Available online at www.sciencedirect.com

ScienceDirect

Journal of Computational Physics 227 (2008) 6574–6590

JOURNAL OF COMPUTATIONAL PHYSICS

www.elsevier.com/locate/jcp

Numerical simulation of filamentary discharges with parallel adaptive mesh refinement

S. Pancheshnyi [a,*], P. Ségur [a], J. Capeillère [a], A. Bourdon [b]

[a] *LAPLACE, Université de Toulouse, CNRS, 118, Route de Narbonne, 31062 Toulouse cedex 9, France*
[b] *EM2C, École Centrale Paris, CNRS, Grande Voie des Vignes, 92295 Châtenay-Malabry, France*

Received 30 September 2007; received in revised form 14 March 2008; accepted 15 March 2008
Available online 24 March 2008

Abstract

Direct simulation of filamentary gas discharges like streamers or dielectric barrier micro-discharges requires the use of an adaptive mesh. The objective of this paper is to develop a strategy which can use a set of grids with suitable local refinements for the continuity equations and Poisson's equation in 2D and 3D geometries with a high-order discretization. The advantages of this approach are presented with a filamentary discharge simulation in plane–plane geometry in nitrogen within the diffusion–drift approximation.

PACS: 52.35.Mw; 52.80.Hc

Keywords: Streamer; Filamentary discharge; Adaptive mesh refinement; Paramesh

---

1. Introduction

It is generally recognized that filamentary discharges (also called streamers) are ionizing waves which occur as a consequence of the high electric field induced by fast variation of the net charge density ahead of an electron avalanche with a large amplification. These ionizing waves propagate in a manner similar to a shock wave in hydrodynamics even if the basic mechanisms are completely different. Usually filamentary discharges appear at high values of the product of p, the gas pressure by d, the gap length, $pd \geqslant (1-3)$ atm mm [1]. Furthermore, as these discharges play a very important role for various industrial applications (ozone generation, gas depollution, surface treatment, sterilisation, etc.), many studies have been made for a long time to improve their understanding.

The possibilities of experimental investigation of filamentary discharges are very limited and the numerical modelling is an important tool to facilitate the knowledge of these discharges [2]. In most works, the motion of electrons, ions and excited molecules is governed by continuity equations coupled to Poisson's equation: this is

---

[*] Corresponding author. Tel.: +33 5 61 55 60 54; fax: +33 5 61 55 63 32.
E-mail address: sergey.pancheshnyi@laplace.univ-tlse.fr (S. Pancheshnyi).

0021-9991/$ - see front matter © 2008 Elsevier Inc. All rights reserved.
doi:10.1016/j.jcp.2008.03.020

the so called drift–diffusion or hydrodynamic approximation. Nowadays it is possible to simulate such a discharge even on standard PC computers. However, to solve accurately this set of equations, many drawbacks have to be overcome.

Due to the fast space and time variation of the main physical quantities (electron and ion densities, electric field, etc.), a specific choice of the numerical method has to be made. This numerical method must have two main properties. First, it has to be sufficiently accurate to describe the fast space variation of charged particles. Second, it must avoid to induce negative values of densities.

It is now well-known that both properties cannot be satisfied with a standard numerical scheme [3]. With a scheme of first-order accuracy (e.g., upwind scheme), the solution is strictly positive, but a strong numerical diffusion occurs. With a scheme of higher accuracy (e.g., Lax-Wendroff scheme), the numerical diffusion is less but the solution is no more positive. It is possible to use a high accuracy scheme by introducing special numerical techniques aimed to control the positivity of the solution. These techniques (e.g., FCT) were first developed in hydrodynamic of neutral gases [4] and later introduced for streamer modelling [5]. Currently, FCT is not the unique technique employed and new modern methods based on flux limiters are also used (e.g., MUSCL scheme [6]). In the following, for reasons which will be explained in Section 3, we will use the so called QUICKEST scheme in connection with universal limiter developed by Leonard [7,8] which has been recently applied to streamer modelling [9,10].

The most common way to solve numerically a partial differential equation is to divide the physical domain of interest into a certain number of cells (or grids). Often, these grids are uniformly distributed over the whole physical domain that is, obviously, has a pure efficiency for non-uniform object simulation and often are not accessible for three-dimensional simulations. Another possibility is to use an adaptive mesh refinement (AMR) method.

The principle of AMR methods is to adapt the distribution of grids in order to increase the number of cells in regions of fast variation and to decrease this number in regions of small variation. To do that, many techniques are available in the literature (see, e.g. [11]). They roughly belong to two large classes. In the first class, the total number of grids is maintained constant and the location of grids changes in order to maintain the accuracy of the solution everywhere. Recently, this so called moving mesh method was applied for two-dimensional streamer simulation [10], however the refinement was only made along the main axis of the discharge in one dimension.

In the second class of methods, the initial uniform grid is successively divided into a set of smaller subgrids which are refined until the chosen accuracy of the solution is reached. In this case, the total number of points is not fixed. Up to now, just a few authors developed adaptative mesh refinement for numerical modelling of streamer propagation. So, a two-dimensional simulation based on logarithmic distribution of points along the main axis of the discharge was used for the first time in [12]. The authors of [13–15] used a concept of an adaptative window in which the number of grids is maintained constant. The points at the front and behind the windows are added or removed as the streamer propagates. A completely different technique was used in [16]. Starting from a uniform grid and introducing a specific weight function, it was possible to construct a non-uniform grid in order to keep the variation of the weighted first and second derivatives of net charge density between two points. Real adaptative meshes in two dimensions were used in [17,18] on non-structured grids with the use of a finite element method. Adaptative mesh refinement techniques were recently used in [19] in two dimensions and in [20] in two and three dimensions. In these cases, structured meshes were used.

Contrary to the moving mesh method, many free libraries corresponding to the second class of methods can be found. The main objectives of these libraries are to manage the creation of the grids, to build and maintain the tree-like structure which tracks the spatial relationships between blocks, to distribute the blocks amongst the available processors and to handle all inter-block and inter-processor communication. They usually can distribute the blocks in ways which maximize block locality and so minimize inter-processor communications. In this work the PARAMESH library [21,22] will be used. This library can be applied to very different physical situations in one, two or three dimensions and includes a set of tools to facilitate the parallelization of a given code with the help of standard MPI parallel library.

In the following, we will first describe the physical model to be considered, together with the principle of the AMR refinement and the numerical method we used. In a second part we will show the results obtained for a filamentary discharge in nitrogen both in two and three dimensions and we will emphasize the importance of the use of an AMR method to increase the efficiency of our numerical modelling.

2. The physical model

2.1. Master equations

In this present paper the space and time variation of charged particles is described by the continuity equation (1) within the drift–diffusion approximation (2). This is a standard formulation which is widely used for simulation of high-pressure discharges (see, e.g., [23–25] and references therein)

$$\frac{\partial n_s(\vec{r},t)}{\partial t} + \mathrm{div}\vec{j}_s(\vec{r},t) = S_s(\vec{r},t) \tag{1}$$

$$\vec{j}_s(\vec{r},t) = \vec{v}_s(\vec{r},t)n_s(\vec{r},t) - D_s(\vec{r},t)\vec{\nabla}n_s(\vec{r},t) \tag{2}$$

where n_s represents the density of each type of charged particles (electrons and ions). In (2), the flux $\vec{j}_s$ consists of an advective part with drift velocity $\vec{v}_s$ and a diffusion part with coefficient D_s.

The S_s term in the right part of the balance equation corresponds to various plasma-chemical processes like ionization, recombination, electron attachment, photoionization, etc. For the simple case of direct ionization by electron impact, the source term is expressed as $S_s = k_i n_e N$ in terms of the constant rate k_i, the electron density n_e and the gas density N. All constant rates and transport parameters are usually expressed as a function of the reduced electric field E/N according to the local field approximation. The validity of this approximation for streamer simulation has been analyzed in [26].

The electric field $\vec{E}(\vec{r},t)$ can be expressed as a sum of the external Laplacian field $\vec{E}^{\mathrm{L}}(\vec{r})$ and of the space charge field $\vec{E}^{\mathrm{chg}}(\vec{r},t)$, i.e. $\vec{E} = \vec{E}^{\mathrm{L}} + \vec{E}^{\mathrm{chg}}$. The module $|\vec{E}^{\mathrm{L}}|$ is equal, in a plane–plane geometry with interelectrode gap distance L_{gap} to $|U_{\mathrm{gap}}|/L_{\mathrm{gap}}$ and the direction of $\vec{E}^{\mathrm{L}}$ is from the anode to the cathode.

In the present formulation, we neglect by the magnetic field intensity and its time dependence due both to low current density (10^{-2}–10^{2} A/cm^2) and to low speed of propagation (10^{7}–10^{8} cm/s) typical for streamer discharges. In this case, the Maxwell equations are satisfied implicitly and the second part of the total electric field $\vec{E}^{\mathrm{chg}}$ is obtained through the distribution of potential φ as $\vec{E}^{\mathrm{chg}}(\vec{r},t) = -\vec{\nabla}\varphi(\vec{r},t)$ and obeys the following Poisson's equation:

$$\Delta\varphi(\vec{r},t) = -\frac{1}{\varepsilon_0}\sum_s q_s n_s(\vec{r},t) \tag{3}$$

Here, q_s is the charge of each type of species and ε_0 is the electric permittivity of free space. In these conditions, Poisson's equation (3) is coupled to the system of equations (1) and (2).

The external electrical current J_{gap} can be obtained by integration of the first Maxwell's equations over the whole computational domain Ω. When the applied voltage U_{gap} is time independent, we obtain the following relation

$$J_{\mathrm{gap}}(t) = \frac{1}{U_{\mathrm{gap}}}\sum_s q_s \int_V \vec{j}_s(\vec{r},t)\vec{E}^{\mathrm{L}}(\vec{r})\,\mathrm{d}^3\vec{r} \tag{4}$$

which is also known as Sato's equation [27].

2.2. Geometry and boundary conditions

The computational domain Ω is limited by two parallel electrodes separated by a distance L_{gap}. The simple parallel geometry allows us to define a bound Σ which coincides with the electrodes and a bound Γ which represents all bounds perpendicular to the electrodes.

The boundary conditions for electrode surfaces Σ are zero potential $\varphi|_{\Sigma} = 0$ on electrode surfaces and linearly extrapolated density of species n_s from the gap space inside the electrodes. The latter is required for high-order approximation of the transport term in the vicinity of the electrodes.

Emission of particles from solid surfaces in secondary processes (electron–ion emission, photoemission, field emission, etc.) is not taken into account in the present model. In these conditions, we assume that the flux j_s of electrons and negative (respectively positive) ions leaving the cathode (respectively the anode) is equal to zero.

The boundary conditions along the external bound Γ are the symmetry conditions $\partial/\partial\vec{r}|_\Gamma = 0$ for the electric potential φ and densities of species n_s, where $\vec{r}$ is a normal to surface Γ.

3. Numerical implementation

In the following, system (1)–(3) is solved over structured meshes by using the control volume method and the fractional step method. We assume that all the data which constitute the solution are located at the center point of grid cells except the vector components of the electric field which are associated with the center of grid cell faces.

The idea of the fractional step method is to decompose equation (1) in a certain number of equations which are successively solved over an elementary time step. For example, in a two-dimensional Cartesian geometry Equation (1) becomes

$$\frac{\partial n(x,y,t)}{\partial t} + \frac{\partial j_x(x,y,t)}{\partial x} = 0 \tag{5}$$

$$\frac{\partial n(x,y,t)}{\partial t} + \frac{\partial j_y(x,y,t)}{\partial y} = 0 \tag{6}$$

$$\frac{\partial n(x,y,t)}{\partial t} = S(x,y,t) \tag{7}$$

The interest of the fractional step method is that an unique complex equation is splitted into a set of simpler equations. In our case, Eqs. (5) and (6) are one-dimensional convection–diffusion equations and (7) is a simple differential equation with respect to time. The latter Eq. (7) will be solved with a standard fourth-order Runge–Kutta method. The numerical resolution of Eqs. (5) and (6) is much more complicated and we will give below the details of our numerical procedure in one dimension for Cartesian and Cylindrical geometries.

The general one-dimensional convection–diffusion equation can be then written under the following form

$$\frac{\partial n(x,t)}{\partial t} + \frac{1}{x^p}\frac{\partial(x^p n(x,t) v^{\mathrm{tot}}(x,t))}{\partial x} = 0 \tag{8}$$

where $p = 0$ ($x^p = 1$) for Cartesian geometry and for axial direction in Cylindrical geometry and $p = 1$ ($x^p = x$) for radial direction in Cylindrical geometry.

In Eq. (8), the $v^{\mathrm{tot}}(x,t)$ mean velocity of particles including convection and diffusion is defined as

$$v^{\mathrm{tot}}(x,t) = v(x,t) - \frac{D(x,t)}{n(x,t)}\frac{\partial n(x,t)}{\partial x} \tag{9}$$

The use in Eq. (8) of the total velocity $v^{\mathrm{tot}}(x,t)$ makes this equation nonlinear with respect to the density $n(x,t)$ but it allows to associate convection and diffusion in the numerical treatment. The use of this total velocity is valid due to the weakness of the diffusion term with respect to the convection term in the case of streamer propagation which makes the drift velocity dominant in Eq. (9).

3.1. Finite volume discretization of the transport equation

The basic principle of the control volume method is to maintain the conservative property of Eq. (8) over every volume element. To do that in 1D, Eq. (8) is integrated over cell element i in the range $\Delta x = x_{i+1/2} - x_{i-1/2}$ and time interval $\Delta t = t^{m+1} - t^m$, that finally gives

$$\bar{n}_i^{n+1} = \bar{n}_i^n - \frac{1}{A_i}(\Phi(x_{i+1/2}, t^m, \Delta t) - \Phi(x_{i-1/2}, t^m, \Delta t)) \tag{10}$$

where $\bar{n}$ is the mean value of the density over the control volume, $\Phi(x,t,\Delta t)$ is the flow function, and A_i is the normalization factor defined as

$$A_i = \int_{x_{i-1/2}}^{x_{i+1/2}} x^p \,\mathrm{d}x \tag{11}$$

$$\bar{n}_i = \frac{1}{A_i}\int_{x_{i-1/2}}^{x_{i+1/2}} x^p n(x,t)\,\mathrm{d}x \tag{12}$$

$$\Phi(x,t^m,\Delta t) = \int_{t^m}^{t^m+\Delta t} x^p n(x,t) v^{\mathrm{tot}}(x,t)\,\mathrm{d}t \tag{13}$$

The function $\Phi(x,t,\Delta t)$ characterizes the number of particles which crosses the interface located at x during the time step Δt. We can note that, as these fluxes represent the input and output of particles at the interface of the control volume, they only strictly depend on the value of the densities at these interfaces and not on the mean value inside. It is then necessary to express the $\Phi(x_{i\pm1/2},\Delta t)$ as a function of $\bar{n}_i$.

3.1.1. High-order approximation of the flow function

As already mentioned in the introduction, the use of a high-order scheme is very necessary to solve the continuity equation (8). Many different techniques can be found in the literature. Most of these techniques are limited to second-order in space and to uniform distribution of points. For a few years, the QUICKEST scheme initially introduced in [28] has been applied to streamer modelling [9,10]. To do that, we have used a procedure also developed by Leonard and called NIRVANA [8]. The first interest of NIRVANA approach is that the accuracy of the numerical scheme can be easily increased, as it is only related to the order of an interpolation polynomial over space. The second interest is that this scheme can be easily written for Cartesian and Cylindrical geometries.

If we consider that the velocity is locally constant in both cells near an interface $i+1/2$ during time step Δt, the total flux $\Phi_{i+1/2}(t^m,\Delta t)$ (13) through this interface can be rewritten for $x = x_{i+1/2}$ as

$$\Phi_{i+1/2}(t^m,\Delta t) = v^{\mathrm{tot}}_{i+1/2}(t^m)\int_{t^m}^{t^m+\Delta t} x^p n(x,t)\,\mathrm{d}t \tag{14}$$

Furthermore, the product $x^p n(x,t)$ is constant along the characteristic line defined by $x(t) = x_{i+1/2} - v^{\mathrm{tot}}_{i+1/2}(t^m)(t-t^m)$. Using latter and changing the variable t to x by the equation of the characteristic line introduced above, we have finally

$$\Phi_{i+1/2}(t^m,\Delta t) = \int_{x-v^{\mathrm{tot}}_{i+1/2}(t^m)\Delta t}^{x} x^p n(x,t^m)\,\mathrm{d}x \tag{15}$$

Let us now introduce a discrete function $\Psi_{i\pm1/2}(t)$ defined on the interfaces of the control volume i at time moment t by the following recurrent relation

$$\Psi_{i+1/2}(t) = \Psi_{i-1/2}(t) + A_i\bar{n}_i(t) \tag{16}$$

As $A_i\bar{n}_i$ product must be positive, the set of $\{\Psi_{i\pm1/2}(t)\}$ values constitutes a strictly non-diminishing set of data. On this set we can define a continuous function $\Psi(x,t)$ which approximates the discrete set $\Psi_{i\pm1/2}(t)$ for every interface $x = x_i \pm 1/2$. Moreover, the derivative of $\Psi(x,t)$ function depends on $n(x,t)$ solution of the initial equation (8) according to definition (12), i.e.

$$x^p n(x,t) = \frac{\mathrm{d}\Psi(x,t)}{\mathrm{d}x} \tag{17}$$

Using the latter with (15), we can express the total flux $\Phi_{i+1/2}(t^m,\Delta t)$ through the introduced function $\Psi(x,t)$

$$\Phi_{i+1/2}(t^m,\Delta t) = \Psi(x_{i+1/2},t^m) - \Psi(x - v^{\mathrm{tot}}_{i+1/2}(t^m)\Delta t, t^m) \tag{18}$$

Then, master equation (10) can be expressed in term of the $\Psi(x,t)$ function as well

$$\bar{n}_i^{m+1} = \bar{n}_i^m - \frac{1}{A_i}[(\Psi(x_{i+1/2},t^m) - \Psi(x^+,t^m)) - (\Psi(x_{i-1/2},t^m) - \Psi(x^-,t^m))] \tag{19}$$

where $x^\pm = x_{i\pm1/2} - v^{\mathrm{tot}}_{i\pm1/2}(t^m)\Delta t$, respectively.

Obviously, $\Psi(x^{\pm})$ can be obtained by using an interpolation function based on the tabulated data $\Psi_{i\pm1/2}$. High-order polynomials can be used to carry out this interpolation. In our calculations, a third-order polynomial was chosen [9]. The numerical scheme obtained in these conditions is an extension to non-uniform meshes of the original QUICKEST scheme [28].

3.1.2. Flux limiter

As we use a high-order polynomial to calculate $\Psi(x)$ function from the knowledge of tabulated data $\Psi_{i\pm1/2}$ defined by (16), it is necessary to control the validity of this approximation. As a high-order polynomial may exhibit spurious oscillations for fast variations of the solution, it is necessary to detect and avoid these oscillations.

According to [8], this can be done by limiting the value of $\Psi(x)$ interpolated inside a cell i, i.e. for any $x \in [x_{i-1/2}, x_{i+1/2}]$, by three straight lines defined by

$$\begin{aligned}
\Psi^0 &= \Psi_{i-1/2} + (\Psi_{i+1/2} - \Psi_{i-1/2})\frac{x - x_{i-1/2}}{h}, \quad h = x_{i+1/2} - x_{i-1/2}\\
\Psi^- &= \Psi_{i-3/2} + (\Psi_{i-1/2} - \Psi_{i-3/2})\frac{x - x_{i-3/2}}{h}, \quad h = x_{i-1/2} - x_{i-3/2}\\
\Psi^+ &= \Psi_{i+3/2} - (\Psi_{i+1/2} - \Psi_{i+3/2})\frac{x - x_{i+3/2}}{h}, \quad h = x_{i+3/2} - x_{i+1/2}
\end{aligned} \tag{20}$$

The limited value $\Psi^*(x)$ is determined by the following relationships

$$\widetilde{\Psi} = \max(\min(\Psi(x), \max(\Psi^0, \Psi^-)), \min(\Psi^0, \Psi^-)) \tag{21}$$

$$\Psi^*(x) = \max(\min(\widetilde{\Psi}, \max(\Psi^0, \Psi^+)), \min(\Psi^0, \Psi^+)) \tag{22}$$

and then used in Eq. (19) instead of $\Psi(x^{\pm})$.

3.2. Poisson's equation

Usually, Poisson's equation is solved by using a centered second-order scheme. To improve the accuracy of the calculation of the electric field, we adopt in this work a fourth-order scheme developed in [29]. This high-order compact scheme only need information from the nearest grid points and therefore do not get into conflict with one-layer boundary conditions for Poisson's equation.

For example, in three-dimensional calculations, this scheme only uses a 19-point stencil for the left-hand side of Eq. (3). In addition, the right-hand-side is also modified. It also takes into account nearest-neighbor source terms, so one can write for a cell (i, j, k)

$$\sum_{l,m,n=-1\ldots+1} L^{l,m,n}_{i,j,k}\varphi_{i+l,j+m,k+n} = \sum_{l,m,n=-1\ldots+1} Q^{l,m,n}_{i,j,k}\rho_{i+l,j+m,k+n} \tag{23}$$

Here, ρ represents the total right-hand-side source in the original equation (3).

Matrix coefficients for 3D Cartesian geometry are position independent and values $L^{l,m,n}$ and $Q^{l,m,n}$ are presented for a uniform mesh in [29].

The same matrix coefficients for a curvilinear geometry can be computed with the control volume method. Below, we present these matrixes for 2D cylindrical geometry (r, z) with h uniform spatial step and $r_{i,j}$ radius-center of the cell (i, j)

$$L^{l,m}_{i,j} = h^{-2}\left(\frac{1}{2}\begin{bmatrix}1 & 1 & 1\\ 1 & -8 & 1\\ 1 & 1 & 1\end{bmatrix} + \frac{1}{4}\begin{bmatrix}-1 & 0 & 1\\ -1 & 0 & 1\\ -1 & 0 & 1\end{bmatrix}\frac{h}{r_{i,j}}\right) \tag{24}$$

$$Q^{l,m}_{i,j} = \frac{1}{8}\begin{bmatrix}0 & 1 & 0\\ 1 & 8 & 1\\ 0 & 1 & 0\end{bmatrix} + \frac{1}{12}\begin{bmatrix}0 & 0 & 0\\ -1 & 0 & 1\\ 0 & 0 & 0\end{bmatrix}\frac{h}{r_{i,j}} \tag{25}$$

The boundary coefficients for $i = 1$ cells corresponding to $r_{i,j} = h/2$ are

$$L_{1,j}^{l,m} = \frac{h^{-2}}{8}\begin{bmatrix} 0 & 5 & 8 \\ 0 & -34 & 8 \\ 0 & 5 & 8 \end{bmatrix}, \quad Q_{1,j}^{l,m} = \frac{1}{24}\begin{bmatrix} 0 & 3 & 0 \\ 0 & 24 & 7 \\ 0 & 3 & 0 \end{bmatrix} \tag{26}$$

Bi-conjugate gradient stabilized method is used here to solve the resulting system of linear equations [30]. It is an iterative algorithm and we consider that the solution is obtained as soon as the maximum of residual defined as $\mathrm{Res} = \max|\Delta\varphi_{i,j,k} - \rho_{i,j,k}|$ becomes less than a value $\mathrm{Res}_{\max} = \delta_\rho \max|\rho_{i,j,k}| + \rho_0$. The values $\delta_\rho = 10^{-5}$ and $\rho_0 = 1$ are used in the present paper and lead typically to 50–100 iterations. In additional 2D tests we have obtained just minor variations of the solution using a much higher constraint, namely $\delta_\rho = 10^{-10}$. This confirms the validity of the solution with the aforementioned values.

As soon as the distribution of potential is known, the components of space charge electric field are then computed by using a second-order central discretization of $\vec{E}^{\mathrm{chg}}$ on the interface of each couple of cells. For example, along the axis with index i it gives

$$E_{i+1/2,j,k}^{\mathrm{chg}} = -h^{-1}(\varphi_{i+1,j,k} - \varphi_{i,j,k}) \tag{27}$$

The face-centered electric field then becomes: $E_{x,z} = E_{x,z}^{\mathrm{chg}}$ and $E_y = E_y^{\mathrm{chg}} + E^{\mathrm{L}}$. Finally, the total cell-centered electric field is computed as

$$|\vec{E}_{i,j,k}| = \frac{1}{2}\left((E_{i+1/2,j,k} + E_{i-1/2,j,k})^2 + (E_{i,j+1/2,k} + E_{i,j-1/2,k})^2 + (E_{i,j,k+1/2} + E_{i,j,k-1/2})^2\right)^{1/2} \tag{28}$$

3.3. Adaptive mesh refinement (AMR) and the PARAMESH library

The PARAMESH software used in this work was developed at the NASA Goddard Space Flight Center and Drexel University under NASA's HPCC and ESTO/CT projects and under Grant NNG04GP79G from the NASA/AISR project [21]. It is a package of Fortran 90 subroutines designed to provide an application developer with an easy route to extend an existing serial code which uses a logically Cartesian structured mesh into a parallel code with AMR. A PARAMESH user guide can be found on the web site [22].

Basically, the computational domain is covered with a hierarchy of numerical sub-grids. These sub-grids form the nodes of a tree data-structure and are distributed among the processors. When a higher spatial resolution is required at some location, the higher resolution sub-grid covering that point spawns child sub-grids (2 in 1D, 4 in 2D and 8 in 3D), which together cover the line, area or volume of their parent, but now with twice its spatial resolution. The block with the highest refinement level at a given physical location is referred to as a "leaf block".

All sub-grids have identical logical structure (i.e. the same number of grid points in each dimension, the same aspect ratios, the same number of guard cells, etc.). They differ from each other in their sizes and locations, and in their lists of neighbors, parents and children. The package assumes that the application will use logically Cartesian grids (i.e. grids that can be indexed i, j, k) for 1D, 2D and 3D.

The package provides routines which perform all the communication required between sub-grids and between processors. It manages the refinement and de-refinement processes, and can balance the workload by reordering the distribution of blocks among the processors. It also provides routines to enforce conservation laws at the interfaces between grid blocks at different refinement levels.

Then, to use PARAMESH, the user only needs to provide the code to advance the solution on a generic grid block with uniform space resolution.

3.3.1. Practical mesh refinement

In our implementation, a block consists of eight cells in every direction and the decision whether a fine or a coarse grid should be used on a certain region is made with respect to $n_i/n_{i\pm1}$ electron density and electric field $E_i/E_{i\pm1}$ ratios to satisfy the following conditions

$$n_i/n_{i\pm1} < C_n^{\mathrm{ref}} \quad \text{and} \quad E_i/E_{i\pm1} < C_E^{\mathrm{ref}} \tag{29}$$

The above mentioned values were fixed in presented simulations to $C_n^{\mathrm{ref}} = 1.40$ and $C_E^{\mathrm{ref}} = 1.15$ for every cell with non-zero values. We have checked that these values are sufficient to diminish the mesh numerical diffusion for numerical algorithms used in the model, at least for the task presented in the next sections.

Nevertheless, simple relations (29) can lead to infinite mesh refinement so a hard limit $h \geqslant h_{\min}$ is used for the grid step.

Conservative first- and second-order interpolations were used for data restriction and prolongation, respectively. Interpolation occurs during prolongation or restriction operations which fills either newly created blocks with data in case of prolongation, or when filling guard cells at block boundaries next to less refined neighbor blocks. For prolongation, the ULTIMATE limiter [7,8] is used to avoid oscillations of the solution in the same way as it is used in transport solver.

3.4. Flowchart of the code

The overall algorithm can be summarized as follows.

We start at some time moment with a known distribution of species and set of grids. The electric field induced by this charge distribution is computed on the initial grid. The known electron and field distributions allow us to refine/derefine the same regions according to criteria described in Section 3.3.1. The procedure repeats as many times as it is needed to get a stable set of grids.

The Δt time step for a time integration is then set in such a way that the stability conditions are met for every cell of the grid. A Courant–Friedrichs–Lewy restriction with max Courant number $Cr_{\max} = 0.1$ is used here: $\Delta t \leqslant Cr_{\max} \times h/v^{\mathrm{tot}}$. We also check that the relative density changes caused by source term in balance equation (1) do not exceed a value of $K = 0.1$ for every cell with non-negligible densities, i.e. $\Delta t \leqslant K \times n/S_s$.

The time step, defined in this way, allows us to update species densities, recalculate the potential and compute new electric fields. So, if we start at a given time moment t_0 with a solution $\{F\}^{t_0}$ (representing, in fact, the species densities, electric potential and fields, i.e. the solutions of equations (1)–(3)), we obtain a first predicted data set $\{\widetilde{F}\}^{t_1}$ at time moment $t_1 = t_0 + \Delta t$. Using this solution, we again compute a new predicted data set $\{\widetilde{F}\}^{t_2}$ corresponding to time moment $t_2 = t_1 + \Delta t$.

As already mentioned, in a multi-dimensional case, the transport equation is split into several one-dimensional equations which are solved by the schemes described above. To improve the accuracy of the splitting method, an alternation of the order between $X - Y - (Z)$ and $(Z) - Y - X$ is used in two successive steps.

Both predictions can be used to make second-order time integration using the trapezoidal rule

$$\{F\}^{t_0+\Delta t} = \{F\}^{t_0} + \frac{1}{2}(\{\widetilde{F}\}^{t_1} + \{\widetilde{F}\}^{t_2}) \tag{30}$$

We note that each time step integration finishes with electric current computation and every 5 steps the mesh adjustment function is called. The latter is computationally not very expensive and is sufficient for mesh optimization.

4. Simulation of a streamer development in nitrogen

An example of streamer propagation in pure nitrogen in $L_{\mathrm{gap}} = 0.5$ cm plane–plane gap in a uniform external field of 196 Td is presented below for both polarities of the voltage. The simulated conditions are identical to those used in paper [23]. This paper was chosen as a well-known "reference" paper widely used in tests for streamer simulations.

The model assumes a cylindrical axisymmetric geometry in 2D case $(r, z) = (0, r_{\max}) \times (0, L_{\mathrm{gap}})$ and full Cartesian in 3D $(x, y, z) = (-x_{\max}, x_{\max}) \times (0, L_{\mathrm{gap}}) \times (-z_{\max}, z_{\max})$. The values of $r_{\max}$, $x_{\max}$ and $z_{\max}$ are chosen below to be $L_{\mathrm{gap}}/2$.

We note that the algorithm used in PARAMESH builds a hierarchy of sub-grids with a doubled space step. We use below values of 10^{-3}, 10^{-4} and 10^{-5} cm which, in fact, represent the values $L_{\mathrm{gap}}/2^9 \simeq 0.98 \times 10^{-3}$, $L_{\mathrm{gap}}/2^{12} \simeq 1.22 \times 10^{-4}$ and $L_{\mathrm{gap}}/2^{15} \simeq 1.52 \times 10^{-5}$ cm, respectively. We have checked that a coarser resolution (e.g., 10^{-2} cm) leads to an artificial diffusive solution.

4.1. The conditions

In this work, two species are taken into account, namely electrons and N_2^+ ions. A detailed analysis confirms the presence of a wide range of charged species, radicals and other active species in streamer plasma (see, e.g. [31]). However, our objective in the present paper is not focused on the simulation of the exhaustive plasma chemistry.

It is assumed that v the drift velocity and k_i the ionization rate are empirically determined functions of the reduced electric field E/N [23]

$$v_e = 1.0 \times 10^5 E/N \tag{31}$$

$$v_i = 9.0 \times 10^2 E/N \tag{32}$$

$$k_i = 1.65 \times 10^{-11} \exp\left(-\frac{745}{E/N}\right) E/N \tag{33}$$

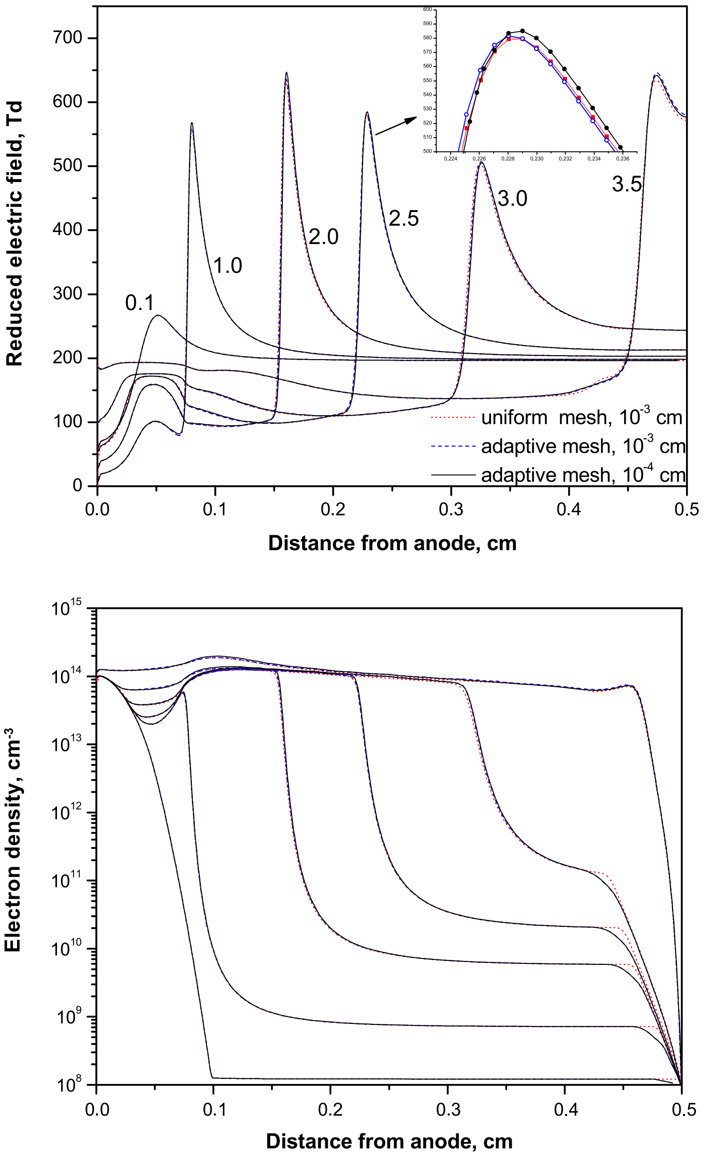


Fig. 1. On-axis electric field and electron density profiles for cathode-directed streamer at time moments of 0.1, 1.0, 2.0, 2.5, 3.0 and 3.5 ns. Results of 2D simulation with a uniform mesh of 10^{-3} cm and adaptive meshes of 10^{-3} and 10^{-4} cm.

Here, E/N, v and k_i are presented in Td ($1\ \mathrm{Td} = 10^{-17}\ \mathrm{V\,cm^2}$), cm/s and $\mathrm{cm^3/s}$, respectively. We assume a constant gas density $N = 2.65 \times 10^{19}\ \mathrm{cm^{-3}}$.

The electron diffusion coefficient $D_e = 2.0 \times 10^3\ \mathrm{cm^2\ s^{-1}}$ is used for both transverse and longitudinal directions. The ion diffusion is not taken into account here according to the assumption made in [23].

Free electrons are necessary for streamer initiation and propagation. These free electrons can be produced in different ways [25]. However, in the present work, as it is assumed in reference paper [23], the discharge is initiated by a quasi-neutral preionization Gaussian spot with $n_{\max} = 10^{14}\ \mathrm{cm^{-3}}$ peak density and $\sigma_{x,z} = 2.1 \times 10^{-2}$, $\sigma_y = 2.7 \times 10^{-2}$ cm spatial dispersion. This spot is initially at the grounded electrode, i.e. $x = y = z = 0$.

The photoionization and photoemission are not considered in the present case according to the conditions used in [23]. Instead of that, a permanent background preionization at the level $10^8\ \mathrm{cm^{-3}}$ is used here, i.e. the electron density is controlled at every time step and we restore the minimal electron density in the cells where it is necessary. Simultaneously, we add the same amount of ions to the cells to keep the space charge density unchanged.

In fact, this approach of minimum electron density is only important for near-cathode region and replaces the phenomena of secondary electron emission on cathode.

With the initial conditions given above, as soon as the voltage is applied, a streamer starts to propagate from the preionization spot toward the opposite electrode. A streamer propagating from anode to cathode is called the cathode-directed (or positive) streamer and a streamer moving in the opposite direction, i.e. from cathode to anode, is the anode-directed (or negative) streamer.

4.2. Cathode-directed streamer

Axial distribution of electric field and electron density during the discharge development are presented in Fig. 1 at various times. At the initial time moment, the preionization spot is at the anode. Three sets of results were obtained in 2D geometry with a uniform grid of 10^{-3} cm and adaptive meshes with 10^{-3} and 10^{-4} cm limit of the smallest mesh.

Two main conclusions can be drawn here. First, the solutions obtained with uniform and adaptive meshes with a reasonable refinement criterion are in a good agreement for the same space resolution. Second, a convergence of the solution is observed using adaptive meshes with different space resolutions. We will show that this conclusion is only correct for a space resolution exceeding some minimum value which is determined by task parameters. Nevertheless, only minor differences can be observed between the solutions and we conclude that a space resolution of 10^{-3} cm is small enough for these test case simulations.

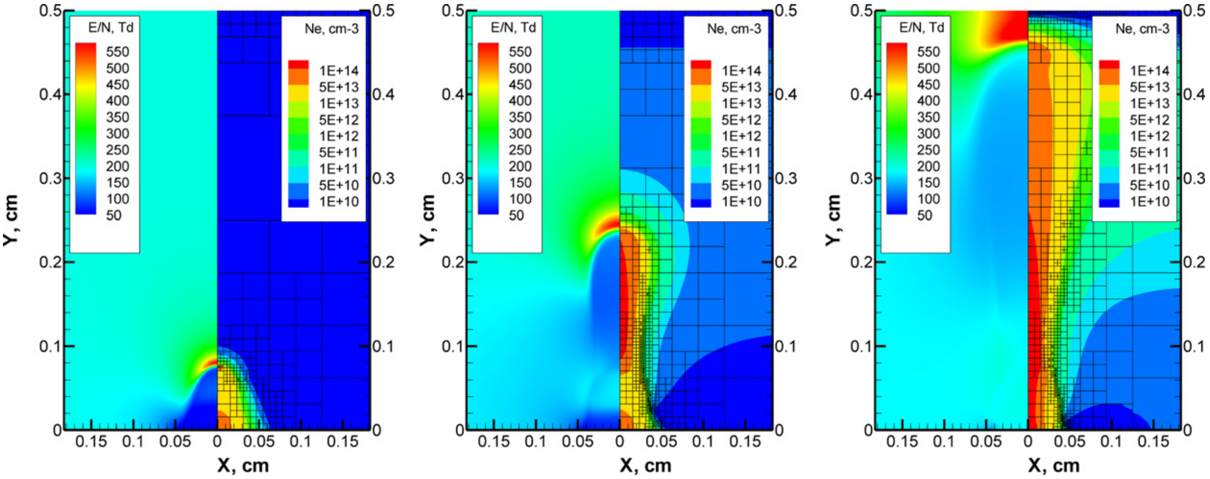


Fig. 2. Electric field and electron density distributions for cathode-directed streamer at time moments of 1.0, 2.6 and 3.5 ns. Results of 2D simulation with adaptive mesh of 10^{-4} cm. Blocks of 8 × 8 cells are shown.

The 2D structure of the streamer during its propagation is presented in Figs. 2 for a set of time moments. The results were obtained with the finest mesh of space step 10^{-4} cm. The mesh structure can be traced in the figures where the instantaneous block structure of 8 × 8 cells is presented.

A high mesh resolution is automatically used in regions with a steep space gradient and, according to the results, the maximum gradient is formed at the external bound of initial preionization spot near anode. The

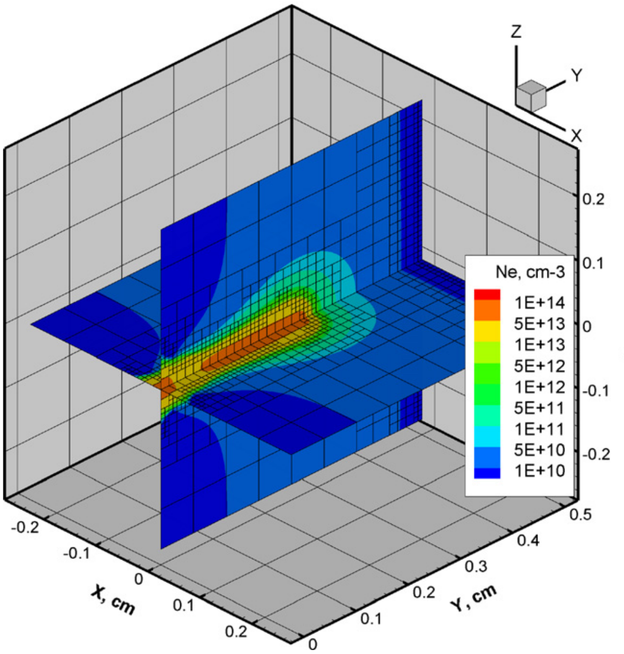


Fig. 3. Electron density distributions for cathode-directed streamer at time moment of 2.6 ns. Results of 3D simulation with adaptive mesh of 10^{-3} cm.

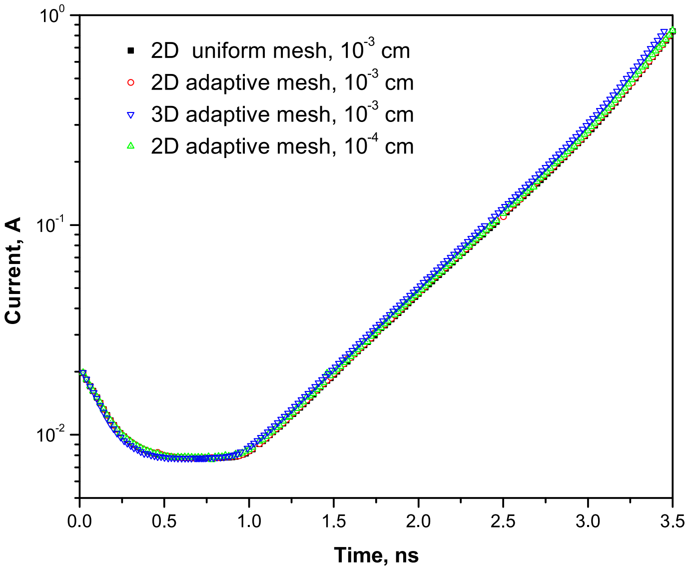


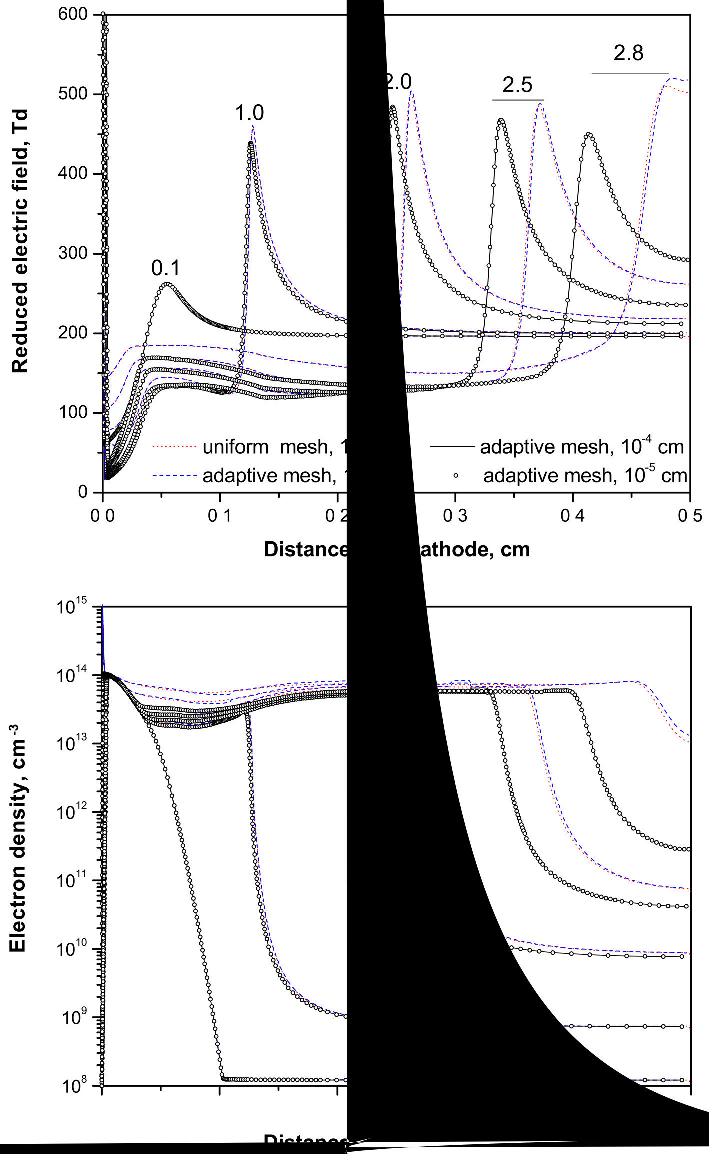
Fig. 4. Gap current for cathode-directed streamer.

mesh in this region is built with blocks of the finest sp[illegible]ce resolution used in the computation, however it has only a slight effect on the streamer structure and prop[illegible]gation in whole. At the same time, a much less detailed mesh of only 10^{-3} cm is required for correct descript[illegible]n in the region of streamer head.

Comparing our results with results published in [2[illegible]] we obtain quite a good agreement between them. A slightly higher speed of streamer at the end of prop[illegible]gation was obtained in our case, however this result was already discussed previously (see, e.g. [32]).

Finally, fully 3D simulation was performed using [illegible]e AMR technique for the mesh limit of 10^{-3} cm. An example of streamer structure is presented in Fig. 3 [illegible] time 2.6 ns corresponding to the moment when the streamer crosses the middle of the gap.

With the same code used, the simulations perform[illegible] in 2D and 3D show similar but not identical results. To demonstrate that, the electric current computed w[illegible] the help of (4) is presented in Fig. 4 for all reported simulations of the cathode-directed streamer. It follo[illegible] from the figure that even for the same space step the



3D simulation differs from 2D simulations with uniform or adaptive meshes. It is important to note that the same algorithms and criteria are used in the model for both 2D and 3D simulations. So, we suppose that an additional numerical diffusion appears in 3D simulations with respect to the one we have in 2D.

4.3. Anode-directed streamer

A similar set of calculations was performed for anode-directed streamer propagation. In this case, the initial preionization spot is located at the cathode. The axial distributions of electric field and electron density for a set of time moments are presented in Fig. 5.

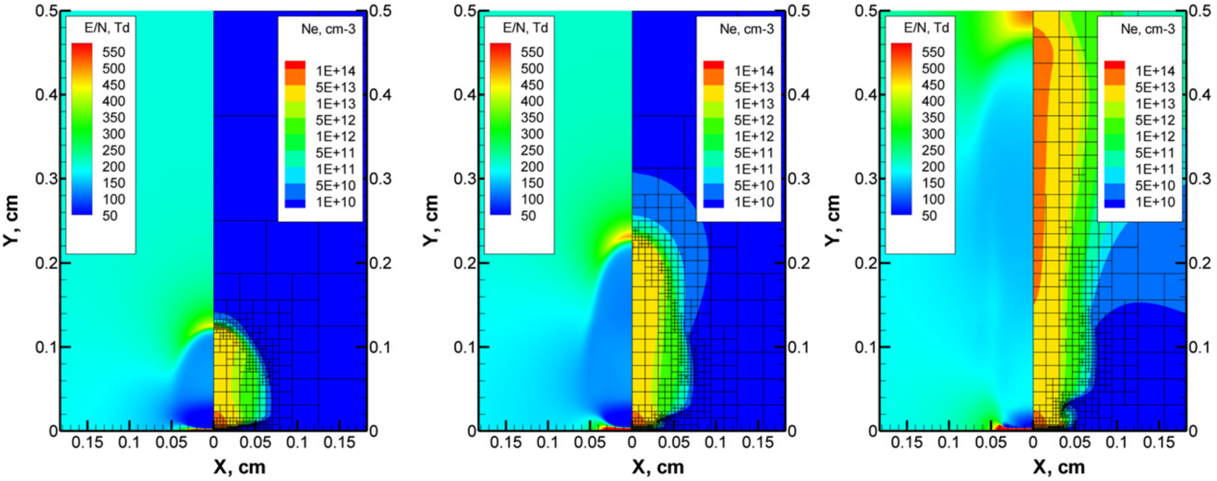


Fig. 6. Electric field and electron density distributions for anode-directed streamer at time moments of 1.0, 1.9 and 3.0 ns. Results of 2D simulation with adaptive mesh of 10^{-4} cm. Blocks of 8 × 8 cells are shown.

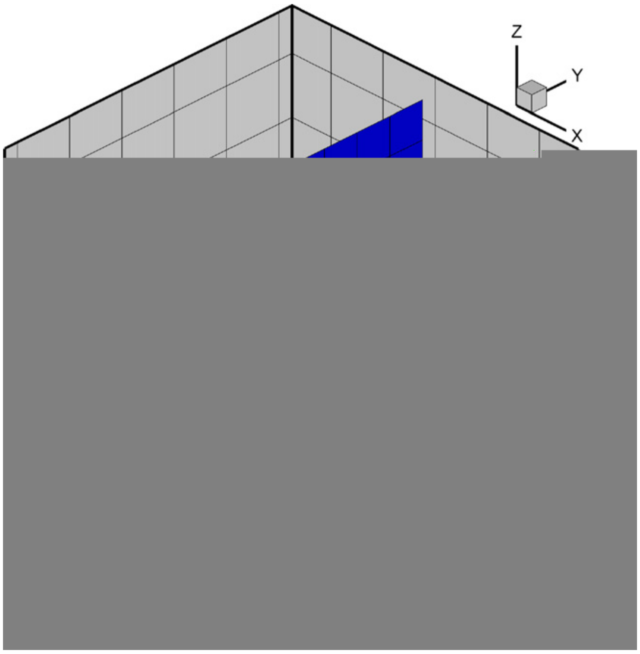


Fig. 7. Electron density distributions for anode-directed streamer at time moment of 1.9 ns. Results of 3D simulation with adaptive mesh of 10^{-3} cm.

As already observed for the cathode-directed streamer, we note that for the anode-directed streamer, the solutions obtained with uniform and adaptive meshes with the same space resolution of 10^{-3} cm are in a good agreement.

It is important to note that a space resolution of 10^{-3} cm was sufficient for cathode-directed streamer simulation under the same conditions, which was not the case for the anode-directed one. A convergence of solution is only observed for results with meshes of 10^{-4} and 10^{-5} cm. Thus, we conclude that a minimum space resolution of 10^{-4} cm minimum is required for anode-directed streamer simulation under the task conditions.

The 2D structure of the streamer during its propagation is presented in Figs. 6 for a set of time moments.

A fully 3D simulation was performed using the AMR technique for mesh limit 10^{-3} cm. An example of streamer structure is presented in Fig. 7 for time 1.9 ns corresponding to the moment when the streamer crosses the middle of the gap.

Electric current computed with the help of (4) is presented in Fig. 8. As already shown for the case of cathode-directed streamer, 2D and 3D give slightly different results for the same space step. Meanwhile, a good agreement is achieved for 2D adaptive mesh simulations with space steps of 10^{-4} and 10^{-5} cm.

4.3.1. The role of boundary conditions at cathode for anode-directed streamer

Similar to the case of cathode-directed streamer, a fine plasma structure appears near the cathode in the region of preionization spot (Fig. 6). Space gradients in that region are even steeper than those in the streamer head and a higher space resolution is required there. Contrary to cathode-directed streamer, this fine near-cathode structure strongly affects all the streamer dynamics.

From the physics point of view, the presence of a region with strong electric field is caused by the absence of electron emission from cathode and can be treated as additional voltage drop which, of course, changes the regime of streamer propagation.

To check this assumption, a set of additional computations was made with "soft" boundary conditions applied at electrodes Σ, namely $\partial j_s/\partial y = 0$. The near-cathode structure of the streamer for both sets of calculations is presented in Fig. 9 at time moment 1 ns for adaptive meshes of 10^{-3} and 10^{-5} cm.

A strong solution drop is observed near the anode for the higher space resolution 10^{-5} cm with original boundary conditions. The lower resolution 10^{-3} cm leads to the disappearance of this region due to extra mesh diffusion. At the same time, the solutions obtained with "soft" boundary conditions exhibit similar behaviour without a cathode drop formation.

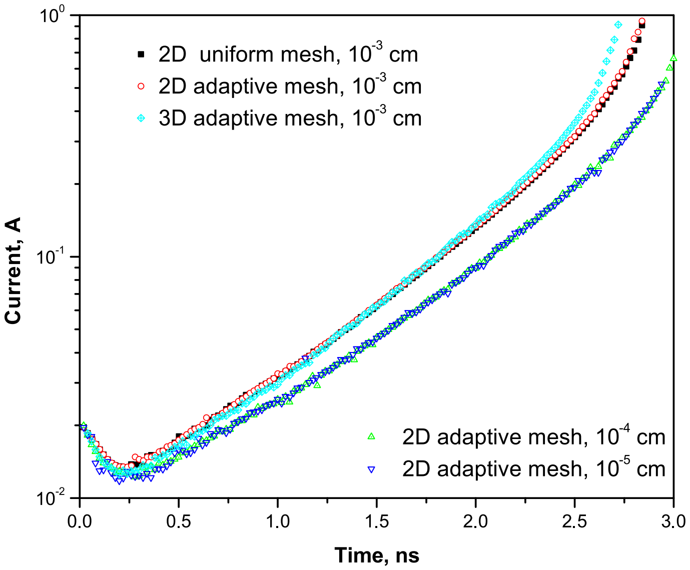


Fig. 8. Gap current for anode-directed streamer.

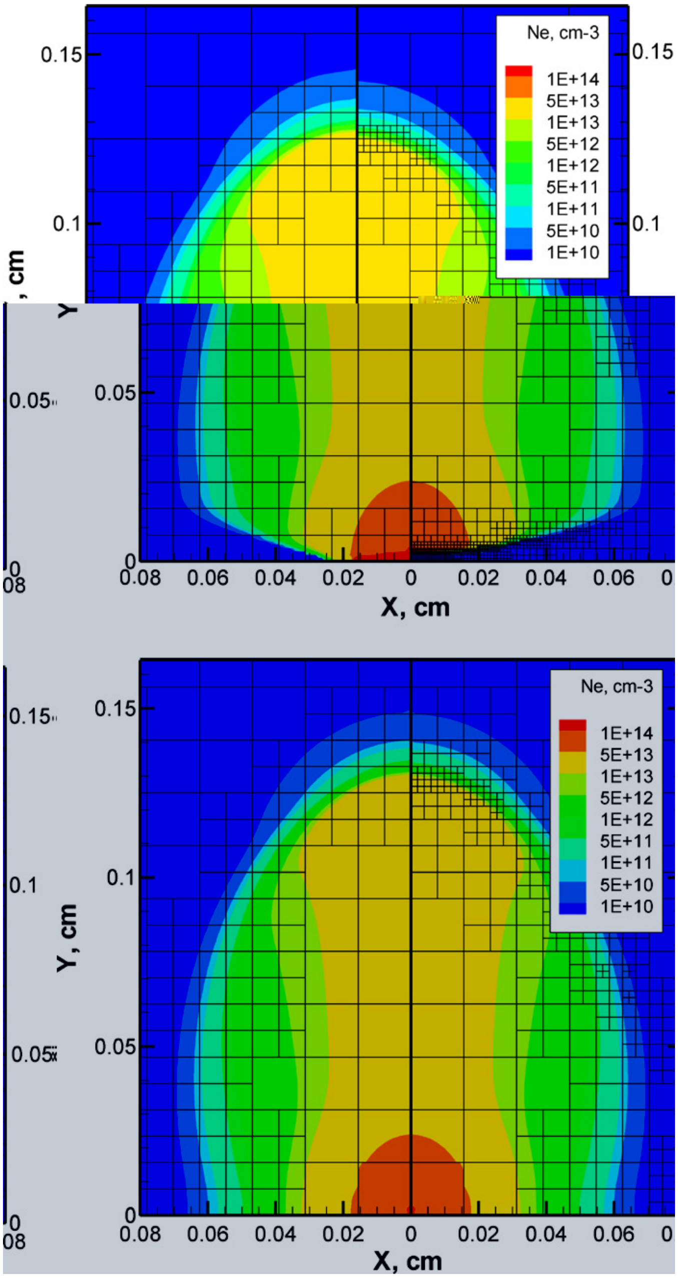


Fig. 9. Electron density at time moment 1.0 ns. Results of 2D simulation with adaptive meshes of 10^{-3} (left) and 10^{-5} cm (right). Normal (top) and modified (bottom) boundary conditions.

So, we conclude that all discrepancies between the solutions with various resolutions are caused mainly by poor accuracy in the cathode fall region and they can be eliminated by using modified "soft" boundary conditions.

4.4. Computational time and parallelization efficiency

The computations presented in the present paper have been performed on a 4-node cluster of double Intel Xeon 2.80 GHz CPU with 16 GB of total physical memory and of about 2 Tb of hard drive space. A Myrinet network and MPICH message-passing interface library are used for internode communication.

Table 1
Computational time in min for the cathode-directed streamer simulation

	Number of nodes						Number of blocks
	1	2	3	4	5	6	
2D uniform	492						2048
2D adaptive	115	74	55	48	44	40	443
3D adaptive						8615	27,263

The principal advantage of the AMR technique is that it minimizes the total number of computational cells without any loss of accuracy. Computational times for cathode-directed streamer simulation with mesh limit 10^{-3} cm are summarized in Table 1 to demonstrate this conclusion.

For a uniform mesh with 2048 blocks (1.3×10^5 cells) in 2D case, the AMR approach gives the same results with only 443 blocks (2.8×10^4 cells). The computational time is almost proportional to the total number of cells. Then it gives an acceleration of about 4.3 times. We note some speed impact of AMR internal procedures, namely solution interpolation, restriction and block reordering, which are not required for uniform meshing.

We also report the typical time branching between different subroutines. The most time-consuming is Poisson's equation solver (95–98%) while the contribution of other modules to computational time is as follows: transport solver (2–4%), data save (0.1–0.5%), mesh modification and block reordering (0.1–0.3%) and kinetics (0.1%).

5. Conclusions

A numerical code for filamentary discharge simulations has been developed. The code is based on balance equations for charged species in self-consistent electric field which is described by Poisson's equation. Plane–plane 2D axisymmetric and fully 3D Cartesian geometries have been considered.

In this work, an adaptive mesh refinement method has been used. The AMR is a method of adaptive meshing in which the continuous domain of interest is discretized into a grid of many individual elements. As the solution proceeds, the regions requiring more resolution are identified by some parameters characterizing the solution, namely the local truncation error. Finer subgrids are superimposed only in these regions. Finer and finer subgrids are added recursively until either a given maximum level of refinement is reached or the local truncation error has dropped below the desired level. Thus, in an adaptive mesh refinement, computation grid spacing is fixed only for the base grid and is determined locally for the subgrids according to the requirements of the problem. We have shown that the use of the AMR accelerates computations by a factor 4–5 and up to 8–12 in 2D and 3D cases, respectively, in comparison with uniform mesh approach.

The results of the test case proposed in [23] are presented for both positive and negative streamer propagation in a 5-mm gap filled by nitrogen at the normal conditions in a constant uniform external field of 196 Td. Quite good agreement is obtained for the case of cathode-directed streamer. The role of boundary conditions and their influence on anode-directed streamer simulation are discussed.

Acknowledgments

This work has been supported by the French National Research Agency (ANR) within the IPER project "Mechanisms of gas-discharge plasma interaction with reactive flows". The authors are grateful to Victoria McBride for the help in preparing this manuscript.

References

[1] Y.P. Raizer, Gas Discharge Physics, Springer-Verlag, Berlin, 1991.
[2] E.M. Veldhuizen (Ed.), Electrical Discharges for Environmental Purposes: Fundamentals and Applications, NOVA Science Publishers, New York, 1999.
[3] C. Hirsch, Numerical Computation of Internal and External Flows, Wiley, New York, 1990.

[4] J.P. Boris, D.L. Book, Flux-corrected transport I. SHASTA, a fluid transport algorithm that works, J. Comput. Phys. 11 (1973) 38.
[5] R. Morrow, Theory of stepped pulses in negative corona discharges, Phys. Rev. A 32 (1985) 3821.
[6] O. Ducasse, L. Papageorghiou, O. Eichwald, N. Spyrou, M. Yousfi, Critical analysis on two-dimensional point-to-plane streamer