SAND2013-4773C

Accommodating Large Temporal, Spatial, and Particle Weighting Demands for Simulating Vacuum Arc Discharge

Matthew M. Hopkins[1], Jeremiah J. Boerner[1], Christopher H. Moore[1], Paul S. Crozier[2], Robert B. Campbell[3], Lawrence C. Musson[4], Matthew T. Bettencourt[2]

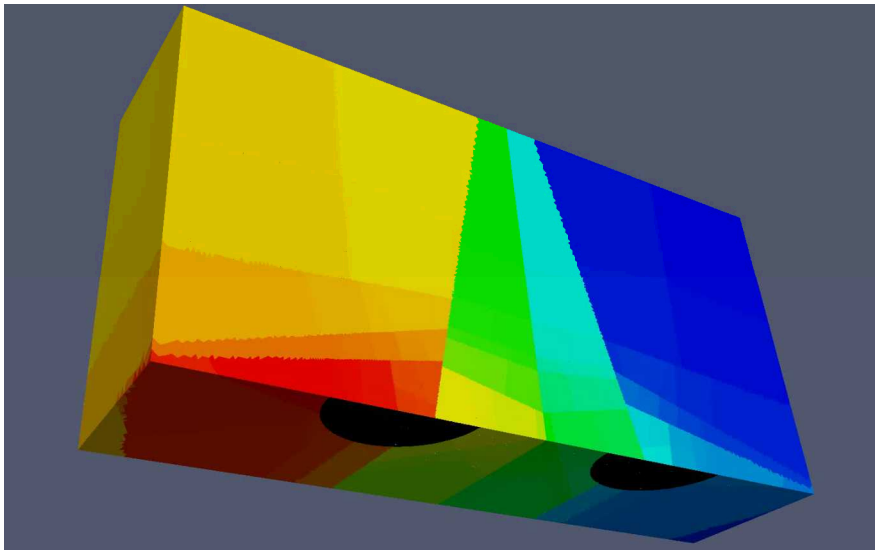
[1]Nanoscale and Reactive Processes,
[2]Scalable Algorithms, [3]HEDP Theory,
[4]Electrical Models & Simulation
Sandia National Laboratories
Albuquerque, NM, USA

19th IEEE Pulsed Power and Plasma Science Conference
40th IEEE International Conference on Plasma Science
June 16 – 21, 2013
San Francisco, CA

SAND2013-xxxx

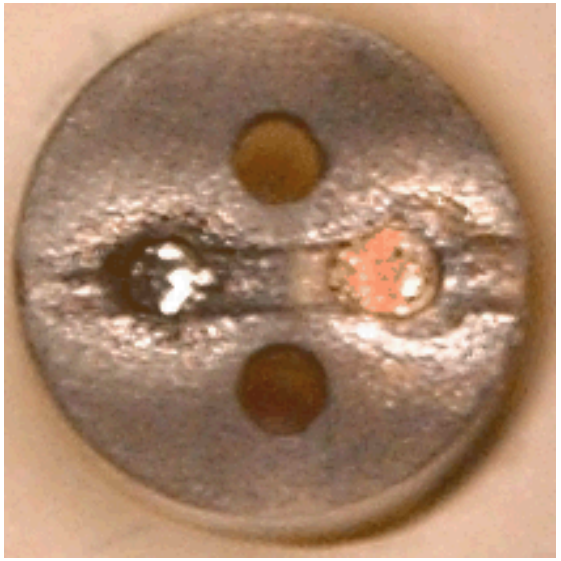
Description of Aleph

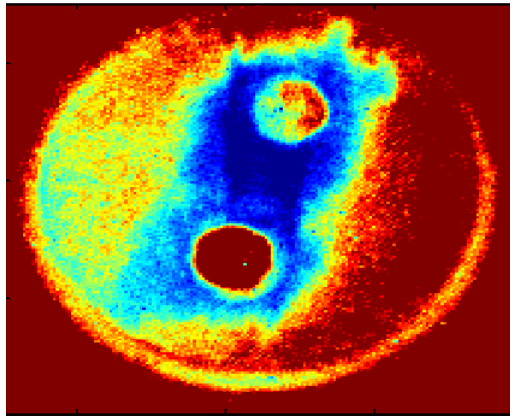
- 1, 2, or 3D Cartesian
- Unstructured FEM (compatible with CAD)
- Massively parallel
- Hybrid PIC + DSMC (PIC-MCC)
- Electrostatics
- Fixed B field
- Solid conduction
- e- approximations (quasi-neutral ambipolar, Boltzmann)
- Dual mesh (Particle and Electrostatics/Output)
- Advanced surface (electrode) physics models
- Collisions, charge exchange, chemistry, excited states, ionization
- Advanced particle weighting methods
- Dynamic load balancing (tricky)
- Restart (with all particles)
- Agile software infrastructure for easily extending BCs, post-processed quantities, etc.
- Currently utilizing up to 64K processors (>1B elements, >1B particles)



256 core particle load balancing example

Target Application: 3D Cu-Cu Arc System





At vacuum or 4 Torr Ar background

1.5 mm inner-to-inner distance

0.75 mm diameter electrodes

Copper electrodes (this picture is Cu-Ti)

2 kV drop across electrodes

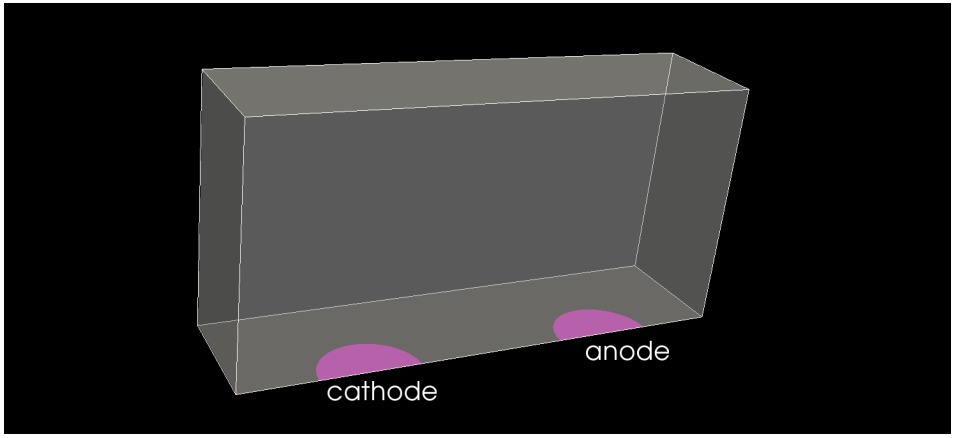
20Ω resistor in series

Steady conditions around 50V, 100A

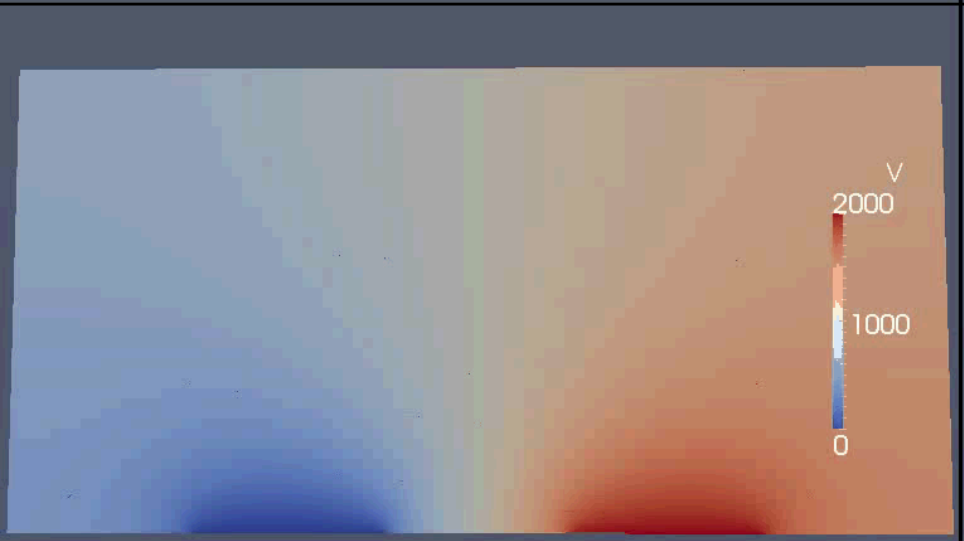
Breakdown time << 100ns

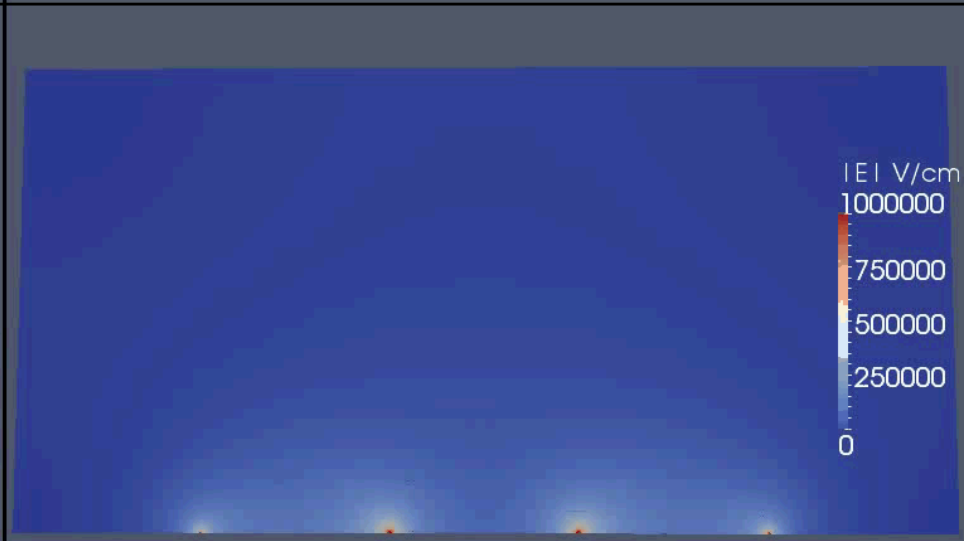
Ionization mfp = 1.5 mm at maximum σ

→ n_i ~ 10^{17} #/cm^3



3D Model of Cu-Cu Arc System





Simulation Requirements

Temporal scales dominated by plasma electron frequency ω_p, CFL, and collision frequency ν_c at different phases of breakdown:

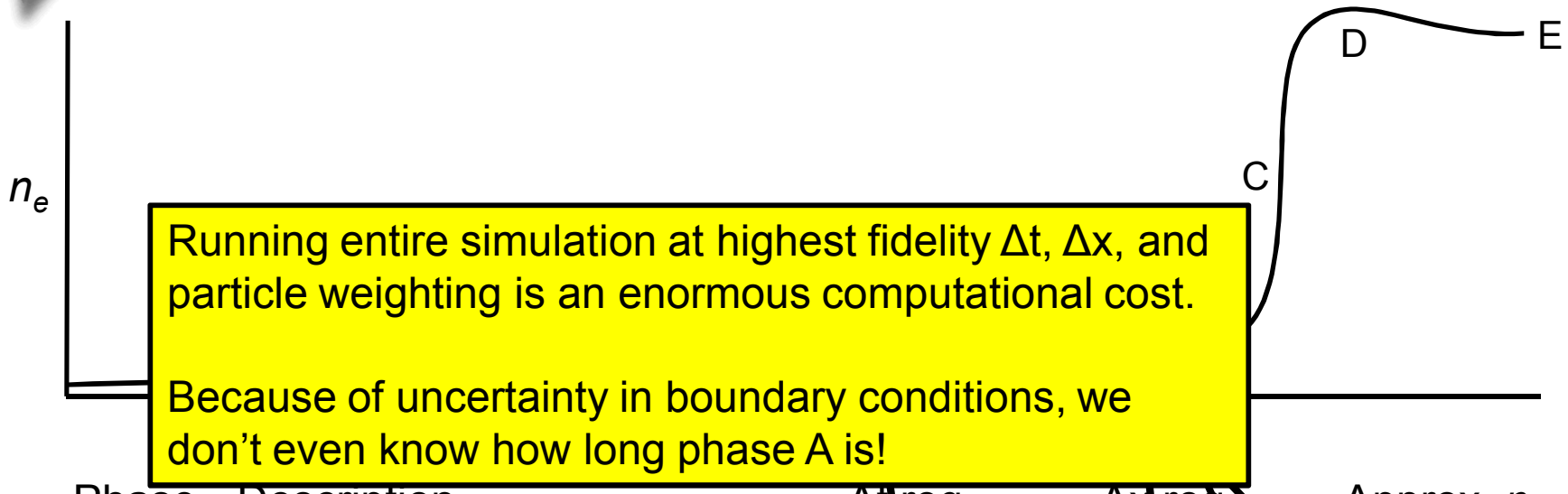
$$\Delta t < \min\left(\frac{2}{\omega_p}, \frac{\Delta x}{\sqrt{\frac{m_e \Delta V}{2q_e}}}, \frac{1}{n_n \sigma \bar{v}}\right)$$

Spatial scales dominated by Debye length λ_D and collision mean free path λ_{mfp} at different phases of breakdown:

$$\Delta x < \min\left(\lambda_D, \frac{1}{n_n \sigma}\right)$$

Number densities increase from "0" to 10^{17} #/cm^3. Using same fixed particle weight p_{weight} isn't an option.

Simulation Phases and Requirements



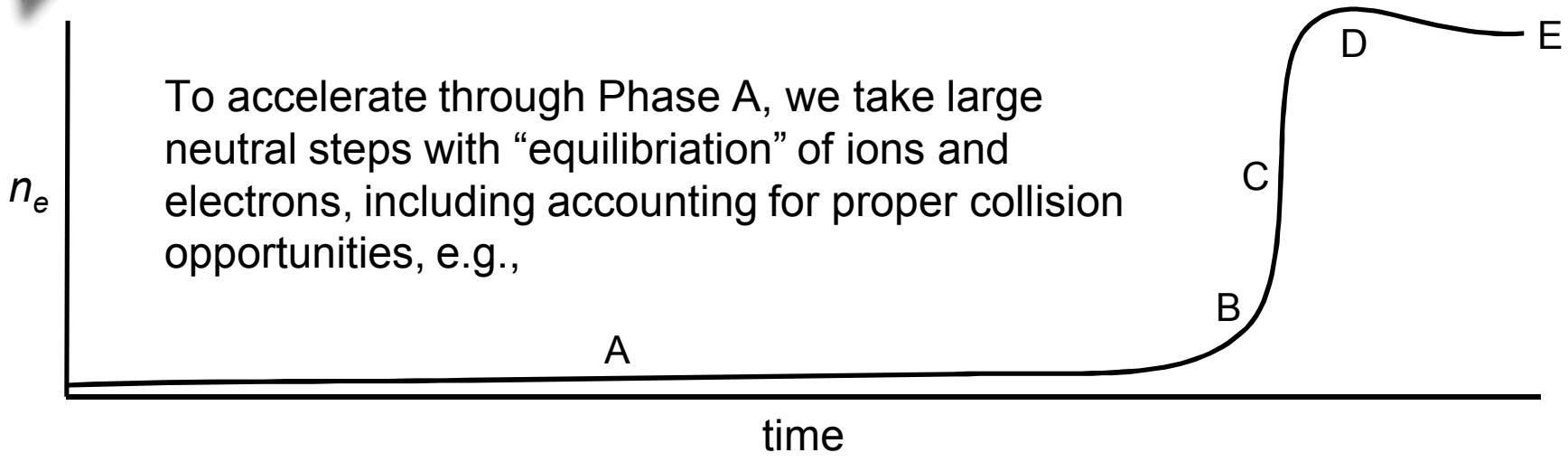
Running entire simulation at highest fidelity Δt, Δx, and particle weighting is an enormous computational cost.

Because of uncertainty in boundary conditions, we don't even know how long phase A is!

Phase	Description	Δt req.	Δx req.	Approx. n_i
A	pre-breakdown neutral growth	CFL	none	0 #/cm^3
B	breakdown initiation	CFL, ω_p	λ_D	10^{11} #/cm^3
C	breakdown	CFL, ω_p, ν_c	λ_D, λ_{mfp}	10^{16} #/cm^3
D	post-breakdown equilibriation	ω_p, ν_c (*)	λ_D, λ_{mfp} (*)	10^{18} #/cm^3 (?)
E	post-breakdown steady arc	ω_p, ν_c (*)	λ_D, λ_{mfp} (*)	10^{17} #/cm^3

(*) Depending on system model, there is a competition between generation and ionization of neutrals. At a minimum, Δx and Δt limits have large spatial variation.

Quasi-static Acceleration



For each of 400 $\Delta t_{neutral}$ steps,
- move neutrals
- neutral-neutral interactions
- for each of 10 Δt_{ion} steps,
  - move ions
  - ion-neutral interactions
  - ion-ion interactions
  - for each of 10 $\Delta t_{electron}$ steps,
    - move electrons
    - electron-* interactions

and

For each of 40 10x$\Delta t_{neutral}$ steps,
- move neutrals
- neutral-neutral interactions
- for each of 100 Δt_{ion} steps,
  - move ions
  - ion-neutral interactions
  - ion-ion interactions
  - for each of 10 $\Delta t_{electron}$ steps,
    - move electrons
    - electron-* interactions

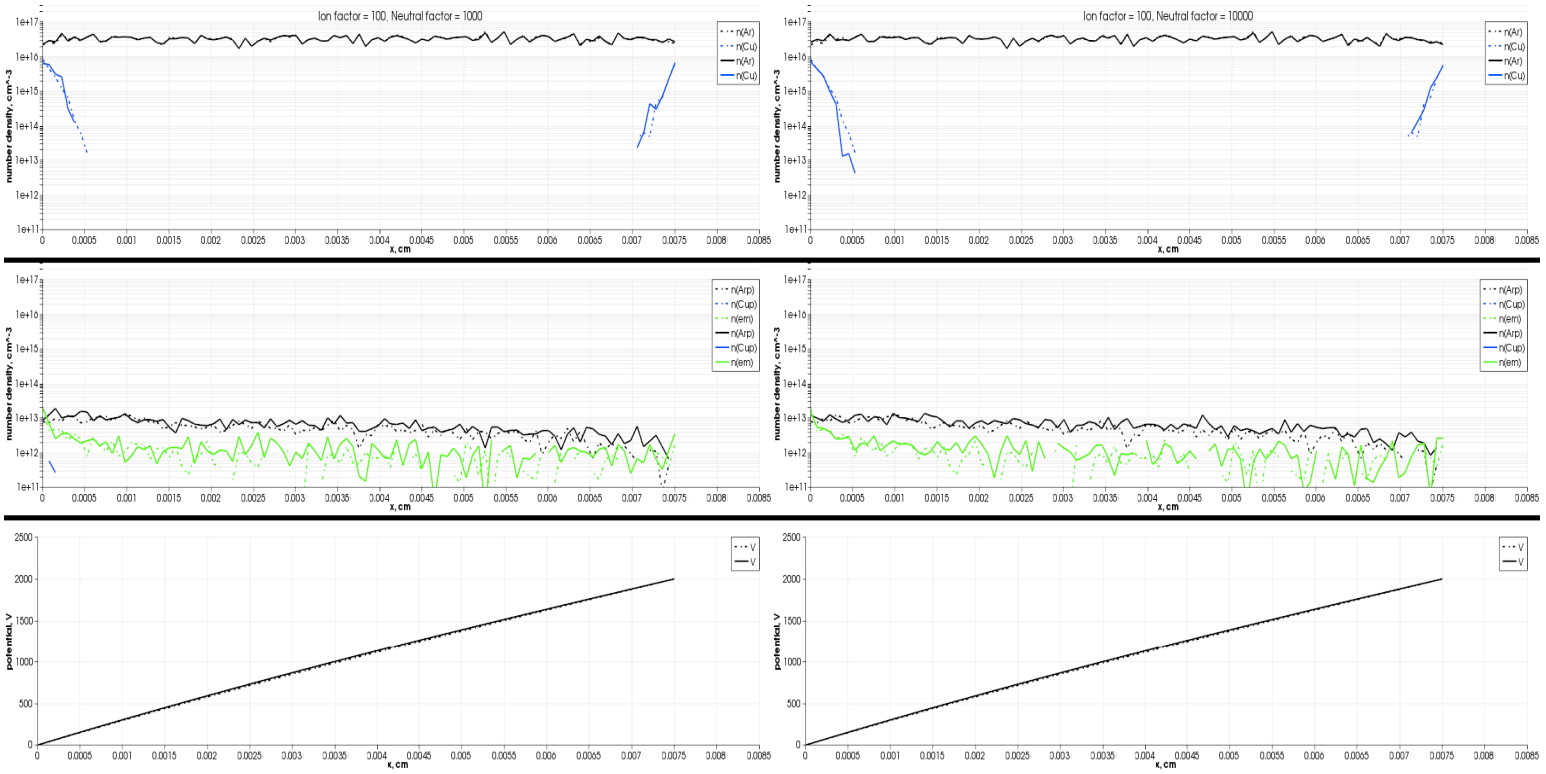
Quasi-static Acceleration

- Dashed lines are no acceleration.
- Neutral sputtering BC's.
- Cathode on left, anode on right.
- Influx of e- from cathode.

n_n
Ar background
Cu from surface

n_i for Ar+, Cu+
n_e

V



400 neutral steps

40 (larger) neutral steps

10x speed up!

Successive (Δx, Δt, p_{weight}) Refinement

Recall constraints become progressively stronger through each phase, and the leading constraint will switch:

	Δt	Δx	N (particle weight)
pre-breakdown neutral growth	CFL	none	0 #/cm^3
breakdown initiation	CFL, ω_p	λ_D	10^{11} #/cm^3
breakdown	CFL, ω_p, ν_c	λ_D, λ_{mfp}	10^{16} #/cm^3
post-breakdown equilibriation	ω_p, ν_c (*)	λ_D, λ_{mfp} (*)	10^{18} #/cm^3 (?)
post-breakdown steady arc	ω_p, ν_c (*)	λ_D, λ_{mfp} (*)	10^{17} #/cm^3

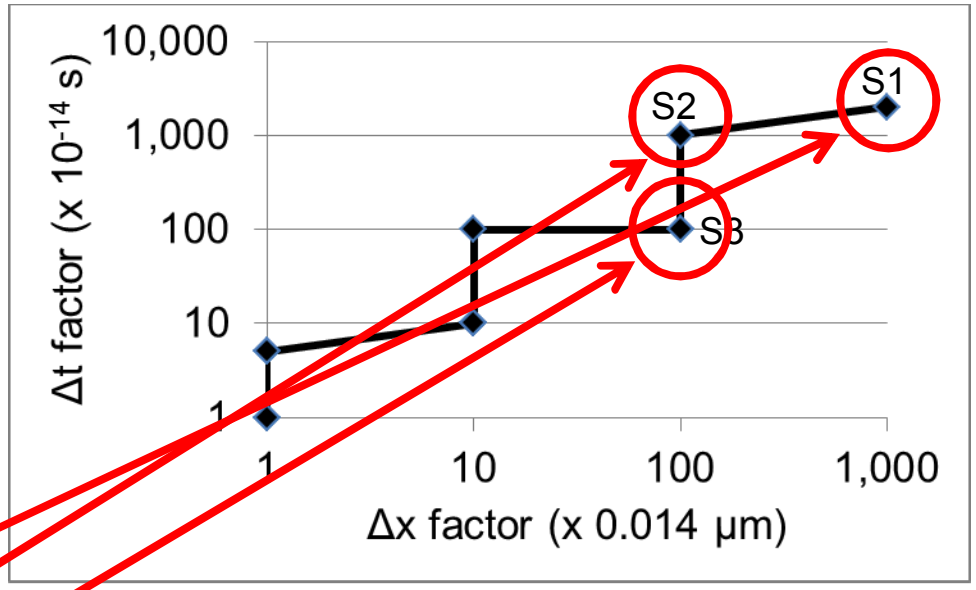
Approach:

Start with (Δx, Δt, p_{weight}) “coarse” simulation and:

- Continually refine in p_{weight} through particle merging
- Discretely refine in (Δx, Δt) by stopping simulation near stability/fidelity limits and perform full particle restart on Δx- and/or Δt-refined simulation.

Successive (Δx, Δt, p_{weight}) Refinement

A typical progression to (Δx, Δt) = (0.014 μm, 10 fs) looks like:



S1: (Δx, Δt) = (0.014 mm, 20 ps), or 2,000,000 x less work than final solution steps.

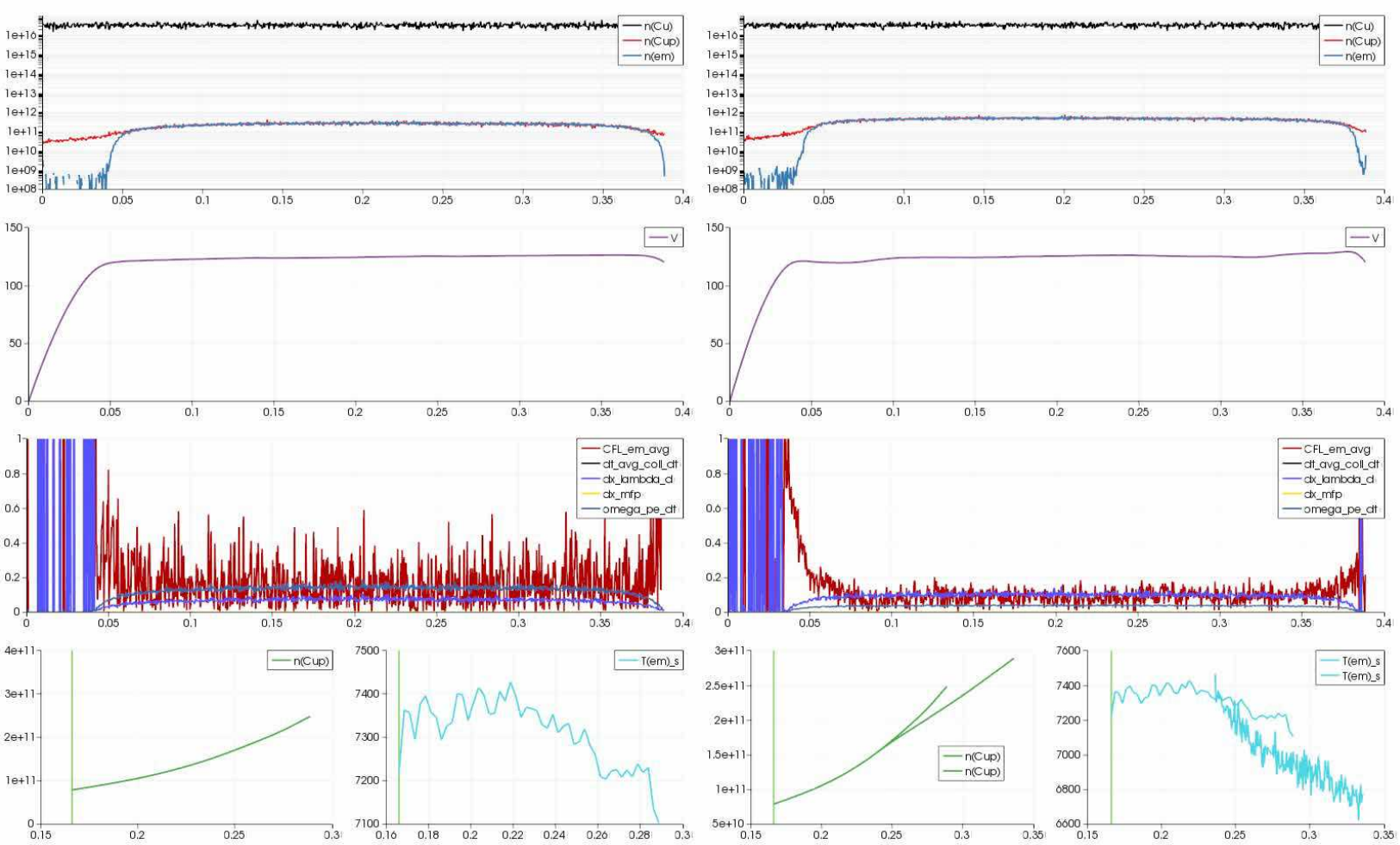
S2: after 160 ns, both Δx and p_{weight} are being challenged, so move to ...

S3: after another 0.990 ns, only p_{weight} is being challenged, so move to ...

... and continue ... (right now this is manual, want to automate termination ...)

Total savings to 1.35 μs (this case) is tremendous, but still need many small steps on small mesh at end...

Successive (Δx, Δt, p_{weight}) Refinement



Conclusions

- Vacuum arc discharge simulations place enormous simulation demands on “vanilla” methods.
- We are developing many approaches to mitigate computational costs, two of which were presented (*).
  - Quasi-static acceleration (*)
  - Successive refinement (*)
  - Dynamic particle weighting / particle merging
  - Implicit kinetic methods
  - Adaptive time-stepping
  - Oct-tree DSMC collision mesh separate from PIC mesh
  - P^3M

Thank You!

Description of Aleph

Basic algorithm for one time step of length Δt:

1. Given known electrostatic field $\mathbf{E}^n$, move each particle for $\frac{\Delta t}{2}$ via:

$$v_i^{n+1/2} = v_i^n + \frac{\Delta t}{2}\left(\frac{q_i}{m_i}\mathbf{E}^n\right)$$

$$x_i^{n+1} = x_i^n + \Delta t v_i^{n+1/2}$$

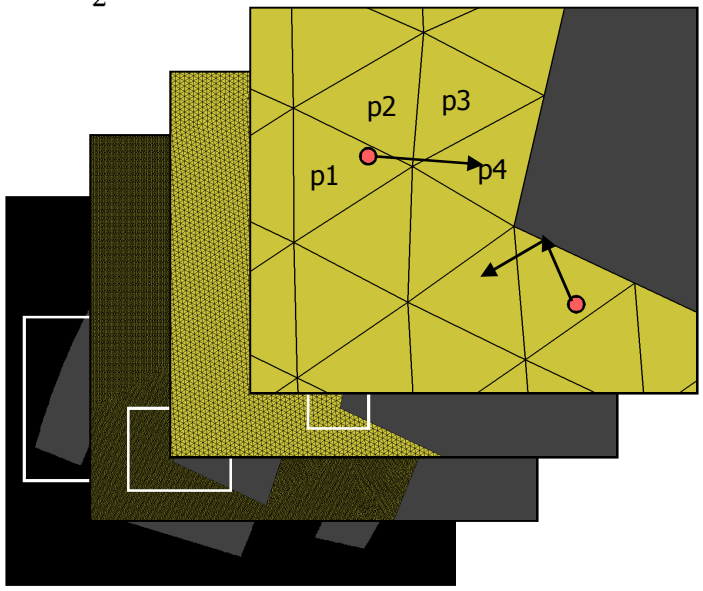
2. Compute intersections (non-trivial in parallel).
3. Transfer charges from particle mesh to static mesh.
4. Solve for $\mathbf{E}^{n+1}$

$$\nabla \cdot \left(\epsilon \nabla V^{n+1}\right) = -\rho(\mathbf{x}^{n-1})$$

$$\mathbf{E}^{n+1} = -\nabla V^{n+1}$$

5. Transfer fields from static mesh to dynamic mesh.
6. Update each particle for another $\frac{\Delta t}{2}$ via:

$$v_i^{n+1} = v_i^{n+1/2} + \frac{\Delta t}{2}\left(\frac{q_i}{m_i}\mathbf{E}^{n-1}\right)$$

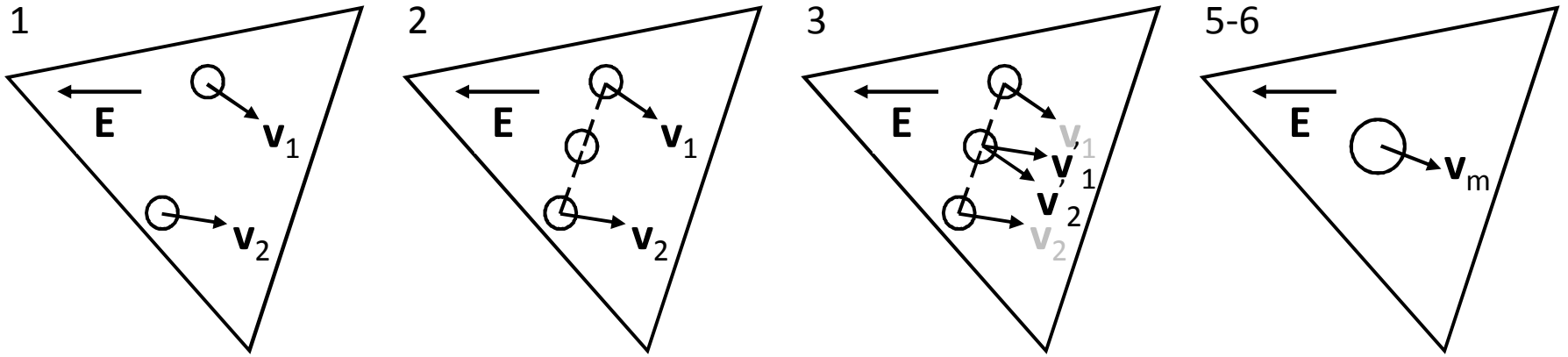


7. Perform DSMC collisions: sample pairs in element, determine cross section and probability of collision. Roll a digital die, and if they collide, re-distribute energy.
8. Perform chemistry: for each reaction, determine expected number of reactions. Sample particles of those types, perform reaction (particle creation/deletion).
9. Reweight particles.
10. Compute post-processing and other quantities and write output.
11. Rebalance particle mesh if appropriate (variety of determination methods).

Merging

1. Choose a random pair of *S* particles.
2. Compute center of mass position.
3. Compute modified velocities at the center of mass by accounting for displacement in the potential field.
4. If velocities are “too different,” reject pair and repeat 1-3.
5. Calculate average velocity, conserving momentum.
6. Adjust (to target) weight and record difference in kinetic energy.

Repeat 1-6 until target number or limiter is met.



Merging

What makes particles “too different” to merge?

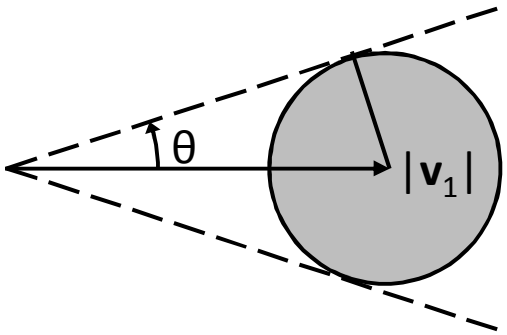
Only approve merge pairs that are close in both position and velocity – close in phase space.

The spatial bin is the element, approves any pair.

The velocity bin has many options. Can use MC sampling to select pairs randomly. (let $|\mathbf{v}_1| < |\mathbf{v}_2|$)

Velocity Sphere

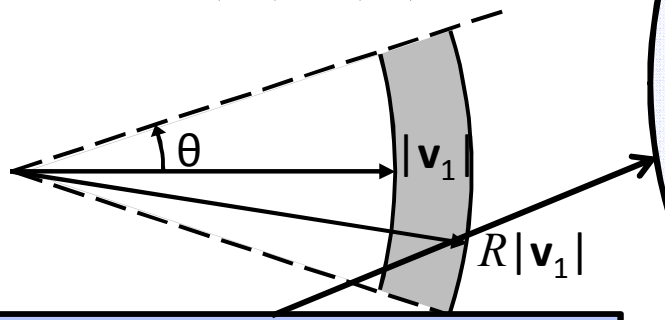
$$|\mathbf{v}_2 - \mathbf{v}_1| < |\mathbf{v}_1|\sin(\theta)$$



Velocity Proportion

$$\mathbf{v}_1 \cdot \mathbf{v}_2 > |\mathbf{v}_1||\mathbf{v}_2|\cos(\theta)$$

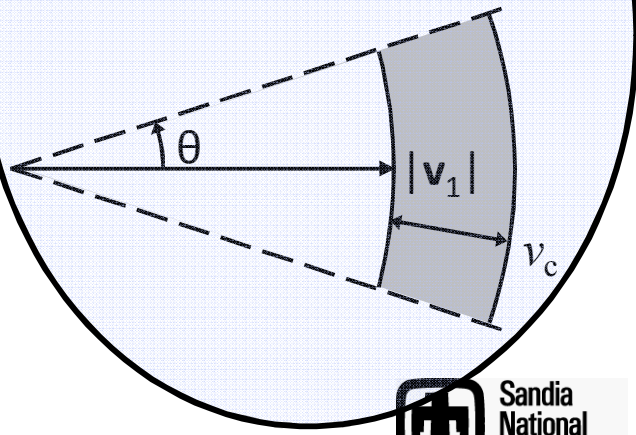
$$|\mathbf{v}_2| < R|\mathbf{v}_1|$$



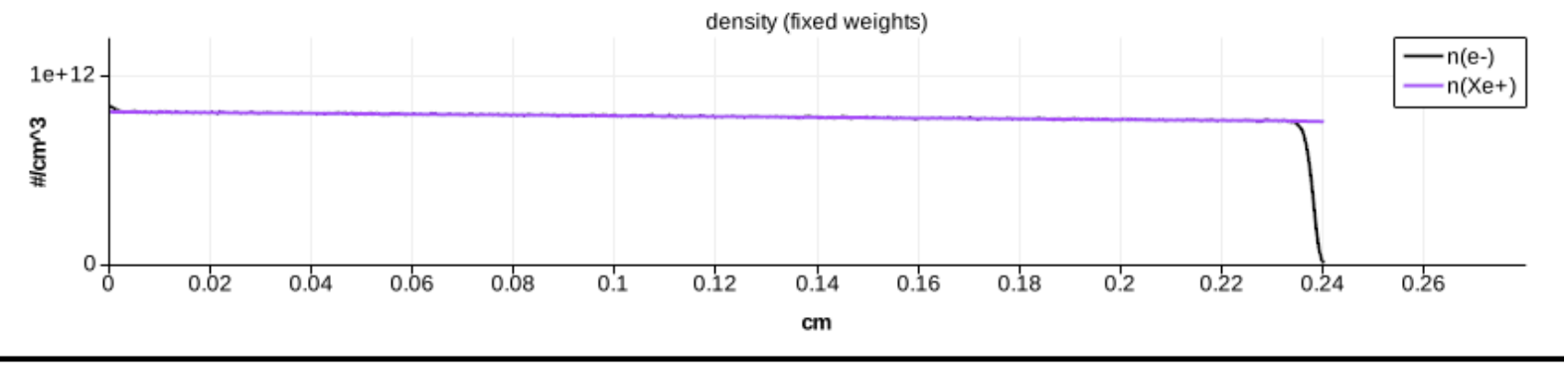
Velocity Interval

$$\mathbf{v}_1 \cdot \mathbf{v}_2 > |\mathbf{v}_1||\mathbf{v}_2|\cos(\theta)$$

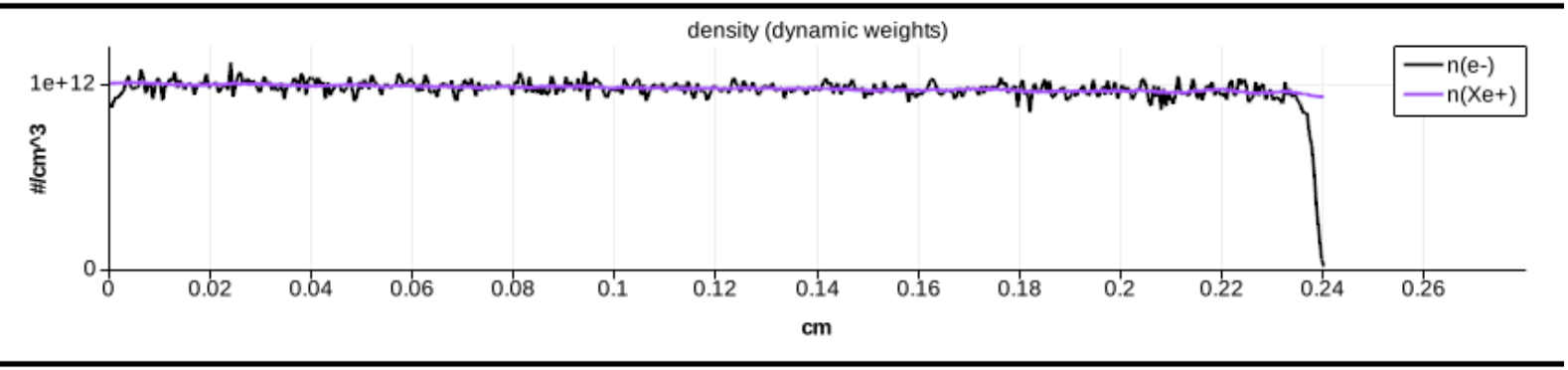
$$|\mathbf{v}_2| - |\mathbf{v}_1| < v_c = \alpha\sqrt{k_B T / m}$$



We use this, plus sorting the S particles by energy in each cell to increase chances of finding merge partner.



147371s,
= 11M



7435s,
= 150k