

polypy - Analysis Tools for Solid State Molecular Dynamics and Monte Carlo Trajectories

Adam R. Symington¹

¹ Department of Chemistry, University of Bath

DOI: [10.21105/joss.02824](https://doi.org/10.21105/joss.02824)

Software

- [Review](#) ↗
- [Repository](#) ↗
- [Archive](#) ↗

Editor: Richard Gowers ↗

Reviewers:

- [@hmacdope](#)
- [@lscalfi](#)

Submitted: 29 September 2020

Published: 04 March 2021

License

Authors of papers retain copyright and release the work under a Creative Commons Attribution 4.0 International License ([CC BY 4.0](https://creativecommons.org/licenses/by/4.0/)).

Summary

A large number of research questions in solid state chemistry can be addressed using molecular dynamics and Monte Carlo simulations. These simulations allow many material properties to be calculated for direct comparison with experiment. These include the diffusion coefficients, ionic conductivities, charge density, electric field, and electrostatic potential. The diffusion coefficient and ionic conductivity are of particular importance for the study of battery materials (e.g., Li-ion / Na-ion diffusion ([A. Symington et al., 2021](#); [A. R. Symington et al., 2020](#))), solid oxide fuel cell materials (e.g. O-ion diffusion ([A. R. Symington, Molinari, Statham, et al., 2019](#))) and many other applications in solid state chemistry. The charge density, electric field, and electrostatic potential are of interest to problems relating to interfaces in solid state chemistry, e.g., space charge theory ([Chiang et al., 1996](#); [Kim & Maier, 2002](#); [Maier, 1984, 1985, 2003](#)) Finally, calculating the distribution of defects in a material is useful for the study of segregation behaviour ([A. R. Symington, Molinari, Brincat, et al., 2019](#); [A. R. Symington, Molinari, Statham, et al., 2019](#)) or adsorption behaviour ([Leeuw & Parker, 1998](#)).

In a molecular dynamics simulation the positions of atoms throughout time are being simulated. A molecular dynamics trajectory is a snapshot of the positions occupied by each atom in the simulation as a function time. For example, the trajectory of a single atom would show, sequentially, all of the positions occupied by that atom throughout the simulation. In a Monte Carlo simulation the positions of atoms are updated randomly to provide a statistical ensemble describing the material. A Monte Carlo trajectory is similar although the simulation is not time resolved and the atom positions are simply a function of simulation step, not simulation timestep. The positions of the atoms allow the particle density of each atom to be determined and from these, the electrostatic potential, electric field and charge density can be calculated. A mean squared displacement can be performed on the molecular dynamics trajectories and from these, the diffusion coefficients and ionic conductivities can be calculated. Diffusion coefficients and ionic conductivities can then be used to estimate the activation energy for diffusion using the Arrhenius relationship.

The polypy code is designed to solve the following problems.

- Read DL_POLY ([Smith & Forester, 1996](#)) and DL_MONTE ([Purton et al., 2013](#)) trajectories.
- Calculate the particle density of all species in a trajectory in one and two dimensions.
- Calculate the charge density in one and two dimensions.
- Calculate the electric field and electrostatic potential in one dimension.
- Calculate the mean squared displacement for a given atom and use this to calculate the diffusion coefficient and ionic conductivity.
- Calculate the volume as a function of simulation timestep.
- Generate publication ready figures.

polypy has been used to study Li-ion diffusion in lithium lanthanum titanate, (A. Symington et al., 2021; A. R. Symington et al., 2020) oxygen diffusion and cation migration in both uranium oxide and cerium oxide, (A. R. Symington, Molinari, Brincat, et al., 2019; A. R. Symington, Molinari, Statham, et al., 2019) thus there is a clear research application.

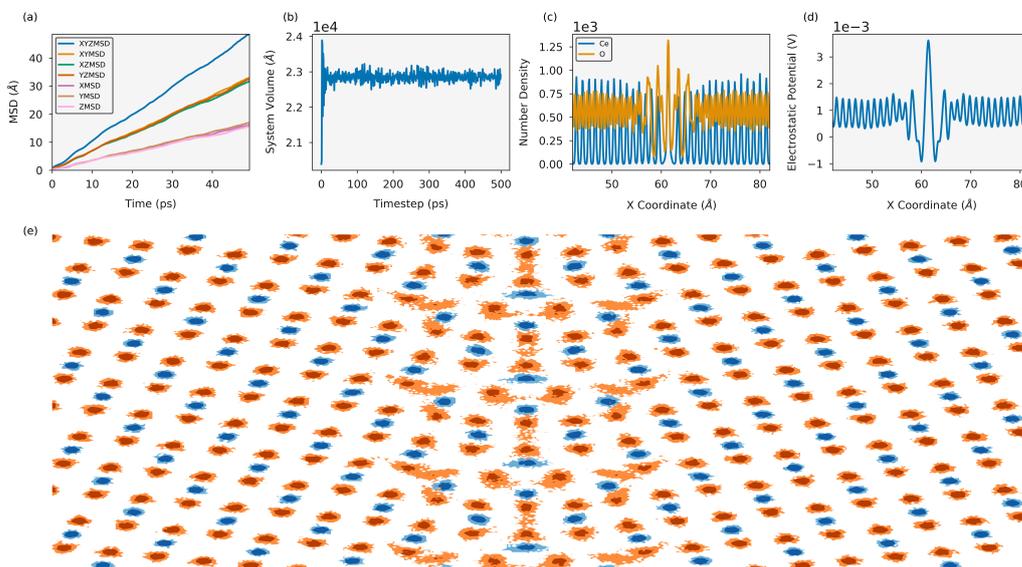


Figure 1: Figure 1 - (a) Mean squared displacement for fluorine diffusion in calcium fluoride. The msd has been plotted in one, two, and three dimensions. (b) The evolution of the system volume during a molecular dynamics simulation. (c) The particle density of cerium (blue) and oxygen (orange) atoms at a grain boundary in cerium oxide. (d) The electrostatic potential at a cerium oxide grain boundary. (e) The center of mass of cerium (blue) and oxygen (orange) atoms in two dimensions, at a grain boundary in cerium oxide.

polypy

polypy is a Python module for analysing molecular dynamics and Monte Carlo trajectories generated from the DL_POLY (Smith & Forester, 1996) and DL_MONTE (Purton et al., 2013) codes. The code reads DL_POLY HISTORY and CONFIG files, DL_MONTE ARCHIVE files, and stores the data in a `polypy.read.Trajectory` object that is then used by the various data analysis modules.

The `polypy.density.Density` module generates a three dimensional grid and counts the number of times a given atom spends at each grid point during the simulation. This is then used to generate the particle density of a given atom in one and two dimensions. From here, the charge density in one and two dimensions, the electric field in one dimension, and electrostatic potential in one dimension can be calculated using the `polypy.analysis` module.

The `polypy.msdc` module performs a mean squared displacement calculation. From the mean squared displacement, the three, two and one dimensional diffusion coefficient, and ionic coefficient can be calculated.

A module allowing easy generation of publication plots from the calculated data is available. The outputs are returned in a sensible form, allowing further manipulation and plotting. The repository contains examples of the core functionality as well as tutorials, implemented in Jupyter notebooks to explain the full theory. Furthermore, a detailed description of theory is also available within the documentation. polypy is aimed towards theoretical solid state physicists and chemists who have a basic familiarity with Python, although the examples contained in the repository are designed to help less experienced users make use of the code.

Statement of Need

There are a number of codes designed to analyse molecular dynamics trajectories that currently exist (Fraux et al., 2020; Michaud-Agrawal et al., 2011; Yong & Todorov, 2018). DL_ANALYSER (Yong & Todorov, 2018) is available under license for the analysis of DL_POLY simulations and chemfiles (Fraux et al., 2020) is available for the analysis of a wide range of file types. MDAnalysis (Michaud-Agrawal et al., 2011) is the most widely used molecular dynamics analysis code and some of the functionality in polypy is already present in MDAnalysis. The MDAnalysis.analysis.lineardensity module calculates the charge density in different dimensions, although according to the documentation, is limited to orthorhombic, fixed volume cells. polypy is designed to work with several simulation ensembles including, NPT, NVT, semi grand, and grand canonical. Furthermore, the calculation of the electric field and electrostatic potential is unique to polypy. MDAnalysis and polypy are both capable of calculating mean squared displacements. polypy goes a step further by allowing the calculation of diffusion coefficients and conductivities within localised regions of a structure, e.g., a grain boundary A. R. Symington, Molinari, Statham, et al. (2019) or local structural environments (A. R. Symington et al., 2020). polypy is also unique in the sense that it is designed for the analysis of both molecular dynamics and Monte Carlo trajectories.

In summary the features that are unique to polypy are as follows

- The analysis of both molecular dynamics and Monte Carlo trajectories.
- The calculation of the electric field and electrostatic potential.
- Regional mean squared displacements.

Acknowledgements

This package was written during a PhD project that was funded by AWE and EPSRC (EP/R010366/1). The polypy software package was developed to analyse data generated using the Balena HPC facility at the University of Bath and the ARCHER UK National Supercomputing Service (<http://www.archer.ac.uk>) via our membership of the UK's HEC Materials Chemistry Consortium funded by EPSRC (EP/L000202). The author would like to thank Andrew R. McCluskey, Benjamin Morgan, Marco Molinari, James Grant, and Stephen C. Parker for their help and guidance during this PhD project.

References

- Chiang, Y. M., Lavik, E. B., Kosacki, I., Tuller, H. L., & Ying, J. Y. (1996). Defect and transport properties of nanocrystalline CeO_{2-x} . *Appl. Phys. Lett.*, 69(2), 185–187. <https://doi.org/10.1063/1.117366>
- Fraux, G., Fine, J., ezavod, Barletta, G. P., Scalfi, L., & Dimura, M. (2020). *Chemfiles/chemfiles: Version 0.9.3* (Version 0.9.3) [Computer software]. Zenodo. <https://doi.org/10.5281/zenodo.3653157>
- Kim, S., & Maier, J. (2002). On the conductivity mechanism of nanocrystalline ceria. *J. Electrochem. Soc.*, 149(10), J73–J83. <https://doi.org/10.1149/1.1507597>
- Leeuw, N. H. de, & Parker, S. C. (1998). Molecular-dynamics simulation of MgO surfaces in liquid water using a shell-model potential for water. *Phys. Rev. B*, 58, 13901–13908. <https://doi.org/10.1103/PhysRevB.58.13901>

- Maier, J. (1984). Enhancement of the ionic conductivity in solid-solid-dispersions by surface induced defects. *Berichte Der Bunsengesellschaft für Physikalische Chemie*, 88(11), 1057–1062. <https://doi.org/10.1002/bbpc.198400007>
- Maier, J. (1985). Space charge regions in solid two-phase systems and their conduction contribution I. Conductance enhancement in the system ionic conductor-inert phase and application on AgCl:Al₂O₃ and AgCl:SiO₂. *J. Phys. Chem. Sol.*, 46(3), 309–320. [https://doi.org/10.1016/0022-3697\(85\)90172-6](https://doi.org/10.1016/0022-3697(85)90172-6)
- Maier, J. (2003). Defect chemistry and ion transport in nanostructured materials: Part II. Aspects of nanoionics. *Solid State Ionics*, 157(1), 327–334. [https://doi.org/10.1016/S0167-2738\(02\)00229-1](https://doi.org/10.1016/S0167-2738(02)00229-1)
- Michaud-Agrawal, N., Denning, E. J., Woolf, T. B., & Beckstein, O. (2011). MDAAnalysis: A toolkit for the analysis of molecular dynamics simulations. *Journal of Computational Chemistry*, 32(10), 2319–2327. <https://doi.org/10.1002/jcc.21787>
- Purton, J. A., Crabtree, J. C., & Parker, S. C. (2013). DL_MONTE: A general purpose program for parallel Monte Carlo simulation. *Molecular Simulation*, 39(14-15), 1240–1252. <https://doi.org/10.1080/08927022.2013.839871>
- Smith, W., & Forester, T. R. (1996). DL_POLY 2.0: A general-purpose parallel molecular dynamics simulation package. *Journal of Molecular Graphics*, 14(3), 136–141. [https://doi.org/10.1016/S0263-7855\(96\)00043-4](https://doi.org/10.1016/S0263-7855(96)00043-4)
- Symington, A., Molinari, M., Dawson, J. A., Statham, J., Purton, J., Canepa, P., & Parker, S. (2021). Elucidating the nature of grain boundary resistance in lithium lanthanum titanate. *J. Mater. Chem. A*, -. <https://doi.org/10.1039/D0TA11539H>
- Symington, A. R., Molinari, M., Brincat, Nicholas. A., Williams, N. R., & Parker, S. C. (2019). Defect segregation facilitates oxygen transport at fluorite UO₂ grain boundaries. *Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences*, 377(2152), 20190026. <https://doi.org/10.1098/rsta.2019.0026>
- Symington, A. R., Molinari, M., Statham, J., Wu, J., & Parker, S. C. (2019). The role of dopant segregation on the oxygen vacancy distribution and oxygen diffusion in CeO₂ grain boundaries. *Journal of Physics: Energy*, 1(4), 042005. <https://doi.org/10.1088/2515-7655/ab28b5>
- Symington, A. R., Purton, J., Statham, J., Molinari, M., Islam, M. S., & Parker, S. C. (2020). Quantifying the impact of disorder on li-ion and na-ion transport in perovskite titanate solid electrolytes for solid-state batteries. *Journal of Materials Chemistry A*, -. <https://doi.org/10.1039/D0TA05343K>
- Yong, C. W., & Todorov, I. T. (2018). DL_ANALYSER notation for atomic interactions (DANAI): A natural annotation system for molecular interactions, using ethanoic acid liquid as a test case. *Molecules*, 23(1). <https://doi.org/10.3390/molecules23010036>