CrySPR: A Python interface for implementation of crystal structure pre-relaxation and prediction using machine-learning interatomic potentials

Wei Nong, Ruiming Zhu, Kedar Hippalgaonkar
School of Materials Science and Engineering, Nanyang Technological University, Singapore 639798, Singapore

Summary

The functional properties of crystalline inorganic materials in a variety of applications including, but not limited to, catalysts, batteries, solar cells, electronics, fundamentally depend on their crystal structures. Discovery of novel materials could be transformative for these fields.[1] In the past few decades, the computational science community has developed crystal structure prediction (CSP) methods with the goal to find the exact symmetry-constrained atomic arrangements in the periodic unit cell, which are globally and/or locally energetically favorable: finding the globally minimal or locally minimal crystal structure for a given chemical formula. The implementation of CSP typically involves an iteration procedure with at least two components: the sampling of the potential energy surface (PES) for generating raw/unrelaxed crystal structures, and the subsequent local energy minimization of generated structures. The latter part is typically carried out through computationally expensive density functional (DFT) calculations. A non-exhaustive but representative list of available CSP codes includes USPEX[2], CALYPSO[3], AIRSS[4], XtalOpt[5,6], IM2ODE[7]; due to the nature of DFT calculations, this CSP process can be very time-consuming. Recent rapid advances of pre-trained machine-learning interatomic potentials (ML-IAPs) based on data from DFT calculations, such as, M3GNet[8], CHGNet[9] and MACE[10] (amongst others) have significantly accelerated the process of local energy minimization but have not thoroughly been tested on CSP tasks. The realization of local energy minimization using ML-IAPs, referred to as pre-relaxation when compared with using DFT calculations, plays a critical role in the CSP implementation. We present here, CrySPR, which stands for **Cry**stal **S**tructure **P**re-**R**elaxation and **PR**ediction, which is specifically designed to serve as a Python package that provides user-friendly application programming interfaces (APIs), functionalities and utilities for crystal structure generation, pre-relaxation of structures using ML-IAPs and structure prediction. The codes are open-source and have been released to the Python Package Index (PyPI).

Statement of need

As far as we are aware, there is no available and/or easily accessible Python code tailored for "all-in-one" structure generation, pre-relaxation and prediction using contemporary ML-IAPs. CrySPR provides three key features:

1. Crystal structure generation for a given chemical formula, which crystallographically follows the space group symmetry of crystals, either by performing random match between Wyckoff positions (along with multiplicities) and the allowed elements and corresponding numbers of atoms, or by enumerating all possible matched cases. The match in this context

is also known as compatibility check.[11]

2. Automatic workflow for stepwise structure pre-relaxation using ML-IAPs.
3. Obtaining (a) the formation energy and (b) energy above the convex hull, from a given reference materials database (illustrated here with the Materials Project), and plotting the phase diagram for predicted materials.

These features are built mainly based on PyXtal[11], ASE[12] and pymatgen[13]. There are also a set of utilities for result visualization and analysis (formation energy calculation, energy data plotting). In this context, CrySPR can play a vital role by providing an efficient framework for structure optimization, prediction and validation, especially in the era of generative algorithms such as variational autoencoders (VAEs)[14,15], generative adversarial networks (GANs)[16], diffusion models[17,18], transformers[19,20], flow-based approaches[21], producing a large number of possible crystal structures that need to be tested for stability and relaxation rapidly.

CrySPR

CrySPR is a modular Python package, which includes three main submodules: calculators, optimization, and utils. **Figure 1** demonstrates the workflow for the CSP task in CrySPR, users are allowed to input several constraints for search space, including chemical formula, the number of formulae per conventional unit cell (Z), space groups (SPG), and some extra constraints including the number of sample structures per SPG for each Z, and Wyckoff positions, etc. The input information then is then processed by CrySPR then used by PyXtal to generate initial unrelaxed crystal structures, which subsequently encounter local structure optimization using a local optimization algorithm and ML-IAP in the ASE framework, from which the final total energies (E) are determined. The package has a built-in function for a developed stepwise strategy. It is specially designed for ML-IAPs and includes two fully symmetry-compliant steps for structure relaxation with an improved accuracy: firstly cell-fixed relaxation and subsequently cell-varying relaxation. The package can easily adopt any ML-IAP that is formulated in the form of ASE calculator class. CrySPR is expandable with more features and functionalities as it evolves, e.g., adapting more advanced and accurate ML-IAPs, comprehensive static reference data of the potential energies, optimization algorithms for efficient PES sampling, and multi-objective oriented CSP, etc.

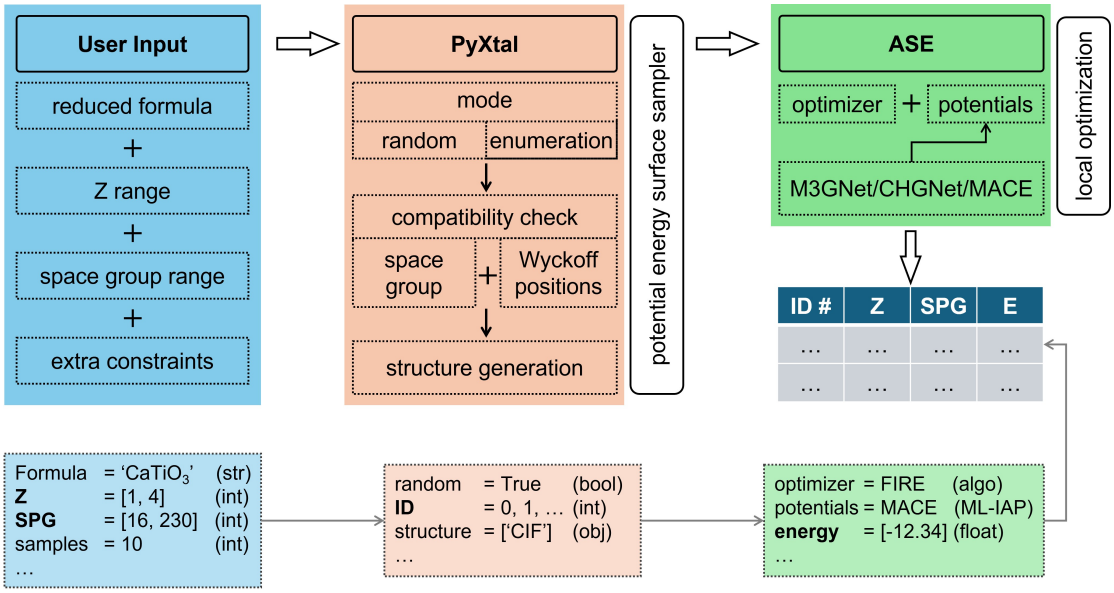

Figure 1: Schematic illustration of the workflow in CrySPR. An example of data types is given.

Figure 2 shows the prediction results that employ one of the ML-IAPs, CHGNet, as the inter-atomic potentials. This demo uses a built-in function, `random_predict`, to perform CSP for chemical spaces with reduced formulae of (A) $CaTiO_3$ and (B) $MgAl_2O_4$, respectively. For $CaTiO_3$, the SPGs are constrained from No. 16 to 230, and Z values from 1 to 4, while for $MgAl_2O_4$, cubic space groups (No. 195 to 230) and Z values from 1 to 10 are set. In **Figure 2**A some representative crystal structures are illustrated with the energy above the hull labeled, and those experimental phases are underscored, while in **Figure 2**B the total energy per atom is shown. In this demo, the ground-state structure of $CaTiO_3$ in tetragonal perovskite phase and $MgAl_2O_4$ in cubic spinel phase are successfully predicted.

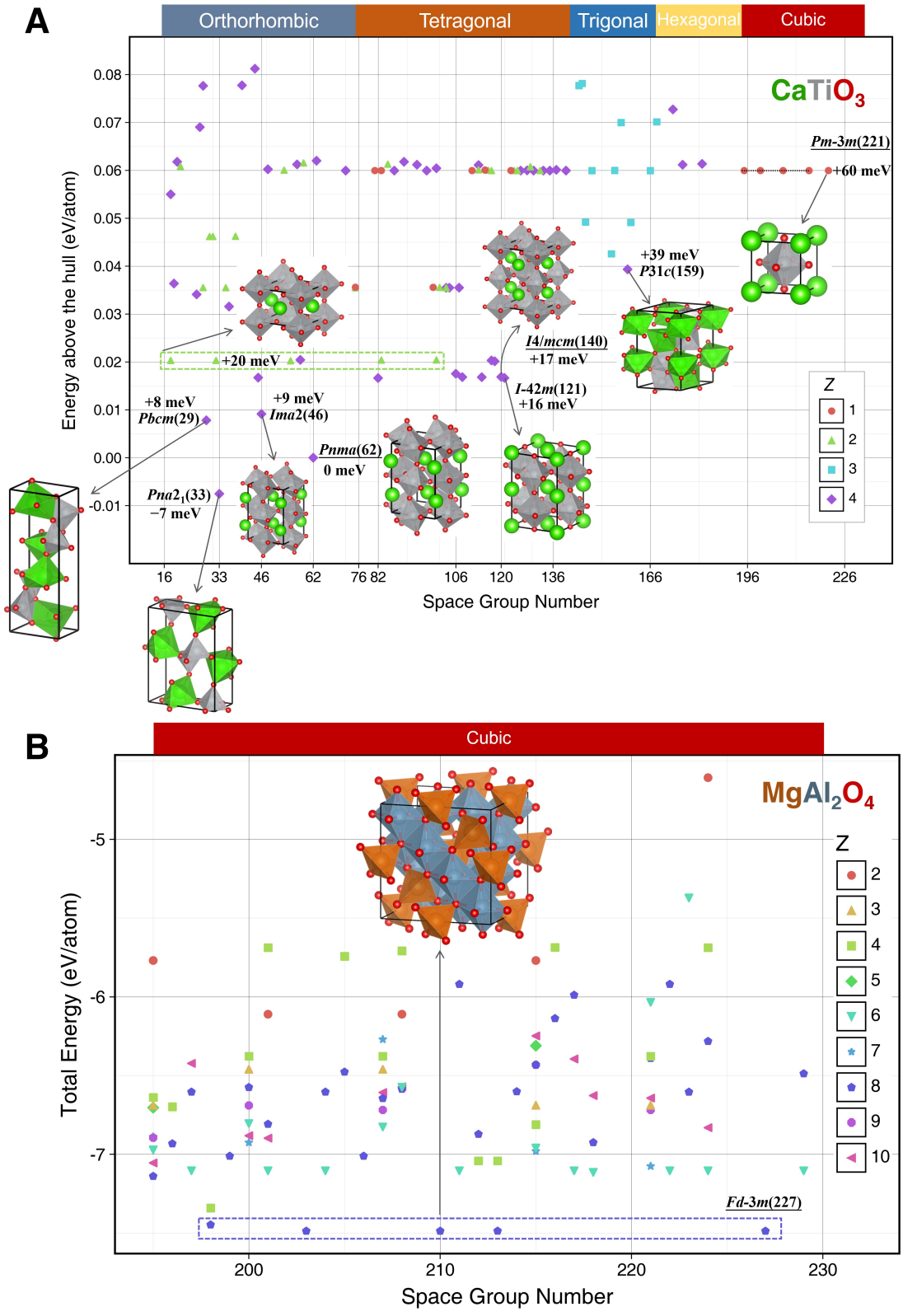

Figure 2: Crystal structure prediction for (A) the perovskite $CaTiO_3$ and (B) the spinel $MgAl_2O_4$.

Acknowledgements

K.H. acknowledges funding from the Accelerated Materials Development for Manufacturing Program at A*STAR via the AME Programmatic Fund by the Agency for Science, Technology and Research under Grant No. A1898b0043. K.H. also acknowledges funding from the NRF Fellowship NRF-NRFF13-2021-0011.

References

1. Oganov, A.R., Pickard, C.J., Zhu, Q., and Needs, R.J. (2019). Structure prediction drives materials discovery. Nature Reviews Materials *4*, 331-348. https://doi.org/10.1038/s41578-019-0101-8.
2. Glass, C.W., Oganov, A.R., and Hansen, N. (2006). USPEX—Evolutionary crystal structure prediction. Computer Physics Communications *175*, 713-720. https://doi.org/10.1016/j.cpc.2006.07.020.
3. Wang, Y., Lv, J., Zhu, L., and Ma, Y. (2012). CALYPSO: A method for crystal structure prediction. Computer Physics Communications *183*, 2063-2070. https://doi.org/10.1016/j.cpc.2012.05.008.
4. Pickard, C.J., and Needs, R.J. (2011). Ab initio random structure searching. Journal of Physics: Condensed Matter *23*, 053201. https://doi.org/10.1088/0953-8984/23/5/053201.
5. Lonie, D.C., and Zurek, E. (2011). XtalOpt: An open-source evolutionary algorithm for crystal structure prediction. Computer Physics Communications *182*, 372-387. https://doi.org/10.1016/j.cpc.2010.07.048.
6. Hajinazar, S., and Zurek, E. (2024). XtalOpt Version 13: Multi-Objective Evolutionary Search for Novel Functional Materials. arXiv preprint. https://doi.org/10.48550/arXiv.2405.02138.
7. Zhang, Y.-Y., Gao, W., Chen, S., Xiang, H., and Gong, X.-G. (2015). Inverse design of materials by multi-objective differential evolution. Computational Materials Science *98*, 51-55. https://doi.org/10.1016/j.commatsci.2014.10.054.
8. Chen, C., and Ong, S.P. (2022). A universal graph deep learning interatomic potential for the periodic table. Nature Computational Science *2*, 718-728. https://doi.org/10.1038/s43588-022-00349-3.
9. Deng, B., Zhong, P., Jun, K., Riebesell, J., Han, K., Bartel, C.J., and Ceder, G. (2023). CHGNet as a pretrained universal neural network potential for charge-informed atomistic modelling. Nature Machine Intelligence *5*, 1031-1041. https://doi.org/10.1038/s42256-023-00716-3.
10. Batatia, I., Kovacs, D.P., Simm, G.N.C., Ortner, C., and Csanyi, G. (2022). MACE: Higher Order Equivariant Message Passing Neural Networks for Fast and Accurate Force Fields. Advances in Neural Information Processing Systems. https://openreview.net/forum?id=YPpSngE-ZU.
11. Fredericks, S., Parrish, K., Sayre, D., and Zhu, Q. (2021). PyXtal: A Python library for crystal structure generation and symmetry analysis. Computer Physics Communications *261*, 107810. https://doi.org/10.1016/j.cpc.2020.107810.
12. Hjorth Larsen, A., Jørgen Mortensen, J., Blomqvist, J., Castelli, I.E., Christensen, R., Dułak, M., Friis, J., Groves, M.N., Hammer, B., Hargus, C., et al. (2017). The atomic simulation environment—a Python library for working with atoms. Journal of Physics: Condensed Matter

29, 273002. https://doi.org/10.1088/1361-648X/aa680e.

13. Ong, S.P., Richards, W.D., Jain, A., Hautier, G., Kocher, M., Cholia, S., Gunter, D., Chevrier, V.L., Persson, K.A., and 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.
14. Zhu, R., Nong, W., Yamazaki, S., and Hippalgaonkar, K. (2024). WyCryst: Wyckoff inorganic crystal generator framework. Matter *7*, 1-20. https://doi.org/10.1016/j.matt.2024.05.042.
15. Ren, Z., Tian, S.I.P., Noh, J., Oviedo, F., Xing, G., Li, J., Liang, Q., Zhu, R., Aberle, A.G., Sun, S., et al. (2022). An invertible crystallographic representation for general inverse design of inorganic crystals with targeted properties. Matter *5*, 314-335. 10.1016/j.matt.2021.11.032.
16. Nouira, A., Sokolovska, N., and Crivello, J.-C. (2018). CrystalGAN: Learning to Discover Crystallographic Structures with Generative Adversarial Networks. arXiv preprint. https://doi.org/10.48550/arXiv.1810.11203.
17. Xie, T., Fu, X., Ganea, O.-E., Barzilay, R., and Jaakkola, T. (2021). Crystal diffusion variational autoencoder for periodic material generation. arXiv preprint. https://doi.org/10.48550/arXiv.2110.06197.
18. Jiao, R., Huang, W., Liu, Y., Zhao, D., and Liu, Y. (2024). Space Group Constrained Crystal Generation. International Conference on Learning Representations. https://openreview.net/forum?id=jkvZ7v4OmP.
19. Antunes, L.M., Butler, K.T., and Grau-Crespo, R. (2023). Crystal structure generation with autoregressive large language modeling. arXiv preprint. https://doi.org/10.48550/arXiv.2307.04340.
20. Kazeev, N., Ustyuzhanin, A., Romanov, I., Zhu, R., Nong, W., Yamazaki, S., and Hippalgaonkar, K. (2024). WyckoffTransformer: Autoregressive Generation of Crystals. Artificial Intelligence for Advanced Materials. https://phantomsfoundation.com/AI4AM/2024/Abstracts/AI4AM2024_Kazeev_Nikita_78.pdf.
21. Miller, B.K., Chen, R.T.Q., Sriram, A., and Wood, B.M. (2024). FlowMM: Generating Materials with Riemannian Flow Matching. arXiv preprint. https://doi.org/10.48550/arXiv.2406.04713.