2D implementation of Kinetic-diffusion Monte Carlo in Eiron

Oskar Lappi*[1], Emil Løvbak*[2], Thijs Steel[3], and Giovanni Samaey[3]

[1]Department of Computer Science, University of Helsinki, Helsinki, Finland
[2]Scientific Computing Center, Karlsruhe Institute of Technology, Karlsruhe, Germany
[3]Department of Computer Science, KU Leuven, Leuven, Belgium

Abstract

Particle-based kinetic Monte Carlo simulations of neutral particles is one of the major computational bottlenecks in tokamak scrape-off layer simulations. This computational cost comes from the need to resolve individual collision events in high-collisional regimes. However, in such regimes, one can approximate the high-collisional kinetic dynamics with computationally cheaper diffusion. Asymptotic-preserving schemes make use of this limit to perform simulations in these regimes, without a blow-up in computational cost as incurred by standard kinetic approaches. One such scheme is Kinetic-diffusion Monte Carlo. In this paper, we present a first extension of this scheme to the two-dimensional setting and its implementation in the Eiron particle code. We then demonstrate that this implementation produces a significant speedup over kinetic simulations in high-collisional cases.

1 Introduction

Particle-based, kinetic Monte Carlo simulation is a key component in modeling of neutral particles in the tokamak scrape-off layer [3]. The EIRENE code [18] is widely used to perform such simulations, as part of, e.g., the SOLPS-ITER software [22]; however, we note the DEGAS code [20] as a widely used alternative. A drawback of kinetic Monte Carlo simulation is that it requires the resolution of individual collision events at the particle level. Resolving all of these collisions can become computationally infeasible, especially in high-collisional regimes. Mathematically, it is known that one can approximate kinetic models by drift-diffusion type models in this regime, assuming a diffusive scaling [13], thus avoiding the need to resolve all collisions.

A variety of hybrid approaches exist that combine kinetic simulation with diffusive simulation [3], including approaches that combine Monte Carlo simulation with deterministic methods, e.g. [2, 8]. However, we make use of asymptotic-preserving Monte Carlo schemes (APMC) [15, 5], which work fully at the Monte Carlo level. Specifically, we make use of the KDMC method [15]. We note that these schemes are a special case of a broader field of asymptotic-preserving schemes, see [9] and references therein.

The general idea behind APMC schemes is that they simulate particle trajectories through time-stepping that alternates between kinetic and diffusive positional increments, weighted in such a way that they produce consistent simulations in both low-collisional (kinetic) regimes and high-collisional (diffusive) regimes. We note that these schemes, in general, do not provide any error guarantees in the intermediate regimes. Any errors in these regimes can however be removed by the combination of APMC with multilevel Monte Carlo techniques [6, 7], see [17, 12, 11]. In the cited works, the APMC approach is shown to significantly reduce the required computational effort for simulating simplified mathematical test problems.

Implementing APMC methods in EIRENE is a significant undertaking, due to the code's complexity and scope. Hence, we assess their feasibility in a simpler code, with limited physics, namely the Eiron code [10]. Eiron is a newly developed `C++` code, following modern software design principles. The code serves as a sandbox in which to do performance analysis and test new numerical and programming approaches at low development effort, before embarking on time intensive implementation in state-of-the-art production codes.

*Equal contribution

In this paper, we present the first two-dimensional implementation of KDMC. We have performed this implementation in Eiron, allowing us to make use of it's efficient, modular implementation to perform HPC-scale simulations. We present the technical challenges of implementing this mathematical algorithm in an efficient and robust way and perform an initial analysis of the resulting simulation accuracy and computational speedup. The remainder of this paper is structured as follows. In Section 2, we provide an introduction to and algorithmic description of the KDMC method. Next, in Section 3, we discuss relevant practical aspects of the Eiron code and the details of the KDMC implementation. In Section 4, we then present some simulation results and compare the convergence behavior of the Eiron implementation to the original 1D results in [15]. We also conduct a performance analysis demonstrating the computational advantage of KDMC in high-collisional simulations. Finally, in Section 5, we draw conclusions from the presented work and comment on future work.

2 Monte Carlo for neutral particle models

We consider the simulation of neutral particles, based on a Boltzmann-BGK [1] equation

$$\partial_t f(x,v,t) + v \cdot \nabla_x f(x,v,t) = R_{\text{cx}} \left(\rho(x,t) \mathcal{M}(v) - f(x,v,t) \right). \tag{1}$$

Here, $f(x,v,t)$ represents the density of particles in a phase space over position x and velocity v, as a function of time t. On the right-hand side of (1), we model charge exchange collisions with a homogeneous event rate R_{cx}. In the BGK collision operator, we make use of $\rho(x,t)$ the particle density (integrating $f(x,v,t)$ over velocity space) and $\mathcal{M}(v)$ a Maxwellian distribution, with a given temperature T and mean drift velocity u. Although we constrain ourselves to homogeneous test-cases, both for the sake of exposition and for performing the initial experiments presented in this work, we note that neither Eiron, nor the KDMC approach are inherently constrained to such cases.

2.1 Kinetic Monte Carlo

Neglecting domain boundary interactions, we can perform a kinetic Monte Carlo simulation of (1) by simulating random particle trajectories with collision events samples according to the Maxwellian $\mathcal{M}(v)$. We present an algorithmic description of such a trajectory simulation in Algorithm 1. Note here that $\mathcal{E}(R_{\text{cx}})$ denotes an exponential distribution with rate R_{cx}. Due to resolving all collision events, kinetic Monte Carlo becomes prohibitively expensive in highly collisional regimes.

Algorithm 1 Simulating a particle with kinetic Monte Carlo until time T.

1: $t \leftarrow 0, \quad n \leftarrow 0, \quad \left\{X_p^0, V_p^0\right\} \leftarrow$ sample_source()
2: **while** $t < T$ **do**
3: $\quad \tau \sim \mathcal{E}\left(R_{\text{cx}}\right)$
4: $\quad X_p^{n+1} \leftarrow X_p^n + \tau V_p^n$
5: $\quad V_p^{n+1} \sim \mathcal{M}(v)$
6: $\quad n \leftarrow n+1, \quad t \leftarrow t + \tau$
7: **end while**

2.2 Kinetic-diffusion Monte Carlo

The kinetic-diffusion Monte Carlo method (KDMC) approximates highly-collisional kinetic Monte Carlo simulations by introducing diffusion at the particle level. The algorithm alternates between kinetic and diffusive updates to a given particle's position. The general idea behind this approach is that, in the infinite collisional limit, one replaces an infinite number of kinetic positional updates (Algorithm 1, line 4) with a single normally distributed increment, through the application of the law of large numbers. We refer to [15] for a detailed description of the algorithm.

We generalize the derivations in [14] to the two-dimensional setting, to observe that a correct implementation requires computing diffusive increments ΔW_p^n that are sampled from a multivariate normal distribution $\mathcal{N}(\mu, \Sigma)$, with mean

$$\mu = u\theta + \frac{V_p^{n+1} - u}{R_{\text{cx}}} \left(1 - e^{-\theta R_{\text{cx}}}\right),$$

and covariance matrix

$$\Sigma = \frac{2T}{R_{\text{cx}}^2\theta}\left(2e^{-\theta R_{\text{cx}}} + \theta R_{\text{cx}}\left(1 + e^{-\theta R_{\text{cx}}}\right) - 2\right) I + \left(1 - 2\theta R_{\text{cx}} e^{-\theta R_{\text{cx}}} - e^{-2\theta R_{\text{cx}}}\right)\left(\frac{V_p^{n+1} - u}{R_{\text{cx}}}\right)^{\otimes 2},$$

where the exponent $\otimes 2$ is used to denote the outer product of a vector with itself. Here, V_p^{n+1} is the velocity of the subsequent kinetic increment, and θ is a diffusive flight time. Algorithm 2 shows a KDMC simulation of a single trajectory until time T.

Algorithm 2 Simulating a particle with KDMC until time T.

1: $t \leftarrow 0, \quad n \leftarrow 0, \quad \left\{X_p^0, V_p^0\right\} \leftarrow$ sample_source()
2: **while** $t < T$ **do**
3: $\quad \tau \sim \mathcal{E}\left(R_{\text{cx}}\right)$
4: $\quad X_p^{n,\prime} \leftarrow X_p^n + \tau V_p^n$
5: $\quad V_p^{n+1} \sim \mathcal{M}(v)$
6: $\quad \theta \leftarrow \Delta t - (\tau \bmod \Delta t)$
7: $\quad \Delta W_p^n \sim \mathcal{N}(\mu, \Sigma)$
8: $\quad X_p^{n+1} \leftarrow X_p^{n,\prime} + \Delta W$
9: $\quad n \leftarrow n + 1, \quad t \leftarrow t + \Delta t$
10: **end while**

We refer to [15] for further details and a full analysis of the scheme, but make the following observations in terms of the charge-exchange rate R_{cx} in relation to the KDMC discretization time step size Δt. In low-collisional regimes, i.e., $R_{\text{cx}} \ll \Delta t$, observe that, in general, $\tau \gg \theta$. Hence, the kinetic component (line 4) of Algorithm 2 will dominate and the results produced by Algorithm 2 will converge to those produced by Algorithm 1. Inversely, in high-collisional regimes, i.e., $R_{\text{cx}} \gg \Delta t$, we have that, in general, $\tau \ll \theta$. Hence, the diffusive component (line 8) of Algorithm 2 will dominate. An intuitive explanation for the introduction of the diffusive step in line 8 is that one substitutes the sum of an infinite number of i.i.d. steps of the form given in line 4 by a single normally distributed increment, making use of the law of large numbers.

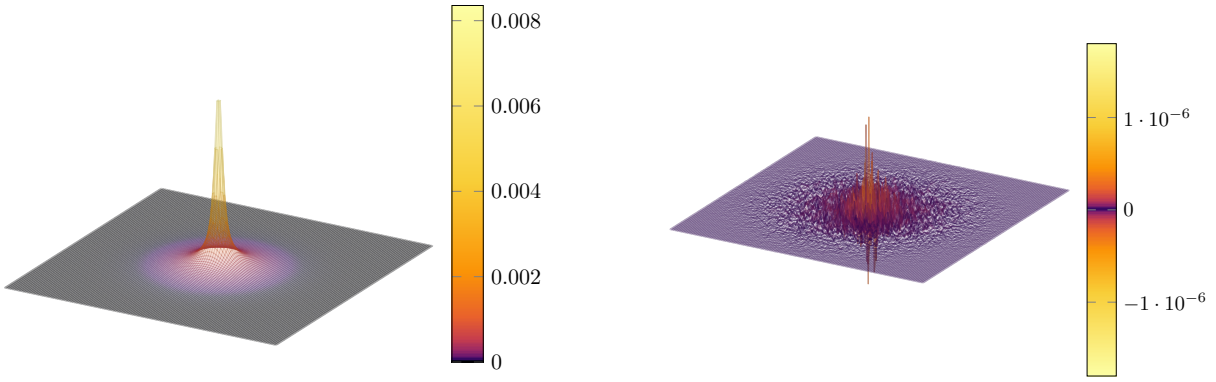
3 Implementation in Eiron

Eiron is a reduced 2D kinetic Monte Carlo simulation code, created to study the performance of the different Monte Carlo parallelization strategies that could be used in EIRENE. We have implemented the KDMC simulation algorithm (Algorithm 2) in Eiron, reusing existing functionality for kinetic simulations. Hence, only the diffusive step required implementation from scratch. We are also able to use the existing parallel algorithms in Eiron to run KDMC simulations, as these are decoupled from the simulation and estimation code.

Eiron supports a variety of shared and distributed-memory parallelism. Ideally, all parallelization strategies would be available to KDMC; unfortunately KDMC breaks some assumptions in Eiron's domain-decomposition algorithm, hence constraining us to a single domain. Unlike kinetic transport, a diffusive transport step is not just a continuous integral that can easily be broken at a subdomain boundary and continued in a neighboring subdomain boundary. When simulating diffusive steps for heterogeneous backgrounds, we approximately evaluate R_{cx} at the midpoint of the step using an estimated endpoint computed with μ evaluated at X_p^n and $\Sigma = 0$. If this estimated endpoint lies in a separate subdomain, this procedure introduces additional communication, not currently supported by Eiron. In future work, we plan to resolve these technical issues and implement domain decomposition for KDMC.

4 Simulation results

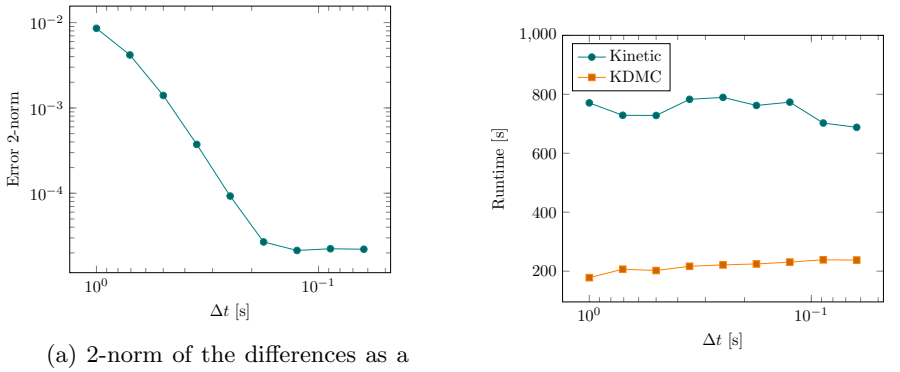
We now present the performance of our implementation, both in terms of accuracy and computational efficiency. In Section 4.1, we consider the low-collisional, kinetic test case ($R_{\text{cx}} \ll \Delta t^{-1}$). Here we show that KDMC and kinetic simulations converge to the same results as $\Delta t \to 0$, with comparable computational cost. In Section 4.2, we consider a high-collisional, diffusive test case ($R_{\text{cx}} \gg \Delta t^{-1}$). Here we show that KDMC and kinetic simulations also converge as $R_{\text{cx}} \to \infty$, with the KDMC simulations



(a) KDMC result

(b) Difference with kinetic simulation

Figure 1: A two-dimensional histogram of a KDMC simulation in the kinetic regime with $\Delta t = 2^{-4}$ s, $R_{\text{cx}} = 0.78125$ s^{-1} and a mean post-collisional speed of 0.013847 ms^{-1}. To compute the pointwise difference we subtract the KDMC result from the kinetic result.



(a) 2-norm of the differences as a function of Δt.

(b) Runtime as a function of Δt.

Figure 2: Behavior of KDMC, compared to kinetic simulation, in the kinetic regime. To reduce variance, we average the 2D results in the x-direction and fold the the results around the origin.

significantly outperforming the kinetic simulations in terms of computational cost. In general, one expects the largest errors to occur when Δt and R_{cx} are of the same order of magnitude.

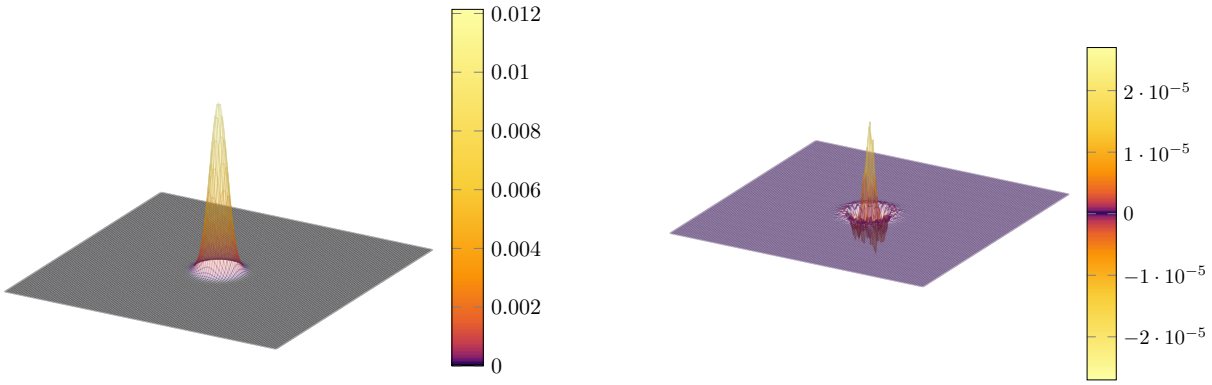
Throughout this Section, our test-case is a two-dimensional slab with ionizing boundaries. In what follows, we normalize all simulation parameters, assuming that the domain has dimensions 1 m $\times$ 1 m. We initialize all particles using an isotropic point-source at the center of the domain with velocities sampled from a 2D-Maxwellian. We then simulate the particles to a given end time t_{end}, at which point we construct a histogram of their final location in the domain using a 128 $\times$ 128-grid. We consider homogeneous collision backgrounds, with a fixed charge-exchange rate R_{cx}. Post-collisional velocities are sampled from an unbiased isotropic Maxwellian. We elaborate on precise numerical parameter values when discussing individual test-cases.

The simulations discussed in this section were run on CSC's Mahti supercomputer [4] using 128 OpenMP threads all running on separate CPU cores.

4.1 Kinetic regime

When refining the time step parameter Δt, KDMC should converge to a kinetic simulation. In the 1D setting, it has been shown that both particle distributions converges $\mathcal{O}(\Delta t^{1.5})$ in Wasserstein-1 distance for $\Delta t \to 0$ [15]. However, we note that this theoretical bound cannot be applied directly to our setting. To verify this limit we set a source mean speed of 0.15625 ms^{-1}, the charge-exchange rate is $R_{\text{cx}} = 0.78125$ and the post-collisional velocities have mean speed 0.013847 ms^{-1}. We then run both kinetic and KDMC simulations with 10^{10} particles until $t_{\text{end}} = 1$ s for a sequence of time step values Δt in the range $[2^{-4}\text{ s}, 1\text{ s}]$.

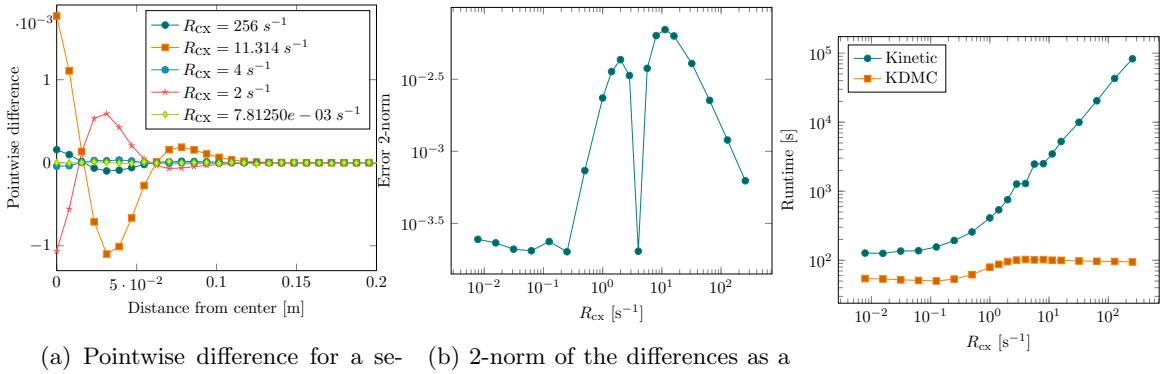
To visualize the results, we show a KDMC simulation for $\Delta t = 2^{-4}$ s in Figure 1, together with a cell-by-cell difference with the reference kinetic simulation. As the two simulations are visually indis-



(a) KDMC result

(b) Difference with kinetic simulation

Figure 3: A two-dimensional histogram of a KDMC simulation in the diffusive regime with $\Delta t = 1$ s, $R_{\text{cx}} = 256$ s^{-1} and a mean post-collisional speed of 0.19817 ms^{-1}. To compute the pointwise difference we subtract the KDMC result from the kinetic result.



(a) Pointwise difference for a selection of values of ε

(b) 2-norm of the differences as a function of R_{cx}

(c) Runtime as a function of R_{cx}

Figure 4: Behavior of KDMC, compared to kinetic simulation, in the diffusive regime. To reduce variance, we average the 2D results in the x-direction and fold the the results around the origin.

tinguishable, we opt not to show the reference separately. Comparing the computed solution with the difference, we see that we observe that we achieve between three and four digits of accuracy in the region of interest at the center of the domain.

In Figure 2, we consider the convergence of the KDMC simulation. To reduce variance, we flatten the solution to one dimension, by averaging over the x-dimension. We then mirror the solution around the source at the middle of the domain. In Figure 2a, we plot the 2-norm of the differences of these curves. We observe that this two norm decreases with roughly an order 3 until it hits a plateau caused by the sampling error due to the finite number of trajectories. This indicates a convergence that is roughly one and a half orders better than that predicted by the theoretical results in [15]. Next to the fact that there is no direct link between the Wasserstein metric on the particles' location and the 2-norm on the histogram, it is likely that the symmetry of the problem is a cause for this increased rate of convergence. In Figure 2b, we show the corresponding run times and observe that the KDMC and kinetic simulations incur a comparable computational cost, with KDMC slightly outperforming the kinetic simulation.

4.2 Diffusive regime

When increasing the collision rate, the modeling error incurred by approximating kinetic increments by diffusion in the KDMC scheme should vanish. In [15], it was show that this convergence should occur as $\mathcal{O}(R_{\text{cx}}^{-1.5})$, in terms of Wasserstein-1 distance of the particle distribution, as $R_{\text{cx}} \to \infty$. To verify the limit we perform simulations with a similar particle source with mean speed 0.0625 ms^{-1}. We fix $\Delta t = 1$ s and consider combinations the mean post-collisional speed $\frac{1}{\varepsilon}\sqrt{\frac{\pi}{10}}\frac{1}{512}$ ms^{-1} and collision rate $R_{\text{cx}} = \frac{1}{\varepsilon^2}\frac{1}{128}$ s^{-1} with ε taking values in the range $[2^{-7.5}, 1]$. In this case we use $2 \cdot 10^9$ particles until time $t_{\text{end}} = 4$ s. We then generate the similar figures to those in Section 4.1.

In Figure 3, we show the KDMC simulation and pointwise error for the case $R_{\text{cx}} = 256\ \text{s}^{-1}$ and the mean post-collisional speed $0.19817\ \text{ms}^{-1}$. Note that the shape of the solution differs from that for the kinetic case, in that it resembles a bell-curve, as opposed to the exponential decay from the center source, observed in Figure 1. We consider the convergence behavior in Figure 4. On the right-hand side of Figure 4b, we empirically observe a convergence rate of with approximate order 1 as R_{cx} increases. Notably, this convergence is slower than the known bound in the Wasserstein-1 distance. However, in parallel work [21] a similar rate was shown in the L_2-norm, when computing time-integrated quantities using KDMC, combined with the diffusive estimator presented in [16].

On the right-hand side of the figure, we see a similar phenomenon to that shown in Figure 2a, i.e., as R_{cx} decreases, the both simulations converge to kinetic simulations until a plateau is achieved due to the sampling error. What is initially surprising, is that we observe a sharp dip approximately $R_{\text{cx}} = 4\ \text{s}^{-1}$. To explain this phenomenon, we plot the pointwise differences of the averaged and folded simulation results for a selection of values of R_{cx} in Figure 4a. In this figure, we see that KDMC underestimates the particles' displacement in kinetic regimes and overestimates the displacement in diffusive regimes (compare the curves for $R_{\text{cx}} = 2\ \text{s}^{-1}$ [peak error on the kinetic side of Figure 4b] and $R_{\text{cx}} = 11.314\ \text{s}^{-1}$ [peak error on the diffusive side of Figure 4b]). Hence, during the transition between the two regimes, there is a value of R_{cx} for which these two errors cancel, causing the sharp dip in error.

In Figure 4c, we show that KDMC achieves a significant speedup in the diffusive regime, growing proportionally to R_{cx}. Here it is clear that the runtime of the kinetic simulations directly scale with the number of collision events that need to be resolved, while that for the KDMC simulations stays more-or-less constant. In the highest-collisional cases, this results in a speedup of between two and three orders of magnitude, showing the strength of KDMC over standard kinetic simulation in this regime.

5 Conclusions

We have introduced the first two-dimensional implementation of the KDMC scheme for the Boltzman-BGK equation in the Eiron code. Using this implementation, we confirmed that the results produced by KDMC converge to those produced by a standard kinetic simulation in both the kinetic limit ($\Delta t \to 0$) and diffusive limit ($R_{\text{cx}} \to \infty$). Our measured 2-norm convergence, does not match the rates given in [15] for the Wasserstein-1 distance. However, we believe the improved rate in Δt to be due to the symmetry in the considered test-case and our rate in the limit $R_{\text{cx}} \to 0$ matches the L_2-norm results shown in parallel work on analyzing time-integrated simulations with KDMC. In the kinetic regime, KDMC simulations attain similar run times to the equivalent kinetic simulations. However, in diffusive regimes, they outperform the kinetic reference simulations. In some cases, by multiple orders of magnitude.

This implementation in Eiron, is a first step towards the use of KDMC for large scale fusion simulations. In future work, we plan to extend the implementation to make full use of the domain-decomposition available in Eiron. We also plan to tackle more generic classes of boundary conditions such as reflective boundaries, in addition to improving the existing (biased) implementation of absorbing boundaries. One strategy for this could be the work presented in [19]. We also plan to run more extensive experiments for non-homogeneous test-cases. Further down the road, we plan to support time-integrated simulations by implementing suitable estimators for the diffusive increments, e.g., the scheme presented in [16].

Author contributions

Oskar Lappi/Emil Løvbak: Methodology, Software, Validation, Investigation, Writing - Original draft, Visualization
Thijs Steel: Methodology, Validation, Investigation, Writing - Review & Editing
Giovanni Samaey: Conceptualization, Supervision, Writing - Review & Editing, Funding acquisition

Acknowledgments

The authors wish to acknowledge CSC – IT Center for Science, Finland, for computational resources. Emil Løvbak was funded by the Deutsche Forschungsgemeinschaft (DFG, German Research Foundation) – Project-ID 563450842. This work has been carried out within the framework of the EUROfusion Consortium, funded by the European Union via the Euratom Research and Training Programme (Grant Agreement No 101052200 — EUROfusion). Views and opinions expressed are however those of the

author(s) only and do not necessarily reflect those of the European Union or the European Commission. Neither the European Union nor the European Commission can be held responsible for them.

Data availability

A copy of the data used to generate all figures, together with documentation on its generation can be found at https://doi.org/10.23729/fd-08ac1189-361d-3350-a66f-99e9db473b30.

References

[1] Prabhu Lal Bhatnagar, Eugene Paul Gross, and Max Krook. A Model for Collision Processes in Gases. I. Small Amplitude Processes in Charged and Neutral One-Component Systems. *Physical Review*, 94(3):511–525, May 1954.

[2] Maarten Blommaert, Niels Horsten, Petra Börner, and Wouter Dekeyser. A spatially hybrid fluid-kinetic neutral model for SOLPS-ITER plasma edge simulations. *Nuclear Materials and Energy*, 19:28–33, May 2019.

[3] Dmitriy Borodin, Friedrich Schluck, Sven Wiesen, Derek Harting, Petra Börner, Sebastijan Brezinsek, Wouter Dekeyser, Stefano Carli, Maarten Blommaert, Wim Van Uytven, Martine Baelmans, Bert Mortier, Giovanni Samaey, Yannick Marandet, Paul Genesio, Hugo Bufferand, Egbert Westerhof, Jorge Gonzalez, Mathias Groth, Andreas Holm, Niels Horsten, and Huw J. Leggate. Fluid, kinetic and hybrid approaches for neutral and trace ion edge transport modelling in fusion devices. *Nuclear Fusion*, 62(8):086051, July 2022.

[4] CSC. *Technical Details about Mahti*, 2025. https://docs.csc.fi/computing/systems-mahti.

[5] Giacomo Dimarco, Lorenzo Pareschi, and Giovanni Samaey. Asymptotic-Preserving Monte Carlo Methods for Transport Equations in the Diffusive Limit. *SIAM Journal on Scientific Computing*, 40(1):A504–A528, January 2018.

[6] Michael B. Giles. Multilevel Monte Carlo Path Simulation. *Operations Research*, 56(3):607–617, June 2008.

[7] Stefan Heinrich. Multilevel Monte Carlo Methods. In Svetozar Margenov, Jerzy Waśniewski, and Plamen Yalamov, editors, *Large-Scale Scientific Computing*, volume 2179 of *Lecture Notes in Scientific Computing*, pages 58–67. Springer Berlin Heidelberg, Berlin, Germany, 2001.

[8] Niels Horsten, Wouter Dekeyser, Maarten Blommaert, Giovanni Samaey, and Martine Baelmans. A hybrid fluid-kinetic neutral model based on a micro-macro decomposition in the SOLPS-ITER plasma edge code suite. *Contributions to Plasma Physics*, 60(5-6):e201900132, June 2020.

[9] Shi Jin. Asymptotic-Preserving Schemes for Multiscale Physical Problems. *Acta Numerica*, 31:415–489, June 2022.

[10] Oskar Lappi, Huw Leggate, Yannick Marandet, Keijo Heljanko, and Dmitriy Borodin. Domain Decomposition and Asynchronous Messaging – A Blueprint for Scalable and Deterministic Monte Carlo Neutral Particle Transport Models in Fusion. In preparation, 2025.

[11] Emil Løvbak and Giovanni Samaey. Accelerated simulation of Boltzmann-BGK equations near the diffusive limit with asymptotic-preserving multilevel Monte Carlo. *SIAM Journal on Scientific Computing*, 45(4):A1862–A1889, August 2023.

[12] Emil Løvbak, Giovanni Samaey, and Stefan Vandewalle. A multilevel Monte Carlo method for asymptotic-preserving particle schemes in the diffusive limit. *Numerische Mathematik*, 148(1):141–186, May 2021.

[13] Peter A. Markowich, Christian A. Ringhofer, and Christian Schmeiser. *Semiconductor Equations*. Springer-Verlag, Vienna, Austria, 1990.

[14] Bert Mortier. *Advanced Monte Carlo Simulation and Estimation for Kinetic Neutral Particles in the Plasma Edge of Fusion Reactors*. PhD thesis, KU Leuven, Leuven, Belgium, December 2020.

[15] Bert Mortier, Martine Baelmans, and Giovanni Samaey. A Kinetic-Diffusion Asymptotic-Preserving Monte Carlo Algorithm for the Boltzmann-BGK Model in the Diffusive Scaling. *SIAM Journal on Scientific Computing*, 44(2):A720–A744, April 2022.

[16] Bert Mortier, Vince Maes, and Giovanni Samaey. Estimation as a post-processing step for random walk approximations of the Boltzmann-BGK model. *Contributions to Plasma Physics*, 62(5-6):e202100197, July 2022.

[17] Bert Mortier, Pieterjan Robbe, Martine Baelmans, and Giovanni Samaey. Multilevel asymptotic-preserving Monte Carlo for kinetic-diffusive particle simulations of the Boltzmann-BGK equation. *Journal of Computational Physics*, 450:110736, February 2022.

[18] Detlev Reiter, Martine Baelmans, and Petra Börner. The EIRENE and B2-EIRENE Codes. *Fusion Science and Technology*, 47(2):172–186, February 2005.

[19] Thijs Steel, Vince Maes, and Giovanni Samaey. Fluid boundary conditions in kinetic-diffusion Monte Carlo. *Contributions to Plasma Physics*, Submitted:arXiv:2509.03942, 2025.

[20] Daren Stotler and Charles Karney. Neutral Gas Transport Modeling with DEGAS 2. *Contributions to Plasma Physics*, 34(2-3):392–397, 1994.

[21] Zhirui Tang, Emil Løvbak, Julian Koellermeier, and Giovanni Samaey. Numerical analysis of fluid estimation for source terms in neutral particles simulation. *Contributions to Plasma Physics*, Submitted:arXiv:2509.11883, 2025.

[22] Sven Wiesen, Detlev Reiter, Vladislav Kotov, Martine Baelmans, Wouter Dekeyser, Alexander S. Kukushkin, Steven W. Lisgo, Richard A. Pitts, Vladimir Rozhansky, Gabriella Saibene, Iirina Veselova, and Sergey Voskoboynikov. The new SOLPS-ITER code package. *Journal of Nuclear Materials*, 463:480–484, August 2015.