

graphenv: a Python library for reinforcement learning on graph search spaces

David Biagioni ², Charles Edison Tripp ², Struan Clark ², Dmitry Duplyakin ², Jeffrey Law ¹, and Peter C. St. John ¹✉

¹ Biosciences Center, National Renewable Energy Laboratory, Golden CO 80401, USA ² Computational Sciences Center, National Renewable Energy Laboratory, Golden CO 80401, USA ✉ Corresponding author

DOI: [10.21105/joss.04621](https://doi.org/10.21105/joss.04621)

Software

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

Editor: Øystein Sørensen  

Reviewers:

- [@iammix](#)
- [@vwxyzjn](#)
- [@Viech](#)
- [@idoby](#)

Submitted: 21 July 2022

Published: 05 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](https://creativecommons.org/licenses/by/4.0/)).

Summary

Many important and challenging problems in combinatorial optimization (CO) can be expressed as graph search problems, in which graph vertices represent full or partial solutions and edges represent decisions that connect them. Graph structure not only introduces strong *relational inductive biases* for learning (Battaglia et al., 2018) – in this context, by providing a way to explicitly model the value of transitioning (along edges) between one search state (vertex) and the next – but lends itself to problems both with and without clearly defined algebraic structure. For example, classic CO problems on graphs such as the Traveling Salesman Problem (TSP) can be expressed as either pure graph search or integer programs. Other problems, however, such as molecular optimization, do not have concise algebraic formulations and yet are readily implemented as a graph search (V. et al., 2022; Zhou et al., 2019). Such “model-free” problems constitute a large fraction of modern reinforcement learning (RL) research owing to the fact that it is often much easier to write a forward simulation that expresses all of the state transitions and rewards, than to write down the precise mathematical expression of the full optimization problem. In the case of molecular optimization, for example, one can use domain knowledge alongside existing software libraries to model the effect of adding a single bond or atom to an existing but incomplete molecule, and let the RL algorithm build a model of how good a given decision is by “experiencing” the simulated environment many times through. In contrast, a model-based mathematical formulation that fully expresses all the chemical and physical constraints is intractable.

In recent years, RL has emerged as an effective paradigm for optimizing searches over graphs and led to state-of-the-art heuristics for games like Go and chess, as well as for classical CO problems such as the TSP. This combination of graph search and RL, while powerful, requires non-trivial software to execute, especially when combining advanced state representations such as Graph Neural Networks (GNN) with scalable RL algorithms.

Statement of need

The graphenv Python library is designed to 1) make graph search problems more readily expressible as RL problems via an extension of the OpenAI gym API (Brockman et al., 2016) while 2) enabling their solution via scalable learning algorithms in the popular RLlib library (Liang et al., 2018). The intended audience consists of researchers working on graph search problems that are amenable to a reinforcement learning formulation, broadly described as “learning to optimize”. This includes those working on classical combinatorial optimization problems such as the TSP, as well as problems that do not have a clear algebraic expression but where the environment dynamics can be simulated, for instance, molecular design.

RLlib provides convenient, out-of-the-box support for several features that enable the application

of RL to complex search problems (e.g., parametrically-defined actions and invalid action masking). However, native support for action spaces where the action *choices* change for each state is challenging to implement in a computationally efficient fashion. The graphenv library provides utility classes that simplify the flattening and masking of action observations for choosing from a set of successor states at every node in a graph search.

Related software efforts have addressed parts of the above need. OpenGraphGym (Zheng et al., 2020) implements RL-based strategies for common graph optimization challenges such as minimum vertex cover or maximum cut, but does not interface with external RL libraries and has minimal documentation. Ecole (Prouvost et al., 2020) provides an OpenAI-like gym environment for combinatorial optimization, but intends to operate in concert with traditional mixed integer solvers rather than directly exposing the environment to an RL agent.

Examples of usage

This package is a generalization of methods employed in the optimization of molecular structure for energy storage applications, funded by US Department of Energy (DOE)'s Advanced Research Projects Agency - Energy (V. et al., 2022). Specifically, this package enables optimization against a surrogate objective function based on high-throughput density functional theory calculations (S. V. et al., 2021; St. John, Guan, Kim, Kim, et al., 2020; St. John, Guan, Kim, Etz, et al., 2020) by considering molecule selection as an iterative process of adding atoms and bonds, transforming the optimization into a rooted search over a directed, acyclic graph. Ongoing work is leveraging this library to enable similar optimization for inorganic crystal structures, again using a surrogate objective function based on high-throughput quantum mechanical calculations (Pandey et al., 2021).

Acknowledgements

This work was authored by the National Renewable Energy Laboratory, operated by Alliance for Sustainable Energy, LLC, for the US Department of Energy (DOE) under Contract No. DE-AC36-08GO28308. The information, data, or work presented herein was funded in part by the Advanced Research Projects Agency-Energy (ARPA-E), U.S. Department of Energy, under Award Number DE-AR0001205. The views and opinions of authors expressed herein do not necessarily state or reflect those of the United States Government or any agency thereof. The US Government retains and the publisher, by accepting the article for publication, acknowledges that the US Government retains a nonexclusive, paid-up, irrevocable, worldwide license to publish or reproduce the published form of this work or allow others to do so, for US Government purposes.

References

- Battaglia, P. W., Hamrick, J. B., Bapst, V., Sanchez-Gonzalez, A., Zambaldi, V., Malinowski, M., Tacchetti, A., Raposo, D., Santoro, A., Faulkner, R., & others. (2018). Relational inductive biases, deep learning, and graph networks. *arXiv Preprint arXiv:1806.01261*. <https://doi.org/10.48550/arXiv.1806.01261>
- Brockman, G., Cheung, V., Pettersson, L., Schneider, J., Schulman, J., Tang, J., & Zaremba, W. (2016). Openai gym. *arXiv Preprint arXiv:1606.01540*. <https://doi.org/10.48550/arXiv.1606.01540>
- Liang, E., Liaw, R., Nishihara, R., Moritz, P., Fox, R., Goldberg, K., Gonzalez, J., Jordan, M., & Stoica, I. (2018). RLlib: Abstractions for distributed reinforcement learning. *International Conference on Machine Learning*, 3053–3062. <https://doi.org/10.48550/arXiv.1712.09381>

- Pandey, S., Qu, J., Stevanović, V., St. John, P., & Gorai, P. (2021). Predicting energy and stability of known and hypothetical crystals using graph neural network. *Patterns*, 2(11), 100361. <https://doi.org/10.1016/j.patter.2021.100361>
- Prouvost, A., Dumouchelle, J., Scavuzzo, L., Gasse, M., Chételat, D., & Lodi, A. (2020). Ecole: A gym-like library for machine learning in combinatorial optimization solvers. *Learning Meets Combinatorial Algorithms at NeurIPS2020*. <https://doi.org/10.48550/arXiv.2011.06069>
- S. V., S. S., St. John, P. C., & Paton, R. S. (2021). A quantitative metric for organic radical stability and persistence using thermodynamic and kinetic features. *Chemical Science*, 12(39), 13158–13166. <https://doi.org/10.1039/d1sc02770k>
- St. John, P. C., Guan, Y., Kim, Y., Etz, B. D., Kim, S., & Paton, R. S. (2020). Quantum chemical calculations for over 200,000 organic radical species and 40,000 associated closed-shell molecules. *Scientific Data*, 7(1). <https://doi.org/10.1038/s41597-020-00588-x>
- St. John, P. C., Guan, Y., Kim, Y., Kim, S., & Paton, R. S. (2020). Prediction of organic homolytic bond dissociation enthalpies at near chemical accuracy with sub-second computational cost. *Nature Communications*, 11(1). <https://doi.org/10.1038/s41467-020-16201-z>
- V., S. S. S., Law, J. N., Tripp, C. E., Duplyakin, D., Skordilis, E., Biagioni, D., Paton, R. S., & John, P. C. St. (2022). Multi-objective goal-directed optimization of de novo stable organic radicals for aqueous redox flow batteries. *Nature Machine Intelligence*. <https://doi.org/10.1038/s42256-022-00506-3>
- Zheng, W., Wang, D., & Song, F. (2020). OpenGraphGym: A parallel reinforcement learning framework for graph optimization problems. In *Lecture notes in computer science* (pp. 439–452). Springer International Publishing. https://doi.org/10.1007/978-3-030-50426-7_33
- Zhou, Z., Kearnes, S., Li, L., Zare, R. N., & Riley, P. (2019). Optimization of molecules via deep reinforcement learning. *Scientific Reports*, 9(1). <https://doi.org/10.1038/s41598-019-47148-x>