The Journal of Open Source Software

PAPRECA: A parallel hybrid off-lattice kinetic Monte Carlo/molecular dynamics simulator

Stavros Ntioudis [1¶], **James P. Ewen** [1], **Daniele Dini** [1], **and C. Heath Turner** [2]

1 Department of Mechanical Engineering, Imperial College London, London, SW7 2BX, United Kingdom **2** Department of Chemical and Biological Engineering, University of Alabama, Tuscaloosa, Alabama 35487, United States of America ¶ Corresponding author

DOI: 10.21105/joss.06714

Software

- Review
- Repository
- Archive

Editor: Sarath Menon

Reviewers:

- @liamhuber
- @jfaraudo

Submitted: 12 April 2024
Published: 13 June 2024

Summary

Kinetic Monte Carlo (kMC) is an atomistic and stochastic simulation technique that captures the temporal evolution of various systems in materials science, chemistry, physics, biology, and engineering. Several open-source kMC packages are currently distributed online. Nevertheless, such implementations are typically lattice-based and are mostly designed to study ordered, crystalline materials. In this work, we present PArallel PREdefined CAtalog (PAPRECA), an easy-to-use and completely lattice-free open-source kMC software suitable for simulations on amorphous materials or systems characterized by a low degree of crystallinity. PAPRECA is a parallel C++ software using the Message Passing Interface (MPI) protocol and coupled with the Large-scale Atomic/Molecular Massively Parallel Simulator (LAMMPS) (Thompson et al., 2022) to enable pure kMC runs as well as hybrid kMC/Molecular Dynamics (MD) simulations.

Statement of need

KMC models have been deployed to investigate non-equilibrium dynamics and properties of thin films (Ntioudis et al., 2023), nanoparticles (Turner et al., 2016), quantum dots (Zhu et al., 2007), semiconductors (Kaap & Koster, 2016), catalysts (Stamatakis & Vlachos, 2012), energy-storage devices (Abbott & Hanke, 2022), interstellar grain chemistry (Cuppen et al., 2013), protein folding (Makarov et al., 2001), and enzyme reactions (Slepoy et al., 2008). Overall, kMC techniques are, on one hand, less accurate, but on the other hand, more computationally efficient than MD. This is justified by the fact that kMC does not describe atomic vibrations explicitly but evolves the system through discrete elementary processes (e.g., diffusion, deposition, reactions etc.) (Fichthorn & Weinberg, 1991; Gillespie, 1976). In any case, the efficiency of kMC models unlocks the possibility for long-timescale simulations with molecular-level resolution beyond the timescales accessible to MD. Note that efficient MD algorithms can be used to simulate systems with 10^4-10^6 atoms from ps to μs (Thompson et al., 2022)).

Typically, on-lattice kMC models select atomistic events from a predefined table and execute them on fixed lattice sites (Andersen et al., 2019). The use of fixed lattice sites contributes to the computational efficiency of on-lattice algorithms but introduces obstacles associated with the study of unordered materials. Several lattice-based open-source kMC software are available, examples include the KMCLib (Leetmaa & Skorodumova, 2014), lattice_mc (Morgan, 2017), KMC_Lattice v2.0 (Heiber, 2019), Excimontec v1.0 (Heiber, 2020), MonteCoffee (Jørgensen & Grönbeck, 2018), and KIMERA (P. Martin et al., 2020). Additionally, the Stochastic Parallel PARticle Kinetic Simulator (SPPARKS) (Mitchell et al., 2023) offers solely on-lattice kMC modeling capabilities, since the only currently available off-lattice solver is a Metropolis Monte

Carlo relaxation scheme. Furthermore, a wide range of on-lattice kMC packages such as kmcos (Reuter et al., 2020), PyCD (Pasumarthi, 2017), VIS-A-VIS (Grabowski & Kochanczyk, 2022), MulSKIPS (Helleboid, 2021), Kimocs (Jansson, 2016), KSOME (Nandipati, 2021), kMCpy (Deng, 2022), and Morphokinetics (Alberdi & Albi, 2018) are available in open-source software repositories.

EON (Chill et al., 2014) is the only identified off-lattice package distributed under an open-source license. This is an Adaptive kinetic Monte Carlo (AkMC) software that discovers as well as stores atomic-scale processes (e.g., reactions, diffusion) throughout the simulation instead of stochastically selecting transition events from a predefined table (Henkelman & Jónsson, 2001). Such feature elevates the accuracy of AkMC approaches but decreases their computational efficiency as well as increases their implementation complexity compared to predefined table kMC schemes.

To the best of our knowledge, a completely lattice-free kMC code with predefined table of events is currently unavailable in open-source repositories. PAPRECA aims to fill that gap by providing the scientific community with a general and easy-to-use solution for performing long-timescale atomistic simulations on complex materials science, chemistry, physics, biology, and engineering problems involving amorphous materials or non-crystalline systems.

PAPRECA (in its initial version) is a parallelized software (uses the MPI protocol) that can capture four distinct classes of predefined transition events: 1) reactions (bonding and scission), 2) deposition (of molecules and atoms), 3) diffusion, and 4) monoatomic desorption. Virtually any system whose temporal evolution can be described by these atomic-scale processes can be effectively simulated by PAPRECA. Example applications of PAPRECA include but are not limited to: adsorption/desorption on catalytic surfaces, amorphous thin films (e.g., phosphate films, solid electrolyte interphases, oxide layers), and modeling self-diffusion of gases. Furthermore, PAPRECA allows for the extension of the source code to include other classes of transition events (e.g., reaction chains).

For accurate simulations of molecular-level resolution, it becomes necessary to model elementary steps of high frequency (10^{13}-10^{14} Hz and above (Van Swygenhoven & Weertman, 2006)) such as atomic vibrations. These events cannot be explicitly included in the table of predefined events because they would dominate the system, thus preventing the simulation from reaching timescales beyond the limits of MD. This can be explained by considering that the higher the rate of a predefined elementary step, the greater its kMC selection probability (Fichthorn & Weinberg, 1991). To circumvent this issue, PAPRECA couples an off-lattice kMC solver with a MD solver (LAMMPS (Thompson et al., 2022)) to enable hybrid off-lattice kMC/MD simulations. Effectively, atomistic processes of elevated activation energies are captured via the off-lattice kMC stage, while fast atomic-scale processes are treated by the MD stage.

py-MCMD (Barhaghi et al., 2022) is a different hybrid MC/MD workflow available in open-source repositories. py-MCMD is a Python-based communication interface between the MC software GOMC(Nejahi et al., 2019) and the MD code NAMD(Phillips et al., 2020). The central difference between PAPRECA and py-MCMD is that the former implements a kMC approach, while the latter utilizes a Metropolis Monte Carlo (MMC) scheme. Effectively, in the kMC stage of PAPRECA predefined event probabilities are calculated based on their rates and an elementary process is selected and executed to overcome a high energy barrier. On the other hand, during the MC phase of py-MCMD, trial moves (e.g., rigid-body displacements/rotations, intra-box swaps (M. G. Martin & Siepmann, 1999)) are attempted and accepted or rejected based on the Metropolis acceptance criterion (Chen & Roux, 2015).

Scalability of PAPRECA

The scalability of PAPRECA was investigated by performing hybrid kMC/MD simulations on thin films grown from the decomposition of lubricant additive tricresyl phosphate (TCP) molecules

on an Fe(110) substrate. For further information regarding the system setup refer to the PAPRECA documentation (Example Applications section) and our previous study (Ntioudis et al., 2023).

Two independent scalability tests were performed. The first scalability test was conducted locally, on a personal computer (CPU: Intel(R) Core(TM) i9-10980XE CPU @ 3.00GHz, RAM: 128 Gb DDR4 @ 3200 MHz). Four runs were performed with 1, 4, 9, and 16 MPI processes, respectively. The local tests simulated 1000 PAPRECA steps, with a PAPRECA step comprising a kMC stage where a predefined event is executed, followed by an MD stage where the system is relaxed. The second scalability test was performed on the CX3 cluster managed by the Research Computing Service at Imperial College London (CPU: 2xAMD EPYC 7742 with 128 cores per node, RAM: 1TB per node, interconnect: 100GbE ethernet). This scalability test was conducted with the same parameters as the local one but with a different number of total PAPRECA steps (i.e., 9000 instead of 1000). Also, five runs were performed with 1, 4, 16, 64, and 144 MPI processes. Since the phosphate thin film grew along the z-direction of the simulation box, an NxNx1 processor grid (along the x-, y-, and z-directions, respectively) was utilized for all local and HPC tests. Figure 1 illustrates the results of both tests:

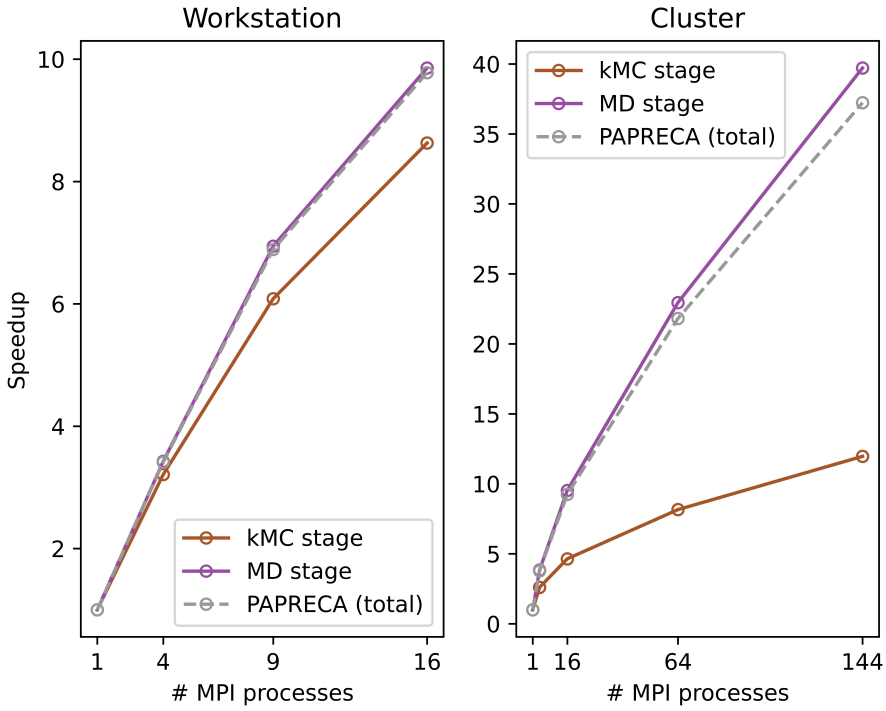

Figure 1: Hybrid kMC/MD scalability tests of PAPRECA for TCP on Fe(110) conducted on a workstation (left) and on the CX3 cluster at Imperial College. (right).

Where the speedup value of N MPI processes was calculated as $t_N = \frac{T_1}{T_N}$, with T_1 being the total walltime of 1 MPI process. For this specific system (i.e., phosphates example), it can be observed that the kMC stage does not scale as effectively as the MD stage (performed in LAMMPS). Nonetheless, the total speedup (i.e., combined kMC and MD) of a hybrid PAPRECA run is comparable to the MD stage speedup. This can be justified by the fact that the kMC stages require significantly less CPU time than the MD stages, regardless of the number of MPI processes. For instance, the total walltimes of the kMC and MD stages of the 64 MPI processes example (second scalability test on the CX3 cluster) were 0.226 and 2.71 hours, respectively. Overall, improving the scalability of the kMC stage will be prioritized in the

upcoming versions of PAPRECA.

Data availability

Scalability test data is available on our software repository (Ntioudis et al., 2024).

Acknowledgements

S.N., J.P.E., and D.D. thank Shell and the Engineering and Physical Sciences Research Council, United Kingdom (EPSRC) for funding via the InFUSE Prosperity Partnership (EP/V038044/1). J.P.E. acknowledges the support of the Royal Academy of Engineering (RAEng) for support through their Research Fellowship scheme. D.D. acknowledges a Shell/RAEng Research Chair in Complex Engineering Interfaces. We acknowledge computational resources and support provided by the Imperial College Research Computing Service.

References

Abbott, J. W., & Hanke, F. (2022). Kinetically Corrected Monte Carlo–Molecular Dynamics Simulations of Solid Electrolyte Interphase Growth. *Journal of Chemical Theory and Computation*, *18*(2), 925–934. https://doi.org/10.1021/acs.jctc.1c00921

Alberdi, J., & Albi, J. (2018). *Morphokinetics*. https://github.com/dipc-cc/Morphokinetics

Andersen, M., Panosetti, C., & Reuter, K. (2019). A Practical Guide to Surface Kinetic Monte Carlo Simulations. *Frontiers in Chemistry*, *7*, 434159. https://doi.org/10.3389/fchem.2019.00202

Barhaghi, M. S., Crawford, B., Schwing, G., Hardy, D. J., Stone, J. E., Schwiebert, L., Potoff, J., & Tajkhorshid, E. (2022). py-MCMD: Python Software for Performing Hybrid Monte Carlo/Molecular Dynamics Simulations with GOMC and NAMD. *Journal of Chemical Theory and Computation*, *18*(8), 4983–4994. https://doi.org/10.1021/acs.jctc.1c00911

Chen, Y., & Roux, B. (2015). Generalized Metropolis acceptance criterion for hybrid non-equilibrium molecular dynamics—Monte Carlo simulations. *Journal of Chemical Physics*, *142*(2). https://doi.org/10.1063/1.4904889

Chill, S. T., Welborn, M., Terrell, R., Zhang, L., Berthet, J.-C., Pedersen, A., Jónsson, H., & Henkelman, G. (2014). EON: software for long time simulations of atomic scale systems. *Modeling and Simulation in Materials Science and Engineering*, *22*(5), 055002. https://doi.org/10.1088/0965-0393/22/5/055002

Cuppen, H. M., Karssemeijer, L. J., & Lamberts, T. (2013). The Kinetic Monte Carlo Method as a Way To Solve the Master Equation for Interstellar Grain Chemistry. *Chemical Reviews*, *113*(12), 8840–8871. https://doi.org/10.1021/cr400234a

Deng, Z. (2022). *kMCpy: Kinetic Monte Carlo Simulation using Python*. https://github.com/caneparesearch/kMCpy

Fichthorn, K. A., & Weinberg, W. H. (1991). Theoretical foundations of dynamical Monte Carlo simulations. *Journal of Chemical Physics*, *95*(2), 1090–1096. https://doi.org/10.1063/1.461138

Gillespie, D. T. (1976). A general method for numerically simulating the stochastic time evolution of coupled chemical reactions. *Journal of Computational Physics*, *22*(4), 403–434. https://doi.org/10.1016/0021-9991(76)90041-3

Grabowski, F., & Kochanczyk, M. (2022). *VIS-A-VIS: agent-based simulator of viral infection spread and viral infection self-containment in a monolayer of cell*. https://github.com/

grfrederic/visavis

Heiber, M. C. (2019). KMC_Lattice v2.0: An Object-Oriented C\texttt{++} Library for Custom Kinetic Monte Carlo Simulations. *Journal of Open Source Software*, *4*(33), 1168. https://doi.org/10.21105/joss.01168

Heiber, M. C. (2020). Excimontec v1.0: An Open-Source Software Tool for Kinetic Monte Carlo Simulations of Organic Electronic Devices. *Journal of Open Source Software*, *5*(53), 2307. https://doi.org/10.21105/joss.02307

Helleboid, R. (2021). *MulSKIPS: A Kinetic Monte Carlo super-Lattice code*. https://github.com/MulSKIPS/MulSKIPS

Henkelman, G., & Jónsson, H. (2001). Long time scale kinetic Monte Carlo simulations without lattice approximation and predefined event table. *Journal of Chemical Physics*, *115*(21), 9657–9666. https://doi.org/10.1063/1.1415500

Jansson, V. (2016). *Kimocs - Kinetic Monte Carlo for Surfaces*. https://gitlab.com/vjansson/Kimocs

Jørgensen, M., & Grönbeck, H. (2018). MonteCoffee: A programmable kinetic Monte Carlo framework. *Journal of Chemical Physics*, *149*(11). https://doi.org/10.1063/1.5046635

Kaap, N. J. van der, & Koster, L. J. A. (2016). Massively parallel kinetic Monte Carlo simulations of charge carrier transport in organic semiconductors. *Journal of Computational Physics*, *307*, 321–332. https://doi.org/10.1016/j.jcp.2015.12.001

Leetmaa, M., & Skorodumova, N. V. (2014). KMCLib: A general framework for lattice kinetic Monte Carlo (KMC) simulations. *Computional Physics Communications*, *185*(9), 2340–2349. https://doi.org/10.1016/j.cpc.2014.04.017

Makarov, D. E., Hansma, P. K., & Metiu, H. (2001). Kinetic Monte Carlo simulation of titin unfolding. *Journal of Chemical Physics*, *114*(21), 9663–9673. https://doi.org/10.1063/1.1369622

Martin, M. G., & Siepmann, J. I. (1999). Novel Configurational-Bias Monte Carlo Method for Branched Molecules. Transferable Potentials for Phase Equilibria. 2. United-Atom Description of Branched Alkanes. *Journal of Physical Chemistry B*, *103*(21), 4508–4517. https://doi.org/10.1021/jp984742e

Martin, P., Gaitero, J. J., Dolado, J. S., & Manzano, H. (2020). KIMERA: A Kinetic Montecarlo Code for Mineral Dissolution. *Minerals*, *10*(9), 825. https://doi.org/10.3390/min10090825

Mitchell, J. A., Abdeljawad, F., Battaile, C., Garcia-Cardona, C., Holm, E. A., Homer, E. R., Madison, J., Rodgers, T. M., Thompson, A. P., Tikare, V., Webb, E., & Plimpton, S. J. (2023). Parallel simulation via SPPARKS of on-lattice kinetic and Metropolis Monte Carlo models for materials processing. *Modeling and Simulation in Materials Science and Engineering*, *31*(5), 055001. https://doi.org/10.1088/1361-651X/accc4b

Morgan, B. J. (2017). lattice_mc: A Python Lattice-Gas Monte Carlo Module. *Journal of Open Source Software*, *2*(13), 247. https://doi.org/10.21105/joss.00247

Nandipati, G. (2021). *KSOME: Kinetic Simulations of Microstructural Evolution*. https://gitlab.osti.gov/giridharnandipati/ksome

Nejahi, Y., Barhaghi, M. S., Mick, J., Jackman, B., Rushaidat, K., Li, Y., Schwiebert, L., & Potoff, J. (2019). GOMC: GPU Optimized Monte Carlo for the simulation of phase equilibria and physical properties of complex fluids. *SoftwareX*, *9*, 20–27. https://doi.org/10.1016/j.softx.2018.11.005

Ntioudis, S., Ewen, J. P., Dini, D., & Turner, C. H. (2023). A hybrid off-lattice kinetic Monte Carlo/molecular dynamics method for amorphous thin film growth. *Computational Materials Science*, *229*, 112421. https://doi.org/10.1016/j.commatsci.2023.112421

Ntioudis, S., Ewen, J. P., Dini, D., & Turner, C. H. (2024). *Papreca*. https://github.com/sntioudis/papreca

Pasumarthi, V. (2017). *Python-based Charge Dynamics (PyCD)*. https://github.com/vpasumarthi/PyCD/

Phillips, J. C., Hardy, D. J., Maia, J. D. C., Stone, J. E., Ribeiro, J. V., Bernardi, R. C., Buch, R., Fiorin, G., Hénin, J., Jiang, W., McGreevy, R., Melo, M. C. R., Radak, B. K., Skeel, R. D., Singharoy, A., Wang, Y., Roux, B., Aksimentiev, A., Luthey-Schulten, Z., … Tajkhorshid, E. (2020). Scalable molecular dynamics on CPU and GPU architectures with NAMD. *Journal of Chemical Physics*, *153*(4). https://doi.org/10.1063/5.0014475

Reuter, K., Savara, Aditya, Hoffmann, M. J., Matera, S., Lorenzi, J. M., Andersen, M., Bajdich, M., & Garhammer, Andreas. (2020). *Kmcos: Kinetic monte carlo of systems*. https://github.com/kmcos/kmcos

Slepoy, A., Thompson, A. P., & Plimpton, S. J. (2008). A constant-time kinetic Monte Carlo algorithm for simulation of large biochemical reaction networks. *Journal of Chemical Physics*, *128*(20). https://doi.org/10.1063/1.2919546

Stamatakis, M., & Vlachos, D. G. (2012). Unraveling the Complexity of Catalytic Reactions via Kinetic Monte Carlo Simulation: Current Status and Frontiers. *ACS Catalysis*, *2*(12), 2648–2663. https://doi.org/10.1021/cs3005709

Thompson, A. P., Aktulga, H. M., Berger, R., Bolintineanu, D. S., Brown, W. M., Crozier, P. S., In 't Veld, P. J., Kohlmeyer, A., Moore, S. G., Nguyen, T. D., Shan, R., Stevens, M. J., Tranchida, J., Trott, C., & Plimpton, S. J. (2022). LAMMPS - a flexible simulation tool for particle-based materials modeling at the atomic, meso, and continuum scales. *Computational Physics Communications*, *271*, 108171. https://doi.org/10.1016/j.cpc.2021.108171

Turner, C. H., Lei, Y., & Bao, Y. (2016). Modeling the atomistic growth behavior of gold nanoparticles in solution. *Nanoscale*, *8*(17), 9354–9365. https://doi.org/10.1039/C6NR01881E

Van Swygenhoven, H., & Weertman, J. R. (2006). Deformation in nanocrystalline metals. *Materials Today*, *9*(5), 24–31. https://doi.org/10.1016/S1369-7021(06)71494-8

Zhu, R., Pan, E., & Chung, P. W. (2007). Fast multiscale kinetic Monte Carlo simulations of three-dimensional self-assembled quantum dot islands. *Physical Review B*, *75*(20), 205339. https://doi.org/10.1103/PhysRevB.75.205339