OuroborosBEM: multi-GPU microscopic simulation for gaseous detectors

Samuel Salvador

Laboratoire de Physique Corpusculaire de Caen

Normandie Univ, ENSICAEN, UNICAEN, CNRS/IN2P3, LPC Caen, 14000 Caen, France

Réseau détecteurs gazeux / 22 mars 2024

OuroborosBEM

Microscopic MC simulation for particle transport in an electric field on GPUs

- Gases available: Ar, CO_2, N_2, CF_4, CH_4, iC_4H_{10}, O_2
- Interaction cross sections for mixture gases
- Electron anisotropy scattering
- Penning transfer probability
- 2 different integration algorithms: Euler or PEFRL
- Tunable time step
- Signal generation using Shockley-Ramo theorem
- Extraction of lots of observables (v_d, <E>, life time,...)
- Creating field, particle and geometry maps every N steps
- Electric field calculation (static, dynamic, RF)

OuroborosBEM

The field

- Calculated using a Boundary Element Method (mesh $\equiv$ surface elements)
  - unknowns: cell surface charge densities $\sigma = Q^{-1}V$ (Q: $\frac{1}{4\pi\epsilon_0}\int\frac{1}{(\vec{r}-\vec{r}')}d^2r'$)
- Common behaviour at start (e.g. 12000 cells):
  1. Fill matrix Q ($\sim$100 ms)
  2. Matrix inversion Q^{-1} ($\sim$1.2 s)

Static option: once at the beginning of the simulation

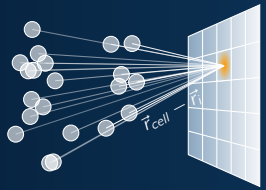
- Creates a field map and a geometry map (and Ramo field maps) at predefined grid points: $Q^{-1}V = \sigma \rightarrow \vec{Q}_{grid}\sigma = \vec{E}$, ($512^3$: $\sim$200 s)

Dynamic option

- Every time step:

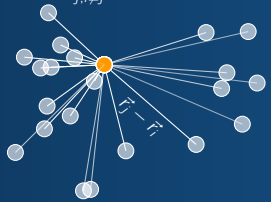
1. Change the potential of each cell
2. Calculate the influence of the particles on the cells:

$$V_{cell\,i} = \sum_{i}^{N_{part}} \frac{1}{4\pi\varepsilon_0} \frac{q}{\| \vec{r}_{cell} - \vec{r}_i \|}$$



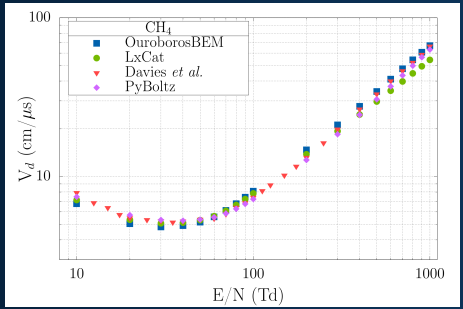
3. $Q^{-1} V = \sigma$
4. $\vec{Q}_{part}\sigma = \vec{E}$
5. Add the contribution of the particles on the field:

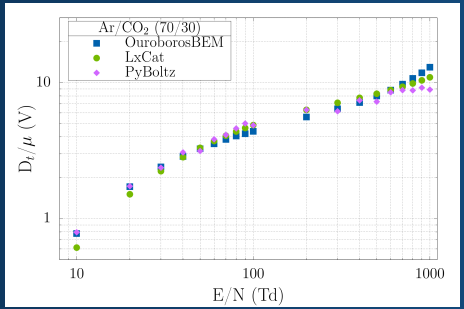
$$\vec{E}_i = \sum_{j,i\neq j}^{N_{part}} \frac{q^2}{4\pi\varepsilon_0} \frac{\vec{r}}{\| \vec{r}_i - \vec{r}_j \|^3}$$



Validation

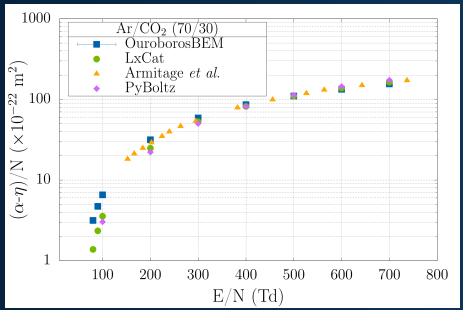
Drift velocities and diffusion coefficient

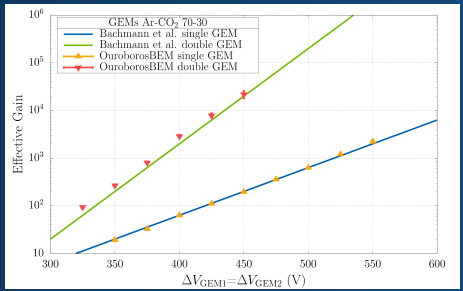




Validation

First Townsend coefficient α and gas gain in GEMs ($r_{Penning} = 0.574$, Şahin et al. NIMA 768, 104-111, 2014)





Results are quite cross section dependent

Special features

Saving data

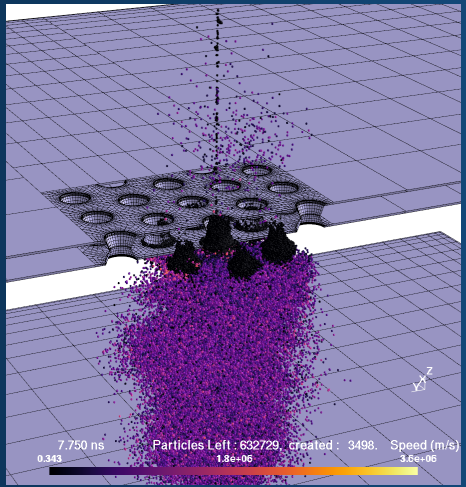
- Autosave time
- Escape time and restart from previous simulation

Speeding-up

- Multi-GPUs
- Space-charge effect: Nearest neighbours algorithm

Physics

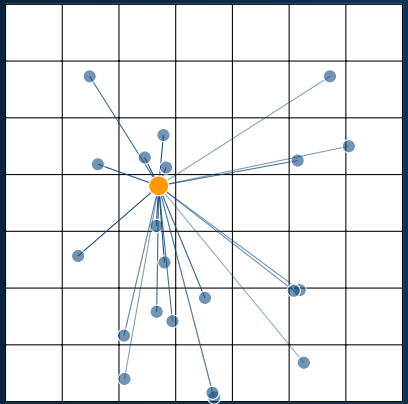
- Charging-up
- Pre-loading dielectric materials
- Microscopic ion transport
- Recombination

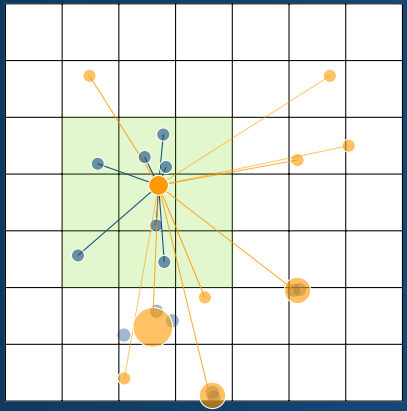


Space-charge effect: ‘Nearest neighbours’

Calculating the N-Body problem

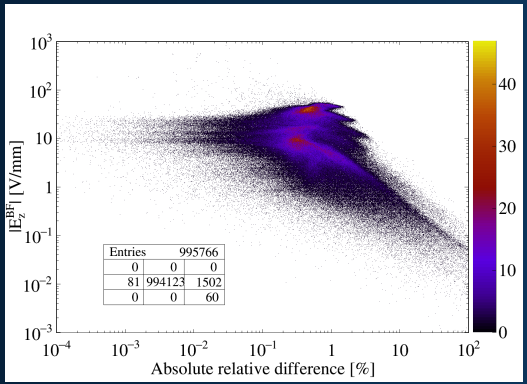
- Far field approximation: exact calculation only for the closest particles





Space-charge effect: ‘Nearest neighbours’

Relative differences compared to the exact calculation of the z component of the field

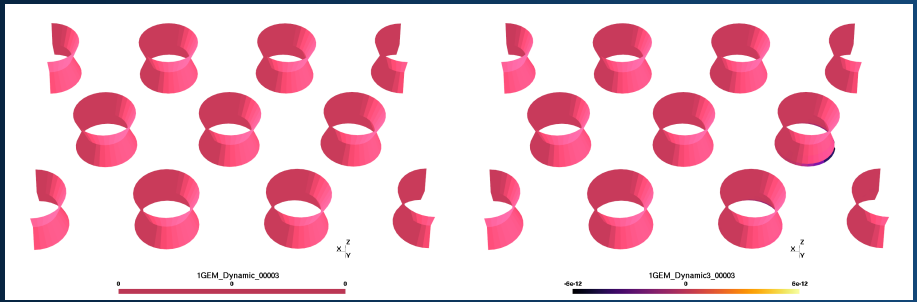


# particles	BF		NN	
	1 GPU	4 GPUs	1 GPU	4 GPUs
10^5	0.121 s	0.05 s	0.005 s	0.004 s
10^6	16 s	3.9 s	0.09 s	0.03 s
10^7	∼1600 s	400 s	1.1 s	0.35 s

Charging-up

Changing the dielectric cells surface charge densities $\sigma_{F,i}(t)$

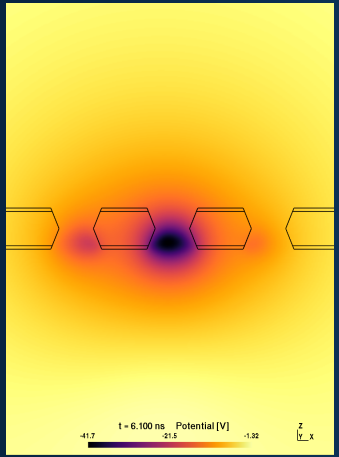
- $\sigma_{F,i}(t) = \sigma_{F,i}(t - \Delta t) + \frac{1}{A_i} \sum_{p=1}^{\mathcal{N}_{p,i}} q_p$
- Delayed by 1 time step for efficiency purposes
- No evacuation for the moment
- Effect is obviously small for $\mathcal{N}_{p,i} < 1 \times 10^{5-6}$
- But can be increased using "Macro" particles



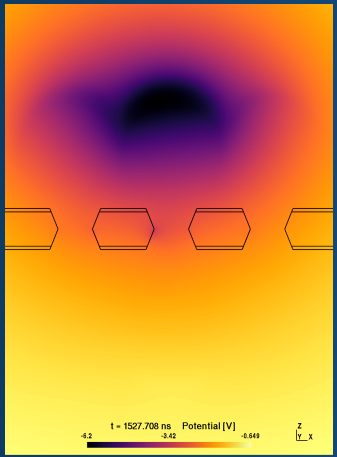
All Space-charge effects

Electric potential from particles: e.g. GEM detector (GEM potential subtracted)

- 6 ns: Electrons



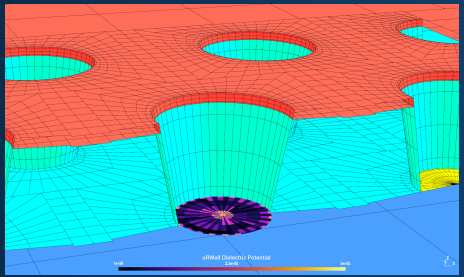
- 1.5 µs: ions



Pre-loading dielectric materials

Adding surface charges on the dielectric cells

- Set the amount of charge in Coulomb
- Creates a 3D gaussian distributed cloud of particles centered on a given position and width
- Thrown to the cells with a given direction
- Adds the surface charges to the cells



Towards full Monte Carlo calculations

Ion scattering cross sections

- Ar^+ + Ar data (Phelps calculations) only
- Isotropic and back-scattering
- Same treatment in the code for electrons and ions

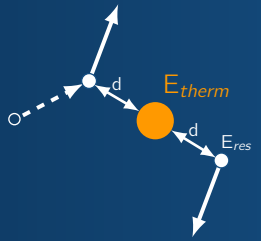
Dynamic microscopic calculations

- Knowledge of the distance between closest electrons and ions
- No difference between anions and cations (created through attachment) due to missing cross sections
- Possibility for recombination processes...

Recombination: Classical representation

1. Creating the electron/ion pair in the ionisation process

   - Thermal energy of the ion (3/2kT)
   - Electron residual energy $\sim$ Recoil energy
   - Both electrons have opposite momentum in the c.m.
   - Distance between the electron and ion from Coulomb barrier:
     $d = \frac{1}{4\pi\varepsilon_0}\frac{q}{E_{res}} \simeq 1.44$ eV·nm$/E_{res}$



Recombination: Classical representation

2. Inverse process for recombination
   - Distance d is known during the 'Nearest neighbours' process
   - If $d < \frac{1}{4\pi\varepsilon_0}\frac{q}{E_{ioni}}$, the particles recombine
   - But what about their kinetic energies?
     - Can be large due to Coulomb attraction
     - Energy is above ionisation potential...
   - If d is too small, might never recombine due to the time step (25 fs)...

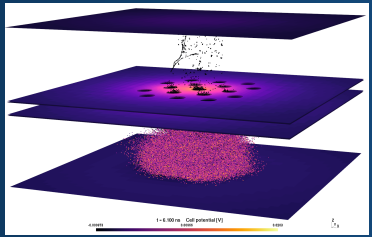
3. Another approach?
   - Statistical one: probability to recombine if certain conditions apply?
   - Semi-quantum one: which calculations and at what cost?

Still to be tested and validated with data (but which ones?)

Conclusion

OuroborosBEM: Garfield++ on GPU and more!

- Full MC simulation and dynamic physical effects
- Available on gitlab (`https://gitlab.in2p3.fr/ouroboros/ouroboros_bem.git`)
- Some geometries already included (1 GEM, 2 GEMS, µRWell, MicroMegas)
- Can include "resistive" materials by adding a life-time to the cell surface charges



A few attention points

- Mesh size (mesh construction): with the static option, cell orientation and geometry maps to be checked
- Grid size for the static option
- Time step size: at low pressure and small geometry entities