UK Atomic Energy Authority

ExCALIBUR

NEPTUNE: Options for Particle Algorithms

M2.3.1

Abstract

This report describes work for ExCALIBUR project NEPTUNE at Milestone 2.3.1: Investigation of algorithms to optimise usage of particle-based information at ExaScale, by for example reducing noise levels in the neutral-plasma fluid interaction. We also propose to study particle-mesh interaction effects with elements of different orders.

UKAEA REFERENCE AND APPROVAL SHEET

	Client Reference:	
	UKAEA Reference:	CD/EXCALIBUR-FMS/000*
	Issue:	1.00
	Date:	10 March 2020

Project Name: ExCALIBUR Fusion Modelling System.

Version 1.00

	Name and Department	Signature	Date
Prepared By:	Wayne Arter Lucian Anton Debasmita Samaddar	N/A N/A N/A	11/03/2020 11/03/2020 11/03/2020
Reviewed By:	Rob Akers	N/A	11/03/2020
Modified By:	Rob Akers MSSC		11/03/2020
Approved By:	Rob Akers MSSC		12/03/2020

Chapter 1

Introduction

The edge region is in parts a very good vacuum, so that the plasma ion species typically do not thermalize fully. Nonetheless it may be adequate for many purposes to treat the majority ionised species as fluids, although this is harder to argue for impurity species that may be present only in relatively small numbers. In fact, collision timescales typically vary $\tau \propto T^{3/2}/N$ where T is the temperature of the species and N is its number density, so that a species may be treatable as a fluid over say microsecond timescales in cooler, denser regions, but not on shorter timescales or in parts closer to the core. There is the additional complication of sheath formation at the edge, where the preferential loss of electrons leads to strong electric fields and consequently flows close to sonic, so that most workers model the sheath plasma using particles, typically with the Particle-in-Cell (PIC) approach, although for the Vlasov equation a wide range of numerical techniques has been examined, see eg. Palmroth et al [1, § 4]. *N.B. Discussion in this note implies usage of particles to model kinetic effects, not schemes such as SPH designed to model advection of classical fluids.*

The cooler parts of the tokamak edge plasma may contain large numbers of neutral atoms. Neutrals are generically less likely than charged species to thermalize. They circulate into hotter and denser regions and collide with charged species. Classical transport coefficients κ, in addition to a $1/\tau$ dependence, are anisotropic because of the strong magnetic field, to the extent that collisions with neutrals may become an important limiting mechanism for transport along the field. Operationally the most important aspect of the neutral species is however that because of the collisions, they represent a source of plasma.

The non-Maxwellian or kinetic aspects of the edge may lead to a need to solve the Boltzmann equation, in fact not only with quadratic source terms representing interaction between two species colliding, but with cubic terms representing chemical reactions. Classical fluid dynamics of course assumes Maxwellian dependence on phase velocity and concentrates on the first few moments density, mean flow and sometimes temperature as well as pressure. It can be helpful to introduce the concept of phase-fluid, where extra dimensions represent the velocity-space dependence at a position in full detail. In the phase-fluid approach, the concept of multiplicatively perturbing a Maxwellian has been explored. The main and important advantage of this approach is that it often facilitates massive simplification of the collision integrals in Boltzmann, to a single point term in significant cases, eg. ref [2].

Kormann & Yurova (private communication, 2019) have recently reviewed the use of the Hermite basis (ie. a Gaussian multiplied by a Hermite polynomial). Two significant works are Vencels et al. [3] describing Spectralplasmasolver and J.T.Parker's thesis [4] which describes software focussed on gyrokinetics, namely SPECTROGK. Note that all these works use a Fourier representation for real space, the so-called Fourier-Hermite method.

Even with the simplification of a multiply-periodic domain, there are nonetheless issues particularly at high velocity values and of course a Maxwellian may not be a particularly good approximation. Hence, borrowing from classical CFD practice on a infinite domain, it might be interesting to look at mappings which expand a set of compact spectral elements to cover the whole of velocity space.

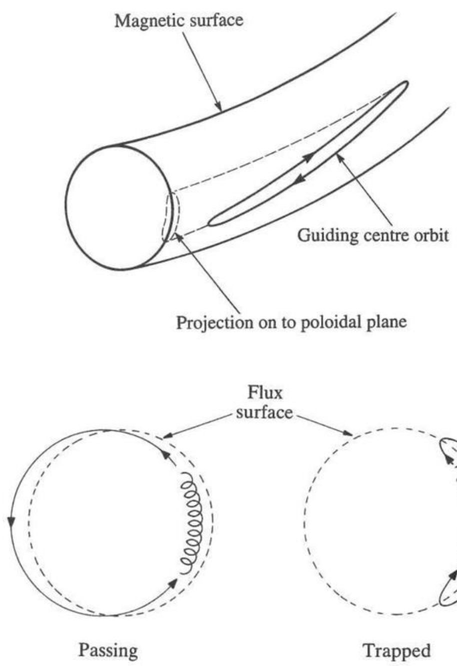

Figure 1.1: Important classes of particle trajectories from Wesson [5, § 3.10].

Alternatively it may be easier to return to basics and look at a particle representation for the ionised species as well as for the neutrals. This may be as efficient as using higher order elements since the distribution function in velocity may not be very smooth. As the sketch in Figure 1.1 indicates, there are two important classes of particle trajectories, depending on the particle energy and where in the magnetic field they begin. The first set approximately follow the fieldlines, gyrating as they go, whereas the second set may have trajectories that bounce. There is lack of smoothness at the trapped-passing boundary in velocity space.

Chapter 2

Task Work

2.1 Fluid and Particle Representations

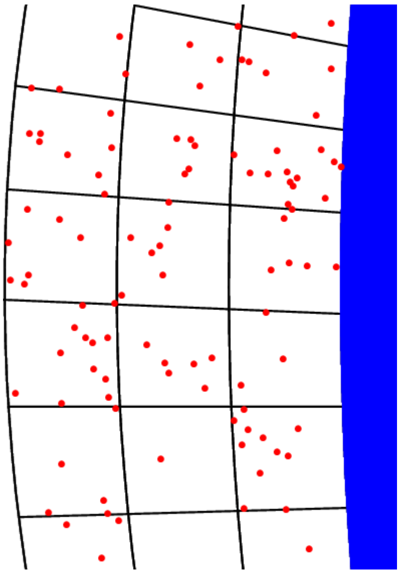

Figure 2.1: Wall at right, schematic finite element (fe) mesh shown as distorted grid. Plasma with fluid properties discretised using fe, interacts with neutral particle species shown as dots.

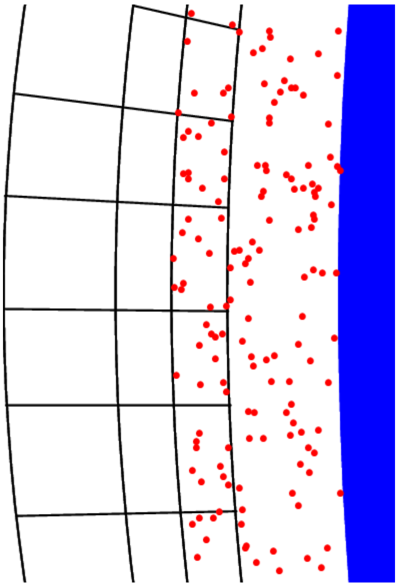

Figure 2.2: Handling a plasma sheath adjacent to wall at right. Plasma has fluid properties discretised using fe indicated using warped grid, but in sheath is better represented as particles indicated by dots.

As described in Section 1, there are two major distinct coupling cases. Figure 2.1 shows is a common case where plasma is represented on a mesh as a fluid whereas the neutrals are represented as particles. Figure 2.2 sketches a case where collisionality changes say because of sheath formation, so there is a need to change representation depending on position. Note that the mesh and particles are nonethelss shown as overlapping. This is believed to be good for numerical robustness in enabling a smoother transition between two very different velocity-space distributions. Both cases are found in engineering fluid dynamics, the first for example when treating combustion products, the second occurs in stratospheric re-entry of space vehicles, although typically the fluid region is adjacent to the solid surface and long mean-free-paths are found at distance. The overlap region in Figure 2.2 has attracted attention in the plasma context, leading to the particle-fluid model of Rönnmark and Hamrin and the fluid-kinetic PIC model both as described in ref [6]. The work of Taitano et al [7] appears to be more sophisticated, and subsequently has been generalised as the multiscale high-order/low-order (HOLO) algorithm which could also address issues around Model Order Reduction [8].

2.2 Advanced Sampling Techniques

The drawback of the particle approach PIC is potentially the low arithmetic intensity relative to accuracy because of the well-known scaling of error$\propto 1/\sqrt{N_p}$ for Monte-Carlo-type methods using N_p particles, which may make PIC very inefficient at Exascale. The control-variates idea of using particles to simulate only deviations from Maxwellian is already well-developed as the so-called δf method, see the review [9]. Present gyrokinetic codes are treating $N_p=10^{11}$ particles in this way, although these calculations appear to be encountering difficulties.

Of course, it will generally be good to reduce the number of particles required for a given error. As illustrated by Figure 2.3, the purely random points from Monte-Carlo sampling tend to cluster. If it is known that lengthscales below a certain critical value are unimportant, then Quasi-Monte-Carlo sampling as indicated by Figure 2.4 may do better, and here errors scale closer to $1/N_p$ than $\propto 1/\sqrt{N_p}$. These more sophisticated sampling techniques are relatively immature in application to neutrals, never mind plasma, but since sampling noise is often the main limiting factor, this simple argument shows that they can be transformative and so cannot be ignored. Ameres [10] has begun to examine this point both analytically and numerically, as earlier, assuming periodicity in physical space.

There are also the hierarchical approaches MLMC (Multi-Level Monte-Carlo) and MIMC (Multi-Level Monte-Carlo) that offer the promise of efficiency gains [11]. PIC is formulated as a set of stochastic differential equations known as a McKean-Vlasov process (meaning the coefficients of the equation depend on expected values of its solution). However the gains appear to be much less dramatic than for QMC [12].

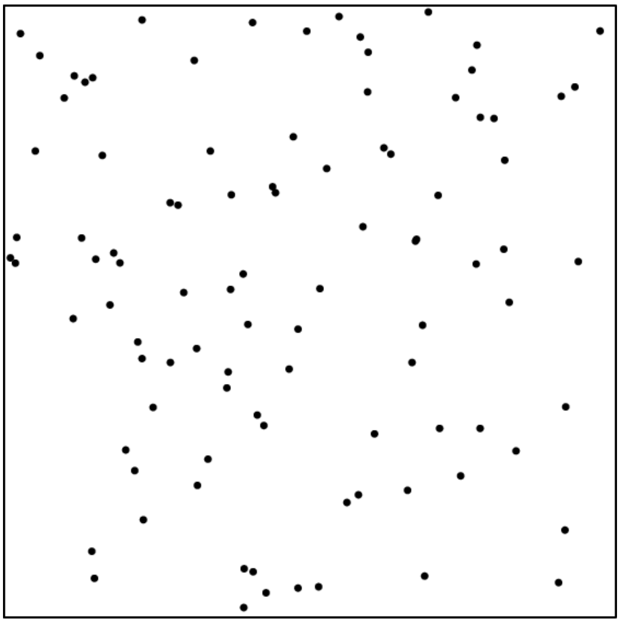

Figure 2.3: Monte-Carlo sampling of points in a square.

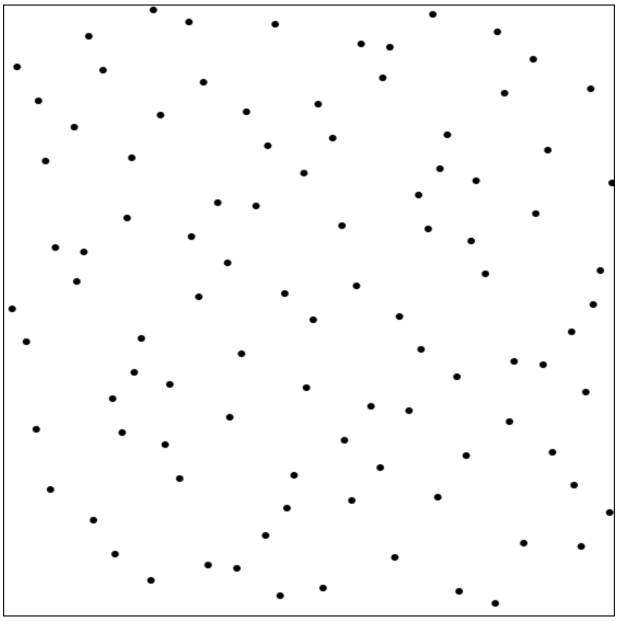

Figure 2.4: Quasi-Monte-Carlo sampling of points in a square.

2.3 Implicit Particle Methods

The HOLO schemes of Section 2.1 above, already require an implicit formulation. In the context of purely PIC schemes, widely differing particle masses (masses of Hydrogen ions and electrons in a ratio of nearly 2000) can also pose a challenge when it comes to coupling the species. Generally, given the possibility of widely different timescales, an implicit approach is indicated which is more natural when there is coupling to a fe model that is also implicit. However it could be important that the implicit particle method requires the solution of matrices with dimensions that scale only with the mesh-size$\propto N_D^p$ where $p_D \leq 3$ is the spatial dimension, not with particle number N_p. This has driven research by the groups of Lapenta [13] and Chacon [14] into approaches variously referred to as ‘particle enslavement’ or ‘kinetic enslavement’. It is significant that the efficiency of the approach depends on how the resulting matrix euquations are solved, with the choice of preconditioner’s being important.

2.4 Timestep-Robust Particle Tracking

For both ease of implementation and speed of execution, many researchers over many years have noted that it would be very desirable to have a ‘stiff’ particle-pusher, viz. an algorithm for integrating the equation of motion of a charged particle in the electromagnetic field, that could produce accurate trajectories regardless of timestep size. The canonical ‘Boris’ pusher is only satisfactory for timesteps significantly shorter than the gyro-period (the time taken for the particle to oscillate about the local field direction). However often, for example leading to the large area of gyrokinetics, details of the gyro-orbit are unimportant in the overall plasma simulation.

In the past two years alone, as a development of ‘Boris’, Hairer [15] has been encouraged to look at use of the ‘filter’ approach he has applied in other ODE integrators. Chacon and Ricketson [16] have examined the replacement of ‘Boris’ by the implicit mid-point rule, which is possible in the context of an implicit PIC code. Burby [17] has tackled the problem from the point-of-view of slow manifold theory.

2.5 Exascale-related Issues

At the SIAM PP20 meeting, it was pointed out that for Exascale, there are important issues which are common to a broad range of particle problems. Reeve in Session MS64 gave a talk highlighting the Co-design center for Particle Applications (CoPA) that addresses issues common to many fields of particle simulation (such as Molecular Dynamics and cosmology as well as PIC). It is worth noting that they see simulation packages as being built in a series of layers, as illustrated by for example the CabanaMD package, to maximise flexibility and the possibility for re-use. A significant point also is their focus on data structures such as array-of-structs (AoS) and structs-of-arrays (SoA), as well as array-of-structs-of-arrays (AoSoA). From the computational physicist’s point-of-view this focus on data structure elements such as arrays is unsatisfactory, in that the physics indicates that all the information in a region of space represented by say a few finite elements should be co-located, which means for NEPTUNE , data from fields and particles' being re-ordered in quite a different manner.

For particle problems, presenters, notably Pope (for Habib in MS64), did mention the important of 'tree'-based storage. Guo in MS22 discussed the effects of using different storage strategies, but found somewhat surprisingly changes in execution time of only approx. 10 %. However, only of order a million particles were present.

Chapter 3

Summary

Examination of the literature and conference attendance described here, plus the Birmingham meeting [18, 19] have indicated a gap in the knowledge of how to transition efficiently between a fluid representation and a particle representation and vice versa, particularly when spectral elements are used to represent the fluid.

Concerning particle methods, there are several potentially disruptive approaches that need evaluating before producing a detailed software design. These are

1. Use of Quasi-Monte-Carlo techniques and related sampling techniques
2. Particle or kinetic enslavement
3. Timestep-robust particle tracking

The aforementioned points require further research and literature analysis, for which indicative references have been provided in Section 2.

There are also practical issues concerning storage and use of a phase-fluid that overlap with other tasks. Although many older PIC codes successfully used 32-bit precision, the use of greatly reduced precision in representing particles is expected to increase 'noise' to unacceptable levels. Hence, for any usage of particles, particularly when fluid approximations are being employed for the main species, it is likely that the major memory cost will be storage of the particles. Hence the demands of particle storage will determine how *all* field data is assigned to memory.

Other issues relating to use of a phase-fluid approach seem worthy of examination. It seems that spectral/hp element might be efficiently used in velocity as well as physical space, and therefore evaluating performance here ought to form part of the general assessment of the performance of spectral elements.

Acknowledgement

The support of the UK Meteorological Office and Strategic Priorities Fund is acknowledged.

Bibliography

[1] M. Palmroth, U. Ganse, Y. Pfau-Kempf, M. Battarbee, L. Turc, T. Brito, M. Grandin, S. Hoilijoki, A. Sandroos, and S. von Alfthan. Vlasov methods in space physics and astrophysics. *Living reviews in computational astrophysics*, 4(1):1, 2018.

[2] V.M. Zhdanov. *Transport processes in multicomponent plasma*. CRC Press, 2002.

[3] J. Vencels, G.L. Delzanno, G. Manzini, S. Markidis, I.B. Peng, and V. Roytershteyn. Spectralplasmasolver: a spectral code for multiscale simulations of collisionless, magnetized plasmas. *Journal of Physics: Conference Series*, 719(1):012022, 2016.

[4] J.T. Parker. *Gyrokinetic simulations of fusion plasmas using a spectral velocity space representation*. PhD thesis, Oxford University, 2015.

[5] J.A. Wesson. *Tokamaks, 3rd Edition*. Clarendon Press, Oxford, 2003.

[6] S. Markidis, P. Henri, G. Lapenta, K. Rönnmark, M. Hamrin, Z. Meliani, and E. Laure. The fluid-kinetic particle-in-cell method for plasma simulations. *Journal of Computational Physics*, 271:415–429, 2014.

[7] W.T. Taitano, D.A. Knoll, L. Chacón, and G. Chen. Development of a Consistent and Stable Fully Implicit Moment Method for Vlasov–Ampère Particle in Cell (PIC) System. *SIAM Journal on Scientific Computing*, 35(5):S126–S149, 2013.

[8] L. Chacon, G. Chen, D.A. Knoll, C. Newman, H. Park, W. Taitano, J.A. Willert, and G. Womeldorff. Multiscale high-order/low-order (holo) algorithms and applications. *Journal of Computational Physics*, 330:21–45, 2017.

[9] W. Arter. Numerical simulation of magnetic fusion plasmas. *Reports on Progress in Physics*, 58:1–59, 1995. http://dx.doi.org/10.1088/0034-4885/58/1/001.

[10] J. Ameres. *Stochastic and Spectral Particle Methods for Plasma Physics*. PhD thesis, Technical University of Munich, 2018. https://mediatum.ub.tum.de/?id=1395335.

[11] A.-L. Haji-Ali and R. Tempone. Multilevel and multi-index monte carlo methods for the mckean–vlasov equation. *Statistics and Computing*, 28(4):923–935, 2018.

[12] L.F. Ricketson. A multilevel Monte Carlo method for a class of McKean-Vlasov processes. arXiv preprint arXiv:1508.02299, 2015.

[13] L. Siddi, E. Cazzola, and G. Lapenta. Comparison of preconditioning strategies in energy conserving implicit particle in cell methods. *Communications in Computational Physics*, 24(3):672–694, 2018.

[14] G. Chen, L. Chacon, and D.C. Barnes. An energy-and charge-conserving, implicit, electrostatic particle-in-cell algorithm. *Journal of Computational Physics*, 230(18):7018–7036, 2011.

[15] E. Hairer, C. Lubich, and B. Wang. A filtered Boris algorithm for charged-particle dynamics in a strong magnetic field. Technical report, arXiv preprint arXiv:1907.07452, 2019. https://arxiv.org/pdf/1907.07452.pdf.

[16] L.F. Ricketson and L. Chacon. An energy-conserving and asymptotic-preserving charged-particle orbit implicit time integrator for arbitrary electromagnetic fields. *arXiv preprint arXiv:1904.09478*, 2019.

[17] J.W. Burby. Guiding center dynamics as motion on a formal slow manifold in loop space. *Journal of Mathematical Physics*, 61(1):012703, 2020.

[18] D. Samaddar. Report on Y1 2019 Internal Workshop. Technical Report CD/EXCALIBUR-FMS/0018-1.00-M1.1.1a, UKAEA, 2020. Report on Y1 2020 External Workshop (REPORT1). Technical Report CD/EXCALIBUR-FMS/0010-1.00-M1.1.1b, UKAEA, 2020.

[19] W. Arter, L. Anton, and D. Samaddar. Year One Summary Report. Technical Report CD/EXCALIBUR-FMS/0011-1.00-M1.2.1, UKAEA, 2020.