Computational Plasma Modeling for Characterization of an Atmospheric Pressure Plasma Jet

Declan George Brick

Follow this and additional works at: https://louis.uah.edu/honors-capstones

Recommended Citation

This Thesis is brought to you for free and open access by the Honors College at LOUIS. It has been accepted for inclusion in Honors Capstone Projects and Theses by an authorized administrator of LOUIS.
Computational Plasma Modeling for Characterization of an Atmospheric Pressure Plasma Jet

by

Declan George Brick

An Honors Capstone

submitted in partial fulfillment of the requirements

for the Honors Diploma

to

The Honors College

of

The University of Alabama in Huntsville

4/29/2022

Honors Capstone Director: Dr. Gabe Xu

Associate Professor of Mechanical and Aerospace Engineering

Student (signature)  Date  4/29/22

Director (signature)  Date  4/29/22

Department Chair (signature)  Date  4/29/2022

William Wilkerson  Digitally signed by William Wilkerson

Honors College Dean (signature)  Date  Date: 2022.05.01 11:35:19 -05'00'
Honors Thesis Copyright Permission

This form must be signed by the student and submitted as a bound part of the thesis.

In presenting this thesis in partial fulfillment of the requirements for Honors Diploma or Certificate from The University of Alabama in Huntsville, I agree that the Library of this University shall make it freely available for inspection. I further agree that permission for extensive copying for scholarly purposes may be granted by my advisor or, in his/her absence, by the Chair of the Department, Director of the Program, or the Dean of the Honors College. It is also understood that due recognition shall be given to me and to The University of Alabama in Huntsville in any scholarly use which may be made of any material in this thesis.

Declan Brick

Student Name (printed)

Student Signature

4/29/22

Date
# Table of Contents

Table of Contents.......................................................................................................................... 3

Abstract............................................................................................................................................... 4

Introduction......................................................................................................................................... 4

Atmospheric Pressure Plasma Jets.................................................................................................... 4

APPJ Research at the University of Alabama in Huntsville ................................................................. 5

Fluid Modeling..................................................................................................................................... 6

Methodology ....................................................................................................................................... 6

Results and Discussion ...................................................................................................................... 8

Plasma Modeling ............................................................................................................................... 13

Methodology ..................................................................................................................................... 13

Results and Discussion ...................................................................................................................... 18

Conclusion .......................................................................................................................................... 23

References.......................................................................................................................................... 24
Abstract

Atmospheric Pressure Plasma Jets (APPJs) are low-temperature plasma devices that have a variety of applications including, but not limited to, water purification, bacteria sterilization, and manufacturing. However, the performance of APPJs can vary from design to design, meaning benchmark parameters must be evaluated to characterize the performance of an APPJ. To this end, the APPJ at UAH’s Plasma and Electrodynamics Research Laboratory has been experimentally characterized across varied operating conditions to understand its performance in water treatment applications. However, this characterization has not investigated the gas mixture ratio, electron number density, and electron temperature of the APPJ. These parameters are important in understanding the APPJs behavior and are easier to determine using computational models rather than experiment. To complete this characterization, we sought to develop a fluid-based computational plasma model. The fluid-based nature of the model allowed for characterization of the mixture ratio. Meanwhile, the plasma and electrodynamic part of the model sought to determine the electron density and temperature. Additionally, for characterization, the model was expected to run for multiple operating conditions to characterize the APPJ performance. With these criteria, we expected the APPJ behavior to be better understood, which allows for optimization of the APPJ use and lessons for future designs.

Introduction

Atmospheric Pressure Plasma Jets

While plasmas are typically thought of as advanced technology in science fiction, many examples occur in nature, such as stars, lightning, and fire. Simply put, plasma is the fourth state of matter where electrons in a gas are separate from their atomic nuclei. Electrons can be separated by a variety of mechanisms such as particle collisions, electric and magnetic fields, or interaction with light. This separation forms a quasi-neutral substance, meaning that while the whole plasma has no net electric charge, localized regions within the plasma can have a net positive or negative charge. Thus, plasmas are subject to a combination of fluid, thermal, chemical, and electromagnetic effects. The variety of effects mean that plasmas have applications ranging from spacecraft propulsion to welding, creating great interest in understanding plasmas to be able to develop new technologies.

Atmospheric Pressure Plasma Jets (APPJs) are low-temperature plasma devices that operate at room temperatures and pressures. Low temperature plasmas have disparate electron and ion temperatures with electron temperatures in the 1-10 eV range (~10,000-100,000 K) and ions and neutrals near room temperature (~300-400 K). Due to the low neutral or gas temperatures, APPJs can interact with materials without burning or destroying them as high temperature plasmas do. Additionally, as APPJs operate at atmospheric pressure, they do not require expensive equipment such as vacuum chambers. These properties result in a variety of applications including, but not limited to, bacteria sterilization, infection treatment, water purification, agricultural products, and materials manufacturing[1,2]. This wide variety of applications have caused APPJs to be the extensively studied.

APPJs are built by flowing a gas, typically a noble gas, in the proximity of powered electrodes that ionize the gas, producing a plasma [3]. Then the plasma will either directly treat the targeted area or form other useful gas species through secondary ionization of the surrounding air. However, the setup of APPJs vary widely from research group to research group. There are wide variations in how power is supplied to electrodes, the number and make of electrodes, and even the gas itself [3]. Due to the wide variation between APPJs, a significant amount of research has focused on developing benchmark parameters to compare different designs and their production of active species used in treatment. By understanding how the design of APPJs affect these parameters, research can find optimal designs for the varied applications.
**APPJ Research at the University of Alabama in Huntsville**

At the University of Alabama in Huntsville, the Plasma and Electrodynamics Research Laboratory (PERL) has developed an APPJ for water purification applications. A schematic of PERL’s APPJ is shown in Figure 1, taken from reference [1]. To give an idea of scale, the quartz tube has a diameter of 6 mm. The working gas in this case is helium, which flows into the Swagelok tee through the inlet labeled Feed Gas. The gas then flows around a capillary quartz tube that encloses a tungsten rod, which acts as the ionizing electrode. The electrode is contained inside the capillary tube to prevent direct electrical contact between the electrode and the plasma. The ionized helium then interacts with the surrounding air to produce reactive oxygen and nitrogen species (RONS) such as OH, O, and O$_3$ which are the active species in water purification. The tungsten electrode is powered by a pulsed DC power supply. The pulsed nature of the power means that the “jet” is actually a series of individual plasma “bullets” repeated at a high rate that they appear to form a jet to human eyes. A steel box encloses the setup and serves as the grounding electrode for the circuit.

![Figure 1: Schematic of PERL’s APPJ](image)

To understand the effects of design choice on APPJs, PERL experimentally characterized its APPJ in 2019 [1]. The parameters of interest were the production of RONS and jet length. The RONS are the species that perform the water purification, so the more species produced, the more effective the APPJ. Jet length is a measure of the effectiveness of ionization, a longer jet means the design is more efficient at plasma generation. The operating conditions investigated were the applied voltage, pulse width of the applied voltage, repetition rate of the applied voltage, and flow rate of helium. The results found increased RONS production at more energetic conditions while the jet length increased at higher applied voltages and flow rates [1]. However, past a 3 slm (standard liters per minute) flow rate, the APPJ entered the turbulent flow regime. This turbulence caused the growth of the jet length to become non-linear [1]. Thus, while jet length is typically a measure of energy efficiency, it does not quite form a 1:1 comparison depending on the effects of turbulence.
While the experimental characterization provided strong insight into the production of active species and plasma by the APPJ, it still left off some key parameters. The most important parameter left off was the electron temperature and density, which is an additional measurement of how energetic the plasma is and where it propagates. Getting accurate measurements of electron temperature and density experimental is difficult as the process is invasive and perturbs the plasma. Therefore, to determine the electron temperature and density we turn to computational models. Models offer to possibility to calculate the electron energy and temperature in a straightforward manner, without having to account for the invasiveness of probes. However, we are not interested in any plasma model, but rather a fluid-based model due to the continuum nature of the atmospheric pressure jet. The flow rate effects on propagation from the experimental results have significant effects on the plasma propagation, which we want to capture and better understand. Additionally, focusing on a fluid model allows a clear research pathway: first develop the fluid portion of the model and then overlay the plasma behavior.

**Fluid Modeling**

**Methodology**

To model the fluid behavior of the plasma, the commercial software ANSYS Fluent was selected. Previous project experience using ANSYS and access to a license gave confidence that the fluid behavior could be successfully modeled. However, it is also worth noting Fluent’s robust modeling capability and diversity of turbulence models to be able to describe the flow behavior past 3 slm. The methodology section first describes the governing equations behind Fluent and the k-ε turbulence model and finishes with a description of the exact setup used to model the various flow rates and their behavior.

Fluent is governed by two conservation laws: the conservation of mass and the conservation of momentum. Thus, any mass or momentum that is in or enters the system must be accounted for within or leaving the system. In addition, conservation of energy was enabled selected to ensure a more accurate solution. Mathematically, these laws are expressed using the Euler equations, which are similar to the Navier-Stokes equations that govern all of fluid mechanics, but it ignores the viscosity of the fluid. The Euler equations for conservation of mass, momentum, and energy are given in Equations (1), (2), and (3) respectively.

\[
\frac{\partial \rho}{\partial t} + \nabla \cdot (\rho \mathbf{v}) = S_m \tag{1}
\]

\[
\frac{\partial}{\partial t}(\rho \mathbf{v}) + \nabla \cdot (\rho \mathbf{vv}) = -\nabla p + \rho \mathbf{g} + \mathbf{F} \tag{2}
\]

\[
\frac{\partial (\rho E)}{\partial t} + \nabla \cdot (\mathbf{v}(\rho E + \rho)) = -\nabla \cdot \sum h_j J_j + S_h \tag{3}
\]

Where \( \rho \) is density, \( \mathbf{v} \) is velocity, \( S_m \) is a mass source term, \( \mathbf{g} \) is the local gravitational constant, \( \mathbf{F} \) is an external force, \( \mathbf{E} \) is energy, \( h \) is enthalpy, \( J \) is a diffusion term, and \( S_h \) is an enthalpy source term.

To model turbulence in the jet, the RNG k-ε model was used. The k-ε model a robust but efficient turbulence model commonly utilized in computational fluid dynamics. The k-ε method models turbulence by considering the turbulent kinetic energy and how it dissipates. The RNG k-ε model is a corrected version of the k-ε model that better accounts for swirls within the flow and wall effects. As the model seeks to describe the mixing between the helium jet and surrounding air, particularly at the exit of the quartz tube, these corrections were needed to ensure accuracy. The two equations that govern the RNG k-ε model are given in Equations (4) and (5).
\[ \frac{\partial}{\partial t}(\rho k) + \frac{\partial}{\partial x_i}(\rho k u_i) = \frac{\partial}{\partial x_j} \left( \alpha_k \mu_{eff} \frac{\partial k}{\partial x_j} \right) + G_k + C_b - \rho \varepsilon - Y_m + S_k \]  

\[ \frac{\partial}{\partial t}(\rho \varepsilon) + \frac{\partial}{\partial x_i}(\rho \varepsilon u_i) = \frac{\partial}{\partial x_j} \left( \alpha_k \mu_{eff} \frac{\partial \varepsilon}{\partial x_j} \right) + C_1 \varepsilon \frac{G_k + C_3 \varepsilon G_b}{k} - C_2 \varepsilon \rho \frac{\varepsilon^2}{k} - R \varepsilon + S \varepsilon \]  

Where \( \rho \) is density, \( k \) is turbulent kinetic energy, \( x_i \) and \( x_j \) are lengths in different dimensions, \( u_i \) is velocity in the \( i \) dimension, \( \alpha_i \) is the inverse effective Prandtl number for \( k \), \( \mu_{eff} \) is the effective dynamic viscosity, \( G_k \) is the turbulent kinetic energy generation due to velocity, \( G_b \) is the turbulent kinetic energy generation due to buoyancy, \( Y_m \) is compressibility effects on the turbulent kinetic energy, \( S_i \) is a turbulent kinetic energy source term, \( \varepsilon \) is the dissipation rate of turbulent kinetic energy, \( \alpha_i \) is the inverse effective Prandtl number for \( \varepsilon \), \( C_{11}, C_{21}, C_{31} \) are model constants, \( R \varepsilon \) is a term that accounts for straining of free-stream lines, and \( S \varepsilon \) is a dissipation rate source term.

The final additional modeling that needed to be included in the ANSYS Fluent analysis was the species transport model. Typically, Fluent only considers one fluid material within the modeling domain. However, the helium and air interaction are a mixture of two unique fluids. To be able to model this interaction the species transport model was selected. The model determines the local mass fraction of each species in a pre-defined mixture (which is air and helium in this model). This mass fraction is a measure of concentration of each species. The transport equation is given in Equation (6), but the model considers \( n-1 \) equations of this form where \( n \) is the number of species.

\[ \frac{\partial}{\partial t}(\rho Y_i) + \nabla \cdot (\rho v Y_i) = -\nabla \cdot J_i + R_i + S_i \]  

Where \( \rho \) is density, \( Y_i \) is the mass fraction of species \( i \), \( v \) is velocity, \( R_i \) is the production of species \( i \), and \( S_i \) is a source term of species \( i \). \( J_i \) is the diffusion flux of species \( i \) and is given by Equation (7)

\[ J_i = -\rho D_{i,m} \nabla Y_i - D_{T,i} \frac{\nabla T}{T} \]  

Where \( \rho \), and \( Y_i \) retain their definition from Equation (6), \( D_{i,m} \) is the mass diffusion coefficient for species \( i \), \( D_{T,i} \) is the thermal diffusion coefficient, and \( T \) is temperature. Equations (1)-(7) and their descriptions were pulled from the ANSYS Fluent theory guide, which is reference [4] for interested readers.

For the model in this work, a 2D domain was designed using DesignModeler, which is ANSYS’s built in CAD environment. As the APPJ is symmetric about the jet axis, a full 3D model is not needed and using a 2D model saves significant computational cost. The domain is displayed in Figure 2 and consists of a tube on top of a 0.4 m x 0.4 m boxed area. The tube at the top of the domain represents the quartz tube from Figure 1 and has the same inner diameter of 4 mm. The top of the tube is just after the end of the inner electrode of the APPJ and serves as the inlet where helium is injected. The 0.4 m x 0.4 m area represents the large surrounding air that the helium diffuses into. The area is large enough to allow the helium to diffuse but small enough for efficient calculation. Fluid can exit the domain at all boundaries except the quartz tube walls. The domain is discretized using a mesh that breaks the domain into small elements in which to calculate fluid properties. This mesh was generated using ANSYS’s meshing tool and checked after loading into Fluent to ensure relative uniformity in the mesh. The domain starts as a purely air domain, and helium is injected into the system. The model runs until a steady state solution is found for the flow, at which point the mixture can be examined.
For mixture analysis the mass fraction was the primary parameter investigated as it shows how concentrated helium is in different regions of the domain. However, as helium and air have significantly different masses, the mole fraction, which is computed from the mass fraction, was utilized and gives a measure of the percentage of helium in a region. To understand the effect of the flow rate on the mole fraction, the minimum mole fraction needed for propagation was determined. This determination was done by placing a line a distance from the quartz tube end based on experimental results in [1]. The minimum value of the helium mole fraction was then increased until the jet no longer crossed the line indicating experimental length. The value just before the jet no longer crossed the line is the minimum mole fraction of helium needed for the plasma to propagate. Measuring this minimum fraction at each flow rate allowed a trend based on flow rate to be determined.

Results and Discussion

The helium mole fraction results are shown for each flow rate in Figure 3 through Figure 8. The black line in each figure placed at a distance away from the exit consistent with the jet lengths in Figure 15 of [1]. Note that the color scales in each image are different to show how the jet appears near each minimum mole fraction. However, each image is centered on the same point.
Figure 3: Helium Mole Fraction for 1 slm

Figure 4: Helium Mole Fraction for 2 slm
Figure 5: Helium Mole Fraction for 3 slm

Figure 6: Helium Mole Fraction for 4 slm
Figure 7: Helium Mole Fraction for 5 slm

Figure 8: Helium Mole Fraction for 6 slm
From these Figures, it is immediately apparent that the jet consists of a strong helium core that displaces the air. Surrounding this core is a small region where the helium quickly diffuses into the surrounding air. However, at the end of the jet, the core tapers to a point, leaving a much larger mixture region. As RONS are produced due to the interaction of the helium with the surrounding air, the results show the expected results that the tip of the jet is the best treatment area. The value of each minimum mole fraction is given in Table 1. Figure 9 plots the minimum helium mole fraction as a function of flow rate to examine the trend.

![Table 1: Minimum Helium Mole Fraction](image)

<table>
<thead>
<tr>
<th>Flow Rate (slm)</th>
<th>Minimum Helium Mole Fraction</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>0.67</td>
</tr>
<tr>
<td>2</td>
<td>0.87</td>
</tr>
<tr>
<td>3</td>
<td>0.88</td>
</tr>
<tr>
<td>4</td>
<td>0.91</td>
</tr>
<tr>
<td>5</td>
<td>0.93</td>
</tr>
<tr>
<td>6</td>
<td>0.93</td>
</tr>
</tbody>
</table>

**Figure 9: Minimum Helium Mole Fraction vs Flow Rate**

As is seen in both Table 1 and Figure 9, the minimum helium mole fraction increases significantly at lower flow rates, but quickly tapers to values near 0.9 past 3 slm, having a behavior similar to a logarithmic function. There are three main explanations for this trend. First is the fact that the mole fraction cannot have a value greater than 1. Thus, as the mole fraction is approaching a mathematical asymptote, it makes sense that the plot behaves like it is approaching an asymptote. Second, turbulence effects must be considered. As it was shown that turbulence past 3 slm resulted in a significant decrease in the growth of the jet length, it is no surprise that the
minimum helium mole fraction, which is calculated based on the experimental jet length, is also impacted. Finally, and perhaps the best explanation, is the growth of higher fraction regions. Comparing Figure 7 and Figure 8 best illustrates the difference as the two plots have the same color scale. As flow rate increases, the higher helium mole fraction regions also see a significant increase in their length. As the same amount of energy is being applied to ionize the plasma in each flow rate, it makes sense that the minimum fraction will appear to increase, even though it may be that the finite energy is getting absorbed by the growth of high fraction regions. Thus, while the growth in minimum helium fraction is consistent with previous results [5], the trend may not be as significant as first appears.

Plasma Modeling

Methodology

While the fluid-based modeling began with ANSYS Fluent, the plasma model utilized COMSOL’s plasma module. The switch to COMSOL was made for two primary reasons. The first is that COMSOL’s plasma module is a robust and well-developed plasma modeling software that was available to PERL. ANSYS, where the previous fluid modeling was performed does not have a robust plasma modeling capability. The second reason is COMSOL is built as a Multiphysics software. Thus, COMSOL is easily able to combine the fluid and plasma behaviors of the APPJ. This ease is again contrasted to ANSYS where a pathway to combining fluid and plasma models was not clear.

COMSOL’s plasma module solves transport equations for the electrons and each heavy species within the domain. Electron transport is governed by the Boltzmann equation, which is shown in Equation (8).

\[
\frac{\partial f}{\partial t} + v \cdot \nabla f - \frac{e}{m} E \cdot \nabla_e f = C[f]
\]  

(8)

Where \(f\) is the electron energy distribution function (EEDF), \(v\) is the velocity of electrons, \(e\) is the elementary charge, \(m\) is the electron mass, \(E\) is the electric field, and \(C\) is a source term due to collisions.

However, there does not currently exist an efficient closed-form solution to the Boltzmann equation, so approximations are used to solve electron transport. COMSOL’s plasma module uses the Drift-Diffusion approximation, which uses fluid-based governing equations for the electron transport. These equations are the electron number density, electron momentum, and mean electron energy and are given in Equations (9), (10), and (11), respectively.

\[
\frac{\partial}{\partial t}(n_e) + \nabla \cdot \Gamma_e = R_e
\]  

(9)

\[
\frac{\partial}{\partial t}(n_e m_e u_e) + \nabla \cdot n_e m_e u_e u_e^T = -(\nabla \cdot \boldsymbol{p}_e) + en_e E - n_e m_e u_e v_m
\]  

(10)

\[
\frac{\partial}{\partial t}(n_e) + \nabla \cdot \Gamma_e = S_{en}
\]  

(11)

Where \(n\) is number of particles, \(\Gamma\) is flux, \(R_e\) is electron source, \(m_e\) is electron mass, \(u_e\) is drift velocity, \(\boldsymbol{p}_e\) is the electron pressure tensor, \(e\) is elementary charge, \(E\) is the electric field, \(v_m\) is the momentum transfer frequency, and \(S_{en}\) is the energy transfer due to inelastic collisions. The subscript \(e\) indicates electron while the subscript \(\varepsilon\) indicates the mean electron energy. The fluxes are calculated using Equation (12).
\[ \Gamma = - (\mu \cdot E)n - \nabla (Dn) \]  

(12)

Where \( \mu \) is the mobility coefficient and \( D \) is the diffusion coefficient. Compared to the significant number of equations for the electron transport, the heavy species transport is solely governed by Equation (13).

\[
\rho \frac{\partial}{\partial t}(\omega_k) + \rho (\mathbf{u} \cdot \nabla) \omega_k = \nabla \cdot \mathbf{j}_k + R_k
\]  

(13)

Where \( \rho \) is the gas density, \( \omega_k \) is the mass fraction of species \( k \), \( \mathbf{u} \) is the velocity field for the gas, \( \mathbf{j}_k \) is the diffusive flux vector, and \( R_k \) is the source term for species \( k \).

From the governing equations, three major steps were taken to develop the model. First, the plasma chemistry and cross sections were imported into COMSOL. The plasma chemistry defines the species and reactions used in the model. The cross sections specify the production from each reaction, which forms the source terms found in the governing equations. The second major step was to import the tabulated transport properties such as the mobility and diffusion coefficients. These coefficients are used in determining the flux that governs much of the electron transport, so accurate tables are needed for a valid model. Finally, the boundary conditions of the model were defined. These conditions include species interactions with walls, species sources, E/M conditions, and fluid inlets and outlets. It should also be noted that a fluid solver is needed to determine the velocity field used in the transport equations. COMSOL uses a similar fluid solver to ANSYS Fluent but turbulence effects were not considered for these models due to the low flow rates.

Collisional cross sections are well tabulated and for this model were pulled from references [6] and [7]. However, the exact values of the mobility and diffusion coefficients depend upon a variety of factors and there is not a wide range of publicly available data. Thus, a pre-processing step must be performed to find these coefficients for the model. To do so we use COMSOL’s Boltzmann Solver, which solves for the EEDF which allows the coefficients to be determined. To solve the EEDF, a two-term approximation is used to write the EEDF in the form found in Equation (14).

\[
f(v, \cos \theta, z, t) = f_0(v, z, t) + f_1(v, z, t) \cos \theta
\]  

(14)

Where \( v \) is the velocity magnitude, \( \theta \) is the angle between the velocity and electric fields, \( z \) is the coordinate along the direction defined by \( \theta \), and \( t \) is time. Separation of variables is then applied to separate the energy dependence of the EEDF from the spatial and temporal dependences. This separated form is given in Equation (15).

\[
\frac{f_{0,1}(\varepsilon, z, t)}{2\pi \gamma F_{0,1}(\varepsilon)n(z, t)}
\]  

(15)

Where most of the variables retain their definitions from Equation (14, \( \varepsilon \) is mean electron energy, and \( \gamma \) is a constant. \( n(z, t) \) is defined by the growth model and allows for \( F_{0,1} \) to be found. From the \( F_{0,1} \) term in Equation 8, the coefficients can then be solved for. The mobility and diffusion coefficients for the electrons and electron energy are given in Equations (16)-(19).
\[\mu_e N = -\frac{\gamma}{3} \int_{0}^{\infty} \frac{\varepsilon}{\sigma_m} \frac{\partial F_0}{\partial \varepsilon} d\varepsilon\]  
\[D_e N = \frac{\gamma}{3} \int_{0}^{\infty} \frac{\varepsilon}{\sigma_m} F_0 d\varepsilon\]  
\[\mu_e N = -\frac{\gamma}{3 \varepsilon} \int_{0}^{\infty} \frac{\varepsilon^2}{\sigma_m} \frac{\partial F_0}{\partial \varepsilon} d\varepsilon\]  
\[D_e N = \frac{\gamma}{3 \varepsilon} \int_{0}^{\infty} \frac{\varepsilon^2}{\sigma_m} F_0 d\varepsilon\]  

Where \(\sigma_m\) represents the cross section of the species.

Additionally, rate and Townsend coefficients for the reactions used in the plasma chemistry can be calculated. Townsend coefficients are another method of specifying the reaction production rates. Townsend coefficients are important as they are typically more numerically stable for DC plasmas, such as PERL's APPJ, than using the cross sections to specify production rates. The rate coefficients are used given by Equation (20)

\[k_k = \gamma \int_{0}^{\infty} \varepsilon \sigma_k F_0 d\varepsilon\]  

Where \(k_k\) is the rate coefficient for reaction \(k\) and \(\sigma_k\) is the cross section for the reaction. The Townsend coefficients have two forms, depending on the growth model used, and are given by Equations (21) and (22).

\[\frac{\alpha_k}{N} = \frac{k_k \alpha}{\bar{v}}\]  
\[\alpha_k \frac{N}{k_k} = \frac{\alpha}{\mu E}\]  

Where \(\alpha_k\) is the Townsend coefficient for reaction \(k\), \(\alpha\) is the overall Townsend coefficient, \(\bar{v}\) is the net production frequency, \(\mu\) is the diffusion coefficient, and \(E\) is the electric field. The descriptions and governing equations of COMSOL's plasma module were pulled from the COMSOL plasma module user's guide, which is reference [8] for interested readers. The description of the Boltzmann solver equations is a summary of reference [9].

Initial attempts at developing the plasma model tried to build a simple helium chemistry from the ground up using the three major steps outlined earlier. However, it was found that this method encountered a number of issues with no clear root cause. Thus, work shifted to reproducing a similar plasma model that could then be augmented to fit PERL's APPJ. An argon-based APPJ model with a simple chemistry was found and is detailed in reference [10]. The simple argon chemistry used is detailed in Table 2. Note that the metastables of argon are combined into a single metastable state, which is represented by Ars.
Table 2: Argon Model Chemistry

<table>
<thead>
<tr>
<th>Reaction</th>
<th>Type</th>
</tr>
</thead>
<tbody>
<tr>
<td>Ar+e-&gt;Ar+e</td>
<td>Elastic</td>
</tr>
<tr>
<td>Ar+e-&gt;Ars+e</td>
<td>Excitation</td>
</tr>
<tr>
<td>Ars+e-&gt;Ar+e</td>
<td>Excitation</td>
</tr>
<tr>
<td>Ar+e-&gt;e+e+Ar+</td>
<td>Ionization</td>
</tr>
<tr>
<td>Ars+e-&gt;e+e+Ar+</td>
<td>Ionization</td>
</tr>
</tbody>
</table>

The computational domain of the model is shown in Figure 10. The model is a 2D axi-symmetric model due to the symmetry of the APPJ and allowing for a reduced computational cost compared to 3D models. The argon enters through the bottom left-hand corner of the domain. The structure in the lower left hand side consists of the ionizing electrode and grounding electrode with a dielectric sandwiched in between. Like in the fluid model, the inlet is surrounded by a large surrounding air domain to allow the argon ample space to expand and be free from computational artifacts. A close up of the inlet region with labeled boundaries is given in Figure 11 while Table 3 details boundary conditions surrounding the inlet.

Figure 10: Argon Model Domain
Figure 11: Close-up of Argon Inlet

Table 3: Argon Inlet Boundary Conditions

<table>
<thead>
<tr>
<th>Boundary</th>
<th>Label</th>
<th>E/M Boundary</th>
<th>Plasma Boundary</th>
<th>Fluid Boundary</th>
</tr>
</thead>
<tbody>
<tr>
<td>A-B</td>
<td>Ionizing Electrode</td>
<td>Metal Contact</td>
<td>Surface Reaction</td>
<td>Wall</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Ar⁺→Ar</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Ars→Ar</td>
<td></td>
</tr>
<tr>
<td>A-F</td>
<td>Gas Inlet</td>
<td>N/A</td>
<td>Electron Density and Energy</td>
<td>Insulation</td>
</tr>
<tr>
<td>B-C</td>
<td>Dielectric Insulator</td>
<td>Dielectric Contact</td>
<td>Surface Reaction</td>
<td>Wall</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Ar⁺→Ar</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Ars→Ar</td>
<td></td>
</tr>
<tr>
<td>C-D</td>
<td>Grounded Electrode</td>
<td>Ground</td>
<td>Surface Reaction</td>
<td>Wall</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Ar⁺→Ar</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Ars→Ar</td>
<td></td>
</tr>
</tbody>
</table>
Results and Discussion

From the Boltzmann solver, the electron transport coefficients and Townsend coefficients for each reaction were calculated as a function of the mean electron energy of the plasma. The electron transport coefficients are shown in Figure 12 while the Townsend coefficients are shown in Figure 13.

Figure 12: Electron Transport Coefficients for Argon Chemistry
Figure 13: Townsend Coefficients for Argon Chemistry

From the plots of these coefficients, the accuracy of the coefficients is apparent. Simplifications to plasma models are typically made by assuming that the electron transport coefficients are a constant value. Thus, the fact that the coefficients quickly reach a near constant value is to be expected. However, the significant variation in these coefficients at very low energy emphasizes the need to accurate tabulated coefficients for a complete model. The accuracy of the Townsend coefficients is best seen in reaction 5. The ionization energy for reaction 5 is 4.427 eV, so it is expected that the reaction rate of reaction 5 will be highest when the mean energy is close to, but slightly higher than 4.427 eV, as it is in Figure 13. While this cutoff may cause confusion as to why there is any ionization with mean energies lower than 4.427 eV, it is important to remember that the x axis is a mean, so some particles will have energy higher than the ionization threshold.

Despite the accurate coefficients however, a complete plasma model was not successfully developed. High-pressure plasmas push the limits of what plasma models are capable of, which means such models can suffer from numerical instabilities that result in the model crashing. To rectify this issue, setting the model to easier conditions and scaling up was tried. More specifically, rather than starting with an applied voltage of 1000 V, starting the simulated at 100 V and then increasing it by 100 V while using the previous solution as an initial condition was attempted. The electron temperature and density plots for 100 V, 200 V, and 300 V are shown in Figure 14 through Figure 19 which are a close up of the argon inlet. In the 100 V and 200 V electron density plots a jet structure can be seen along the axis of symmetry, indicating that the coupling between fluid results and plasma model are working as expected as the fluid flow is displacing electrons. In both cases, there is also a much lower density near the electrodes than elsewhere in the domain, which means more plasma is being neutralized than is being created. For the lower applied voltages this is to be expected as the applied voltage is not strong enough to produce significant amounts of plasma. The electron temperature plots between the 100 V and 200 V cases also indicate the development of the plasma model. The temperature plots see the formation of a plasma
bullet, which is to be expected [1]. Additionally, this bullet grows from the 100 V case to 200 V case, as more energy is being applied to the system.

Figure 14: Electron Density with 100 V Applied
Figure 15: Electron Temperature with 100 V Applied

Figure 16: Electron Density with 200 V Applied
Figure 17: Electron Temperature with 200 V Applied

Figure 18: Electron Density with 300 V Applied
Despite the encouraging results from the 100 V and 200 V cases, the 300 V case demonstrates that the model is not fully developed. The main indicator is the time stamp of 2.93e-5 s while the 100 V and 200 V cases ran for the full time of 0.001 s. The 300 V case crashed before reaching the full run time. Analysis of both Figure 18 and Figure 19 don’t point to a clear root cause. Figure 18 as a small high-density region of plasma along the axis of symmetry which could indicate that the model is calculating an electron density that is non-physical and causes a singularity. However, from the electron temperature plot, it appears that the plasma is arcing from the ionizing to the grounding electrode. While both plots could provide some explanation, it is worth noting that neither of these behaviors are expected at 300 V, which is not significantly more power than the 100 V and 200 V cases. Further investigation into these discrepancies is required to complete the model development.

Conclusion

In this work multiple fluid and plasma models were developed to characterize the performance of an atmospheric pressure plasma jet. Fluid models of the APPJ were able to successfully capture the turbulence and mixing behavior. The mixture behavior indicated that the tip of the jet is the best location for plasma treatment. The minimum helium mole fraction indicated that while previous research [5] seemed to indicate a relationship between flow rate and minimum fraction, there are likely other energy conversation effects to cause the observed trend. Plasma modeling of the APPJ was able to successfully calculate transport and reaction coefficients but did not capture the electron temperature and density of the APPJ. Lower voltage results indicate that scaling the voltage was resulting in a more accurate model, but an undetermined cause or causes prevented the model from being scaled to the operating voltage. If there are readers interested in completing the model development, it is recommended to first solve the scaling issue to replicate the argon model. From there, the model can be adjusted to a helium chemistry and a geometry that matches PERL’s APPJ. The final add on is to include reactions between the helium of the jet and the surrounding air to be able to determine the production of RONS and complete model
development. With the model developed, characterizing the APPJ can be done through varying input conditions and analyzing the differences as a function of those conditions.

References


4. ANSYS Inc., (2022) Fluent Theory Guide. ANSYS.


