Zapdos

A Low Temperature Plasma Simulation Tool in MOOSE

Corey DeChant, Steven Shannon

North Carolina State University - Nuclear Engineering

The 71st Annual Gaseous Electronics Conference

Zapdos

Contents of Zapdos

- Zapdos is a open source application created by Alexander Lindsay.
- Zapdos focus on low temperature plasma simulations utilizing the MOOSE framework.
- Zapdos uses the the continuity equations based on the drift-diffusion approximation.
- Zapdos can be either alone from simple modeling of electrons and single ions or can be coupled with CRANE for more diverse chemical reactions.

Solving Using The Drift-Diffusion Approximation

- By taking the moments in velocity space of the Boltzmann equation, the following equations can be obtained:

$$\frac{\partial n_i}{\partial t} + \nabla \cdot \vec{\Gamma_i} = S_{iz}$$
$$\frac{\partial n_e}{\partial t} + \nabla \cdot \vec{\Gamma_e} = S_{iz}$$
$$\Gamma_i = \mu_i \vec{E} - D_i \nabla n_i$$
$$\Gamma_e = \mu_e \vec{E} - D_e \nabla n_e$$

  where μ is the mobility, D the diffusivity, S_{iz} is the source term, Γ the species flux, n the species density, and $\vec{E}$ the electric field.
- Each separate term is a kernel in Zapdos
  - Time Derivative
  - Advection
  - Diffusion
  - Source Term

The Drift-Diffusion Approximation: Source Term

- The source term in the continuity equations can be solved in one of two ways:
  - It can be solved within Zapdos with the use of:

$$S_{iz} = -\alpha_{iz}\overrightarrow{\Gamma_e}$$

    where α_{iz} is the Townsend ionization coefficient.
  - The other way is to have CRANE coupled to Zapdos and have it solve the ionization source.

The Drift-Diffusion Approximation: Poisson Equation

- The electric field is solved by equaling it to $\nabla\ V$ and using the Poisson equation to solving for the potential:

$$-\nabla^2 V = \frac{e(n_i - n_e)}{\epsilon_0}$$

  where e is the Coulombic charge and ϵ_0 is the permittivity of free space.
- There should be at less three separate kernels. One for the potential and one for each charge particle.

The Drift-Diffusion Approximation: Electron Energy

- In order to obtain the correct mobility, diffusion and Townsend coefficient, the electron energy also needs to be solve. The equation for the electron energy is:

$$\frac{\partial n_e \epsilon}{\partial t} + \nabla \cdot \vec{\Gamma_\epsilon} = -e\vec{\Gamma} \cdot \vec{E} - \vec{\Gamma_\epsilon}(\alpha_{iz}\epsilon_{iz} + \alpha_{ex}\epsilon_{ex} + 3\frac{m_e}{m_i}\alpha_{el}T_e)$$

$$\vec{\Gamma_\epsilon} = \frac{5}{3}\epsilon\vec{\Gamma_e} - \frac{5}{3}n_e D_e \nabla\epsilon$$

where ϵ is the mean electron energy, ϵ_{iz} the electron energy lost in ionization collision, ϵ_{ez} the electron energy lost in excitation collision, m_i and m_e the ion and electron masses respectively, α_{el} the Townsend elastic collision coefficient, and T_e the electron temperature.

Boundary Condition

- The boundary conditions for the ion and electron flux are based on the work of GJM Hagelaar *et al.* [1] and the boundary condition for the electron energy flux is based on the work of Y Sakiyama and D Graves [2].
- All three boundary conditions require a boundary reflection coefficient.
- Along with reflection coefficient, a secondary electron emission coefficient can be introduced.

Boundary Condition (cont.)

- There are two ways of setting the boundary condition for the potential been if the discharge is rf or dc.
- For a dc discharge, Kirchoff's voltage law is used, which is:

$$V_{source} + V_{cathode} = (e\vec{\Gamma}_i - e\vec{\Gamma}_e)AR$$

where A is the cross-sectional area of the plasma and R is the ballast resistance.

- For a rf discharge, a sinusoidal function of the applied voltage is used:

$$V_{cathode} = V_0 sin(ft)$$

where f is the applied frequency and t is the time.

Interface Kernels

- Zapdos has the capability of treating the transport of electrons through an interface.
- This is achieved by setting the flux in one material equal to the flux in the adjacent material at the interface.

Log From of the Densities

- In order to over come possible negative values that are associated with ion, electron and electron energy densities when solving, these values are converted into the log form and converted back after the simulation is finished.
- An example of this is:

$$N_i = \ln n_i$$
$$N_e = \ln n_e$$
$$E_n = \ln(n_e \epsilon)$$

- The units of each variable do not charge, e.g. if n_e has units of $\frac{\#}{m^3}$ the e^{N_e} also has units of $\frac{\#}{m^3}$.

Scaling

- To solving performance, it is desirable to have all variables and coefficients as close to unity as possible.
- To help accomplish this, scaling is added as an option.
  - The position can be scaled to any value for micro discharges
  - The voltage can be set to units of V or kV
  - The number density can be set to moles

Installing Zapdos

- Ensure MOOSE is installed correctly and the MOOSE environment is begin used.
- Clone from source repository

```
cd ~/projects
git clone https://github.com/shannon-lab/zapdos
```

- Compile Zapdos

```
cd ~/projects/zapdos
make -jn  #where n is the number of processors
```

- Run tests

```
./run_tests
```

Coupling CRANE

- To couple CRANE, modify the Zapdos Makefile by adding:

```
# Use the CRANE submodule if it exists and CRANE_DIR is not set
CRANE_SUBMODULE := $(CURDIR)/crane
ifneq ($(wildcard $(CRANE_SUBMODULE)/Makefile),)
  CRANE_DIR         ?= $(CRANE_SUBMODULE)
else
  CRANE_DIR         ?= $(shell dirname `pwd`)/crane
endif

# crane
APPLICATION_DIR     := $(CRANE_DIR)
APPLICATION_NAME    := crane
include             $(FRAMEWORK_DIR)/app.mk
```

Coupling CRANE (cont.)

- The file "zapdos/src/base/Zapdos.C" must also by modify by adding:

```
#include "CraneApp.h"

ZapdosApp::ZapdosApp(InputParameters parameters) : MooseApp(parameters)
{
  Moose::registerObjects(_factory);
  ModulesApp::registerObjects(_factory);
  SquirrelApp::registerObjects(_factory);
  CraneApp::registerObjects(_factory);
  ZapdosApp::registerObjects(_factory);

  Moose::associateSyntax(_syntax, _action_factory);
  ModulesApp::associateSyntax(_syntax, _action_factory);
  SquirrelApp::associateSyntax(_syntax, _action_factory);
  CraneApp::associateSyntax(_syntax, _action_factory);
  ZapdosApp::associateSyntax(_syntax, _action_factory);
}
```

Coupling CRANE (cont.)

- There are additional kernels, materials and BC require to use the CRANE Actions feature:
  - kernels
    - ElectronEnergyTermElasticRate
    - ElectronEnergyTermElasticTownsend
    - ElectronEnergyTermRate
    - ElectronEnergyTermTownsend
    - ElectronImpactReactionProduct
    - ElectronImpactReactionReactant
  - materials
    - HeavySpeciesMaterial

  Conditions
    - EnergyBC2
- These files can to found at

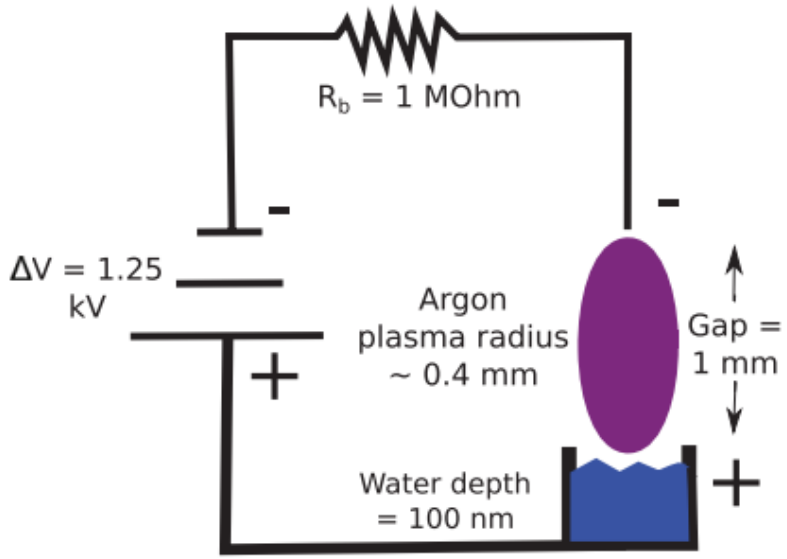
  `https://github.com/keniley1/zapdos/tree/rxn_network`

Works Using Zapdos

- Currently Zapdos has mainly been used to model atmospheric mirco plasmas.
- Previous applications of Zapdos are:
  - Liquid-Plasma interactions
  - Microscale Gas Discharges to Enhance Thermionic Energy Conversion
  - Modeling the COST-Jet (an atmospheric micro-plasma jet reference source)

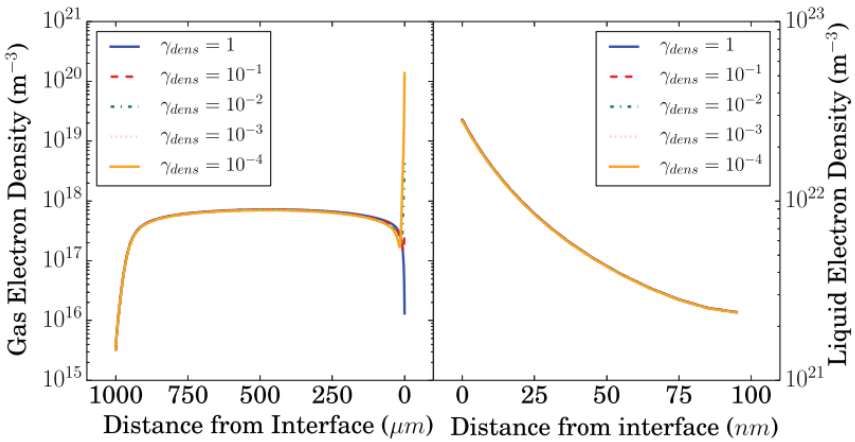
Liquid-Plasma Interactions

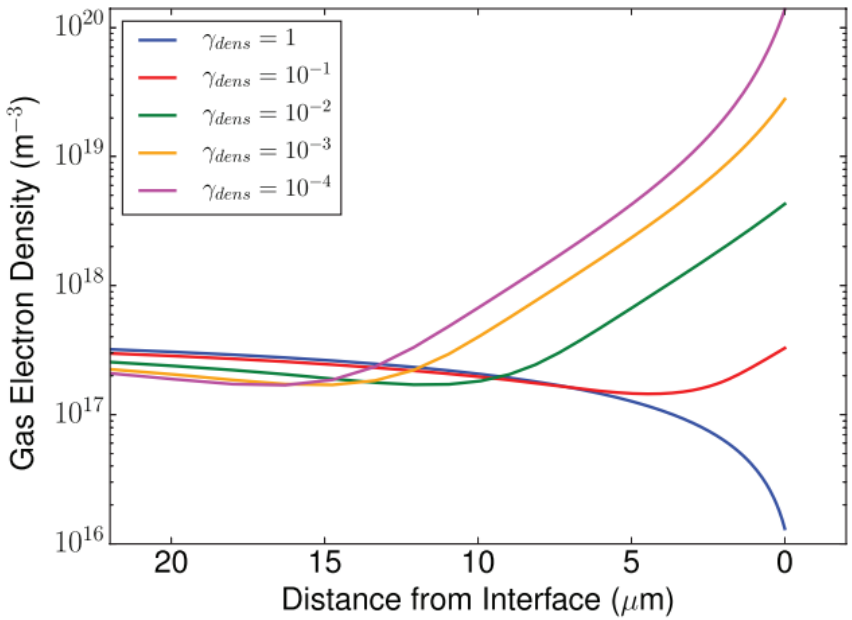
- Work by Alexander Lindsay used Zapdos to show the effects of modifying the surface losses on the interface of an argon plasma and water [3].
  - The circuit schematic of the coupled plasma was as such:



Liquid-Plasma Interactions (cont.)

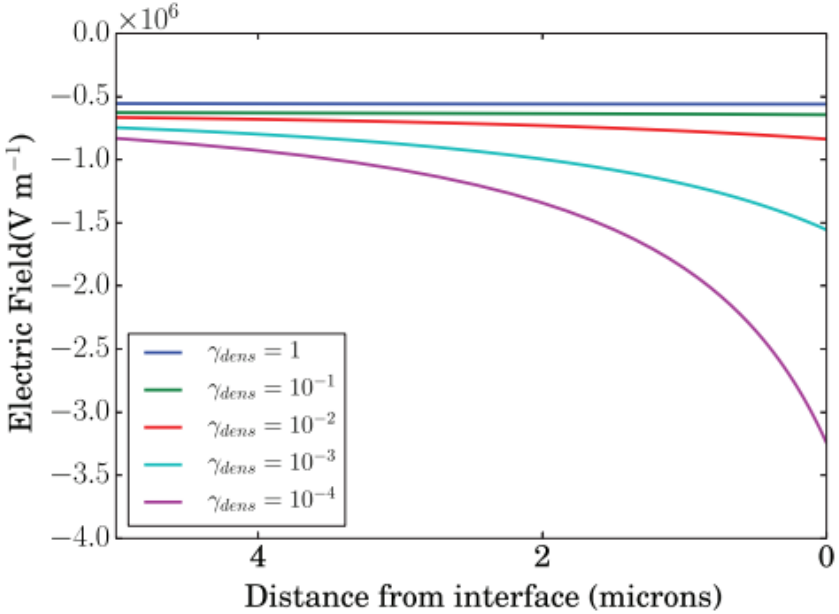
- Results of electron density in both the plasma and water were simulated at different interfacial surface loss coefficient.
- As can be seen in the figures, the change in the interfacial surface loss coefficient plays a little role in the liquid, by the drastically change the electron density in the gas.

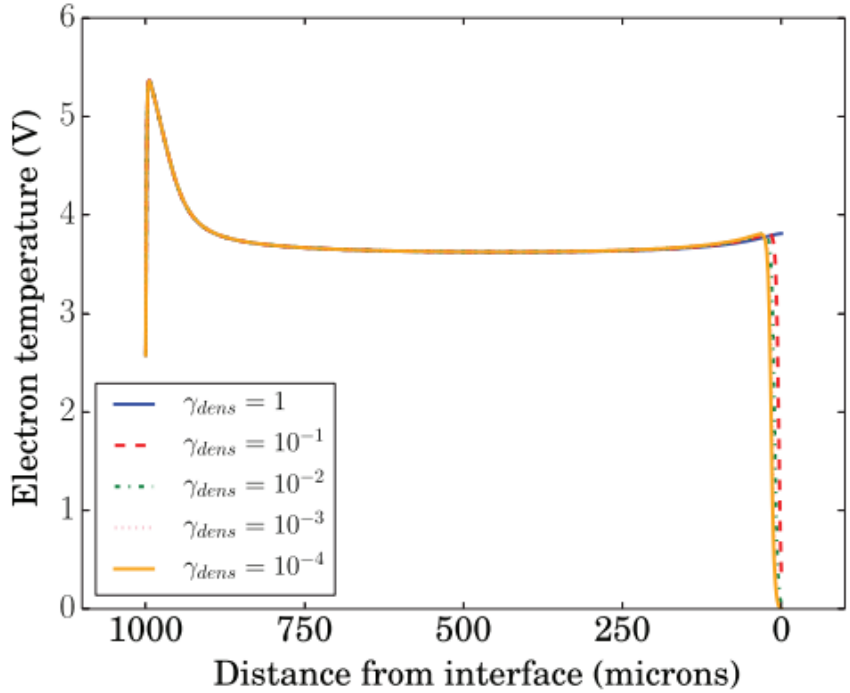




Liquid-Plasma Interactions (cont.)

- With the Auxiliary Kernels in Zapdos, we can see how the parameters (such as electric field and gas temperature) are effected by the change in coefficient.

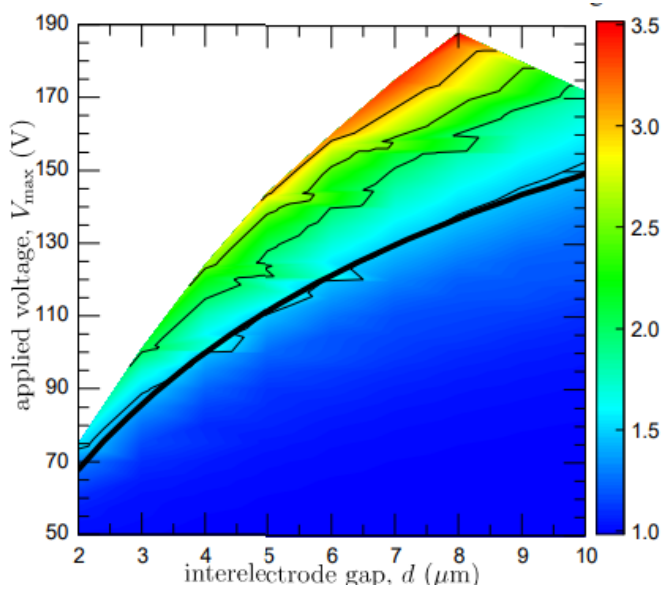




Microscale Gas Discharges to Enhance Thermionic Energy Conversion

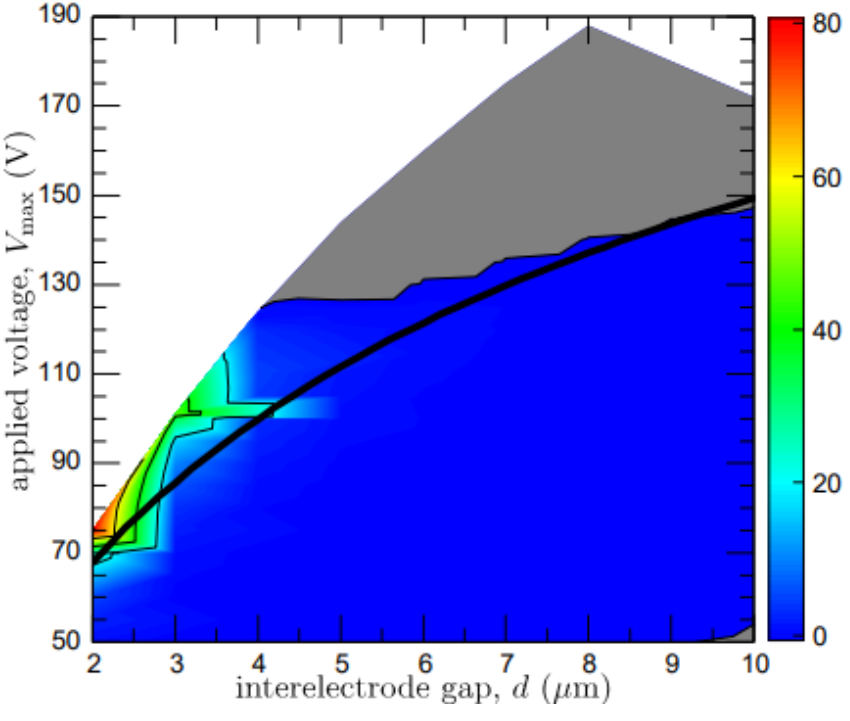
- Works by John Hasse and David Go used Zapdos to plot current and power as a function of applied voltage and position [4].
- This was done to better understand to effects plasma can have a removing negative space charge for a thermionic energy converters (TEC).
  - A TEC is where a hot cathode emits electrons that are then collected by a cold anode.

- The figure is Contours of enhanced emission current relative to emission when there is no space charge for an emitter work function of 3.5 eV averaged over one cycle with an on sub-cycle of 0.5 ns and an off sub-cycle of 20 ns. The black curve shows the breakdown voltage.

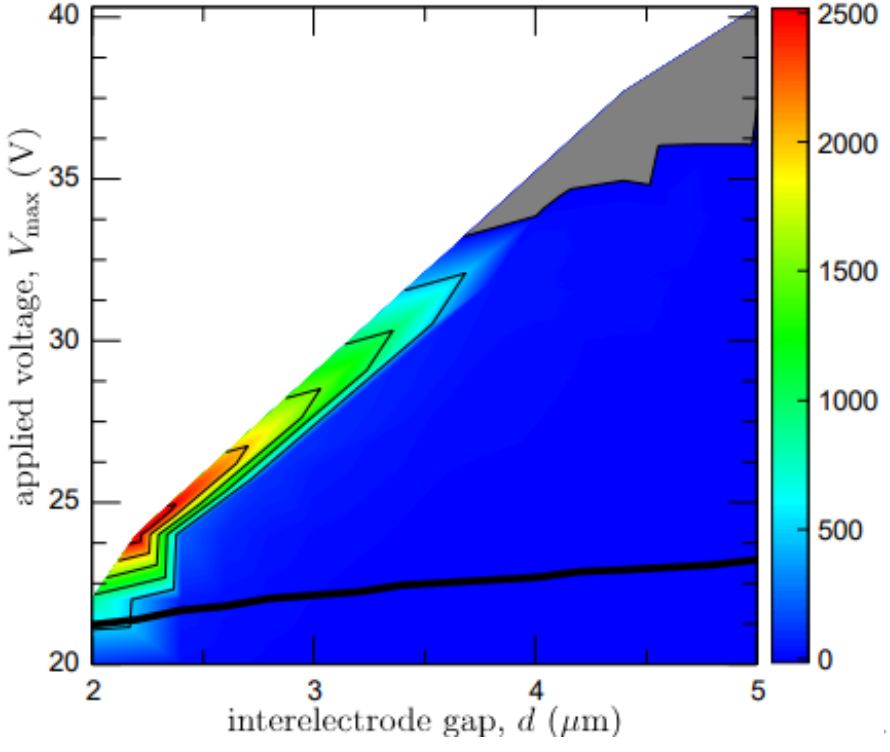


Microscale Gas Discharges to Enhance Thermionic Energy Conversion (cont.)

- Contours of net power produced by a microplasma-enhanced TEC in W/cm2, for an emitter work function of 3.5 eV averaged over one cycle with an on sub-cycle of 0.5 ns and an off sub-cycle of 20 ns. The black curve shows the breakdown voltage, and gray regions indicate no net power.

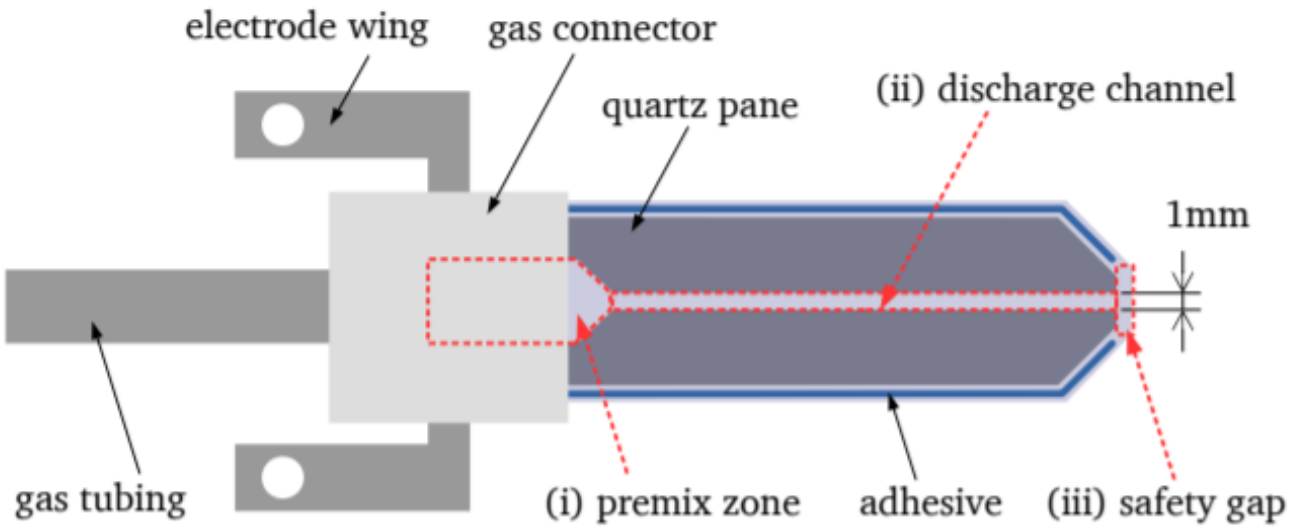


- Contours of net power produced by a microplasma-enhanced TEC in W/cm2, for an emitter work function of 2.5 eV averaged over one cycle with an on sub-cycle of 0.5 ns and an off sub-cycle of 20 ns. The black curve shows the breakdown voltage, and gray regions indicate no net power.



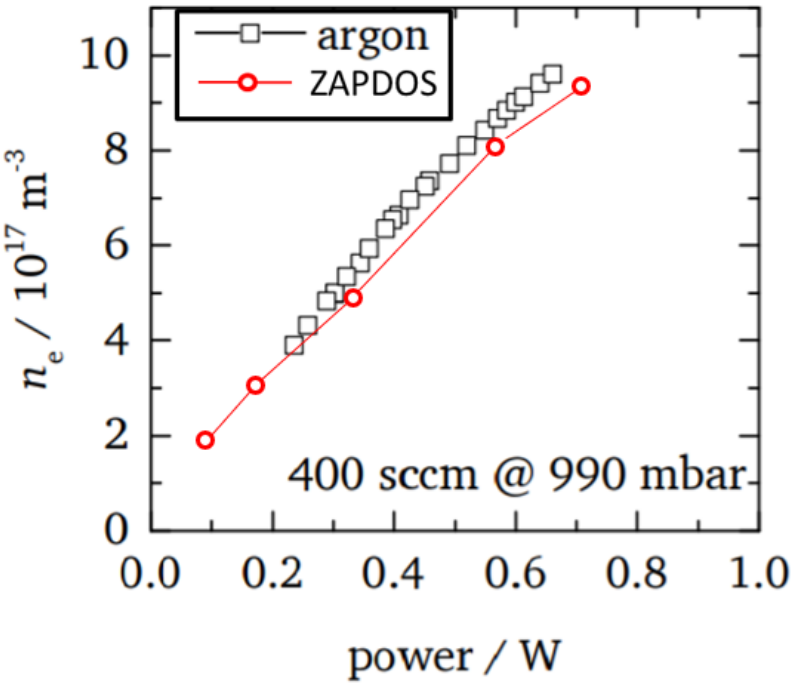
Atmospheric Pressure Plasma Jet

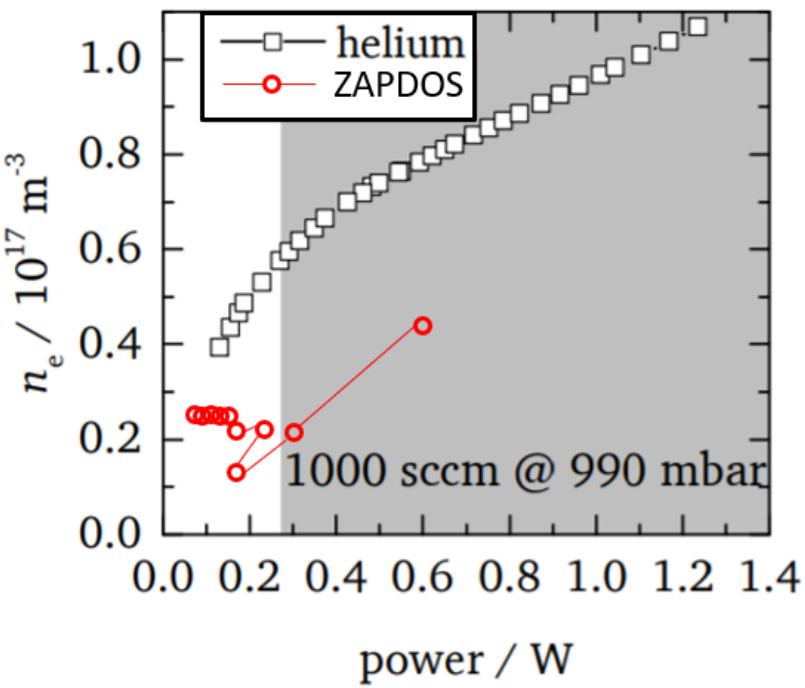
- Current work with Zapdos is focus on modeling the COST-Jet, a reference source for atmospheric pressure plasma jet.
  - The COST-Jet forms plasma by applying a rf voltage at 13.56MHz to a 30mm x 1mm cathode.
- When coupled to CRANE, Zapdos has shown comparative results for both experimental and PIC results.



Atmospheric Pressure Plasma Jet (cont.)

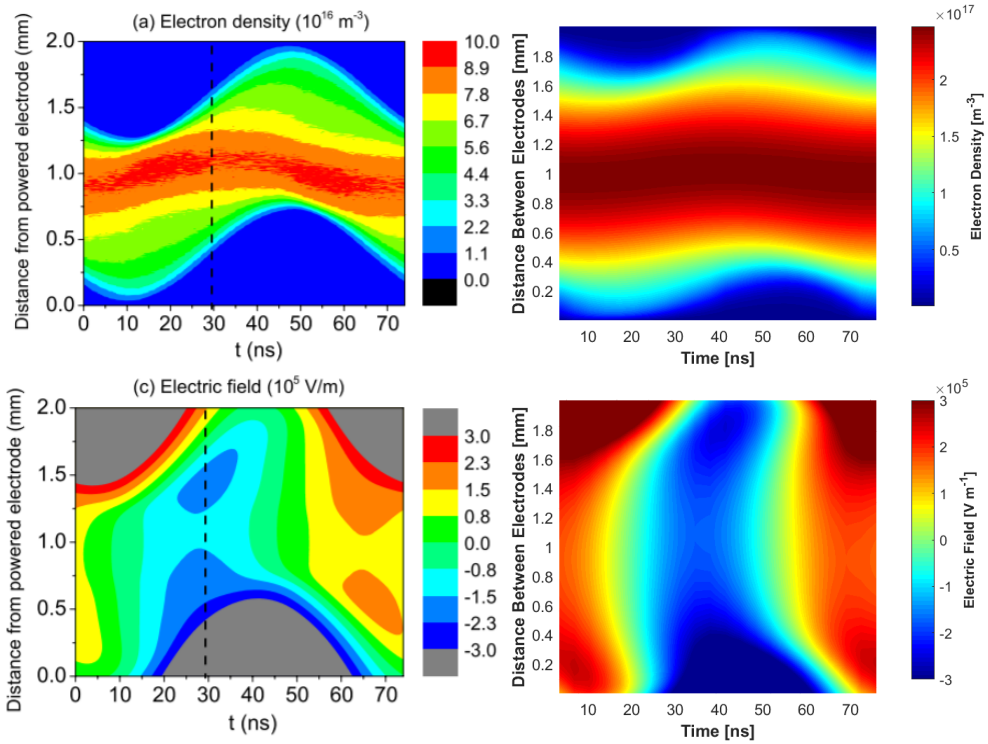
- The following is comparison to experimental work done by Judith Golda [5] (on the right) to Zapods (on the left).
- The argon agreed relatively, while Zapdos underpredicted the electron density of helium.
  - This is mostly due to the transition to γ model, depnoted by the gray section in the experimental work. To over come this, most simulations set a desired power instead of applied voltage.
  - Currently, Zapdos lacks this option but work has begin to add it.



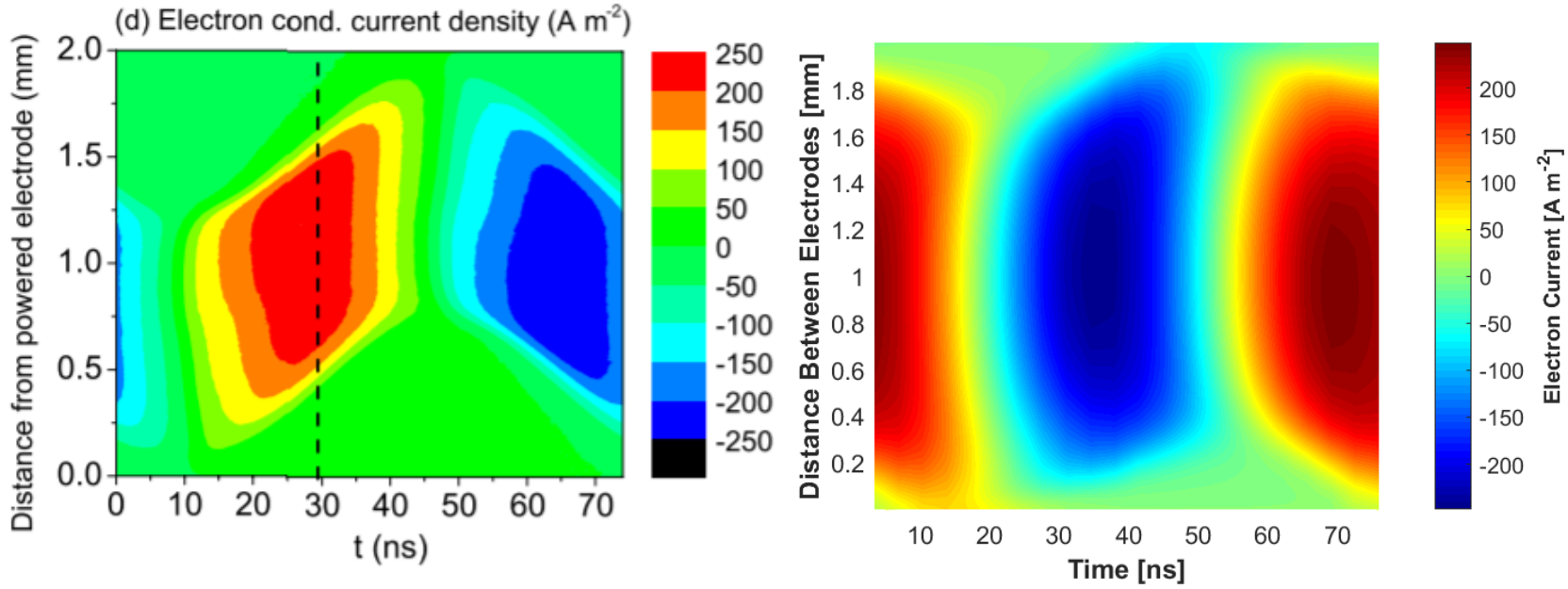


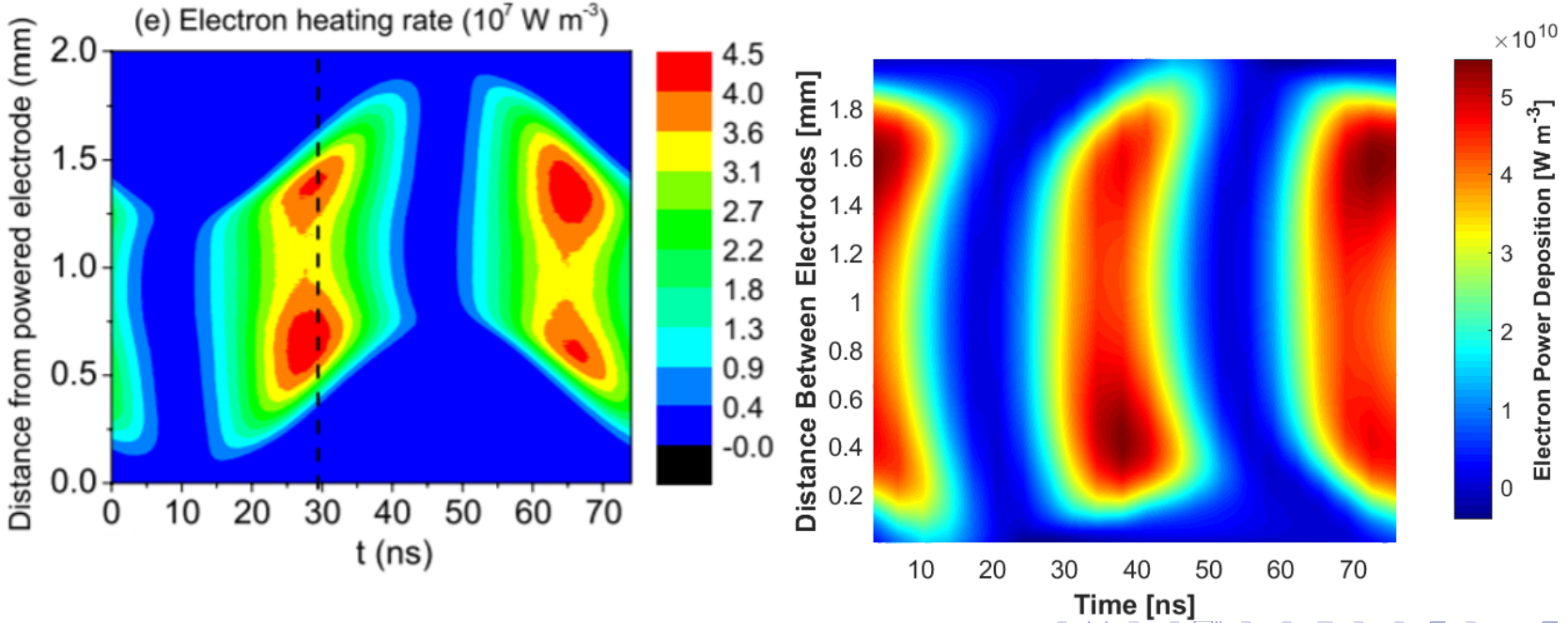
Atmospheric Pressure Plasma Jet (cont.)

- Zapdos (on the right) was also compared with a PIC simulation done by T Hemke *et al* [6] (on the left).
  - The simulations was done for helium gas with an applied rf voltage of 500V at 13.56MHz.
  - The behaviors between models were similar, while Zapdos often predicted higher magnitudes.



Atmospheric Pressure Plasma Jet (cont.)





Contents of Zapdos

- Zapdos is available at:

  `https://github.com/lindsayad/zapdos`

- Join the zapdos user Google Group. Asking any question and the Zapdos community will answer it!

  `https://groups.google.com/forum/#!forum/zapdos-users`

- The MOOSE google group is useful for questions about the MOOSE framework in general (feedback directly from the MOOSE developers is generally on the order of hours):

  `https://groups.google.com/forum/#!forum/moose-users`

Works Cited

1. Hagelaar G J M, De Hoog F J and Kroesen G M W "Boundary conditions in fluid models of gas discharges" *Phys. Rev. E* **62** (2000) 14524
2. Sakiyama Y and Graves D B "Nonthermal atmospheric RF plasma in one-dimensional spherical coordinates: asymmetric sheath structure and the discharge mechanism" *J. Appl. Phys.* **101** (2007) 073306
3. Alexander D Lindsay *et al* "Fully coupled simulation of the plasma liquid interface and interfacial coefficient effects" *J. Phys. D: Appl. Phys.* **49** (2016) 235204
4. John R. Hasse and David B. Go "Designing microscale gas discharges to enhance thermionic energy conversion," Proceedings of the 2017 30th International Vacuum Nanoelectronics Conference (IVNC), July 2017.
5. Judith Golda, Cross-correlating discharge physics, excitation mechanisms and plasma chemistry to describe the stability of an RF-excited atmospheric pressure argon plasma jet (2017) Unpublished Thesis
6. T Hemke *et al.* Ionization by bulk heating of electrons in capacitive radio frequency atmospheric pressure microplasmas, *Plasma Sources Sci. Technol.* **22** (2013) 015012 (8pp)

Kernels

- List of all Kernels current available:

Kernel Name	Governing Eqn. Term	Description
ElectronTimeDerivative	$\psi_i e^u \partial u/\partial t$	Generic accumulation term
EFieldAdvectionElectrons	$-\nabla\psi_i \mu_e(N_e, E_n) e^{N_e}$	Electron specific electric field driven advection term
CoeffDiffusionElectons	$\nabla\psi_i D_e(N_e, E_n) e^{N_e} \nabla N_e / l_c^2$	Electron specific diffusion terms
ElectronsFromIonization	$-\psi_i \alpha_{iz}(N_e E_n) \vec{\Gamma_e}$	Rate of production of electrons from ionization
LogStabilizationMoles	$-\psi_i e^{-(b+u)}$	Kernel stabilizes solution variable *textit*(u) in places where *textit*(u) 0; *textit*(b) is the offset value specified by the user. A typical value for *textit*(b) is 20.
EFieldAdvection	$-\nabla\psi_i(N_e, E_n) e^{N_e} \nabla V / l_c^2$	Generic electric field driven advection term.
CoeffDiffusion	$\nabla\psi_i D_e(N_e, E_n) e^{N_e} \nabla N_e / l_c^2$	Generic diffusion term
ReactantFirstOrderRxn	$\psi_i k e^u$	Generic first order reaction sink term for *textit*(u) (*textit*(u) is the reactant); *textit*(k) is the reaction rate coefficient

Kernel Name	Governing Eqn. Term	Description
ReactantAARxn	$2\psi_i k e^{2u}$	Generic second order reaction sink term for $textit(u)$ in which two molecules of $textit(u)$ u are consumed
CoeffDiffusionLin	$\nabla\psi_i D_e(N_e, E_n) e^{N_e} \nabla N_e / l_c^2$	Generic $textitlinear$ diffusion term, e.g. this is a diffusion term for solution variables not cast in a logarithmic form
ChargeSourceMoles$_K V$	$-\psi_i esgn(q) N_A e^v / V_c$	Used for adding charged sources to Poissons equation; $(e^v) represents the charged particle density of species$. This kernel assumes that densities are measured in units of mol/volume as opposed to /volume.
IonsFromIonization	$-\psi_i \alpha_{iz}(N_e, E_n) \vec{\Gamma_e}$	Same governing term/residual as ElectronsFromIonization; however, the Jacobian structure is different. Ri Ne will be on-diagonal for ElectronsFromIonization and off-diagonal for IonsFromIonization
ProductFirstOrderRxn	$\psi_i k e^u$	Generic first order reaction source term for $textit(u)$ ($textit(v)$ is the reactant)

Kernel Name	Governing Eqn. Term	Description
ProductAABBRxn	$2\psi_i k e^{2u}$	Generic second order reaction source term in which two molecules of $textit(v)$ are produced from two molecules of $textit(u)$
EFieldAdvectionEnergy	$-\nabla\psi_i \mu_e(N_e, E_n) e^{N_e} \nabla V / l_c^2$	Electron energy specific electric field driven advection term
CoeffDiffusionEnergy	$\nabla\psi_i D_e(N_e, E_n) e^{N_e} \nabla N_e / l_c^2$	Electron energy specific diffusion term
JouleHeating	$-\psi_i \nabla V V_c / l_c \cdot \vec{\Gamma_e}$	Joule heating term for electrons
ElectronEnergyLossFromIonization	$-\psi_i \alpha_{iz}(N_e, E_n) \vec{\Gamma_e} E_{iz}$	Electron energy loss term for inelastic ionization collisions; Eiz is the energy lost in Volts in a single ionization collision
ElectronEnergyLossFromExcitation	$-\psi_i \alpha_{iz}(N_e, E_n) \vec{\Gamma_e} E_{iz}$	Electron energy loss term for inelastic excitation collisions; Eex is the energy lost in Volts in a single excitation collision
ElectronEnergyLossFromElastic	$-\psi_i \alpha_{iz}(N_e, N_n) \vec{\Gamma_e} E_{iz}$	Electron energy loss term for elastic collisions. el is the elastic Townsend coefficient; me is the electron mass; mn is the mass of the neutral background gas; Te = 2 3 is the electron temperature

AuxKernels

- List of all Auxiliary Kernels current available:

AuxKernel Name	**Governing Eqn. Term**	**Description**
PowerDep	$\mathrm{sgn}(q)eN_A\cdot(sgn\mu\cdot\text{-}\nabla Ve^{N_k}\text{-}De^{N_k}\nabla N_k)\cdot\text{-}\nabla VV_c/l_c^2$	Amount of power deposited into a user specified specie by Joule Heating
ProcRate	$N_A\mu\cdot\text{-}\nabla Ve^{N_k}\text{-}De^{N_k}\nabla N_k\cdot\alpha/l_c$	Reaction rate for electron impact collisions in units of m3s . User can pass choice of elastic, excitation, or ionization
ElectronTemperature	$\frac{2}{3}e^{E_n-N_e}$	The electron temperature
Position	xl_c	Produces an elemental auxiliary variable useful for plotting against other elemental auxiliary variables.
Density	$e^{N_k}N_A$	Returns physical densities in

InterfaceKernels

- List of all InterFace Kernels current available:

InterfaceKernel Name	Governing Eqn. Term	Description
InterfaceAdvection	$-\psi_{i,el}$ $\mu_{k,n} sgn(q_k) e^{N_{k,n}} V_n \cdot \vec{n} / (l_{c,n} l_{c,el})$	Used to include the electric field driven advective flux of species k into or out of a neighboring subdomain. The subscript el denotes the subdomain to which the InterfaceAdvection residual is being added. The subscript n denotes the neighboring subdomain. Currently this interface kernel is specific to electrons because the transport coefficients are assumed to be a function of the mean electron energy. A generic interface kernel with constant transport coefficients will have a much simpler Jacobian
InterfaceLogDiffusionElectrons	$-\psi_{i,el} D_{k,n} e^{N_{k,n}} \cdot \vec{n} / (l_{c,n} l_{c,el})$	Used to include the diffusive flux of species k into or out of a neighboring subdomain. Also currently specific to electrons.

Boundary Conditions

- List of all Boundary conditions currently available:

Boundary Condition Name	Governing Eqn. Term	Description
HagelaarIonAdvectionBC	$\psi_i(1-r)/(1+r)(-2a-1)\mu\nabla Ve^u\vec{n}l_c^2$	Kinetic advective ion boundary condition
HagelaarIonDiffusionBC	$\psi_i(1-r)/(1+r)0.5v_{th}e^u$	Kinetic diffusive ion boundary condition
HagelaarElectronBC	$\psi_i(1-r)/(1+r)(-2a-1)\mu\nabla Ve^u\vec{n}l_c^2+0.5v_{th}e^u s$	Kinetic electron boundary condition
HagelaarEnergyBC	$\psi_i/(6(r+1))(10\vec{\Gamma}_i)$	Kinetic electron energy boundary condition
DCIonBC	$\psi_i a\mu\nabla Ve^u\vec{n}l_c^2$	Electric field driven outflow boundary condition
NeumannCircuitVoltageMoles$_K V$	Based of KLV	Circuit boundary condition for potential
MatchedValueLogBC	$He^u e^v$	Henrys Law, like thermodynamic boundary condition for specifying a specie concentration ratio at the gas-liquid interface