streamer is initiated by means of a spherical non-neutral, positive-ion Gaussian seed with a peak density $N_0 = 5 \times 10^{18} \text{ m}^{-3}$ and a e -folding length $\sigma = 0.4 \text{ mm}$. In this setup photo-ionization is neglected and the streamer propagates due to a background ionization density of 10^{13} m^{-3} . [28] provides further details.

In our case, since our code is purely three-dimensional, we replaced the domain geometry by a square prism with side length $2L_r$ and located the initial seed in the center. Figure A1 shows the results of our simulation. The axial field as well as the location of the streamer head as a function of time closely reproduce the results of [28].

ORCID iD

A Luque  <https://orcid.org/0000-0002-7922-8627>

References

- [1] Nijdam S, Teunissen J and Ebert U 2020 *Plasma Sources Sci. Technol.* **29** 103001
- [2] Kostinskiy A Y, Syssoev V S, Bogatov N A, Mareev E A, Andreev M G, Bulatov M U, Sukharevsky D I and Rakov V A 2018 *J. Geophys. Res.* **123** 5360–75
- [3] Malagón-Romero A and Luque A 2019 *Geophys. Res. Lett.* **46** 4029–38
- [4] Franz R C, Nemzek R J and Winckler J R 1990 *Science* **249** 48–51
- [5] Stenbaek-Nielsen H C and McHarg M G 2008 *J. Phys. D* **41** 234009
- [6] Malagón-Romero A, Teunissen J, Stenbaek-Nielsen H C, McHarg M G, Ebert U and Luque A 2020 *Geophys. Res. Lett.* **47** e85776
- [7] Gordillo-Vázquez F J and Pérez-Invernón F J 2021 *Atmos. Res.* **252** 105432
- [8] Liu N, Dwyer J R, Tilles J N, Stanley M A, Krehbiel P R, Rison W, Brown R G and Wilson J G 2019 *J. Geophys. Res. Atmos.* **124** 10,134–53
- [9] Li D, Luque A, Gordillo-Vázquez F J, Liu F, Lu G, Neubert T, Chanrion O, Zhu B, Østgaard N and Reglero V 2021 *J. Geophys. Res. Atmos.* **126** e2021JD035013
- [10] Rison W, Krehbiel P R, Stock M G, Edens H E, Shao X-M, Thomas R J, Stanley M A and Zhang Y 2016 *Nat. Commun.* **7** 10721
- [11] Tilles J N, Liu N, Stanley M A, Krehbiel P R, Rison W, Stock M G, Dwyer J R, Brown R and Wilson J 2019 *Nat. Commun.* **10** 1648
- [12] Li D, Luque A, Gordillo-Vázquez F J, Silva C d, Krehbiel P R, Rachidi F and Rubinstein M 2022 *Geophys. Res. Lett.* **49** e97452
- [13] Le Vine D M 1980 *J. Geophys. Res.* **85** 4091–5
- [14] Willett J C, Bailey J C and Krider E P 1989 *J. Geophys. Res.* **94** 16255–67

- [15] Smith D A, Shao X M, Holden D N, Rhodes C T, Brook M, Krehbiel P R, Stanley M, Rison W and Thomas R J 1999 *J. Geophys. Res.* **104** 4189–212
- [16] Soler S et al 2020 *J. Geophys. Res. Atmos.* **125** e2020JD032708
- [17] Li D, Luque A, Lehtinen N G, Gordillo-Vázquez F J, Neubert T, Lu G, Chanrion O, Zhang H, Østgaard N and Reglero V 2022 *Geophys. Res. Lett.* **49** e98938
- [18] Luque A and Ebert U 2014 *New J. Phys.* **16** 013039
- [19] González M, Gordillo-Vázquez F J and Luque A 2019 *Plasma Sources Sci. Technol.* **28** 115007
- [20] Teunissen J and Ebert U 2017 *J. Phys. D: Appl. Phys.* **50** 474001
- [21] Marskar R 2019 *J. Comput. Phys.* **388** 624–54
- [22] Niknezhad M, Chanrion O, Köhn C, Holboll J and Neubert T 2021 *Plasma Sources Sci. Technol.* **30** 045012
- [23] LeVeque R 2002 *Finite Volume Methods for Hyperbolic Problems (Cambridge Texts in Applied Mathematics)* (Cambridge University Press)
- [24] Trottenberg U, Oosterlee C W and Schüller A 2001 *Multigrid (Texts in Applied Mathematics. Bd)* vol 33 (Academic) with contributions by Brandt A., Oswald P. and Stüben K.
- [25] Teunissen J and Ebert U 2018 *Comput. Phys. Commun.* **233** 156–66
- [26] Dhali S K and Williams P F 1987 *J. Appl. Phys.* **62** 4696–707
- [27] Luque A, Ratushnaya V and Ebert U 2008 *J. Phys. D: Appl. Phys.* **41** 234005
- [28] Bagheri B et al 2018 *Plasma Sources Sci. Technol.* **27** 095002
- [29] Zhelezniak M B, Mnatsakanian A K and Sizykh S V 1982 *High Temp. Sci.* **20** 357–62
- [30] Luque A, Ebert U, Montijn C and Hundsdoerfer W 2007 *Appl. Phys. Lett.* **90** 081501
- [31] Bourdon A, Pasko V P, Liu N Y, Célestin S, Ségur P and Marode E 2007 *Plasma Sources Sci. Technol.* **16** 656–78
- [32] Luque A, Brau F and Ebert U 2008 *Phys. Rev. E* **78** 016206
- [33] Luque A Front1D. jl: One-dimensional model of a streamer front (available at: <https://github.com/aluque/Front1D.jl>)
- [34] Naidis G V 2009 *Phys. Rev. E* **79** 057401
- [35] Kochkin P O, Nguyen C V, van Deursen A P J and Ebert U 2012 *J. Phys. D: Appl. Phys.* **45** 425202
- [36] Briels T M P, van Veldhuizen E M and Ebert U 2008 *J. Phys. D: Appl. Phys.* **41** 234008
- [37] Wang Z, Dijcks S, Guo Y, van der Leege M, Sun A, Ebert U, Nijdam S and Teunissen J 2022 arXiv:2208.07279
- [38] Dijcks S, der Leege M v and Nijdam S 2023 *Plasma Sources Sci. Technol.* **32** 045004
- [39] Nijdam S, van de Wetering F M J H, Blanc R, van Veldhuizen E M and Ebert U 2010 *J. Phys. D: Appl. Phys.* **43** 145204