BEHAVIORAL CLONING FOR CRYSTAL DESIGN

Prashant Govindarajan[*1,4], **Santiago Miret**[3], **Jarrid Rector-Brooks**[1,2], **Mariano Phielipp**[3], **Janarthanan Rajendran**[1,2], **Sarath Chandar**[1,4]
[1]Mila, [2]Université de Montréal, [3]Intel, [4]Polytechnique Montréal

ABSTRACT

Solid-state materials, which are made up of periodic 3D crystal structures, are particularly useful for a variety of real-world applications such as batteries, fuel cells and catalytic materials. Designing solid-state materials, especially in a robust and automated fashion, remains an ongoing challenge. To further the automated design of crystalline materials, we propose a method to learn to design valid crystal structures given a crystal skeleton. By incorporating Euclidean equivariance into a policy network, we pose the problem of designing new crystals as a sequential prediction task suited for imitation learning. At each step, given an incomplete graph of a crystal skeleton, an agent assigns an element to a specific node. We adopt a behavioral cloning strategy to train the policy network on data consisting of curated trajectories generated from known crystals.

1 INTRODUCTION

Crystalline substances form a major class of naturally occurring materials, from salts and precious stones to metallic objects. Despite recent advancements in simulation algorithms for computational solid-state chemistry (Corpinot & Bučar, 2019), designing novel crystalline materials with desired structural and chemical properties remains a cumbersome and specialized task. The highly relevant applications of solid-state materials, including energy-efficient semiconductors, biosensors, and technologies to mitigate climate change, make designing automated discovery methods even more impactful. The past few years have witnessed significant progress in leveraging artificial intelligence for the design of structural entities such as small molecules and polymers (Chen et al., 2020; Cheng et al., 2022; Adams & Coley, 2022; Gebauer et al., 2019). Recent works have attempted to use generative models and reinforcement learning for material design (Xie et al., 2021; Pan et al., 2022).

Solid-state crystals consist of unit cells that repeat infinitely in a periodic manner. By considering crystal design as a sequential prediction task, we can train an agent to construct crystal structures based on partially constructed crystals in a unit cell. At each step of the sequence, the agent can predict a missing atom (action) in this partially completed structure (state). In a sequential learning framework, reinforcement learning (RL) methods can help in learning an optimal policy by exploring the vast chemical space with the aim of maximizing a reward function (Arulkumaran et al., 2017). However, this is an extremely challenging task for online RL techniques considering the discrete and high dimensional action space: With more than 100 elements in the periodic table, a simple cubic structure with 8 atoms leads to combinatorial search space of 10^{16}, which only increases with more complex crystal skeletons. This design challenge is further complicated by the fact that a large fraction of those trajectories leads to invalid crystals, which do not provide useful learning signals. On the other hand, offline RL and imitation learning methods can learn behaviors from large databases of transitions obtained through experiments and simulations. In particular, behavioral cloning (BC), learns to imitate an expert behavior policy from an existing dataset without the help of a reward function (Kanervisto et al., 2020).

We train a BC agent (which is known to be successful when trained on good-quality datasets (Torabi et al., 2018)) on a sizeable set of transitions generated from an existing material dataset consisting of state-action pairs. We model the policy using neural networks and maximize its likelihood. Precisely, given a graph of a partially filled crystal (skeleton) containing positional and

[*] Corresponding author: prashant.govindarajan@mila.quebec

lattice information, the policy predicts the type of atom to be placed in a given location (Figure 1a). The primary goal of this work is to examine whether BC can be used to sequentially construct crystals. We use ablation studies to compare the performance of the agent by varying the size of the data and the model.

2 Methods

2.1 Dataset

In this work, we use the Perov-5 database (Castelli et al., 2012), which consists of 18,928 perovskite materials that have the same structure but different compositions. All crystals in the dataset have 5 atoms in their unit cells, with 56 unique elements altogether. The generic chemical formula for perovskites is ABX_3 (e.g. $CaTiO_3$) and their unit cells generally have a cubic structure. As explained in Section 2.3, we use this dataset to generate trajectories of state-action transitions of partially constructed crystal skeletons and the corresponding atom type at a given location.

2.2 Crystal Representation

We represent the unit cell of a crystal as a graph $\mathcal{G} = (V, E)$, where the nodes V with $|V| = N$ are the individual atoms of the unit cell and neighboring atoms form an edge E ($|E| = M$). For the edges, we applied the CrystalNN method (Pan et al., 2021), which determines the neighbors of nodes in a crystal graph. Each node $v \in V$ consists of two features - the first represents the one-hot encoding of the atom type ($\boldsymbol{h_v}$) and the second denotes the position of the atom in the 3-dimensional space ($\boldsymbol{x_v}$). $\boldsymbol{h_v}$ has an extra dimension to indicate whether node v is already filled with an atom or not. The global features of the crystal are the lengths ($\boldsymbol{l}$) and angles ($\boldsymbol{\alpha}$) of the 3D lattice. An additional graph feature ($\boldsymbol{f}$) that indicates the node that the agent has to focus on for prediction at a particular step is also given as input to the policy network.

2.3 Trajectory Generation

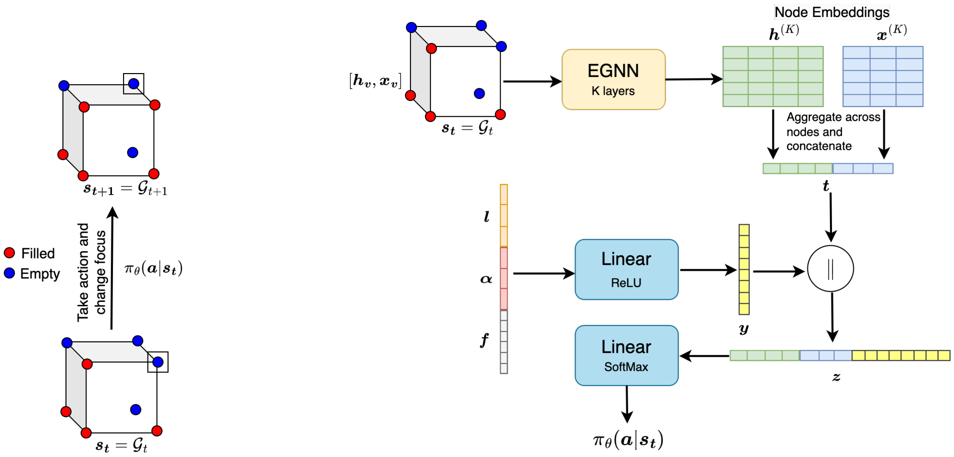
We generated a dataset of trajectories consisting of sequential state-action transitions from Perov-5 crystals, which we utilized to train the BC agent. Concretely, the trajectories consist of state-action pairs obtained from a deterministic policy $\pi_\beta(\boldsymbol{a}|\boldsymbol{s})$, which are actions that correspond to the true identity of the atom at a given position in an incomplete crystal. Here, states are crystal skeleton graphs and actions are one of the 56 elements in the action space $\mathcal{A}$. The initial state $\boldsymbol{s_0}$ is a graph $\mathcal{G}_0$ of a crystal skeleton (containing lattice and positional information) with all the nodes being unfilled, and the focus feature $\boldsymbol{f}$ indicating the first node to focus for atom type prediction. Next, we leverage a breadth-first search traversals of the graph from multiple source nodes to determine the order of nodes to focus on in the trajectory. In this manner, we can obtain multiple trajectories for each crystal, and we generated up to 5 trajectories (with different source nodes) per crystal for our experiments.

2.4 Policy Network

The policy network $\pi_\theta(\boldsymbol{a}|\boldsymbol{s})$ is used to transform a given state into an effective representation and predict the action. In this study, we use an equivariant graph neural network, or EGNN (Satorras et al., 2021) backbone which consists of message passing layers that respect equivariance to the Euclidean group (group of translations, rotations, and reflections). The embeddings obtained from the EGNN were aggregated across the nodes of the graph to obtain graph-level embeddings. Further, each of the graph-level features, i.e., the lengths ($\boldsymbol{l}$), angles ($\boldsymbol{\alpha}$), and focus ($\boldsymbol{f}$) were passed through a linear network and concatenated with the outputs of EGNN, before passing through an output layer for action prediction (Figure 1b).

Algorithm 1 Training Behavioral Cloning Agent

```
Generate data $\mathcal{D}$ of size $N_{\mathcal{D}}$ consisting of transitions $(\boldsymbol{s_i}, \boldsymbol{a_i})$ using existing crystals
Initialize parameters of policy network $\pi_{\theta_0}, t = 0$
for $j = 1$ to max_epochs do
    for $i = 1$ to $N_{\mathcal{D}}$ do
        Compute $\hat{\boldsymbol{a}}_{\boldsymbol{i}} = \pi_{\theta_t}(\boldsymbol{s_i})$
        Calculate loss $L(\theta_t) = \text{CrossEntropy}(\hat{\boldsymbol{a}}_{\boldsymbol{i}}, \boldsymbol{a_i})$
        Compute gradients and backpropogate: $\theta_{t+1} \leftarrow \theta_t - \alpha\nabla L(\theta_t),\ \alpha$ is the learning rate
        $t \leftarrow t + 1$
    end for
end for
```


$\boldsymbol{s_{t+1}} = \mathcal{G}_{t+1}$
Filled
Empty
Take action and change focus
$\pi_\theta(\boldsymbol{a}|\boldsymbol{s_t})$
$\boldsymbol{s_t} = \mathcal{G}_t$
$[\boldsymbol{h_v}, \boldsymbol{x_v}]$
$\boldsymbol{s_t} = \mathcal{G}_t$
EGNN
K layers
Node Embeddings
$\boldsymbol{h}^{(K)}$
$\boldsymbol{x}^{(K)}$
Aggregate across nodes and concatenate
$\boldsymbol{t}$
$\boldsymbol{l}$
$\boldsymbol{\alpha}$
$\boldsymbol{f}$
Linear
ReLU
$\boldsymbol{y}$
$\|$
Linear
SoftMax
$\boldsymbol{z}$
$\pi_\theta(\boldsymbol{a}|\boldsymbol{s_t})$

(a) Predicting the atom type at a given node of a crystal skeleton graph using learned policy π_θ

(b) Policy network architecture using E(n) Equivariant GNN (EGNN). Concretely, the policy applies the following operations: 1) $\boldsymbol{h_v}^{(K)}, \boldsymbol{x_v}^{(K)} = EGNN(\boldsymbol{s}, \boldsymbol{h_v}, \boldsymbol{x_v})$, 2) $\boldsymbol{t} = AGG_v\left(\left[\boldsymbol{h_v}^{(K)}, \boldsymbol{x_v}^{(K)}\right]\right)$, 3) $\boldsymbol{y} = \sigma_1([\boldsymbol{l} \parallel \boldsymbol{\alpha} \parallel \boldsymbol{f}])$, 4) $\hat{\boldsymbol{a}} = \sigma_2([\boldsymbol{t} \parallel \boldsymbol{y}])$, where σ_1, σ_2 are linear layers.

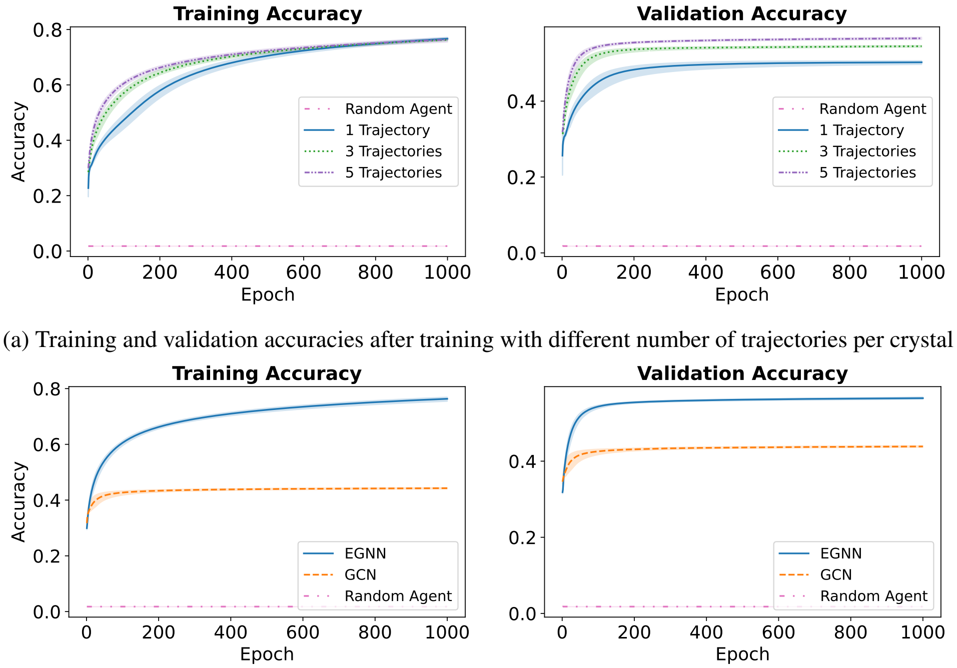
Figure 1: Procedure for sequentially constructing crystals via behavioral cloning with trajectory described in a) and the policy network described in b).

3 Experiments

3.1 Training and Experiments

For our first set of experiments, we trained the policy network on the dataset by varying the number of trajectories per crystal, and the learning performances were compared to a random agent. The models were trained for 1000 epochs with the cross-entropy loss and accuracy[1] (fraction of correctly predicted atoms) being recorded for the training (11,356 crystals) and validation set (3,787 crystals), as shown in Figure 2a. Next, we ran experiments by replacing EGNN with a Graph Convolutional Network (GCN) (Kipf & Welling, 2016) to determine if Euclidean equivariance plays a role in the performance of the agent. The accuracy curves for both experiments are shown in Figure 2.

[1] For the training and validation set, the average accuracy was computed by determining the average number of correctly predicted actions. For the test set, the accuracy was computed after sequentially reconstructing the crystal starting from the initial state using the learned policy.



(a) Training and validation accuracies after training with different number of trajectories per crystal

(b) Training and validation accuracies for EGNN and GCN in the policy network (5 trajectories)

Figure 2: Results for crystal structure building using behavioral cloning (averaged over 5 runs). From the above data, we observe: 1) BC policy clearly outperforms a random policy; 2. Training a policy on more trajectories generally performs better; 3. EGNN outperforms GCN indicating the benefits of incorporating E(n) equivariance into the policy network architecture.

3.2 Evaluation

We use the learned policy π_θ to sequentially fill nodes with atoms for the crystal skeleton graphs in the test dataset, i.e., starting from the initial state s_0 and given the order of traversal, the atoms are sequentially filled in the crystal. The generated crystals were evaluated based on the test accuracy (fraction of correctly predicted atom types for the crystal skeletons) and compositional validity - composition is valid if the crystal's overall charge is neutral, as computed by SMACT (Davies et al. (2019)). Table 1 shows the performance of our BC agent trained on datasets containing different number of trajectories for each crystal, and also with graph convolutional network (GCN) instead of EGNN, compared with a random agent. Our models perform better than the random baseline. The models trained on more samples appeared to perform better on test data, indicating that having more transitions helps an agent to better model the complexity of the problem. Likewise, the agent with EGNN outperformed the agent with GCN by a significant margin, suggesting the importance of capturing the equivariance properties of crystal structures.

	EGNN			*GCN*	*Random*
# Trajectories	1	3	5	5	5
Test Accuracy (%)	44.12	47.65	49.53	37.06	17.8
Validity (%)	96.08	96.48	96.75	95.03	82.20

Table 1: Results for crystals constructed by our BC agent evaluated on validity and coverage. EGNN outperforms GCN and higher number of trajectories increases performance with best highlighted

3.3 Discussion

The learning curves indicate the prevalence of bias and non-generalizability. This could be due to the choice of crystal representation, insufficient model complexity, optimization objective, or the large and discrete nature of the problem. Further, the simple graph representation that we adopt does not capture information about the periodicity in crystals. Also, in our experiments, the policy is not constrained to any specific property, and our evaluation metrics (accuracy and validity) is generally not sufficient while dealing with the crystal design problem.

4 Conclusion

The goal of this study is to demonstrate a way to formulate the problem of crystal design as a sequential prediction task using behavioral cloning. We show that by using an effective policy network and sufficient samples of transitions during training, the agent can learn to mimic the examples in the data. Despite the agent not being able to reach high performance in terms of accuracy, we claim that such a framework could be used as a baseline for testing RL algorithms on the same task, or the learned BC policy could serve as an initial policy for fine-tuning with online RL methods. Our future directions are - *de novo* crystal design using RL approaches with energy-based reward functions, using more appropriate evaluation metrics; improving the complexity and expressiveness of our policy network, such as multigraphs to capture periodicity and advanced models like DimeNet (Gasteiger et al., 2020) or GemNet (Gasteiger et al., 2021) that capture geometric properties of molecular and crystal structures.

5 Acknowledgements

We wish to thank Sékou-Oumar Kaba for his valuable inputs. We acknowledge the people responsible for the smooth functioning of the compute resources of Mila. We also wish to acknowledge funding from Intel and CIFAR. Janarthanan Rajendran is supported by IVADO postdoctoral fellowship.

References

Keir Adams and Connor W Coley. Equivariant shape-conditioned generation of 3d molecules for ligand-based drug design. *arXiv preprint arXiv:2210.04893*, 2022.

Kai Arulkumaran, Marc Peter Deisenroth, Miles Brundage, and Anil Anthony Bharath. A brief survey of deep reinforcement learning. *arXiv preprint arXiv:1708.05866*, 2017.

Ivano E Castelli, David D Landis, Kristian S Thygesen, Søren Dahl, Ib Chorkendorff, Thomas F Jaramillo, and Karsten W Jacobsen. New cubic perovskites for one-and two-photon water splitting using the computational materials repository. *Energy & Environmental Science*, 5(10):9034–9043, 2012.

Guang Chen, Zhiqiang Shen, Akshay Iyer, Umar Farooq Ghumman, Shan Tang, Jinbo Bi, Wei Chen, and Ying Li. Machine-learning-assisted de novo design of organic molecules and polymers: Opportunities and challenges. *Polymers*, 12(1):163, 2020.

Austin Cheng, Andy Cai, Santiago Miret, Gustavo Malkomes, Mariano Phielipp, and Alán Aspuru-Guzik. Group selfies: A robust fragment-based molecular string representation. *arXiv preprint arXiv:2211.13322*, 2022.

Mérina K. Corpinot and Dejan-Krešimir Bučar. A practical guide to the design of molecular crystals. *Crystal Growth & Design*, 19(2):1426–1453, 2019. doi: 10.1021/acs.cgd.8b00972. URL `https://doi.org/10.1021/acs.cgd.8b00972`.

Daniel W Davies, Keith T Butler, Adam J Jackson, Jonathan M Skelton, Kazuki Morita, and Aron Walsh. Smact: Semiconducting materials by analogy and chemical theory. *Journal of Open Source Software*, 4(38):1361, 2019.

Johannes Gasteiger, Janek Groß, and Stephan Günnemann. Directional message passing for molecular graphs. *arXiv preprint arXiv:2003.03123*, 2020.

Johannes Gasteiger, Florian Becker, and Stephan Günnemann. Gemnet: Universal directional graph neural networks for molecules. *Advances in Neural Information Processing Systems*, 34:6790–6802, 2021.

Niklas Gebauer, Michael Gastegger, and Kristof Schütt. Symmetry-adapted generation of 3d point sets for the targeted discovery of molecules. *Advances in neural information processing systems*, 32, 2019.

Anssi Kanervisto, Janne Karttunen, and Ville Hautamäki. Playing minecraft with behavioural cloning. In *NeurIPS 2019 Competition and Demonstration Track*, pp. 56–66. PMLR, 2020.

Thomas N Kipf and Max Welling. Semi-supervised classification with graph convolutional networks. *arXiv preprint arXiv:1609.02907*, 2016.

Elton Pan, Christopher Karpovich, and Elsa Olivetti. Deep reinforcement learning for inverse inorganic materials design. *arXiv preprint arXiv:2210.11931*, 2022.

Hillary Pan, Alex M. Ganose, Matthew Horton, Muratahan Aykol, Kristin A. Persson, Nils E. R. Zimmermann, and Anubhav Jain. Benchmarking coordination number prediction algorithms on inorganic crystal structures. *Inorganic Chemistry*, 60(3):1590–1603, 2021. doi: 10.1021/acs.inorgchem.0c02996. URL `https://doi.org/10.1021/acs.inorgchem.0c02996`. PMID: 33417450.

Vıctor Garcia Satorras, Emiel Hoogeboom, and Max Welling. E (n) equivariant graph neural networks. In *International conference on machine learning*, pp. 9323–9332. PMLR, 2021.

Faraz Torabi, Garrett Warnell, and Peter Stone. Behavioral cloning from observation. In *Proceedings of the 27th International Joint Conference on Artificial Intelligence*, IJCAI'18, pp. 4950–4957. AAAI Press, 2018. ISBN 9780999241127.

Tian Xie, Xiang Fu, Octavian-Eugen Ganea, Regina Barzilay, and Tommi Jaakkola. Crystal diffusion variational autoencoder for periodic material generation. *arXiv preprint arXiv:2110.06197*, 2021.

A Appendix

A.1 Model and Training Details

For all the experiments, the following set of hyperparameters were used

1. **Number of layers in GCN / EGNN**: 4
2. **Hidden size of each GNN layer**: 256
3. **Output dimension of GNN**: 128
4. **Output dimension of linear layer that processes graph features**: 8
5. **Loss function**: Categorical cross-entropy
6. **# Epochs**: 1000
7. **Training batch size**: 2048
8. **Optimizer**: Adam
9. **Learning rate**: 10^{-3}
10. **Weight decay**: 10^{-5}
11. **Model selection**: Final model after last epoch

From the random agent and GCN agent, we trained on the dataset with 5 trajectories per crystal.