The Journal of Open Source Software

pyscal: A python module for structural analysis of atomic environments

Sarath Menon[1], **Grisell Díaz Leines**[1], **and Jutta Rogal**[1]

1 Interdisciplinary Centre for Advanced Materials Simulation, Ruhr-Universität Bochum, Universitätsstr. 150, 44801 Bochum, Germany.

DOI: 10.21105/joss.01824

Software

- Review
- Repository
- Archive

Submitted: 17 October 2019
Published: 01 November 2019

Summary

The structural characterization of local atomic environments is essential in our understanding and design of materials with customized properties. Atomistic simulations have become a powerful tool to provide insight into the structural patterns and mechanisms that signify transformations between different crystalline and liquid phases, or the formation and dynamics of point and extended defects. The development and availability of methods that provide a reliable and accurate structural analysis of atomistic simulation data constitute an indispensable task that continues to be an active field of research in materials science.

Several techniques have been proposed over the years to identify local structural motifs in crystalline materials (Stukowski, 2012). To distinguish solid- and liquid-like environments, Auer and Frenkel (Auer & Frenkel, 2005) introduced a criterion that measures the structural correlations between a particle and its neighbors using the bond orientational order parameters suggested by Steinhardt et al. (Steinhardt, Nelson, & Ronchetti, 1983). The Steinhardt parameters are based on the spherical harmonics in the local neighborhood of a particle within a cutoff distance, and are rotationally and translationally invariant. They have been used extensively for the identification of crystal structures (Mickel, Kapfer, Schröder-Turk, & Mecke, 2013 and references within). At finite temperatures the Steinhardt parameters for a given crystal structure exhibit a certain distribution of values instead of a single one due to thermal vibrations of the atoms. This can lead to an overlap in the distributions for different phases which makes an accurate identification difficult or even impossible. An averaged version of the Steinhardt bond order parameters introduced by Lechner and Dellago (Lechner & Dellago, 2008) significantly reduces this overlap and improves the characterization of common crystal structures. An additional complication at finite temperatures is the use of a fixed cutoff distance which might cause ambiguity in the definition of the local neighborhood of a particle. An adaptive definition of the local environment can be achieved by methods such as the solid angle based nearest neighbor (SANN) algorithm (van Meel, Filion, Valeriani, & Frenkel, 2012) or the adaptive common neighbor analysis (a-CNA) (Stukowski, 2012). Furthermore, Voronoi tessellation can be employed for a parameter-free definition of the local neighborhood of a particle. Mickel et al. showed (Mickel et al., 2013) that the identification of crystal structures can be improved by weighting the contribution of each neighbor to the Steinhardt parameters by the area of the Voronoi facet shared between the central atom and its corresponding neighbor. Higher exponents of the facet area to weight the bond orientational order parameters can further improve the recognition of common crystal phases (Haeberle, Sperl, & Born, 2019).

Originally inspired by the C++ code `BondOrderAnalysis` (Lechner, 2016) for the calculation of bond order parameters, `pyscal` is a Python module designed for the computation of local structural order parameters during post-processing of atomistic simulation data. In addition to the calculation of Steinhardt parameters which is offered by tools like `BondOrderAnalysis`

and pyboo (Leocmach, 2017), pyscal brings together the various approaches for structure identification discussed above in a single Python module. While Python offers the advantages of extensibility and flexibility, the module ensures the speed and efficiency of the calculations by using a core code written in C++, which is ported to Python using pybind11 (Jakob, Rhinelander, & Moldovan, 2016). The pyscal module includes the following functionality:

- calculation of Steinhardt bond orientational order parameters and their averaged version.
- weighted bond order parameters using face areas of Voronoi polyhedra; the calculation of the Voronoi polyhedra is enabled using the Voro++ code (Rycroft, 2009) integrated into pyscal.
- distinction of atoms in solid or liquid environments based on the criterion by Auer and Frenkel (Auer & Frenkel, 2005).
- flexible local neighborhood definition using an adaptive cutoff distance.
- clustering algorithm for particles based on user defined properties.
- inbuilt functions for additional structural features including radial distribution functions, Voronoi volumes of individual particles, and coordination numbers.
- calculation of further parameters based on Voronoi tesselation such as the number of vertices and face areas.

pyscal uses a list of particle positions and simulation cell vectors as input, which can also be read in from a file. Currently supported file formats include the dump format of the molecular dynamics code LAMMPS (Plimpton, 1995) and the POSCAR format used by the *ab initio* simulation package VASP (Kresse & Furthmüller, 1996a, 1996b; Kresse & Hafner, 1993). Extensions to other file formats are easily implemented. The module can handle arbitrary simulation cells. pyscal also offers a set of supporting tools including those for reading, writing, and splitting atomic trajectory files and the setup of simulation cells for perfect crystal structures. In general, the pyscal module acts as a tool for the calculation of quantities based on atomic positions, and can easily be extended using either C++ for computationally intensive operations or Python for other tasks that wrap over the existing features. pyscal includes a documentation and various usage examples, available on the pyscal website. Further examples, including installation of the package and the calculation of bond orientational order parameters, are also available. Alternatively, pyscal provides a binder environment to test the example cases before installation.

Acknowledgements

S.M. acknowledges a scholarship from the International Max Planck Research School for Interface Controlled Materials for Energy Conversion. The authors acknowledge support by the Mexican National Council for Science and Technology (CONACYT) through project 232090 and by the German Research Foundation (DFG) through project 262052203.

References

Auer, S., & Frenkel, D. (2005). *Numerical Simulation of Crystal Nucleation in Colloids: Advanced computer simulation. Advances in polymer science*. (C. Dr. Holm & K. Prof. Dr. Kremer, Eds.) (Vol. 173). Springer Berlin Heidelberg. doi:10.1007/b99429

Haeberle, J., Sperl, M., & Born, P. (2019). Distinguishing noisy crystalline structures using bond orientational order parameters. Retrieved from http://arxiv.org/abs/1906.08111

Jakob, W., Rhinelander, J., & Moldovan, D. (2016). Pybind11—seamless operability between C++ 11 and Python. Retrieved from https://github.com/pybind/pybind11

Kresse, G., & Furthmüller, J. (1996a). Efficiency of ab-initio total energy calculations for metals and semiconductors using a plane-wave basis set. *Computational Materials Science*, *6*, 15. doi:10.1016/0927-0256(96)00008-0

Kresse, G., & Furthmüller, J. (1996b). Efficient iterative schemes for ab initio total-energy calculations using a plane-wave basis set. *Physical Review B*, *54*, 11169. doi:10.1103/PhysRevB.54.11169

Kresse, G., & Hafner, J. (1993). Ab initio molecular dynamics for liquid metals. *Physical Review B*, *47*, 558. doi:10.1103/PhysRevB.47.558

Lechner, W. (2016). Bond order analysis. Retrieved from https://github.com/WolfgangLechner/StructureAnalysis

Lechner, W., & Dellago, C. (2008). Accurate determination of crystal structures based on averaged local bond order parameters. *The Journal of Chemical Physics*, *129*, 114707. doi:10.1063/1.2977970

Leocmach, A. (2017). Pyboo: A Python package to compute bond orientational order parameters. doi:10.5281/zenodo.1066568

Mickel, W., Kapfer, S. C., Schröder-Turk, G. E., & Mecke, K. (2013). Shortcomings of the bond orientational order parameters for the analysis of disordered particulate matter. *The Journal of Chemical Physics*, *138*, 044501. doi:10.1063/1.4774084

Plimpton, S. (1995). Fast Parallel Algorithms for Short - Range Molecular Dynamics. *Journal of Computational Physics*, *117*, 1. doi:10.1006/jcph.1995.1039

Rycroft, C. H. (2009). VORO++: A three-dimensional Voronoi cell library in C++. *Chaos: An Interdisciplinary Journal of Nonlinear Science*, *19*, 041111. doi:10.1063/1.3215722

Steinhardt, P. J., Nelson, D. R., & Ronchetti, M. (1983). Bond-orientational order in liquids and glasses. *Physical Review B*, *28*, 784. doi:10.1103/PhysRevB.28.784

Stukowski, A. (2012). Structure identification methods for atomistic simulations of crystalline materials. *Modelling and Simulation in Materials Science and Engineering*, *20*, 045021. doi:10.1088/0965-0393/20/4/045021

van Meel, J. A., Filion, L., Valeriani, C., & Frenkel, D. (2012). A parameter-free, solid-angle based, nearest-neighbor algorithm. *The Journal of Chemical Physics*, *136*, 234107. doi:10.1063/1.4729313