

Ionic Wind Simulation Notes

July 22, 2023

1 Objectives

The objective of this research is to replicate the results of Chen et al. (2017), "A Self-Consistent Model of Ionic Wind Generation by Negative Corona Discharges in Air With Experimental Validation".

2 Governing Equations

The following equations constitute equations 1-5 in Chen et al. They will be solved simultaneously in Exasim using the Convection-Diffusion transport model (Model D). The newly-introduced DAE "subproblem" module will be used to separately solve the species transport/diffusion problem and the electrostatic problem.

$$\frac{\partial n_e}{\partial t} + \nabla \cdot (-\mu_e \vec{E} n_e - D_e \nabla n_e) = \alpha n_e |\mu_e \vec{E}| - \eta n_e |\mu_e \vec{E}| - k_{ep} n_e n_p \quad (1)$$

$$\frac{\partial n_n}{\partial t} + \nabla \cdot (-\mu_n \vec{E} n_n - D_n \nabla n_n) = \eta n_e |\mu_e \vec{E}| - k_{np} n_n n_p \quad (2)$$

$$\frac{\partial n_p}{\partial t} + \nabla \cdot (\mu_p \vec{E} n_p - D_p \nabla n_p) = \alpha n_e |\mu_e \vec{E}| - k_{np} n_n n_p - k_{ep} n_e n_p \quad (3)$$

$$\nabla^2 \Phi = -\frac{e(n_p - n_e - n_n)}{\epsilon} \quad (4)$$

$$\vec{E} = -\nabla \Phi \quad (5)$$

Additionally, the one-way coupling equations from the EHD model to the gas dynamics model (N-S) is provided in Eqns. 6-8:

$$f_{ehd} = e(n_p - n_e - n_n) \vec{E} \quad (6)$$

$$\nabla \cdot \vec{u} = 0 \quad (7)$$

$$\rho_g \left(\frac{\partial \vec{u}}{\partial t} + \vec{u} \cdot \nabla \vec{u} \right) = -\nabla p + \mu_v \nabla^2 \vec{u} + f_{ehd} \quad (8)$$

3 Cylindrical Coordinates

The gradient and divergence operators in cylindrical coordinates (r, θ, z) are given, for a scalar function f and a vector field \mathbf{f} , by

$$\begin{aligned} \nabla f &= \frac{\partial f}{\partial r} \mathbf{e}_r + \frac{1}{r} \frac{\partial f}{\partial \theta} \mathbf{e}_\theta + \frac{\partial f}{\partial z} \mathbf{e}_z \\ \nabla \cdot \mathbf{f} &= \frac{1}{r} \frac{\partial(r f_r)}{\partial r} + \frac{1}{r} \frac{\partial f_\theta}{\partial \theta} + \frac{\partial f_z}{\partial z} \end{aligned}$$

If we assume axial symmetry all variable are function of (r, z) only.

4 Fully Conservative Form

Assuming axial symmetry, the above equations can be written as

$$\mathbf{M} \frac{\partial \mathbf{U}}{\partial t} + \frac{1}{r} \frac{\partial (r \mathbf{F}_r)}{\partial r} + \frac{\partial \mathbf{F}_z}{\partial z} - \mathbf{S} = \mathbf{0}$$

Where

$$\mathbf{U} = \begin{pmatrix} n_e \\ n_n \\ n_p \\ \Phi \end{pmatrix}, \quad \mathbf{F}_r = \begin{pmatrix} -\mu_e n_e E_r - D_e(q_e)_r \\ -\mu_n n_n E_r - D_n(q_n)_r \\ \mu_p n_p E_r - D_p(q_p)_r \\ -E_r \end{pmatrix}, \quad \mathbf{F}_z = \begin{pmatrix} -\mu_e n_e E_z - D_e(q_e)_z \\ -\mu_n n_n E_z - D_n(q_n)_z \\ -\mu_p n_p E_z - D_p(q_p)_z \\ -E_z \end{pmatrix}, \quad (9)$$

$$\mathbf{S} = \begin{pmatrix} \alpha n_e |\mu_e \vec{E}| - \eta n_e |\mu_e \vec{E}| - k_{ep} n_e n_p \\ \eta n_e |\mu_e \vec{E}| - k_{np} n_n n_p \\ \alpha n_e |\mu_e \vec{E}| - k_{np} n_n n_p - k_{ep} n_e n_p \\ -\frac{e(n_p - n_e - n_n)}{\epsilon} \end{pmatrix}, \quad \mathbf{M} = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 \end{pmatrix} \quad (10)$$

and

$$\mathbf{Q} = \begin{pmatrix} (q_e)_r & (q_e)_z \\ (q_n)_r & (q_n)_z \\ (q_p)_r & (q_p)_z \\ -E_r & -E_z \end{pmatrix}, \quad (11)$$

with

$$\mathbf{Q} = \nabla \mathbf{U}. \quad (12)$$

Note that under the axisymmetry assumption, the gradient operators in cylindrical and cartesian coordinates are the same.

5 Weak form and the Divergence Theorem

The elemental volume becomes $dV = r dr dz$ and for any \mathbf{W} we can write the following weighted residual form

$$\int_V \left(\mathbf{M} \frac{\partial \mathbf{U}}{\partial t} + \frac{1}{r} \frac{\partial (r \mathbf{F}_r)}{\partial r} + \frac{\partial \mathbf{F}_z}{\partial z} + \mathbf{S} \right) \mathbf{W} r dr dz = 0,$$

or, integrating by parts,

$$\int_V \left((Mr) \frac{\partial \mathbf{U}}{\partial t} + r\mathbf{S} \right) \mathbf{W} \, drdz + \int_S (r\mathbf{F}_r \mathbf{n}_r + r\mathbf{F}_z \mathbf{n}_z) \mathbf{W} \, dS$$

$$- \int_V \left(r\mathbf{F}_r \frac{\partial \mathbf{W}}{\partial r} + r\mathbf{F}_z \frac{\partial \mathbf{W}}{\partial z} \right) \, drdz = \mathbf{0}$$

Note that in the integrals in the second and third lines the *effective* dV becomes $drdz$. The only differences between cylindrical and cartesian coordinates thus

- Multiply M and S by r . Note that this will require a modified mass matrix.
- Multiply F_r and F_z by r

6 Nondimensionalization

Because of the wide range of scales used, it is important to nondimensionalize the problem to prevent numerical instability. The solution variables as well as spatial and temporal variables were nondimensionalized.

6.1 Nondimensional groups

The following nondimensional groups were chosen:

- $n_{ref} = \frac{\epsilon_0 E_{bd}}{er_{tip}}$
- $E_{ref} = E_{bd}$
- $\Phi_{ref} = E_{bd} r_{tip}$
- $t_{ref} = \frac{r_{tip}}{\mu_{e,ref} E_{bd}}$
- $l_{ref} = r_{tip}$

$\mu_{e,ref}$ is taken at the reduced electric field value of $\frac{E_{bd}}{N}$, where N , the neutral number density, is computed using the ideal gas equation of state at standard temperature and pressure:

- $pV = Nk_B T$
- $P = 101325 \, Pa$
- $V = 1 \, m^3$
- $T = 273.15 \, K$
- $k_B = 1.380649 \times 10^{-23} \frac{m^2 kg}{s^2 K}$

- $\Rightarrow N = 2.6868 \times 10^{25} \frac{\text{particles}}{\text{m}^3}$

Thus, the independent and dependent variables may be expressed as the following. The asterisk indicates a nondimensional quantity.

- $n_e = \frac{n_e^* \epsilon_0 E_{bd}}{e r_{tip}}$
- $n_p = \frac{n_p^* \epsilon_0 E_{bd}}{e r_{tip}}$
- $n_n = \frac{n_n^* \epsilon_0 E_{bd}}{e r_{tip}}$
- $\vec{E} = \vec{E}^* E_{bd}$
- $\Phi = \Phi^* E_{bd} r_{tip}$
- $t = \frac{t^* r_{tip}}{\mu_{e,ref} E_{bd}}$
- $r = r^* r_{tip}$
- $z = z^* r_{tip}$

Where r_{tip} is the needle tip radius of curvature, $220 \mu m$, and E_{bd} is the breakdown electric field strength in air, $3 \times 10^6 \frac{V}{m}$.

We can re-write the governing equations (section 2) using the nondimensional groups, taking care to nondimensionalize the partial derivatives with respect to space and time as well:

$$\begin{aligned} \frac{\partial(n_e^* n_{ref})}{\partial \left(\frac{t^* r_{tip}}{\mu_e E_{bd}} \right)} + \nabla \cdot \frac{\left(-\mu_e \vec{E}^* E_{bd} (n_e^* n_{ref}) - D_e \nabla \left(\frac{n_e^* n_{ref}}{r_{tip}} \right) \right)}{r_{tip}} = \\ (\alpha - \eta)(n_e^* n_{ref}) |\mu_e (\vec{E}^* E_{bd})| - k_{ep} n_e^* n_p^* n_{ref}^2 \end{aligned}$$

$$\begin{aligned} \frac{\partial(n_p^* n_{ref})}{\partial \left(\frac{t^* r_{tip}}{\mu_e E_{bd}} \right)} + \nabla \cdot \frac{\left(\mu_p \vec{E}^* E_{bd} (n_p^* n_{ref}) - D_p \nabla \left(\frac{n_p^* n_{ref}}{r_{tip}} \right) \right)}{r_{tip}} = \\ \alpha n_e^* n_{ref} |\mu_e (\vec{E}^* E_{bd})| - n_p^* n_{ref}^2 (k_{np} n_n^* + k_{ep} n_e^*) \end{aligned}$$

$$\begin{aligned} \frac{\partial(n_n^* n_{ref})}{\partial \left(\frac{t^* r_{tip}}{\mu_e E_{bd}} \right)} + \nabla \cdot \frac{\left(-\mu_n \vec{E}^* E_{bd} (n_n^* n_{ref}) - D_n \nabla \left(\frac{n_n^* n_{ref}}{r_{tip}} \right) \right)}{r_{tip}} = \\ \eta n_e^* n_{ref} |\mu_e (\vec{E}^* E_{bd})| - k_{np} n_n^* n_p^* n_{ref}^2 \end{aligned}$$

$$\nabla^2 \left(\frac{\Phi^* E_{bd} r_{tip}}{r_{tip}^2} \right) = - \frac{\epsilon_0 e E_{bd}}{\epsilon_0 e r_{tip}} (n_p^* - n_e^* - n_n^*)$$

Simplifying:

$$\frac{\partial n_e^*}{\partial t^*} + \nabla \cdot \left(- \frac{\mu_e}{\mu_{e,ref}} \vec{E}^* n_e^* - \frac{D_e}{\mu_{e,ref} E_{bd} r_{tip}} \nabla(n_e^*) \right) = (\alpha - \eta) \frac{\mu_e}{\mu_{e,ref}} r_{tip} n_e^* |\vec{E}^*| - \frac{k_{ep} \epsilon_0}{e \mu_{e,ref}} n_e^* n_p^*$$

$$\frac{\partial n_p^*}{\partial t^*} + \nabla \cdot \left(\frac{\mu_p}{\mu_{e,ref}} \vec{E}^* n_p^* - \frac{D_p}{\mu_{e,ref} E_{bd} r_{tip}} \nabla(n_p^*) \right) = \alpha \frac{\mu_e}{\mu_{e,ref}} r_{tip} n_e^* |\vec{E}^*| - \left(\frac{\epsilon_0 n_p^*}{e \mu_{e,ref}} \right) (k_{np} n_n^* + k_{ep} n_e^*)$$

$$\frac{\partial n_n^*}{\partial t^*} + \nabla \cdot \left(- \frac{\mu_n}{\mu_{e,ref}} \vec{E}^* n_n^* - \frac{D_n}{\mu_{e,ref} E_{bd} r_{tip}} \nabla(n_n^*) \right) = \eta \frac{\mu_e}{\mu_{e,ref}} r_{tip} n_e^* |\vec{E}^*| - \left(\frac{k_{np} \epsilon_0}{e \mu_{e,ref}} \right) n_n^* n_p^*$$

$$\nabla^2 \Phi^* = n_e^* + n_n^* - n_p^*$$

7 Flux and source functions

The source and flux functions as they are entered into the code generation script are presented below, accounting for the axisymmetry condition by multiplying by r . The jacobian of each are taken automatically and assembled into the appropriate output format using functions in the MATLAB symbolic toolbox and its symbolic optimization capabilities such as Common Subexpression Elimination (CSE).

Note that for succinctness, the diffusion coefficient for all three species has been normalized as follows:

$$D_{xx}^* = \frac{D_{xx}}{\mu_{e,ref} E_{bd} r_{tip}}$$

To improve the computational efficiency, the linearity of the Poisson equation is leveraged and the electric field is split into two components. The first component is obtained using a homogeneous solution to the electrostatic equation, i.e. the Laplace equation $\nabla \Phi = 0$, and an inhomogeneous solution accounting for the space charge created by the charged species. The homogeneous solution is solved in a separate simulation and loaded into the simulation during the initialization steps. The boundary conditions for the homogeneous solution reflect the voltage applied to the emitter tip and grounded cylinder and floor, whereas the boundary condition for the inhomogeneous case solved at runtime is uniformly homogeneous dirichlet, with the exception of the axis of symmetry.

The electric field is thus comprised of two components:

$$E = E_0 + E'$$

With E_0 computed beforehand, and E' representing the (negative) gradient of the electric potential, the fourth solution variable. Also note that since the gradient provided in the code is the negative of the true gradient of the solution variable, the values for the viscous flux appear to be negated.

Flux terms

Equation 1: electron number density

$$F_{1,r} = - \left(\frac{\mu_e}{\mu_{e,ref}} \right) E_r n_e + D_e^* \frac{dn_e}{dr}$$

$$F_{1,z} = - \left(\frac{\mu_e}{\mu_{e,ref}} \right) E_z n_e + D_e^* \frac{dn_e}{dz}$$

Equation 2: negative ion number density

$$F_{2,r} = - \left(\frac{\mu_n}{\mu_{e,ref}} \right) E_r n_n + D_n^* \frac{dn_n}{dr}$$

$$F_{2,z} = - \left(\frac{\mu_n}{\mu_{e,ref}} \right) E_z n_n + D_n^* \frac{dn_n}{dz}$$

Equation 3: positive ion number density

$$F_{3,r} = \left(\frac{\mu_p}{\mu_{e,ref}} \right) E_r n_p + D_p^* \frac{dn_p}{dr}$$

$$F_{3,z} = \left(\frac{\mu_p}{\mu_{e,ref}} \right) E_z n_p + D_p^* \frac{dn_p}{dz}$$

Equation 4: Electric potential

$$F_{4,r} = E'_r$$

$$F_{4,z} = E'_z$$

Assembly:

$$f_{assembled} = r [\mathbf{F}_r, \mathbf{F}_z]$$

Source terms

$$\begin{aligned}
s_e &= (\alpha - \eta) \left(\frac{\mu_e}{\mu_{e,ref}} \right) r_{tip} n_e |E| - \frac{K_{ep} \epsilon_0}{e \mu_{e,ref}} n_e n_p \\
s_n &= eta \left(\frac{\mu_e}{\mu_{e,ref}} \right) r_{tip} n_e |E| - \frac{K_{np} \epsilon_0}{e \mu_{e,ref}} n_n n_p \\
s_p &= alpha \left(\frac{\mu_e}{\mu_{e,ref}} \right) r_{tip} n_e |E| - \frac{n_p \epsilon_0}{e \mu_{e,ref}} (K_{np} n_n + K_{ep} n_e) \\
s_\Phi &= n_p - n_e - n_n
\end{aligned}$$

Assembly:

$$s_{assembled} = r [s_e, s_n, s_p, s_\Phi]$$

Swarm parameters

The swarm parameters governing the forward and reverse ionization rates, as well as the electron mobility and diffusivity coefficients, are taken as curve fits of data obtained from the BOLSIG+ solver hosted by the LXCAT website (<https://us.lxcat.net/solvers/BolsigPlus/>). All four parameters are a function of the electric field. For comparison with the literature, they are presented here as a function of the reduced electric field, $\frac{E}{N}$, where N is the neutral number density. Here, N is calculated using the ideal gas equation of state at STP conditions: $P = 1\text{atm}$ and $T = 300\text{K}$. The same conditions were used as inputs to the BOLSIG+ solver. The following curve fits were obtained, with the reduced electric field $\frac{E}{N}$ in Townshend (Td). For most of these equations, a hyperbolic tangent step function was used to transition between different curve fits. This was to minimize a piecewise function while remaining differentiable through the simulation. Here, the logarithm used is base 10.

Ionization coefficient α

$$\begin{aligned}
\text{exponent} \left(\frac{E}{N} \right) &= \begin{cases} 0 & \frac{E}{N} \leq 20.1\text{Td} \\ \frac{-212.327 + 304.251 \log \left(\frac{E}{N} \right)}{-186.287 * \log \left(\frac{E}{N} \right)^2 + 51.501 * \log \left(\frac{E}{N} \right)^3 - 5.362 * \log \left(\frac{E}{N} \right)^4} & \frac{E}{N} > 20.1\text{Td} \end{cases} \\
\alpha &= 10^{\text{exponent}}
\end{aligned}$$

Recombination coefficient η

$$\text{exponent} \left(\frac{E}{N} \right) = \begin{cases} \frac{-40.129 - 1.044 \log \left(\frac{E}{N} \right)}{-125.180 + 168.336 \log \left(\frac{E}{N} \right)} & \frac{E}{N} \leq 1.1\text{Td} \\ \frac{-103.649 \log \left(\frac{E}{N} \right)^2 + 28.456 \log \left(\frac{E}{N} \right)^3 - 2.942 \log \left(\frac{E}{N} \right)^4}{-125.180 + 168.336 \log \left(\frac{E}{N} \right)} & \frac{E}{N} > 1.1\text{Td} \end{cases}$$

$$\eta = 10^{\text{exponent}}$$

Mobility coefficient μ

$$\mu_e = 10^{(24.767 - 0.348 \log(\frac{E}{N}))}$$

Diffusion coefficient D

$$D\left(\frac{E}{N}\right) = \begin{cases} \frac{3.477 \times 10^{23} \log\left(\frac{E}{N}\right) + 1.413 \times 10^{24}}{4.633 \times 10^{24} \log\left(\frac{E}{N}\right) + -6.700 \times 10^{24}} & \frac{E}{N} \leq 1.9 \text{Td} \\ & \frac{E}{N} > 1.9 \text{Td} \end{cases}$$

8 Initial Conditions

A gaussian IC is used for the electrons and positive ions. Using the same distribution ensures that charge conservation is obeyed, but the number density is high enough to initiate the discharge process at the emitter tip. The initial number density for the negative ions is zero across the entire domain.

$$\begin{array}{l|l} n_e, n_p & N_{max}^* \exp\left(\frac{-1}{2\sigma_0^{*2}}((r - r_0^*)^2 + (z - z_0^*)^2)\right) \\ \frac{dn_e}{dr}, \frac{dn_p}{dr} & N_{max}^* \exp\left(\frac{-1}{2\sigma_0^{*2}}((r - r_0^*)^2 + (z - z_0^*)^2)\right) \frac{r - r_0^{*2}}{\sigma_0^{*2}} \\ \frac{dn_e}{dz}, \frac{dn_p}{dz} & N_{max}^* \exp\left(\frac{-1}{2\sigma_0^{*2}}((r - r_0^*)^2 + (z - z_0^*)^2)\right) \frac{z - z_0^{*2}}{\sigma_0^{*2}} \end{array}$$

Table 1: Boundary conditions for the electrons and positives.

The following values are used to define the initial condition:

- $N_{max}^* = \frac{N_{max}}{n_{ref}} = \frac{N_{max}}{\frac{\epsilon_0 E_{bd}}{er_{tip}}}$
- $\sigma_0^* = \frac{\sigma_0}{r_{tip}}$

9 Boundary Conditions

Note: Boundary numbering follows the boundary numbering in the paper.

9.1 Boundary 1: Needle Tip

Equation	Boundary condition
1	Total flux $-\vec{n} \cdot \left(-\mu_e \vec{E} - D_e \nabla n_e \right) = \gamma n_p \mu_p \vec{E} $
2	Outflow, $\vec{n} \cdot (-D_p \nabla n_p) = 0$
3	$n_n = 0$
4	$\Phi = -U_a$

9.2 Boundary 2: Axis of Symmetry

Equation	Boundary condition
1	Axial symmetry $\frac{\partial n_e}{\partial r} = 0$
2	Axial symmetry $\frac{\partial n_p}{\partial r} = 0$
3	Axial symmetry $\frac{\partial n_n}{\partial r} = 0$
4	Axial symmetry $\frac{\partial \phi}{\partial r} = 0$

9.3 Boundary 3: Outflow

Equation	Boundary condition
1	Open boundary $\vec{n} \cdot (-D_e \nabla n_e) = 0; \vec{n} \cdot (-\mu_e \vec{E}) \geq 0$ $n_e = 0; \vec{n} \cdot (-\mu_e \vec{E}) < 0$
2	Open boundary $\vec{n} \cdot (-D_p \nabla n_p) = 0; \vec{n} \cdot (-\mu_p \vec{E}) \geq 0$ $n_p = 0; \vec{n} \cdot (-\mu_p \vec{E}) < 0$
3	Open boundary $\vec{n} \cdot (-D_n \nabla n_n) = 0; \vec{n} \cdot (-\mu_n \vec{E}) \geq 0$ $n_n = 0; \vec{n} \cdot (-\mu_n \vec{E}) < 0$
4	Ground $\phi = 0$

9.4 Boundary 4: Grounded surfaces - cylinder and ground plane

Equation	Boundary condition
1	Outflow $\vec{n} \cdot (-D_e \nabla n_e) = 0$
2	$n_p = 0$
3	Outflow $\vec{n} \cdot (-D_n \nabla n_n) = 0$
4	Ground $\phi = 0$

9.5 Boundary 5 and 6: Farfield (Insulated)

Total flux = convective + diffusive

Equation	Boundary condition
1	Total flux = 0
2	Total flux = 0
3	Total flux = 0
4	Total flux = 0