










smol: A Python package for cluster expansions and beyond

Luis Barroso-Luque ^{1,2¶}, Julia H. Yang ^{1,2}, Fengyu Xie ^{1,2}, Tina Chen ^{1,2}, Ronald L. Kam^{1,2}, Zinab Jadidi^{1,2}, Peichen Zhong ^{1,2}, and Gerbrand Ceder ^{1,2¶}

¹ Department of Materials Science and Engineering, University of California Berkeley, Berkeley CA, 94720, USA ² Materials Sciences Division, Lawrence Berkeley National Laboratory, Berkeley CA, 94720, USA ¶ Corresponding author

DOI: [10.21105/joss.04504](https://doi.org/10.21105/joss.04504)

Software

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

Editor: [Jarvist Moore Frost](#) 

Reviewers:

- [@TomTranter](#)
- [@zhubonan](#)

Submitted: 27 April 2022

Published: 29 September 2022

License

Authors of papers retain copyright and release the work under a Creative Commons Attribution 4.0 International License ([CC BY 4.0](#)).

Summary

The growing research focus on multi-principal element materials—spanning a variety of applications, such as electrochemical ([Lun et al., 2020](#)), structural ([George et al., 2019](#)), semiconductor, thermoelectric, magnetic, and superconducting ([Gao et al., 2018](#)) materials—necessitates the development of computational methodology capable of resolving details of atomic configuration and resulting thermodynamic properties. The cluster expansion (CE) method is a formal and effective way to construct functions of atomic configuration by coarse-graining materials properties, such as formation energies, in terms of species occupancy lattice models ([Sanchez et al., 1984](#)). The cluster expansion method coupled with Monte Carlo sampling (CE-MC) is an established and effective way to resolve atomic details underlying important thermodynamic properties ([Van der Ven et al., 2018](#)).

smol (Statistical Mechanics on Lattices) is a Python package for constructing generalized applied lattice models, and performing Monte Carlo sampling of associated thermodynamic ensembles. The representation of lattice models in smol is based largely on the CE formalism ([Sanchez et al., 1984](#)). However, the package is designed to allow easy implementation of extensions to the formalism, such as redundant representations ([Barroso-Luque et al., 2021](#)). smol also includes flexible and extensible functionality to run Monte Carlo (MC) sampling from canonical and semigrand-canonical ensembles associated with the generated lattice models. smol has been intentionally designed to be lightweight and include a minimal set of dependencies to enable smooth installation, use, and development. smol was conceived primarily to enable development and implementation of novel CE-MC methodology but is now sufficiently mature that it is already being used in applied research of relevant material systems. ([Chen et al., submitted 2022](#); [Jadidi et al., in prep. 2022](#); [Yang et al., 2022](#); [Yang & Ceder, in prep. 2022](#))

Statement of need

Several high-quality software packages implementing CE-MC methodology, such as ATAT ([A. van de Walle et al., 2002](#)), CASM ([Thomas et al., 2015/2022](#)), CLEASE ([Chang et al., 2019](#)), and icet ([Ångqvist et al., 2019](#)) are readily available either open source or by request. However, smol is distinct from existing CE-MC packages in both vision and implementation for the following three main reasons:

1. smol has been designed to easily develop, implement and test new methodology for the representation, fitting, and inference of applied lattice models beyond standard CE-MC methodology. The package has a heavily object-oriented and modular design that

closely follows mathematical and methodological abstractions, which enables hassle-free implementation of methodology extensions. Furthermore, `smol` is written in pure Python (with a few critical components implemented in Cython to maintain performance) making it particularly developer friendly.

2. `smol` is the only package implemented using `pymatgen`—a widely used Python materials analysis library (Ong et al., 2013). This allows seamless use of `pymatgen` functionality for pre and post-processing. Additionally, several other Materials Project (Jain et al., 2013) packages, such as `Fireworks` (Jain et al., 2015), `atomate/atomate2` (Mathew et al., 2017; Rosen et al., 2020/2022), database creation, and management tools can be leveraged alongside `smol` to include configuration thermodynamic calculations as part of more elaborate materials analysis workflows.
3. `smol` is designed to be intentionally lightweight and dependency lean by delegating much of the non-core functionality to already well-established Python packages, for example, general structure manipulations, enumeration, and linear regression. This makes `smol` easy to install, easy to use, easy to develop, easy to extend, and easy to test.

`smol` should be considerably more user and developer friendly than standalone C++ packages `ATAT` and `CASM`. In comparison to other Python implementations, in particular `icet`—which is superbly well documented and user-friendly—`smol` stands out as largely more developer friendly and easier to extend. In the context of all available packages, `smol` is geared towards efficient and open development of new methodology that is also user-friendly, thus allowing quick development-to-application turnaround time.

Formalism overview

The atomic configuration of a crystalline material can be represented by a string of occupation variables, $\sigma = (\sigma_1, \sigma_2, \dots, \sigma_N)$. Where the value of each occupation variable σ_i represents the atomic species occupying the i -th site in an N -site supercell. Accordingly, any generalized lattice model of the atomic configuration can be written as a sum of multi-site (cluster) interaction functions,

$$H(\sigma) = \sum_{S \subseteq [N]} H_S(\sigma_S) \quad (1)$$

Where $[N] = \{1, 2, \dots, N\}$ is the set of all site indices, and σ_S is the set of all occupation variables for the sites in a cluster S .

Two important considerations enable practical representations for effective fitting of applied lattice models:

1. A general procedure to construct function sets that span the function space over configurations.
2. Leveraging the symmetries of the underlying crystal structure to reduce the total function space to a subspace of symmetrically invariant functions only.

These two considerations are at the foundation of the original CE method (Sanchez et al., 1984), however, these considerations have been limited only to a small number of variations of the same formal representation. For example, consideration (1) has been limited only to a handful of different basis sets. In `smol` we have sought to implement a generalized version of the original CE method, where any symmetrically invariant lattice model is represented as,

$$H(\sigma) = \sum_{\beta} m_{\beta} J_{\beta} \Theta_{\beta}(\sigma) \quad (2)$$

where m_β are crystallographic multiplicities and J_β are expansion coefficients. The correlation functions Θ_β take as input different sets of clusters of sites S that are symmetrically equivalent under permutations corresponding to the symmetries of the underlying crystal structure's space group. The set of all correlation functions $\{\Theta_\beta\}$, unlike the classical CE method, is not limited only to those that represent a basis set but can be any complete set of functions (linearly independent or redundant) that spans the symmetry invariant function subspace over configurations σ .

Following the original CE method formalism, the correlation functions Θ_β are constructed from symmetrically adapted averages of cluster product functions,

$$\Phi_\alpha(\sigma) = \prod_{i=1}^N \phi_{\alpha_i}(\sigma_i) \quad (3)$$

$$\Theta_\beta(\sigma) = \frac{1}{|\beta|} \sum_{\alpha \in \beta} \Phi_\alpha(\sigma) \quad (4)$$

Where the site functions ϕ_{α_i} are single variable functions of each occupation variable; such that the set of all included site functions for each site i span the associated space of all possible occupations of a given site. The multi-indices α ($|\alpha| = N$), serve as indices for the site function corresponding to each site in the supercell; and the orbits β are sets of symmetrically equivalent multi-indices.

Functions represented by an appropriately truncated version of Equation 2 can be fitted to properties calculated with computationally intensive methods, such as first-principles electronic structure methods. The fitting procedure is predominantly done with linear regression using advanced regularization techniques. Subsequently, the resulting lattice model can be used in MC simulations to sample configurations for a corresponding statistical mechanical ensemble in order to efficiently compute thermodynamic functions and properties of atomic configuration.

Package overview

The `smol` Python package is deliberately designed to be easily extensible and provide useful abstractions such that new methodology will rarely need to be implemented from scratch.

Classes and functions for representation and construction of functions of configuration (i.e. defining terms in a cluster expansion) are included in the `smol.cofe` module. Notably, the following object-oriented abstractions allow flexible definitions and the ability to easily implement extensions:

- Classes and functions to define site function sets, which make up the basic building blocks for an expansion as detailed in Equation 3. The package includes functionality to generate both basis and redundant sets with any of the commonly used site function sets, (polynomial (Sanchez et al., 1984), trigonometric (Axel van de Walle, 2009), and occupancy indicator (Zhang & Sluiter, 2016)), as well as abstractions to effortlessly implement new function sets.
- Classes to represent clusters of sites S and groupings of symmetrically equivalent cluster functions to represent the terms in the sum of Equation 2. Additionally, the package includes functionality to automatically generate these objects based on a given disordered structure—that may include neutral species, ionic species with assigned oxidation states, or vacancies—by leveraging `pymatgen`'s established and flexible representations of structures and associated symmetries.
- Classes to include additional interaction terms to a CE-based lattice model to improve training convergence. Currently, the package only includes an electrostatic pair potential for ionic structures (Richards, 2017), but the concept applies to any simple interaction

model such as the reciprocal space CE constituent strain interaction (Laks et al., 1992), or any other empirical or fitted pair potential.

- Classes and functions to preprocess and generate feature matrices and fitting data corresponding to a defined set of correlation functions, datasets of relaxed structures, and computed energies from any first-principle, machine learning, or empirical potential calculations.

Additionally, the functionality to sample thermodynamic properties for a fitted lattice model under both canonical and semi-grand canonical ensembles is included in the `smol.moca` module. The `smol.moca` module includes flexible object-oriented abstractions, including the following:

- Classes and functions to quickly evaluate a cluster expansion for a given configuration and local configuration changes over a predefined supercell size and shape. Critical functions are implemented in Cython so that MC performance is not compromised.
- Classes to implement complex MC algorithms. The different components of MC are implemented as independent objects and utilities, that include classes to define configuration transition proposals, statistical ensembles, sampled value traces, and various Monte Carlo algorithm kernels. This enables customization of MC sampling methods, ensembles, and computed properties without the need to re-write the sample generation, saving, and streaming to file functionality.

All classes and functions included in `smol` are thoroughly documented and several usage examples are available in the documentation.

Acknowledgements

The development of `smol` was primarily funded by the U.S. Department of Energy, Office of Science, Office of Basic Energy Sciences, Materials Sciences and Engineering Division under Contract No. DE-AC02-05-CH11231 (Materials Project program KC23MP). L.B.L, Z.J., and T.C. also gratefully acknowledge support from the National Science Foundation Graduate Research Fellowship under Grant No. DGE 1752814 and DGE 1106400.

References

- Ångqvist, M., Muñoz, W. A., Rahm, J. M., Fransson, E., Durniak, C., Rozyczko, P., Rod, T. H., & Erhart, P. (2019). ICET – A Python Library for Constructing and Sampling Alloy Cluster Expansions. *Advanced Theory and Simulations*, 2(7), 1900015. <https://doi.org/10.1002/adts.201900015>
- Barroso-Luque, L., Yang, J. H., & Ceder, G. (2021). Sparse expansions of multicomponent oxide configuration energy using coherency and redundancy. *Physical Review B*, 104(22), 224203. <https://doi.org/10.1103/PhysRevB.104.224203>
- Chang, J. H., Kleiven, D., Melander, M., Akola, J., Garcia-Lastra, J. M., & Vegge, T. (2019). CLEASE: A versatile and user-friendly implementation of cluster expansion method. *Journal of Physics: Condensed Matter*, 31(32), 325901. <https://doi.org/10.1088/1361-648X/ab1bbc>
- Chen, T., Yang, J. H., Barroso-Luque, L., & Ceder, G. (submitted 2022). *Removing the two-phase transition in spinel LiMn_2O_4 through cation disorder*.
- Gao, M. C., Miracle, D. B., Maurice, D., Yan, X., Zhang, Y., & Hawk, J. A. (2018). High-entropy functional materials. *Journal of Materials Research*, 33(19), 3138–3155. <https://doi.org/10.1557/jmr.2018.323>
- George, E. P., Raabe, D., & Ritchie, R. O. (2019). High-entropy alloys. *Nature Reviews Materials*, 4(8), 515–534. <https://doi.org/10.1038/s41578-019-0121-4>

- Jadidi, Z., Yang, J. H., Chen, T., Barroso-Luque, L., & Ceder, G. (in prep. 2022). *Ab-initio study of short-range-ordering in vanadium-based disordered rocksalt structures*.
- Jain, A., Ong, S. P., Chen, W., Medasani, B., Qu, X., Kocher, M., Brafman, M., Petretto, G., Rignanese, G.-M., Hautier, G., Gunter, D., & Persson, K. A. (2015). FireWorks: A dynamic workflow system designed for high-throughput applications. *Concurrency and Computation: Practice and Experience*, 27(17), 5037–5059. <https://doi.org/10.1002/cpe.3505>
- Jain, A., Ong, S. P., Hautier, G., Chen, W., Richards, W. D., Dacek, S., Cholia, S., Gunter, D., Skinner, D., Ceder, G., & Persson, K. A. (2013). Commentary: The Materials Project: A materials genome approach to accelerating materials innovation. *APL Materials*, 1(1), 011002. <https://doi.org/10.1063/1.4812323>
- Laks, D. B., Ferreira, L. G., Froyen, S., & Zunger, A. (1992). Efficient cluster expansion for substitutional systems. *Physical Review B*, 46(19), 12587–12605. <https://doi.org/10.1103/PhysRevB.46.12587>
- Lun, Z., Ouyang, B., Kwon, D.-H., Ha, Y., Foley, E. E., Huang, T.-Y., Cai, Z., Kim, H., Balasubramanian, M., Sun, Y., Huang, J., Tian, Y., Kim, H., McCloskey, B. D., Yang, W., Clément, R. J., Ji, H., & Ceder, G. (2020). Cation-disordered rocksalt-type high-entropy cathodes for Li-ion batteries. *Nature Materials*, 1–8. <https://doi.org/10.1038/s41563-020-00816-0>
- Mathew, K., Montoya, J. H., Faghaninia, A., Dwarakanath, S., Aykol, M., Tang, H., Chu, I., Smidt, T., Bocklund, B., Horton, M., Dagdelen, J., Wood, B., Liu, Z.-K., Neaton, J., Ong, S. P., Persson, K., & Jain, A. (2017). Atomate: A high-level interface to generate, execute, and analyze computational materials science workflows. *Computational Materials Science*, 139, 140–152. <https://doi.org/10.1016/j.commatsci.2017.07.030>
- Ong, S. P., Richards, W. D., Jain, A., Hautier, G., Kocher, M., Cholia, S., Gunter, D., Chevrier, V. L., Persson, K. A., & Ceder, G. (2013). Python Materials Genomics (pymatgen): A robust, open-source python library for materials analysis. *Computational Materials Science*, 68, 314–319. <https://doi.org/10.1016/j.commatsci.2012.10.028>
- Richards, W. D. (William. D. (2017). *Ab initio investigations of solid electrolytes for lithium- and Sodium-ion batteries* [Thesis, Massachusetts Institute of Technology]. <https://dspace.mit.edu/handle/1721.1/108967>
- Rosen, A., Shen, J.-X., & Riebesell, J. (2022). Atomate2. Materials Project. <https://github.com/materialsproject/atomate2> (Original work published 2020)
- Sanchez, J. M., Ducastelle, F., & Gratias, D. (1984). Generalized cluster description of multicomponent systems. *Physica A: Statistical Mechanics and Its Applications*, 128(1), 334–350. [https://doi.org/10.1016/0378-4371\(84\)90096-7](https://doi.org/10.1016/0378-4371(84)90096-7)
- Thomas, J. C., Puchala, B., Goiri, J., Nataraja, A., & Van der Ven, A. (2022). Prisms-center/CASMcode. PRISMS Center. <https://github.com/prisms-center/CASMcode> (Original work published 2015)
- van de Walle, Axel. (2009). Multicomponent multisublattice alloys, nonconfigurational entropy and other additions to the Alloy Theoretic Automated Toolkit. *Calphad*, 33(2), 266–278. <https://doi.org/10.1016/j.calphad.2008.12.005>
- van de Walle, A., Asta, M., & Ceder, G. (2002). The alloy theoretic automated toolkit: A user guide. *Calphad*, 26(4), 539–553. [https://doi.org/10.1016/S0364-5916\(02\)80006-2](https://doi.org/10.1016/S0364-5916(02)80006-2)
- Van der Ven, A., Thomas, J. c., Puchala, B., & Natarajan, A. r. (2018). First-Principles Statistical Mechanics of Multicomponent Crystals. *Annual Review of Materials Research*, 48(1), 27–55. <https://doi.org/10.1146/annurev-matsci-070317-124443>
- Yang, J. H., & Ceder, G. (in prep. 2022). *Structural understanding of partially-disordered spinel materials with high rate performance*.

- Yang, J. H., Chen, T., Barroso-Luque, L., Jadidi, Z., & Ceder, G. (2022). Approaches for handling high-dimensional cluster expansions of ionic systems. *Npj Computational Materials*, 8(1), 1–11. <https://doi.org/10.1038/s41524-022-00818-3>
- Zhang, X., & Sluiter, M. H. F. (2016). Cluster Expansions for Thermodynamics and Kinetics of Multicomponent Alloys. *Journal of Phase Equilibria and Diffusion*, 37(1), 44–52. <https://doi.org/10.1007/s11669-015-0427-x>