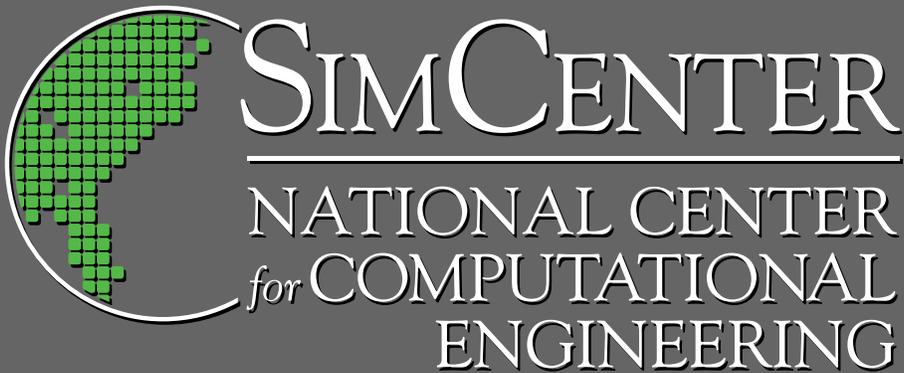


THE UNIVERSITY of TENNESSEE at CHATTANOOGA
COLLEGE of ENGINEERING and COMPUTER SCIENCE



Using Sensitivity Derivatives for the Purpose of Design in a High Pressure Glow Discharge Simulation

A Technical Report by

Kyle Lange and W. Kyle Anderson

UTC-CECS-SimCenter-2008-05

November 2008

GRADUATE SCHOOL OF COMPUTATIONAL ENGINEERING
701 East M.L. King Boulevard • Chattanooga, TN 37403

Contents

1	Acknowledgments	2
2	Abstract	3
3	Introduction	4
4	Description of the Model	6
4.1	Governing Equations	6
4.2	Transport Properties	9
4.3	Chemical Reactions	9
4.4	Boundary Conditions	12
5	Numerical Method	13
5.1	Mesh Discretization	13
5.2	Spatial Discretization	14
5.3	Temporal Discretization	16
5.4	Solving the Linear System	16
6	Computing Sensitivity Derivatives	18
6.1	Perturbation Methods	18
6.2	Differentiation Methods	18
7	Computational Results and Discussion	23
7.1	Verification	23
7.2	Using Sensitivity Derivatives to Increase a Given Cost Function	23
7.3	Using Sensitivity Derivatives to Obtain a Better Estimate for Uncertain Parameters	25
7.4	Using Sensitivity Derivatives to Compute a Bound on Computational Error	28
8	Summary	29
	References	30

1 Acknowledgments

In 1984, the state of Tennessee created a Center of Excellence program for public institutions of higher education within the state. The purpose of these centers, known as THEC (Tennessee Higher Education Commission) Centers of Excellence, is to build on the strength of the State's universities, expand the State's research base, and increase its national and international stature and economic competitiveness. The research reported herein was carried out under the auspices of the THEC Center of Excellence for Applied Computational Science and Engineering under the direction of Dr. Kyle Anderson at the University of Tennessee at Chattanooga under the project entitled "Analysis and Sensitivity Derivatives for Plasma Simulations". This support is gratefully acknowledged.

2 Abstract

The problem of applying sensitivity analysis to a one-dimensional high pressure radio frequency plasma discharge simulation is considered. The derivations of forward mode direct differentiation and the reverse mode adjoint method are presented. Sensitivity derivatives computed from these methods are then shown to match derivatives computed using complex perturbations. It is then demonstrated how sensitivity derivatives can be used within a design cycle to change experimentally variable quantities so as to increase or decrease a given cost function. It is also shown how sensitivity analysis could be used in conjunction with experimental data to obtain better estimates for rate and transport parameters. Finally, it is described how sensitivity derivatives of design variables can be used to compute error bounds for the computation of a given cost function.

3 Introduction

Since the seminal work of Graves and Jensen [1], there have been numerous efforts to accurately simulate plasma discharges [2] [3] [4] [5] [6] [7] [8], which are used in a variety of applications including microelectronics manufacturing, lighting, and plasma display panels. Many of these simulations have been motivated by a need to better understand the underlying physics in plasma discharges. A variety of approaches have been used including particle-in-cell methods, fluid simulations, and hybrid models.

There have also been efforts to use simulations of plasma discharges to perform parameter studies, or qualitative sensitivity analysis [9] [10] [11] [12]. The typical approach that is used is to change a given parameter for a plasma discharge (e.g. pressure, temperature) and run the simulation with the new value for the parameter to see what changes have taken place. This can provide valuable information about the sensitivity of the plasma behavior to various parameters and can be useful for optimization or design. However, this can be computationally expensive, as the simulation must be run every time that a parameter is changed. In addition, although this approach can provide qualitative information, it does not yield any quantitative information about sensitivity derivatives.

Quantitative sensitivity analysis, using direct differentiation or adjoint methods, can be used to compute numerical sensitivity derivatives of cost functions to given design variables. In recent years, sensitivity derivatives have been computed for unsteady time-dependent problems [13] [14] [15]. In a previous paper, the authors used time-dependent sensitivity analysis to compute sensitivity derivatives for a one-dimensional low-pressure helium discharge simulation [16], where only electrons and helium monomer ions were modeled. To the knowledge of the authors, this was the first time that quantitative sensitivity derivatives were computed for a plasma discharge simulation.

High pressure glow (HPG) discharge systems are more economical than low pressure discharge systems since no vacuum pump is needed. For this reason, there has been a plethora of research done on HPG plasmas in the last decade. Many applications of HPG discharges have been reported including etching and deposition [17] [18], decontamination of chemical and biological warfare agents [19], the treatment of dental cavities [20], and as a means for sterilization [21] [22].

In the current paper, quantitative sensitivity analysis has been applied to a one-dimensional high pressure helium/nitrogen glow discharge simulation. The paper is organized in the following manner. Section 4 describes the equations and boundary conditions used in the simulation. Section 5 describes the nu-

merical methods and discretization used to obtain a periodic solution. Section 6 presents the algorithms for forward mode direct differentiation and reverse mode adjoint, which are used in sensitivity analysis. Section 7 presents the computed sensitivity derivatives and several applications of sensitivity analysis for a plasma discharge simulation. The paper is summarized in Section 8.

4 Description of the Model

A one-dimensional fluid model accounting for ten chemical species is used to simulate a plasma discharge that was modeled in a previous work [11]. All values of coefficients for chemical reactions, diffusion coefficients, and mobilities used in this simulation are the same as those used in this previous work.

4.1 Governing Equations

The discharge gas is assumed to be predominantly helium with a very small amount of nitrogen. It is assumed that the nitrogen has an impurity mole fraction of 5×10^{-7} . The following chemical species are used in the model: electrons(e), helium atoms (He), monomer helium ions (He^+), dimer helium ions (He_2^+), monomer helium metastables (He^*), dimer helium metastables (He_2^*), monomer nitrogen atoms (N), dimer nitrogen molecules (N_2), monomer nitrogen ions (N^+), and dimer nitrogen ions (N_2^+). A continuity equation of the following form is used for each species:

$$\frac{\partial n_i}{\partial t} + \frac{\partial \Gamma_i}{\partial x} = \dot{G}_i, \quad (1)$$

where n_i is the density of species i , Γ_i is the flux for species i , and \dot{G}_i is the net production or destruction of species i through chemical reactions. Time is denoted as t and the interelectrode axial distance is denoted as x . A drift-diffusion approximation is used to compute the particle fluxes, where the flux for each species is composed of a term corresponding to drift from the electric field and a term that corresponds to diffusion. This is formally denoted as

$$\Gamma_k = -D_i \frac{\partial n_i}{\partial x} + \mu_i n_i E, \quad (2)$$

where D_i is the diffusion coefficient for species i , μ_i is the mobility for species i (the mobility is zero for neutral species), and E is the electric field which is defined by

$$E = -\frac{\partial V}{\partial x}, \quad (3)$$

where V is defined to be the potential. The ions and neutral particle species are assumed to be in thermal equilibrium with the background gas temperature.

However, as the plasma is nonthermal, one must have an equation for the electron temperature. The electron energy equation is given as

$$\frac{\partial}{\partial t} \left(\frac{3}{2} n_e k_B T_e \right) + \frac{\partial Q_e}{\partial x} = -e \Gamma_e E - e \sum_{j=1}^r \Delta E_j^e G_j - 3 n_e k_B \frac{m_e}{m_{He}} (T_e - T_{gas}) \bar{\nu}_e, \quad (4)$$

where k_B is the Boltzmann constant, T_e is the electron temperature, Q_e is the electron energy flux, ΔE_j^e is the energy lost per electron for reaction j , m_e is the electron mass, m_{He} is the mass of a helium atom, and $\bar{\nu}_e$ is the electron momentum transfer collision frequency for elastic collisions between electrons and the background gas. The electron energy flux is composed of a conductive and a convective term and is given as

$$Q_e = -\eta_e \frac{\partial T_e}{\partial x} + \frac{5}{2} k_B T_e \Gamma_e, \quad (5)$$

where η_e is the electron thermal conductivity, which is computed as

$$\eta_e = \frac{5}{2} n_e k_B D_e. \quad (6)$$

To complete the set of equations, the total current density in the discharge is specified, allowing the electric potential to be determined using the equation

$$\epsilon_0 \frac{\partial}{\partial t} \left(\frac{\partial V}{\partial x} \right) = -j_{tot} + \sum_{i=1}^{i=n_s} e Z_i \Gamma_i, \quad (7)$$

where ϵ_0 is the permittivity of free space, j_{tot} is the specified current density, n_s is the total number of charged particle species, and Z_i is the charge number of species i . It should be noted that j_{tot} varies sinusoidally during each RF cycle.

Bringing all the equations together allows one to express them in conservative form as

$$\frac{\partial \mathbf{Q}}{\partial t} + \nabla \cdot \mathbf{F} = \mathbf{S}, \quad (8)$$

where

$$\mathbf{Q} = \begin{pmatrix} n_e \\ n_{He} \\ n_{He^+} \\ n_{He_2^+} \\ n_{He^*} \\ n_{He_2^*} \\ n_N \\ n_{N_2} \\ n_{N^+} \\ n_{N_2^+} \\ \frac{3}{2}n_e k_B T_e \\ \frac{\partial V}{\partial x} \end{pmatrix}, \quad \mathbf{F} = \begin{pmatrix} \Gamma_e \\ \Gamma_{He} \\ \Gamma_{He^+} \\ \Gamma_{He_2^+} \\ \Gamma_{He^*} \\ \Gamma_{He_2^*} \\ \Gamma_N \\ \Gamma_{N_2} \\ \Gamma_{N^+} \\ \Gamma_{N_2^+} \\ Q_e \\ 0 \end{pmatrix}, \quad (9)$$

$$\mathbf{S} = \begin{pmatrix} \dot{G}_e \\ \dot{G}_{He} \\ \dot{G}_{He^+} \\ \dot{G}_{He_2^+} \\ \dot{G}_{He^*} \\ \dot{G}_{He_2^*} \\ \dot{G}_N \\ \dot{G}_{N_2} \\ \dot{G}_{N^+} \\ \dot{G}_{N_2^+} \\ -e\Gamma_e E - e \sum_{j=1}^r \Delta E_j^e G_j - 3n_e k_B \frac{m_e}{m_{He}} (T_e - T_{gas}) \\ -j_{tot} + \sum_{i=1}^{i=n_s} eZ_i \Gamma_i \end{pmatrix}. \quad (10)$$

4.2 Transport Properties

The values for transport properties are found in the literature [23] [24] [25] and are summarized in a previous work [11]. Diffusion coefficients can be computed from reduced diffusion coefficients as $D_k = D'_k \left(\frac{760}{p} \right)$, while mobilities can be computed from reduced mobilities as $\mu_k = \mu'_k \left(\frac{760}{p} \right)$, where p is measured in Torr.

The values for reduced diffusion coefficients and reduced mobilities are shown in Table 1.

Species	D'_k (m ² /sec, K, eV)	μ'_k (m ² /V-sec)
e	$1.737 \times 10^{-1} \left(\frac{T_e}{17406} \right)$	-1.132×10^{-1}
He ⁺	5.026×10^{-5}	1.482×10^{-3}
He ₂ ⁺	$8,148 \times 10^{-5}$	2.403×10^{-3}
N ⁺	9.710×10^{-5}	2.863×10^{-3}
N ₂ ⁺	1.015×10^{-4}	2.993×10^{-3}
He*	4.116×10^{-4}	-
He ₂ *	2.029×10^{-4}	-
He	4.116×10^{-4}	-
N	1.955×10^{-4}	-
N ₂	1.075×10^{-4}	-

Table 1
Transport Properties for species in plasma discharge model

4.3 Chemical Reactions

The values used for reaction rate coefficients and other coefficients in determining production and destruction of chemical species are found in previous works [6] [26] [27] [28] and are compiled in a previous work [11]. The reactions are summarized in Table 2. The electron energy gains and losses for given reactions are shown in Table 3.

Reaction	Numerical Model (m, molecules, sec, K)
$e + \text{He} \rightarrow \text{He}^* + e$	$G_1 = (2.308 \times 10^{-16})n_e n_{\text{He}} T_e^{0.31} e^{-2.297 \times 10^5 / T_e}$
$e + \text{He}^* \rightarrow \text{He} + e$	$G_2 = (1.099 \times 10^{-17})n_e n_{\text{He}^*} T_e^{0.31}$
$e + \text{He} \rightarrow \text{He}^+ + 2e$	$G_3 = (2.584 \times 10^{-18})n_e n_{\text{He}} T_e^{0.68} e^{-2.854092 \times 10^5 / T_e}$
$e + \text{He}^* \rightarrow \text{He}^+ + 2e$	$G_4 = (4.661 \times 10^{-16})n_e n_{\text{He}^*} T_e^{0.6} e^{-5.546 \times 10^4 / T_e}$
$e + \text{He}_2^* \rightarrow \text{He}_2^+ + 2e$	$G_5 = (1.268 \times 10^{-18})n_e n_{\text{He}_2^*} T_e^{0.71} e^{-3.945 \times 10^4 / T_e}$
$e + \text{He}_2^+ \rightarrow \text{He}^* + \text{He}$	$G_6 = (5.386 \times 10^{-13})n_e n_{\text{He}_2^+} T_e^{-0.5}$
$\text{He}^* + \text{He}^* \rightarrow \text{He}^+ + \text{He} + e$	$G_7 = (2.7 \times 10^{-16})n_{\text{He}^*} n_{\text{He}^*}$
$\text{He}^* + 2\text{He} \rightarrow \text{He}_2^* + \text{He}$	$G_8 = (1.3 \times 10^{-45})n_{\text{He}^*} n_{\text{He}} n_{\text{He}}$
$\text{He}^+ + 2\text{He} \rightarrow \text{He}_2^+ + \text{He}$	$G_9 = (1.0 \times 10^{-43})n_{\text{He}^+} n_{\text{He}} n_{\text{He}}$
$\text{He}^* + \text{N}_2 \rightarrow \text{N}_2^+ + \text{He} + e$	$G_{10} = (7.0 \times 10^{-17})n_{\text{He}^*} n_{\text{N}_2}$
$\text{He}_2^* + \text{N}_2 \rightarrow \text{N}_2^+ + 2\text{He} + e$	$G_{11} = (7.0 \times 10^{-17})n_{\text{He}_2^*} n_{\text{N}_2}$
$\text{He}^+ + \text{N}_2 \rightarrow \text{N}_2^+ + \text{He}$	$G_{12} = (5.0 \times 10^{-16})n_{\text{He}^+} n_{\text{N}_2}$
$\text{He}^+ + \text{N}_2 \rightarrow \text{N}^+ + \text{N} + \text{He}$	$G_{13} = (7.0 \times 10^{-16})n_{\text{He}^+} n_{\text{N}_2}$
$\text{He}_2^+ + \text{N}_2 \rightarrow \text{N}_2^+ + 2\text{He}$	$G_{14} = (5.0 \times 10^{-16})n_{\text{He}_2^+} n_{\text{N}_2}$
$\text{He}_2^+ + \text{N}_2 \rightarrow \text{N}^+ + \text{N} + 2\text{He}$	$G_{15} = (7.0 \times 10^{-16})n_{\text{He}_2^+} n_{\text{N}_2}$
$2e + \text{N}_2^+ \rightarrow \text{N}_2 + e$	$G_{16} = (5.651 \times 10^{-39})n_e n_e n_{\text{N}_2^+} T_e^{-0.8}$
$e + \text{N}_2^+ \rightarrow \text{N} + \text{N}$	$G_{17} = (2.540 \times 10^{-12})n_e n_{\text{N}_2^+} T_e^{-0.5}$
$e + \text{N}_2 \rightarrow \text{N} + \text{N} + e$	$G_{18} = (1.959 \times 10^{-12})n_e n_{\text{N}_2} T_e^{-0.7} e^{-1.132 \times 10^5 / T_e}$
$e + \text{N} \rightarrow \text{N}^+ + 2e$	$G_{19} = (8.401 \times 10^{-11})n_e n_{\text{N}} e^{-1.682 \times 10^5 / T_e}$
$e + \text{N}_2 \rightarrow \text{N}_2^+ + 2e$	$G_{20} = (4.483 \times 10^{-13})n_e n_{\text{N}_2} T_e^{-0.3} e^{-1.81 \times 10^5 / T_e}$

Table 2

Reaction Mechanisms Used in Plasma Discharge Model

Reaction	Energy Gain or Loss (eV)
$e + \text{He} \rightarrow \text{He}^* + e$	$\Delta E_1^e = 19.8$
$e + \text{He}^* \rightarrow \text{He} + e$	$\Delta E_2^e = -19.8$
$e + \text{He} \rightarrow \text{He}^+ + 2e$	$\Delta E_3^e = 24.6$
$e + \text{He}^* \rightarrow \text{He}^+ + 2e$	$\Delta E_4^e = 4.78$
$e + \text{He}_2^* \rightarrow \text{He}_2^+ + 2e$	$\Delta E_5^e = 3.4$
$\text{He}^* + \text{He}^* \rightarrow \text{He}^+ + \text{He} + e$	$\Delta E_7^e = -15.0$
$e + \text{N}_2 \rightarrow \text{N} + \text{N} + e$	$\Delta E_{18}^e = 9.757$
$e + \text{N} \rightarrow \text{N}^+ + 2e$	$\Delta E_{19}^e = 14.5$
$e + \text{N}_2 \rightarrow \text{N}_2^+ + 2e$	$\Delta E_{20}^e = 15.6$

Table 3
Electron Energy Gains and Losses Used in Plasma Discharge Model

4.4 Boundary Conditions

The fluxes for the electrons, metastable species and monomer nitrogen atoms at the boundaries are assumed to be kinetically limited thermal fluxes. The ion fluxes are assumed to be mobility limited. Recombination is assumed to occur at the boundaries, and all helium and nitrogen particles are assumed to return to the discharge as either He or N₂ particles. The electron temperature is set equal to 0.5 eV at each boundary. At one boundary of the discharge, the potential is grounded. At both boundaries, the voltage is computed using the specified boundary fluxes of the charged particles. It is assumed that there is no secondary electron emission at the boundaries. The boundary conditions for each particle species are summarized in Table 4.

Species	Boundary Condition
e	$\Gamma_e = \mp \frac{1}{4} n_e \sqrt{\frac{8k_B T_e}{\pi m_e}}$
He ⁺	$\Gamma_{He^+} = \begin{cases} \mu_{He^+} n_{He^+} E & \text{if } \vec{E} \cdot \hat{n} > 0 \\ 0 & \text{if } \vec{E} \cdot \hat{n} \leq 0 \end{cases}$
He ₂ ⁺	$\Gamma_{He_2^+} = \begin{cases} \mu_{He_2^+} n_{He_2^+} E & \text{if } \vec{E} \cdot \hat{n} > 0 \\ 0 & \text{if } \vec{E} \cdot \hat{n} \leq 0 \end{cases}$
He*	$\Gamma_{He^*} = \mp \frac{1}{4} n_{He^*} \sqrt{\frac{8k_B T_{gas}}{\pi m_{He}}}$
He ₂ *	$\Gamma_{He_2^*} = \mp \frac{1}{4} n_{He_2^*} \sqrt{\frac{8k_B T_{gas}}{2\pi m_{He}}}$
N	$\Gamma_N = \mp \frac{1}{4} n_N \sqrt{\frac{8k_B T_{gas}}{\pi m_N}}$
N ⁺	$\Gamma_{N^+} = \begin{cases} \mu_{N^+} n_{N^+} E & \text{if } \vec{E} \cdot \hat{n} > 0 \\ 0 & \text{if } \vec{E} \cdot \hat{n} \leq 0 \end{cases}$
N ₂ ⁺	$\Gamma_{N_2^+} = \begin{cases} \mu_{N_2^+} n_{N_2^+} E & \text{if } \vec{E} \cdot \hat{n} > 0 \\ 0 & \text{if } \vec{E} \cdot \hat{n} \leq 0 \end{cases}$
He	$\Gamma_{He} = - \left(\Gamma_{He^+} + 2\Gamma_{He_2^+} + \Gamma_{He^*} + 2\Gamma_{He_2^*} \right)$
N ₂	$\Gamma_{N_2} = - \left(\frac{1}{2}\Gamma_{N^+} + \Gamma_{N_2^+} + \frac{1}{2}\Gamma_N \right)$

Table 4
Boundary conditions for species in plasma discharge model

5 Numerical Method

5.1 Mesh Discretization

The domain is discretized into control volumes by distributing mesh points across the domain according to an algorithm that clusters the points close to the boundaries, which has been used in previous simulations [29]. The faces of the control volumes are designated to be halfway in between the mesh points except for at the boundaries, where the mesh points are coincident with the cell faces. The mesh setup is shown in figure 1.

The spatial discretization is represented with subscripts by the index k . Integer values such as k and $k+1$ denote a value within a given cell. Half-integer values such as $k + \frac{1}{2}$ and $k - \frac{1}{2}$, denote the values at cell faces. The distance between points k and $k+1$ is denoted as Δx_k while the distance between faces $k + \frac{1}{2}$ and $k - \frac{1}{2}$, the cell volume, is represented by Ω_k .

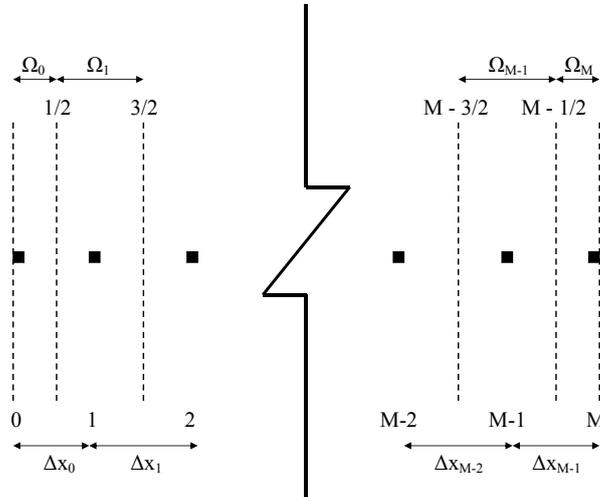


Fig. 1. Discretization of domain into M control volumes with $M + 1$ grid points

5.2 Spatial Discretization

A finite volume approach is used to discretize the equations. A staggered mesh and non-staggered mesh approach have each been implemented for the simulation. Although both approaches yield the same results, only results for the staggered grid approach are presented here. In this approach, the values for the potential V are stored at cell faces, and values for the remaining variables (species densities and electron energy) are stored at cell centers. For M control volumes, there are M unknowns for each species densities and the electron energy, with $M + 1$ unknowns for the voltage. However, since the voltage is grounded at one electrode, and its value does not change during the simulation, this results in M unknowns for the voltage, making this a straightforward system of equations to solve since each of the M control volumes has 12 unknown variables associated with it, corresponding to the 10 particle species densities, the electron energy and the voltage.

Applying a finite volume method to the governing equations and applying Green's theorem to Eq. 8 results in the following discretization:

$$\Omega_k \frac{\partial \mathbf{Q}_k}{\partial t} + \mathbf{F}_{k+\frac{1}{2}} - \mathbf{F}_{k-\frac{1}{2}} = \Omega_k \mathbf{S}_k. \quad (11)$$

The species continuity equation is discretized for species i as:

$$\frac{\partial n_{i,k}}{\partial t} \Omega_k = - \left(\Gamma_{i,k+\frac{1}{2}} - \Gamma_{i,k-\frac{1}{2}} \right) + \Omega_k \dot{G}_i. \quad (12)$$

For neutral and metastable particles, the flux is entirely diffusive and is discretized as

$$\Gamma_{i,k+\frac{1}{2}} = -D_i \left(\frac{n_{i,k+1} - n_{i,k}}{\Delta x_k} \right). \quad (13)$$

For charged particles, the flux is composed of a diffusive term and convective term and is discretized as

$$\Gamma_{i,k+\frac{1}{2}} = -D_{i,k+\frac{1}{2}} \left(\frac{n_{i,k+1} - n_{i,k}}{\Delta x_k} \right) + \mu_i n_{i,k+\frac{1}{2}} \left(\frac{1}{2} (E_k + E_{k+1}) \right). \quad (14)$$

In Eq. 14, the species density that is used at the electric field is computed differently based on the sign of the electric field and the species charge. It is computed as:

$$n_{i,k+\frac{1}{2}} = \begin{cases} n_{i,k} + \frac{1}{2}\Delta x_k \left(\frac{\partial n}{\partial x}\right)_{i,k} & \text{if } Z_i E \geq 0 \\ n_{i,k+1} - \frac{1}{2}\Delta x_k \left(\frac{\partial n}{\partial x}\right)_{i,k+1} & \text{if } Z_i E < 0 \end{cases} \quad (15)$$

The electric field in Eq. 14 is computed as

$$E_k = -\frac{(V_{k+\frac{1}{2}} - V_{k-\frac{1}{2}})}{\Omega_k}. \quad (16)$$

The electron energy equation is discretized as

$$\begin{aligned} \frac{\partial}{\partial t} \left(\frac{3}{2}n_e k T_e\right)_{i,k} \Omega_k = & - (Q_{e,k+\frac{1}{2}} - Q_{e,k-\frac{1}{2}}) - \\ & - \frac{1}{2} (e\Gamma_{e,k+\frac{1}{2}} E_{k+\frac{1}{2}} \Delta x_k + e\Gamma_{e,k-\frac{1}{2}} E_{k-\frac{1}{2}} \Delta x_{k-1}) - \\ & - e \sum_{i=1}^r \Delta E_i^e G_i \Omega_k + 3k_B n_{e,k} \frac{m_e}{m_{He}} (T_{e,k} - T_{gas}) \bar{v}_e \Omega_k, \end{aligned} \quad (17)$$

where the electron energy flux is discretized as

$$Q_{e,k+\frac{1}{2}} = -\eta_{e,k+\frac{1}{2}} \left(\frac{T_{e,k+1} - T_{e,k}}{\Delta x_k}\right) + \frac{5}{2} k_B \Gamma_{e,k+\frac{1}{2}} \left[\frac{1}{2} (T_{e,k+1} + T_{e,k})\right], \quad (18)$$

and the electron thermal conductivity is computed as

$$\eta_{e,k+\frac{1}{2}} = \frac{5}{2} D_e \left(\left[\frac{1}{2} (T_{e,k} + T_{e,k+1})\right]\right) \left[\frac{1}{2} (n_{e,k} + n_{e,k+1})\right] k_B. \quad (19)$$

The current conservation equation is discretized as

$$\begin{aligned} \frac{\partial}{\partial t} \left(\frac{V_{k+\frac{1}{2}} - V_{k-\frac{1}{2}}}{\Omega_k} \right) \Omega_k = -j_{tot} \Omega_k + \\ + \sum_{i=1}^{i=n_s} e Z_i \left(\frac{1}{2} \Delta x_k \Gamma_{i,k+\frac{1}{2}} + \frac{1}{2} \Delta x_{k-1} \Gamma_{i,k-\frac{1}{2}} \right). \end{aligned} \quad (20)$$

All of the above equations are nondimensionalized for stable and efficient computation. They are nondimensionalized in a way that is similar to previous computational models [7] [29].

5.3 Temporal Discretization

All time derivatives are computed using a second order backwards difference formula as

$$\left(\frac{\partial a}{\partial t} \right)_k^n = \frac{3a_k^n - 4a_k^{n-1} + a_k^{n-2}}{2\Delta t}, \quad (21)$$

where Δt is the time step used in the simulation. A fully implicit method is used, meaning that all fluxes and source terms are evaluated at the current time step n .

5.4 Solving the Linear System

Newton's method is used to solve the discretized linear equations at each time step. At each time step, the residual is driven to zero and is defined using Eq. 11 as

$$\mathbf{R}_k^n(\mathbf{Q}) = \Omega_k \left(\frac{\partial \mathbf{Q}_k}{\partial t} \right)^n + \mathbf{F}_{k+\frac{1}{2}}^n - \mathbf{F}_{k-\frac{1}{2}}^n - \Omega_k \mathbf{S}_k^n = 0 \quad (22)$$

For a given Newton iteration $m+1$ for a given time step n , the update ($\Delta \mathbf{Q}^{m,n}$) to the variable vector from the previous Newton iteration m is computed as:

$$\left(\frac{\partial \mathbf{R}}{\partial \mathbf{Q}} \right)^{m+1,n} (\Delta \mathbf{Q})^{m,n} = -\mathbf{R}(\mathbf{Q}^{m+1,n}), \quad (23)$$

$$\Delta \mathbf{Q}^{m,n} = \mathbf{Q}^{m+1,n} - \mathbf{Q}^{m,n} \quad (24)$$

The Newton iterations continue within a given time step until the norm of the residual in a given Newton iteration falls below a specified tolerance. This indicates that the updated value \mathbf{Q}^n has been computed, and the solution is advanced to the next time step. This process is repeated until a periodic solution has been computed.

In this simulation, the logarithms of the charged particle densities and the logarithm of the electron energy are used as fundamental variables. This is done because these quantities are inherently positive, and because second order spatial accuracy in the finite volume scheme is achieved by extrapolating values from cell centers to cell faces. In regions where there are steep gradients, this extrapolation can lead to negative densities or a negative electron energy.

Newton's method requires the computation of the Jacobian matrix $\frac{\partial \mathbf{R}}{\partial \mathbf{Q}}$, the linearization of the residual vector \mathbf{R} with respect to the variable vector \mathbf{Q} . The Jacobian matrix is computed analytically for computational efficiency. For this one-dimensional simulation with second-order spatial accuracy, the Jacobian matrix is block pentadiagonal. The problem is parallelized to run on four processors and uses GMRES [30] with an approximate ILU preconditioner [31].

6 Computing Sensitivity Derivatives

6.1 Perturbation Methods

Perturbation methods rely on a perturbation in a design variable β to compute the sensitivity with regard to a cost function I . Using a Taylor series approximation, a sensitivity derivative can be computed using finite differences as

$$\frac{dI}{d\beta} = \frac{I(\beta + \Delta\beta) - I(\beta)}{\Delta\beta}. \quad (25)$$

This can be useful in some cases, but it is subject to cancellation error and can be sensitive to the size of the perturbation $\Delta\beta$. One way to avoid the cancellation error is to use a complex perturbation [32] [33] [34]. In this case, the entire computer code is converted to complex numbers and the design variable is perturbed in the complex plane, allowing the sensitivity derivative to be computed as the imaginary part of the computed cost function as

$$\frac{dI}{d\beta} = \text{Imag} \left(\frac{I(\beta + i\Delta\beta)}{\Delta\beta} \right). \quad (26)$$

The cancellation error is eliminated in this case, but using complex numbers can significantly increase the computational load for a simulation. Both perturbation methods can compute a sensitivity derivative vector for multiple cost functions with respect to one design variable, but they suffer from the fact that the process must be repeated for each design variable.

6.2 Differentiation Methods

Two different types of differentiation methods are used to compute sensitivity derivatives. The first method is forward mode direct differentiation [35], which can be used to compute the sensitivity derivatives of an unlimited number of cost functions with respect to one design variable. In direct differentiation, the sensitivity derivatives are computed as the solution advances in time. The second method is the reverse mode adjoint method [36] [37] [13], which computes the sensitivity of one cost function to an unlimited number of design variables. In this case, after the periodic solution has been reached, the sensitivity derivatives are computed using solution values at each previous time step, going all the way back to the initial conditions.

Differentiation methods can be implemented continuously [38] [39] [40] or discretely [41] [42] [43]. The continuous sensitivity method differentiates the governing equations first with respect to the design variables, and then discretizes them, while a discrete sensitivity method discretizes the governing equations first and then differentiates them with respect to design variables. Discrete sensitivity methods are used in this work.

Much of the following nomenclature for time-dependent sensitivity analysis has been expressed in previous papers [15] [16].

6.2.1 Forward Mode Direct Differentiation

For a periodic solution, a given cost function depends on data at each time step during the periodic cycle. For this reason, it is convenient to define a global cost function I^g along with a local cost function I^n for each time step, where

$$I^g = \frac{1}{N} \sum_{n=1}^{n=N} I^n, \quad (27)$$

and there are N time steps in the given period. Note that the local cost function I^n is only computed at time steps during the periodic cycle when the cost function is computed, and is not computed at every time step during the solution process. For a time-dependent simulation that runs for n_{tot} time steps, the global sensitivity derivative with respect to a design variable (ignoring mesh sensitivity) can be expressed in terms of the local sensitivity derivatives as

$$\frac{dI^g}{d\beta} = \sum_{n=1}^{n=n_{tot}} \frac{dI^g}{dI^n} \left[\frac{\partial I^n}{\partial \mathbf{Q}^n} \frac{\partial \mathbf{Q}^n}{\partial \beta} + \frac{\partial I^n}{\partial \beta} \right] \quad (28)$$

At each time step, the residual function \mathbf{R} is driven down to zero. Differentiating the residual function with respect to β , one obtains

$$\frac{d\mathbf{R}^n}{d\beta} = \frac{\partial \mathbf{R}^n}{\partial \mathbf{Q}^n} \frac{\partial \mathbf{Q}^n}{\partial \beta} + \frac{\partial \mathbf{R}^n}{\partial \mathbf{Q}^{n-1}} \frac{\partial \mathbf{Q}^{n-1}}{\partial \beta} + \frac{\partial \mathbf{R}^n}{\partial \mathbf{Q}^{n-2}} \frac{\partial \mathbf{Q}^{n-2}}{\partial \beta} + \frac{\partial \mathbf{R}^n}{\partial \beta} = 0. \quad (29)$$

Thus, one can solve for $\frac{\partial \mathbf{Q}^n}{\partial \beta}$ as

$$\frac{\partial \mathbf{Q}^n}{\partial \beta} = - \left[\frac{\partial \mathbf{R}^n}{\partial \mathbf{Q}^n} \right]^{-1} \left(\frac{\partial \mathbf{R}^n}{\partial \mathbf{Q}^{n-1}} \frac{\partial \mathbf{Q}^{n-1}}{\partial \beta} + \frac{\partial \mathbf{R}^n}{\partial \mathbf{Q}^{n-2}} \frac{\partial \mathbf{Q}^{n-2}}{\partial \beta} + \frac{\partial \mathbf{R}^n}{\partial \beta} \right). \quad (30)$$

Substituting Eq. 30 into Eq. 28 results in the following expression for the global sensitivity derivative:

$$\begin{aligned}
\frac{dI^g}{d\beta} = & \sum_{n=1}^{n=n_{tot}} \frac{dI^g}{dI^n} \left[-\frac{\partial I^n}{\partial \mathbf{Q}^n} \left[\frac{\partial \mathbf{R}^n}{\partial \mathbf{Q}^n} \right]^{-1} \frac{\partial \mathbf{R}^n}{\partial \mathbf{Q}^{n-1}} \frac{\partial \mathbf{Q}^{n-1}}{\partial \beta} \right] + \\
& + \sum_{n=1}^{n=n_{tot}} \frac{dI^g}{dI^n} \left[-\frac{\partial I^n}{\partial \mathbf{Q}^n} \left[\frac{\partial \mathbf{R}^n}{\partial \mathbf{Q}^n} \right]^{-1} \frac{\partial \mathbf{R}^n}{\partial \mathbf{Q}^{n-2}} \frac{\partial \mathbf{Q}^{n-2}}{\partial \beta} \right] + \\
& + \sum_{n=1}^{n=n_{tot}} \frac{dI^g}{dI^n} \left[-\frac{\partial I^n}{\partial \mathbf{Q}^n} \left[\frac{\partial \mathbf{R}^n}{\partial \mathbf{Q}^n} \right]^{-1} \frac{\partial \mathbf{R}^n}{\partial \beta} + \frac{\partial I^n}{\partial \beta} \right].
\end{aligned} \tag{31}$$

Forward mode direct differentiation computes sensitivity derivatives by computing $\frac{\partial I^n}{\partial \mathbf{Q}^n}$ and $\frac{\partial I^n}{\partial \beta}$ at each time step. In addition, the quantity $\frac{\partial \mathbf{Q}}{\partial \beta}$ is computed at each time step using Eq. 30. Once the solution has converged to a periodic state, the sensitivity derivatives of a vector of cost functions \mathbf{I}^g with respect to one design variable can be computed over the course of one period. It is most useful in situations where there are many cost functions and only a few design variables. The algorithm is as follows:

- (1) Compute the sensitivity of initial conditions to the design variable $\frac{\partial \mathbf{Q}^0}{\partial \beta}$.
- (2) Advance the solution by one time step.
- (3) Compute $\frac{\partial \mathbf{R}^n}{\partial \beta}$ at the current time step.
- (4) Compute $\frac{\partial \mathbf{Q}^n}{\partial \beta}$ for the current time step using Eq. 30.
- (5) Repeat steps 2-4 until the solution has converged to a periodic state. At convergence, begin the cycle to compute the cost function and sensitivity derivatives.
- (6) Advance the solution by one time step.
- (7) Compute $\frac{\partial I^n}{\partial \beta}$ for the current time step.
- (8) Compute $\frac{\partial I^n}{\partial \mathbf{Q}^n}$ for the current time step.
- (9) Compute $\frac{\partial \mathbf{Q}^n}{\partial \beta}$ for the current time step using Eq. 30.
- (10) Compute $\frac{dI^g}{dI^n}$ for the current time step.
- (11) Using Eq. 28, add the contribution $\frac{dI^g}{dI^n} \left[\frac{\partial I^n}{\partial \mathbf{Q}^n} \frac{\partial \mathbf{Q}^n}{\partial \beta} + \frac{\partial I^n}{\partial \beta} \right]$ to the global sensitivity derivative $\frac{dI^g}{d\beta}$.
- (12) Repeat steps 6 - 12 until the period has been completed.

Direct differentiation does not require much storage for saving the complete time history of $\frac{\partial \mathbf{Q}}{\partial \beta}$ because the quantities $\frac{\partial \mathbf{Q}^n}{\partial \beta}$, $\frac{\partial \mathbf{Q}^{n-1}}{\partial \beta}$, and $\frac{\partial \mathbf{Q}^{n-2}}{\partial \beta}$ can be overwritten as the algorithm proceeds. However, if there are multiple design variables, direct differentiation must be repeated for each design variable.

6.2.2 Reverse Mode Adjoint

For time-dependent problems with cost functions that are computed over a given period, reverse mode adjoint involves advancing the solution to a terminal time step, and from there stepping backwards in time to compute the sensitivity derivatives. It allows for the efficient computation of sensitivity derivatives where there are few cost functions I^g and numerous design variables expressed in the vector $\boldsymbol{\beta}$. Thus, the reverse mode adjoint method is useful in applications where many design variables are present.

Taking the transpose of Eq. 28 and Eq. 30, one obtains

$$\frac{dI^g T}{d\boldsymbol{\beta}} = \sum_{n=0}^{n=n_f} \left[\frac{\partial \mathbf{Q}^{n T}}{\partial \boldsymbol{\beta}} \frac{\partial I^{n T}}{\partial \mathbf{Q}^n} + \frac{\partial I^{n T}}{\partial \boldsymbol{\beta}} \right] \frac{dI^g T}{dI^n} \quad (32)$$

$$\frac{\partial \mathbf{Q}^{n T}}{\partial \boldsymbol{\beta}} = - \left(\frac{\partial \mathbf{Q}^{n-1 T}}{\partial \boldsymbol{\beta}} \frac{\partial \mathbf{R}^{n T}}{\partial \mathbf{Q}^{n-1}} + \frac{\partial \mathbf{Q}^{n-2 T}}{\partial \boldsymbol{\beta}} \frac{\partial \mathbf{R}^{n T}}{\partial \mathbf{Q}^{n-2}} + \frac{\partial \mathbf{R}^{n T}}{\partial \boldsymbol{\beta}} \right) \left[\frac{\partial \mathbf{R}^n}{\partial \mathbf{Q}^n} \right]^{-T} \quad (33)$$

Since reverse mode differentiation starts at the final time step and goes backwards, it is convenient to rewrite Eq. 32 as

$$\frac{dI^g T}{d\boldsymbol{\beta}} = \sum_{n=n_{tot}}^{n=1} \left[\frac{\partial \mathbf{Q}^{n T}}{\partial \boldsymbol{\beta}} \frac{\partial I^{n T}}{\partial \mathbf{Q}^n} + \frac{\partial I^{n T}}{\partial \boldsymbol{\beta}} \right] \frac{dI^g T}{dI^n} \quad (34)$$

Substituting Eq. 33 into Eq. 34 results in

$$\begin{aligned} \frac{dI^g T}{d\boldsymbol{\beta}} = & \sum_{n=n_{tot}}^{n=1} \left[-\frac{\partial \mathbf{Q}^{n-1 T}}{\partial \boldsymbol{\beta}} \frac{\partial \mathbf{R}^{n T}}{\partial \mathbf{Q}^{n-1}} \right] \left[\frac{\partial \mathbf{R}^n}{\partial \mathbf{Q}^n} \right]^{-T} \frac{\partial I^{n T}}{\partial \mathbf{Q}^n} \frac{dI^g T}{dI^n} + \\ & + \sum_{n=n_{tot}}^{n=1} \left[-\frac{\partial \mathbf{Q}^{n-2 T}}{\partial \boldsymbol{\beta}} \frac{\partial \mathbf{R}^{n T}}{\partial \mathbf{Q}^{n-2}} \right] \left[\frac{\partial \mathbf{R}^n}{\partial \mathbf{Q}^n} \right]^{-T} \frac{\partial I^{n T}}{\partial \mathbf{Q}^n} \frac{dI^g T}{dI^n} + \\ & + \sum_{n=n_{tot}}^{n=1} \left[-\frac{\partial \mathbf{R}^{n T}}{\partial \boldsymbol{\beta}} \left[\frac{\partial \mathbf{R}^n}{\partial \mathbf{Q}^n} \right]^{-T} \frac{\partial I^{n T}}{\partial \mathbf{Q}^n} + \frac{\partial I^{n T}}{\partial \boldsymbol{\beta}} \right] \frac{dI^g T}{dI^n} . \end{aligned} \quad (35)$$

Due to the considerable expense of computing each term in Eq. 35 at each time step, the terms are computed recursively. The algorithm is as follows:

- (1) Converge the solution to a periodic state, storing the solution vector \mathbf{Q} at each time step.

- (2) Starting with the final time step, compute the first flow adjoint vector Λ^n as

$$\Lambda^n = - \left[\frac{\partial \mathbf{R}^n}{\partial \mathbf{Q}^n} \right]^{-T} \frac{\partial I^n}{\partial \mathbf{Q}^n} \frac{dI^g}{dI^n} \quad (36)$$

- (3) Compute the vector $\frac{\partial I^n}{\partial \beta}$ for the current time step.
(4) Add the contribution $\left(\frac{\partial \mathbf{R}^n}{\partial \beta} \Lambda^n + \frac{\partial I^n}{\partial \beta} \frac{\partial I^g}{\partial I^n} \right)$ from the current time step to the global sensitivity derivative $\frac{dI^g}{d\beta}$.
(5) Compute the second flow adjoint vector λ^n as

$$\lambda^n = \frac{\partial \mathbf{R}^n}{\partial \mathbf{Q}^{n-1}} \Lambda^n + \frac{\partial I^{n-1}}{\partial \mathbf{Q}^{n-1}} \frac{dI^g}{dI^{n-1}} \quad (37)$$

- (6) Compute the first flow adjoint vector for the previous time step as

$$\Lambda^{n-1} = - \left[\frac{\partial \mathbf{R}^{n-1}}{\partial \mathbf{Q}^{n-1}} \right]^{-T} \lambda^n \quad (38)$$

- (7) Compute $\frac{\partial I^{n-1}}{\partial \beta}$.
(8) Add the contribution $\left(\frac{\partial \mathbf{R}^{n-1}}{\partial \beta} \Lambda^{n-1} + \frac{\partial I^{n-1}}{\partial \beta} \frac{\partial I^g}{\partial I^{n-1}} \right)$ from the current time step to the global sensitivity derivative.
(9) Compute the second flow adjoint vector for the previous time step as

$$\lambda^{n-1} = \frac{\partial \mathbf{R}^{n-1}}{\partial \mathbf{Q}^{n-2}} \Lambda^{n-1} + \frac{\partial \mathbf{R}^n}{\partial \mathbf{Q}^{n-2}} \Lambda^n + \frac{\partial I^{n-2}}{\partial \mathbf{Q}^{n-2}} \frac{dI^g}{dI^{n-2}} \quad (39)$$

- (10) Continue stepping backwards, computing Λ and λ at each time step as well as adding the contribution to the global sensitivity derivative. Note that once the time step number is less than the time step number at which the computation of the cost function began, the sensitivity of the global cost function with respect to the local cost function $\left(\frac{dI^g}{dI^n} \right)$ is equal to zero.
(11) Once λ^1 has been computed, add $\frac{\partial \mathbf{Q}^0}{\partial \beta} \lambda^1$ to the global sensitivity derivative $\frac{dI^g}{d\beta}$. Note that $\frac{\partial \mathbf{Q}^0}{\partial \beta}$ is computed from the known initial conditions.

Thus, the expression for the total sensitivity derivative is computed as

$$\frac{dI^g}{d\beta} = \frac{\partial \mathbf{Q}^0}{\partial \beta} \lambda^1 + \sum_{n=1}^{n=n_{tot}} \left[\frac{\partial \mathbf{R}^n}{\partial \beta} \Lambda^n + \frac{\partial I^n}{\partial \beta} \frac{\partial I^g}{\partial I^n} \right] \quad (40)$$

The main drawback to reverse mode adjoint is that it requires more storage than forward mode direct differentiation since the solution vector must be stored or recomputed at every single time step in the solution process. This can be accomplished either by storing the solutions in memory or by writing out the solutions to a file.

7 Computational Results and Discussion

The simulation is run on a 161 point grid with 400 time steps per RF cycle. The design variables considered are the reaction rate coefficients for each of the 20 reactions (k_1, k_2, \dots, k_{20}), the nitrogen impurity fraction f_N , the gas pressure p , the discharge gap length l , the rms current density j_{rms} , the frequency f , and the gas temperature T_{gas} . The cost functions considered are the average peak electron density and the rms voltage during the final RF cycle. The simulation is defined to be converged when the average change in the solution values at the middle of the discharge is less than 1×10^{-6} compared to the values in the previous cycle. Using this criteria, it can take more than 60000 RF cycles to reach full convergence. However, it is observed that the cost functions considered obtain a value within five percent of the converged value after 3500 RF cycles. Thus, in the interests of computational efficiency, the simulation is not run to a fully converged state. Simulation results were compared for cases that were run with 41, 81, 161 and 321 grid points, and the results showed good agreement.

7.1 Verification

To verify the accuracy of the sensitivity analysis, sensitivity derivatives are computed using adjoint, direct differentiation, and complex perturbations. In this case, the cost function is defined as the average peak electron density over one RF cycle. The simulation is run with the reaction rate values listed in Table 2, a nitrogen impurity mole fraction of 5×10^{-7} , a gas pressure of 600 Torr, a discharge gap length of 2.4 mm, an rms current density of 21.2 mA/cm², a frequency of 13.56 MHz, and a gas temperature of 393 K. The sensitivity derivatives are computed after 3500 RF cycles. Good agreement is shown between different methods as shown below in Table 5. The values computed using adjoint and direct differentiation are identical, while the values computed using complex perturbations are accurate to five decimal places.

7.2 Using Sensitivity Derivatives to Increase a Given Cost Function

Sensitivity derivatives can be used to increase or decrease a cost function. There are many steepest descent or trust region optimization methods that can be used to find a global maximum or minimum of a cost function. In this case, open source optimization routines [44] which are based on trust region methods are used to vary the physical design variables so as to maximize the

β	Complex Perturbation	Direct Differentiation	Adjoint
k_1	-7.957085×10^{30}	-7.957095×10^{30}	-7.957095×10^{30}
k_{10}	7.54887×10^{28}	7.54890×10^{28}	7.54890×10^{28}
k_{18}	-3.355051×10^{43}	-3.355063×10^{43}	-3.355063×10^{43}
p	4.4412508×10^{13}	4.4412507×10^{13}	4.4412507×10^{13}
j_{rms}	9.4717367×10^{15}	9.4717361×10^{15}	9.4717361×10^{15}

Table 5
Sensitivity derivatives computed using various methods for the average peak electron density

peak electron density. Bounds on the physical design variables are put in place so as to prevent arcing, and are based on a previous experimental parametric study [45]. The initial values and bounds are listed below in Table 6.

Design Variable	Initial Value	Lower Bound	Upper Bound
l (m)	2.4×10^{-3}	1.6×10^{-3}	3.2×10^{-3}
j_{rms} (mA/cm ²)	21.2	10.0	30.0
p (Torr)	600	550	650
f (MHz)	13.56	12.0	25.0
T_{gas} (K)	393	250	450

Table 6
Bounds for physical design variables

The simulation is run for 10000 RF cycles. The initial peak electron density is found to be 2.07×10^{17} /m³, and after two design cycles, this had increased to its maximum value of 3.28×10^{17} /m³. This maximum value for the peak electron density is found when the discharge gap length, rms current density, and pressure are at the upper bound and when the frequency and gas temperature are at the lower bound.

7.3 *Using Sensitivity Derivatives to Obtain a Better Estimate for Uncertain Parameters*

It is well known that there is a good degree of uncertainty in the computation of reaction rate coefficients and other parameters used to simulate reaction chemistry [12] [46]. Sensitivity derivatives can be used to obtain better estimates of these parameters and coefficients, so that results from computational models more closely match experimental data. This is accomplished by using the reaction rate coefficients and parameters as design variables, and by defining the cost function to be a function that computes the magnitude of the difference between experimental and computational results. The known uncertainty in a given design variable is used to determine the maximum and minimum value of that design variable during the design process. Once the sensitivity derivatives are computed, the design variables are then changed to minimize the cost function.

In this particular case, it is demonstrated how quantitative sensitivity analysis might be used in conjunction with experiments to obtain better estimates for reaction rates. Three experimental data points on a V-I plot are taken from a previous work [47], and the reaction rate coefficients and nitrogen impurity

mole fraction are used as design variables and are changed in order to make the computational data better match the experimental data. The cost function is defined to be

$$I = \sum_{i=1}^3 (V_{i,e} - V_{i,c})^2, \quad (41)$$

where $V_{i,e}$ is the experimental rms voltage for a given rms current density, and $V_{i,c}$ is the computational rms voltage for a given rms current density. Reducing this cost function will result in matching the computational data with the experimental data.

Because cross-section data often has a fifty percent uncertainty associated with it [46], the reaction rates are bounded between 50 percent and 150 percent of the values in Table 2. The nitrogen mole impurity fraction is bounded between 5×10^{-11} and 5×10^{-6} , which is consistent with the experimental data [47]. Five design iterations using the adjoint method are run for 3500 RF cycles. The results are shown below in figure 2.

The cost function is reduced from 3410 to 1494 over five design cycles. Although the cost function is reduced and the computational data points better match the experimental data points, the differential impedance (slope of the V-I curve) remains nearly constant for each design cycle and limits the amount by which the cost function can be reduced. It is thought that if mobilities and diffusion coefficients are included as design variables, perhaps the differential impedance would be changed to better match the impedance on the experimental V-I curve.

However, even if the parameters were changed in such a way that the computational data matched the experimental data very closely, there is no guarantee that the results obtained by one set of parameters cannot be obtained by an entirely different set of parameters. In this case, there are 20 reaction rate coefficients, 17 additional reaction rate parameters, 10 diffusion coefficients and 5 mobilities to consider. There could be multiple local minima present in the design surface. This analysis should be used as a tool that can be used in conjunction with experimental data to help get better estimates for uncertain parameters.

Design Cycles Matching Experimental Data

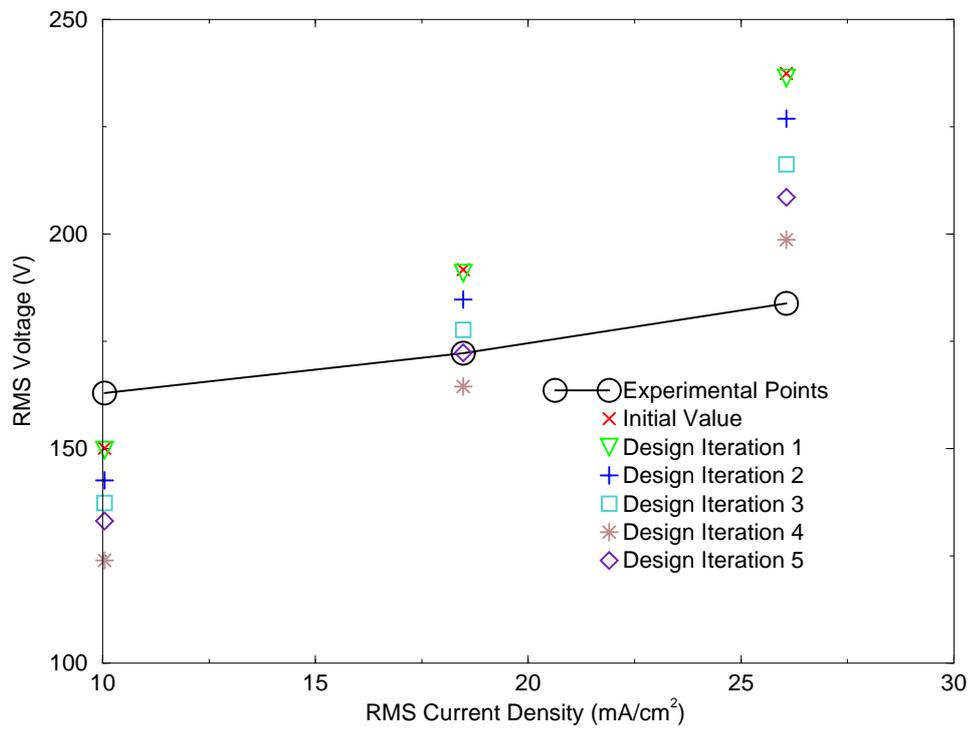


Fig. 2. Design cycles matching computational data with experimental data on a V-I plot

7.4 *Using Sensitivity Derivatives to Compute a Bound on Computational Error*

Sensitivity derivatives can be used to assess the effects of uncertainties in input data on the resulting output from a computational model. Reaction rate coefficients and parameters, mobilities, diffusion coefficients and physical parameters (e.g. gas temperature, pressure) all have an associated uncertainty with them. Sensitivity analysis can be used to compute a maximum bound on computational uncertainty for a given cost function. This has been done previously in various fields [48] [49] [50].

For a given cost function I , with n_d design variables, the maximum bound on computational uncertainty can be computed as

$$\max |\hat{I} - I| = \max |\Delta I^g| = \sum_{i=1}^{n_d} \left| \frac{dI^g}{d\beta_n} \Delta\beta_i \right|, \quad (42)$$

where $\Delta\beta_i$ is the known physical uncertainty in design variable i . To get an accurate measure of the upper bound on the uncertainty in this simulation one must obtain sensitivity derivatives and uncertainties for 57 parameters. Many of the uncertainties in these parameters were not reported in the literature, and thus no bound on the uncertainty was computed. However, with the knowledge of all uncertainties, this could be a useful tool for determining the maximum uncertainty of a simulation. It should be noted that this analysis shows how much results from the computational model are bounded, and is not a measure of the quality of the model itself.

8 Summary

Sensitivity derivatives are computed for a one-dimensional simulation of a high pressure helium glow discharge. The computed derivatives show good agreement between complex perturbations, direct differentiation, and adjoint. Several applications for quantitative sensitivity analysis are presented. Sensitivity derivatives can be used to increase or decrease a given cost function, to obtain better estimates of uncertain parameters, and to compute a maximum bound on computational uncertainty.

In the future, it is hoped that quantitative sensitivity analysis will be extended to multi-dimensional plasma discharge simulations. However, there will need to be significant improvements in efficiency to the computational model. Unfortunately, increasing the number of processors did not improve efficiency. This is because the approximate ILU preconditioner in the GMRES routine becomes an increasingly worse approximation of the Jacobian as more processors are used, requiring more GMRES iterations and increasing the time that it takes to compute a solution.

The convergence of the computational model will need to be accelerated. There are several approaches that could be used for this. Some authors have used different time steps for the electrons and the ions, due to their disparate characteristic time scales [3] [7]. Other authors have directly solved for the periodic solution of the discharge [51]. It is possible that sensitivity analysis could be used to solve for the periodic solution simply by reducing the norm of the solution vector from one cycle to the next.

Related to the issue of efficiency is the issue of disk storage. Using adjoint for the current problem, it requires the storage of 93 gigabytes of memory. Extending this problem to multiple dimensions will increase these memory requirements even further.

References

- [1] D. Graves, K. Jensen, A continuum model of dc and rf discharges, *IEEE Transactions on Plasma Science* PS 14 (1986) 78–91.
- [2] E. Gogolides, H. Sawin, Continuum modeling of radio-frequency glow discharges. i. theory and results for electropositive and electronegative gases, *Journal of Applied Physics* 72 (1992) 3971–3987.
- [3] T. Nitschke, D. Graves, A comparison of particle in cell and fluid model simulations of low-pressure radio frequency discharges, *Journal of Applied Physics* 76 (10) (1994) 5646–5660.
- [4] T. R. Govindan, M. Meyyappan, One-dimensional modeling studies of the gaseous electronics conference rf reference cell, *J. Res. Natl. Inst. Stand. Technol.* 100 (1995) 463–472.
- [5] D. Lymberopoulos, D. Economou, Two-dimensional self-consistent radio frequency plasma simulations relevant to the gaseous electronics conference rf reference cell, *Journal of Research of the National Institute of Standards and Technology* 100 (4) (1995) 473–494.
- [6] S. Rauf, M. Kushner, Dynamics of a coplanar-electrode plasma display panel cell. i. basic operation, *Journal of Applied Physics* 85 (1999) 3460–3469.
- [7] E. Hammond, K. Mahesh, P. Moin, A numerical method to simulate radio-frequency plasma discharges, *Journal of Computational Physics* 176 (2002) 402–429.
- [8] Q. Wang, D. Economou, V. Donnelly, Simulation of a direct current microplasma discharge in helium at atmospheric pressure, *Journal of Applied Physics* 100 (2006) 023301.
- [9] E. Gogolides, H. Sawin, Continuum modeling of radio-frequency glow discharges. ii. parametric studies and sensitivity analysis, *Journal of Applied Physics* 72 (1992) 3988–4002.
- [10] S. Rauf, M. Kushner, Dynamics of a coplanar-electrode plasma display panel cell. ii. cell optimization, *Journal of Applied Physics* 85 (1999) 3470–3476.
- [11] X. Yuan, L. Raja, Computational study of capacitively coupled high-pressure glow discharges in helium, *IEEE Transactions on Plasma Science* 31 (2003) 495–503.
- [12] D. Bose, M. Rao, T. Govindan, M. Meyyappan, Uncertainty and sensitivity analysis of gas-phase chemistry in a CHF_3 plasma, *Plasma Sources Science and Technology* 12 (2003) 225–234.
- [13] S. Nadarajah, A. Jameson, Optimal control of unsteady flows using a time accurate method, in: *Proceedings of the 9th AIAA/ISSMO Symposium on Multidisciplinary Analysis and Optimization Conference*, AIAA paper 2002-5436, Reno, NV, 2002.

- [14] K. Ghayour, O. Baysal, Limit-cycle shape optimization using time-dependent transonic equation, in: Proceedings of the 14th Computational Fluid Dynamics Conference, AIAA Paper 1999-3375, Norfolk, VA, 1999.
- [15] K. Mani, D. Mavriplis, An unsteady discrete adjoint formulation for two-dimensional flow problems with deforming meshes, in: Proceedings of the 45th AIAA Aerospace Sciences Meeting and Exhibit, 2007.
- [16] K. Lange, W. K. Anderson, Sensitivity derivatives for plasma discharge simulations, in: Proceedings of the 12th AIAA/ISSMO Multidisciplinary Analysis and Optimization Conference, AIAA Paper 2008-5930, Victoria, BC, Canada, 2008.
- [17] S. Kanazawa, M. Kogoma, T. Moriwaki, S. Okazaki, Stable glow plasma at atmospheric pressure, *Journal of Physics D, Applied Physics* 21 (1988) 838–840.
- [18] J. Jeong, S. Babayan, A. Schutze, V. Tu, J. Park, I. Henins, R. Hicks, G. Selwyn, Etching materials with an atmospheric-pressure plasma jet, *Plasma Sources of Science and Technology* 7 (1993) 282–285.
- [19] H. Herrmann, I. Henins, J. Park, G. Selwyn, Decontamination of chemical and biological warfare (cbw) agents using an atmospheric pressure plasma jet (appj), *Physics of Plasmas* 6 (5) (1999) 2284–2289.
- [20] R. Sladek, E. Stoffels, R. Walraven, P. Tielbeek, R. Koolhoven, Plasma treatment of dental cavities: A feasibility study, *IEEE Transactions on Plasma Science* 32 (4) (2004) 1540–1543.
- [21] K. Kelly-Wintenberg, D. Sherman, P. Tsai, R. Ben Gadri, F. Karakaya, Z. Chen, J. Roth, T. Montie, Air filter sterilization using a one atmosphere uniform glow discharge plasma (the volfilter), *IEEE Transactions on Plasma Science* 28 (1) (2000) 64–71.
- [22] J. Roth, D. Sherman, R. Ben Gadri, F. Karakaya, Z. Chen, T. Montie, K. Kelly-Wintenberg, P. Tsai, A remote exposure reactor (rer) for plasma processing and sterilization by plasma active species at one atmosphere, *IEEE Transactions on Plasma Science* 28 (1) (2000) 56–63.
- [23] Y. P. Raizer, *Gas Discharge Physics*, Springer-Verlag, New York, NY, 1991.
- [24] H. Ellis, R. Pai, E. McDaniel, E. Mason, L. Viehland, Transport properties of gaseous ions over a wide energy range, *Atomic Data and Nuclear Data Tables* 17 (1976) 177–210.
- [25] R. Kee, G. Dixon-Lewis, J. Warnatz, M. Coltrin, J. Miller, A fortran computer code package of gas-phase multicomponent transport properties, Tech. Rep. SAND86-8246, Sandia National Lab, Livermore, CA (1995).
- [26] J. Shon, M. Kushner, Excitation mechanism and gain modeling of the high pressure atomic ar laser in he/ar mixtures, *Journal of Applied Physics* 75 (1994) 1883–1890.

- [27] T. Sommerer, M. Kushner, Numerical investigation of the kinetics and chemistry of rf glow discharge plasmas sustained in he, n₂, o₂, he/n₂/o₂, he/cf₄/o₂ and sih₄/nh₃ using a monte carlo-fluid hybrid model, *Journal of Applied Physics* 71 (1992) 1654–1673.
- [28] C. Laux, L. Yu, D. Packan, R. Gessman, L. Pierrot, C. Kruger, R. Zare, Ionization mechanisms in two-temperature air plasmas, in: *Proceedings of the 30th AIAA Plasmadynamics and Lasers Conference*, Norfolk, VA, 1999, aIAA Paper 99-3476.
- [29] K. Lange, A fully implicit characteristic-based algorithm for a one dimensional radio frequency glow discharge fluid simulation, Master’s thesis, The University of Tennessee at Chattanooga (July 2007).
- [30] Y. Saad, M. Schultz, Gmres: A generalized minimal residual algorithm for solving nonsymmetric linear systems, *SIAM J. Sci. Statist. Comput.* 7 (1986) 856–869.
- [31] O. Axelsson, *Iterative Solution Methods*, Cambridge University Press, 1994.
- [32] J. Newman, W. Anderson, D. Whitfield, Multidisciplinary sensitivity derivatives using complex variables, Tech. Rep. MSSU-EIRS-ERC-98-08, Mississippi State University, Mississippi State, MS (1998).
- [33] W. Anderson, J. Newman, D. Whitfield, E. Nielsen, Sensitivity analysis for the navier-stokes equations on unstructured meshes using complex variables, in: *Proceedings of the 14th Computational Fluid Dynamics Conference*, AIAA Paper 1999-3294, Norfolk, VA, 1999.
- [34] J. Newman, W. Anderson, D. Whitfield, E. Nielsen, Step-size independent approach for multidisciplinary sensitivity analysis, *Journal of Aircraft* 40 (3) (2003) 566–573.
- [35] G. J.-W. Hou, A. I. Taylor, V. Korivi, Discrete shape sensitivity equations for aerodynamic problems, in: *Proceedings of the 27th AIAA/SAE/ASME/ASEE Joint Propulsion Conference*, Sacramento, CA, 1991, aIAA-91-2259.
- [36] O. Pironneau, On optimum design in fluid mechanics, *Journal of Fluid Mechanics* 64 (1974) 97–110.
- [37] A. Jameson, Aerodynamic design via control theory, *Journal of Scientific Computing* 3 (1988) 233–260.
- [38] A. I. Taylor, G. Hou, V. Korivi, Methodology for calculating aerodynamic sensitivity derivatives, *AIAA Journal* 30 (10) (1992) 2411–2419.
- [39] S. Ta’asan, G. Kuruvila, M. Salas, Aerodynamic design and optimization in one shot, in: *Proceedings of the 30th Aerospace Sciences Meeting*, Reno, NV, 1992, aIAA-92-0025.
- [40] W. Anderson, V. Venkatakrishnan, Aerodynamic design optimization on unstructured grids with a continuous adjoint formulation, *Computers and Fluids* 28 (4-5) (1999) 443–480.

- [41] J. Elliott, J. Peraire, Practical three-dimensional aerodynamic design and optimization, *AIAA Journal* 35 (9) (1997) 1479–1485.
- [42] E. Nielsen, W. Anderson, Aerodynamic design optimization on unstructured meshes using the navier-stokes equations, *AIAA Journal* 37 (11) (1999) 957–964.
- [43] S. Nadarajah, A. Jameson, A comparison of the continuous and discrete adjoint approach to automatic aerodynamic optimization, in: *Proceedings of the AIAA 38th Aerospace Sciences Meeting and Exhibit, Reno, NV, 2000*, aIAA-2000-0667.
- [44] J. Dennis, D. Gay, R. Welsch, Algorithm 573: An adaptive nonlinear least-squares algorithm, *ACM Transactions on Mathematical Software* 7 (3) (1981) 367–383.
- [45] J. Park, I. Henins, H. Herrman, G. Selwyn, Discharge phenomena an atmospheric pressure radio-frequency capacitive plasma source, *Journal of Applied Physics* 89 (1) (2001) 20–28.
- [46] W. Janev, K. J. Langer, D. J. Post, *Elementary Processes in Hydrogen-Helium Plasmas*, Springer, Berlin, 1987.
- [47] J. Park, I. Henins, G. Selwyn, Gas breakdown in an atmospheric pressure radio-frequency capacitive plasma source, *Journal of Applied Physics* 89 (1) (2001) 15–19.
- [48] B. Blackwell, K. Dowding, R. Cochran, D. Dobranich, Utilization of sensitivity coefficients to guide the design of a thermal battery, in: *Proceedings of the 1998 ASME International Mechanical Engineering Congress and Exposition, Vol. 361, Anaheim, CA, 1998*, pp. 73–82.
- [49] M. Putko, P. Newman, A. Taylor III, L. Green, Approach for uncertainty propagation and robust design in cfd using sensitivity derivatives, in: *Proceedings of the 15th AIAA Computational Fluid Dynamics Conference, AIAA Paper 2001-2528, Anaheim, CA, 2001*.
- [50] E. Turgeon, D. Pelletier, S. Etienne, J. Borggaard, Sensitivity and uncertainty analysis for turbulent flows, in: *Proceedings of the 40th AIAA Aerospace Sciences Meeting and Exhibit, AIAA Paper 2002-0985, Reno, NV, 2002*.
- [51] E. Gogolides, H. Sawin, R. Brown, Direct calculation of time-periodic states of continuum models of radio-frequency plasmas, *Chemical Engineering Science* 47 (1992) 3839–3855.