Developments for a Novel Moment-Based Particle-In-Cell Code

Finlay Gunneberg

f.gunneberg1@lancaster.ac.uk

Supervised by Jonathan Gratus

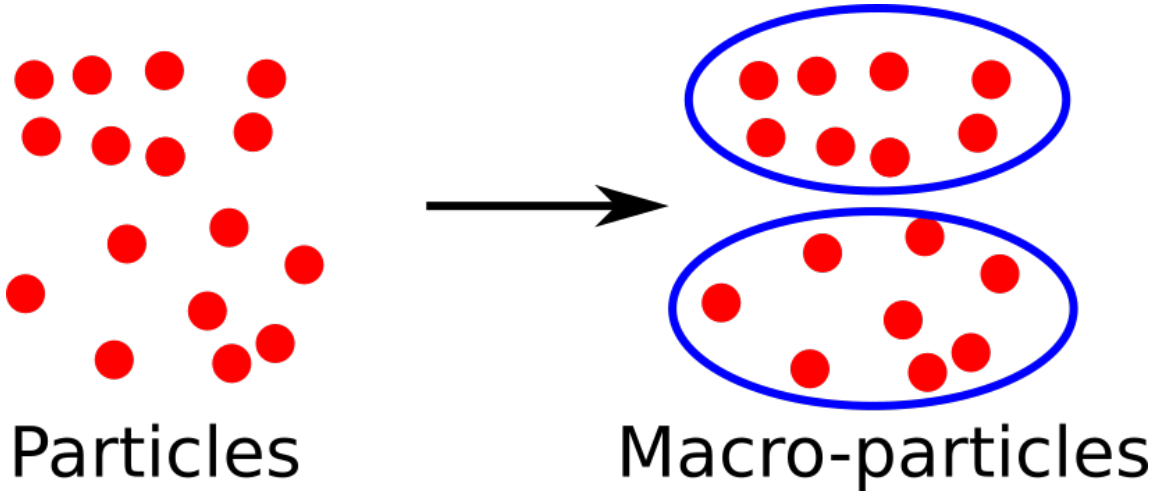
Lancaster University

Particle In-Cell codes

What is a Particle In-Cell (PIC) code?

PIC codes simulate charged matter e.g. beams in particle accelerators. They use a series of tricks to speed up computation:

- Simulate macro-particles instead of every particle.
- Grid the simulation space and solve the Maxwell equations on the grid.



PIC Schematic

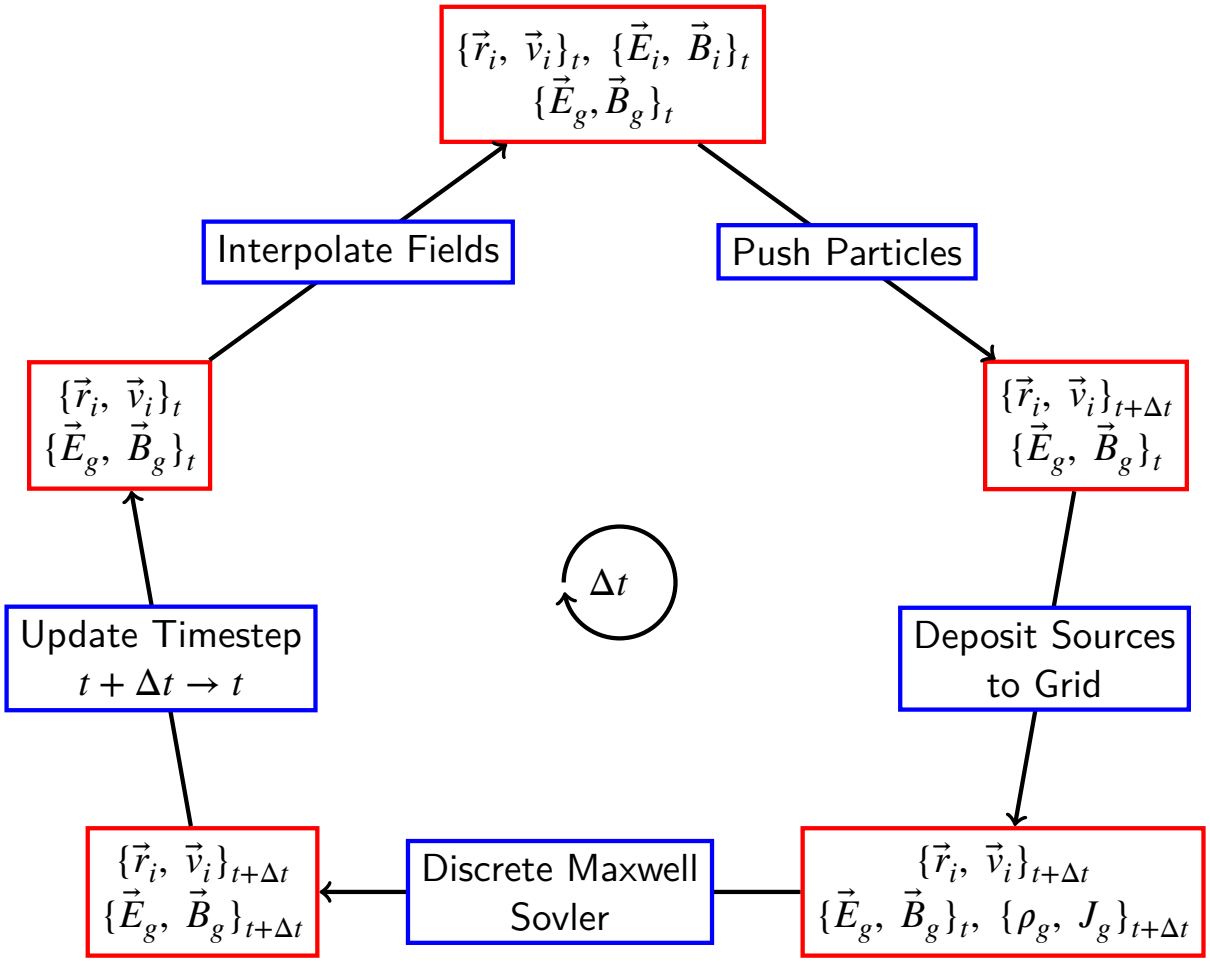
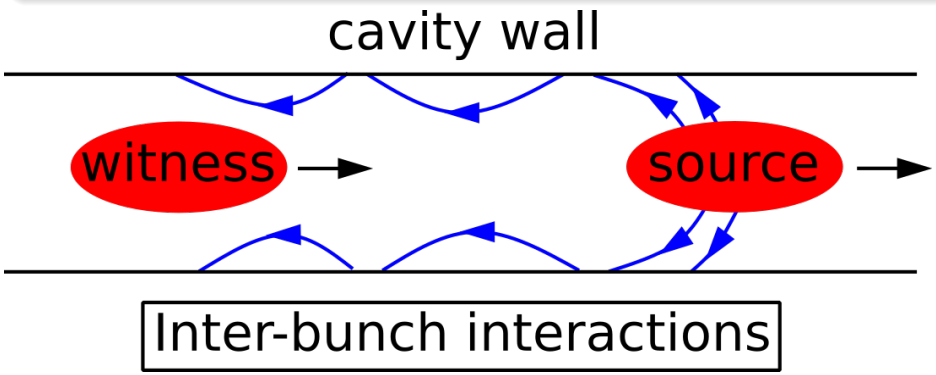


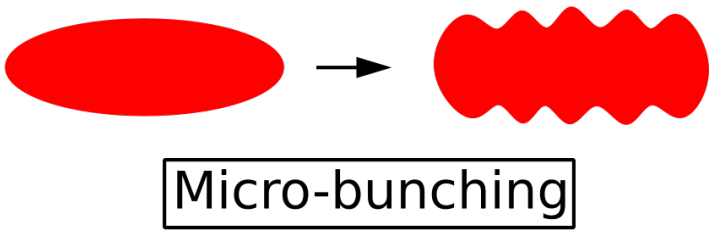
Figure: Schematic for a simple PIC code. Particle quantities e.g. $\vec{r}_i$ are denoted by an i subscript, while grid quantities e.g. $\vec{E}_g$ are denoted with a g subscript, where g denotes the index of the grid node.

Moment-Based PIC Codes

What's the big idea?

- The jump from particles to macro-particles loses simulation detail.
- Adding an underlying distribution and moment data to macro-particles allows their structure to change through time. We call these objects super-macro-particles (SMPs).
- There are instances in accelerator physics where the internal structure of bunches is important e.g. inter bunch interactions or micro-bunching.





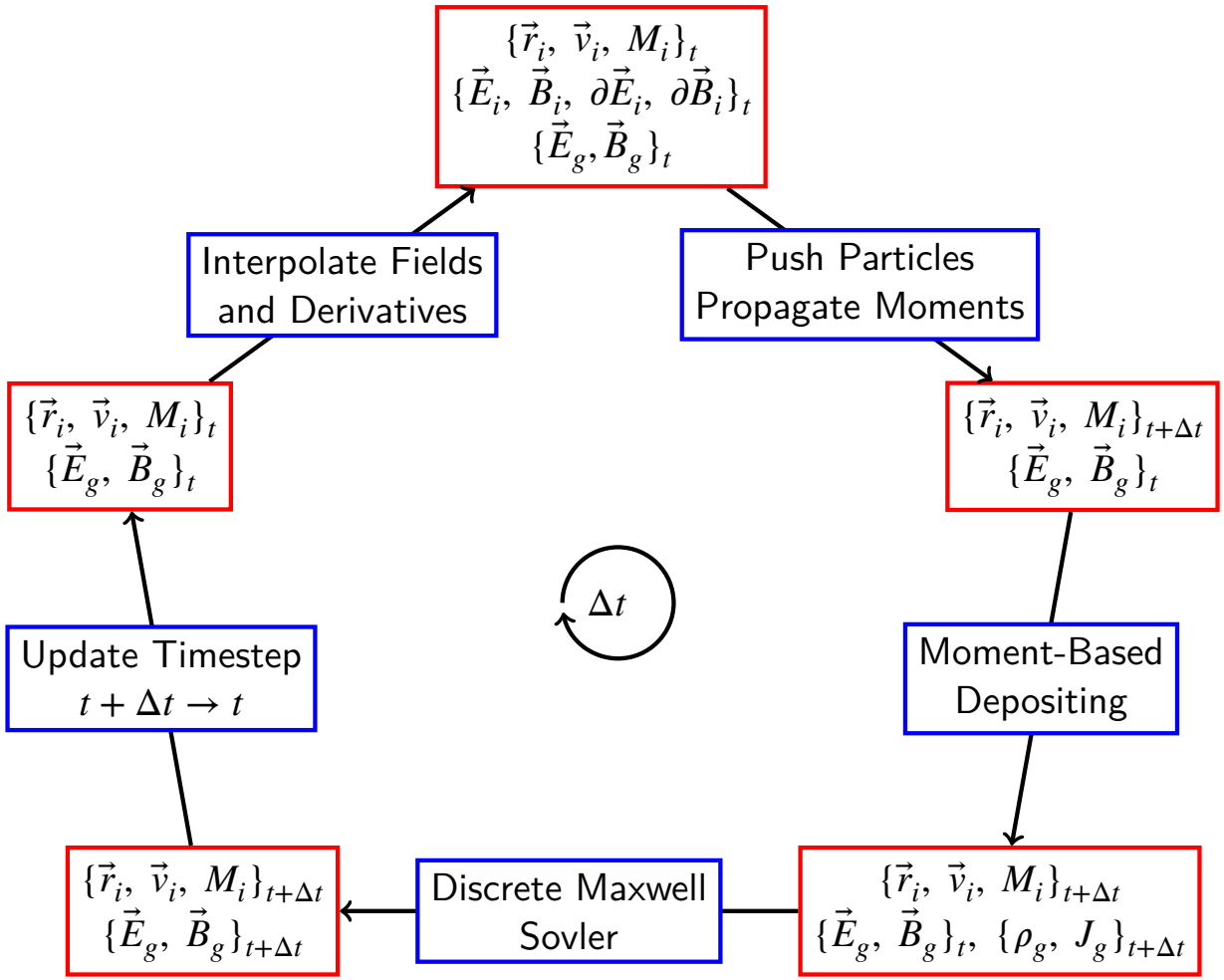


Figure: Schematic for the moment-based PIC code. Particle quantities are denoted by an i subscript, grid quantities are denoted by a g subscript.

Pushing Particles and Propagating Moments

How do we update SMPs?

- Position and velocity coordinates of SMPs are updated in the same way as regular macro-particles.
- The moments of each SMP are updated by propagating equations [1],

$$\left.\frac{dM^{a_1\ldots a_m}}{dt}\right|_{\xi_0} = \sum_{k=1}^{m}\sum_{\ell=1}^{\infty}\frac{1}{m!}M^{a_1\ldots a_{k-1}a_{k+1}\ldots a_m b_1\ldots b_\ell}\,\partial_{b_1}\cdots\partial_{b_\ell}W^{a_k}|_{\xi_0}, \quad (1)$$

where W^{a_k} is the corresponding coefficient of the Vlasov field,

$$W = \partial_t + \vec{v}\cdot\vec{\nabla}_r + \frac{q}{m}\left(\vec{E}+\vec{v}\times\vec{B}\right)\cdot\vec{\nabla}_v. \quad (2)$$

Reconstructing Distributions from Finite Sets of Moments

Why?

- In order to perform moment-based depositing, we need a method for estimating distributions from a finite set of moments.
- We introduce an algorithm which, given a choice of model functions $\varphi_{k_x,k_v}(x,v)$ and a set of moments $\{M_i\}$, we may deduce a set of structure coefficients c_{k_x,k_v} such that

$$\hat{f}(x,v) = \sum_{k_x+k_v \leq M} c_{k_x,k_v} \varphi_{k_x,k_v}(x,v). \tag{3}$$

- This algorithm is similar to that found in [2].

Reconstructing Distributions from Moments

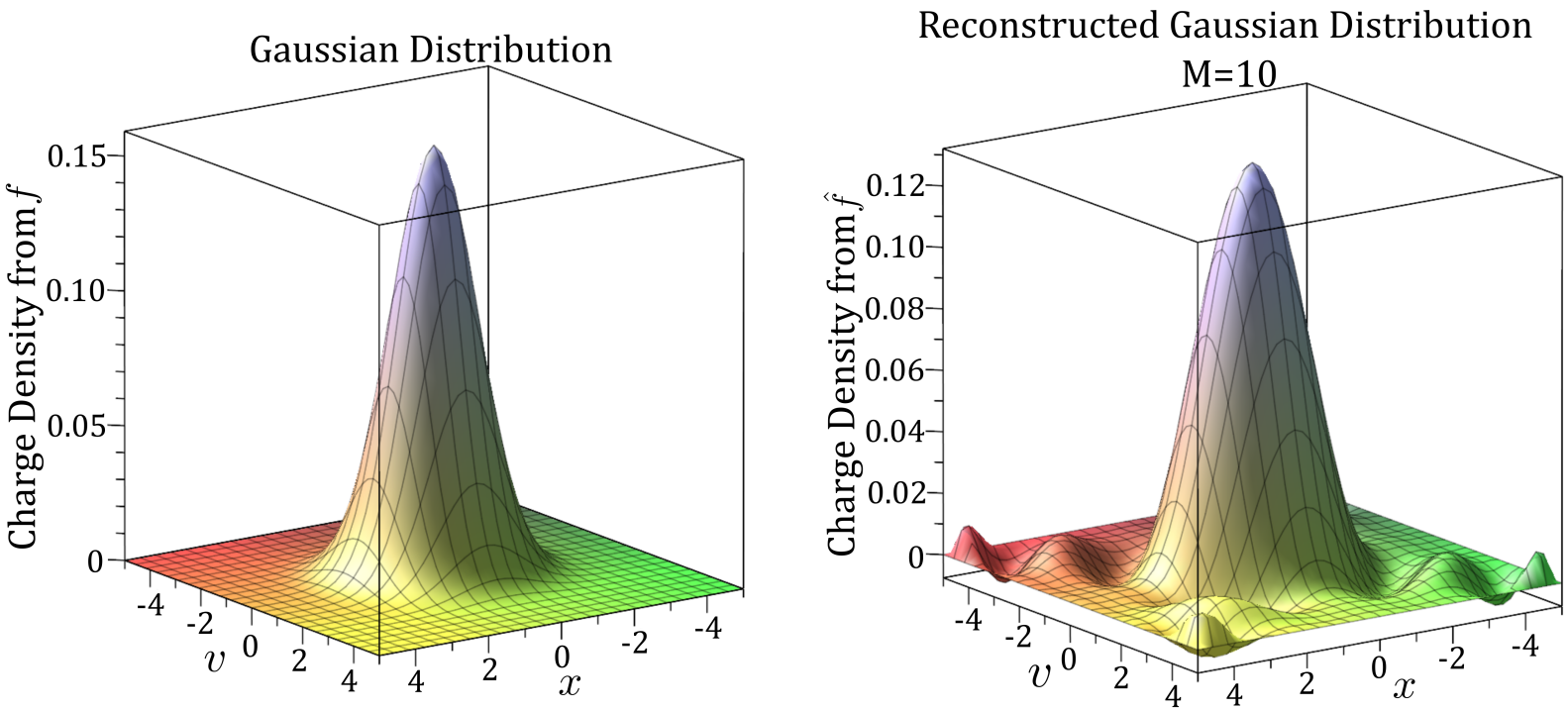


Figure: Left: plot of a Gaussian distribution over xv plane. Right: Reconstruction of Gaussian distribution using moments up to order $M = 10$ and a tophat model function.

Moment-Based Depositing

Weights for charge density.

- Moment-based depositing is performed in terms of weights. The fraction of total charge density due to the i-th SMP contributed to the j-th grid node is denoted $W_j^{(i)}$.
- Given the structure coefficients of the i-th SMP, $c_{k_x,k_v}^{(i)}$, the charge density weight for the j-th grid node is

$$W_j^{(i)} = \sum_{k_x+k_v \leq M} c_{k_x,k_v}^{(i)} \left[\int_{x_j-\Delta x/2}^{x_j+\Delta x/2} \int_{v^-}^{v^+} \varphi_{k_x,k_v}(x,v) dv dx \right]. \quad (4)$$

- Since the choice of model function remains constant, the quantity in the square brackets can be computed before the simulation is run.

Moment Based vs Standard Depositing

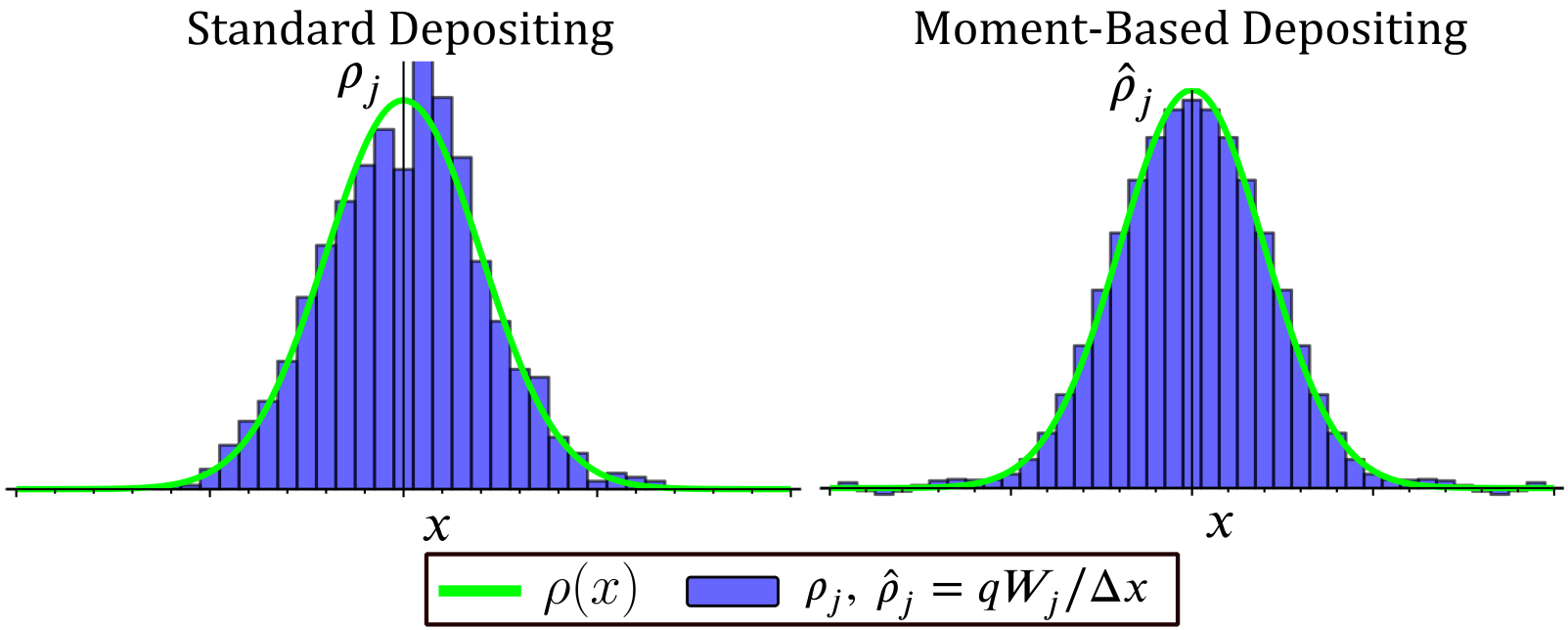


Figure: Left: Moment based depositing order $M = 10$ for a single SMP with Gaussian distribution. Right: Depositing sample ($N = 1000$) from Gaussian distribution using a 0-order shape function.

Moment-Based Interpolation

The final obstacle

- We have shown that in the unmagnetised, periodic, electrostatic regime that depositing and interpolating with eq. (4) conserves momentum.
- Interpolation when current density is present is less obvious: if the SMPs are not sufficiently homogenous in velocity space then the weights for depositing current density are different to the weights for depositing charge density:

$$\vec{W}_j^{(i)} = \sum_{k_x+k_v \leq M} c_{k_x,k_v}^{(i)} \left[\int_{x_j-\Delta x/2}^{x_j+\Delta x/2} \int_{v^-}^{v^+} \vec{v} \varphi_{k_x,k_v}(x,v) dv dx \right]. \quad (5)$$

- Hence, there is a choice to be made while interpolating.

Optimisations and Future Research

Current Research

- The sources for the Maxwell equations are expressed entirely in terms of quantities on the base space (i.e. $\vec{j} = \vec{j}(t, x)$), so it may be possible to write eq. (4) and eq. (5) in terms of positional moments only.
- This is ongoing research conducted by E. Williams.
- Research is underway to deduce how to simulate the beam of a klystron in a moment-based simulation, but our priority is a minimum-viable-model.

Illustration of Klystron Beam SMP

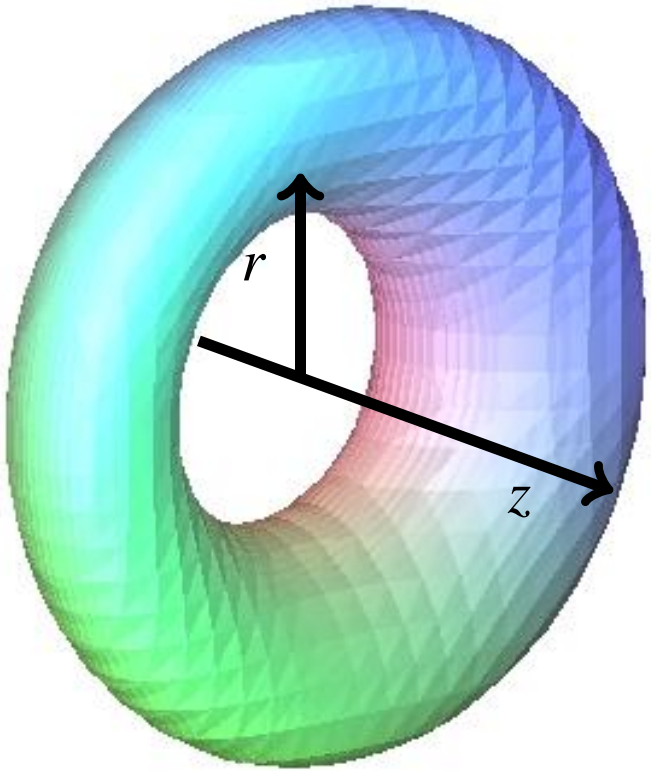


Figure: Schematic of possible SMP shape for a moment-based klystron code: an oblique ellipse in the r-z plane rotated about the z axis. By adding information about angular moments, a congruence of these particles may describe scalloping.

Thank you for your time!
Are there any questions?

References

[1] Moment tracking and their coordinate transformations for macroparticles with an application to plasmas around black hole, A. Warwick, J. Gratus, December 2023, Plasma Physics and Controlled Fusion, https://iopscience.iop.org/article/10.1088/1361-6587/ad11fc/meta.

[2] Reconstruction of a distribution from a finite number of moments with an adaptive spline-based algorithm, L.G.M de Souza, G. Janiga, V. John, D. Thevenin, May 2010, Chemical Engineering Science.