UNIVERSITY OF South Carolina

Molinaroli College of Engineering and Computing

Caltech

UCI University of California, Irvine

AI Institute UNIVERSITY OF SOUTH CAROLINA

Finding Reaction Mechanism Pathways with Deep Reinforcement Learning and Heuristic Search

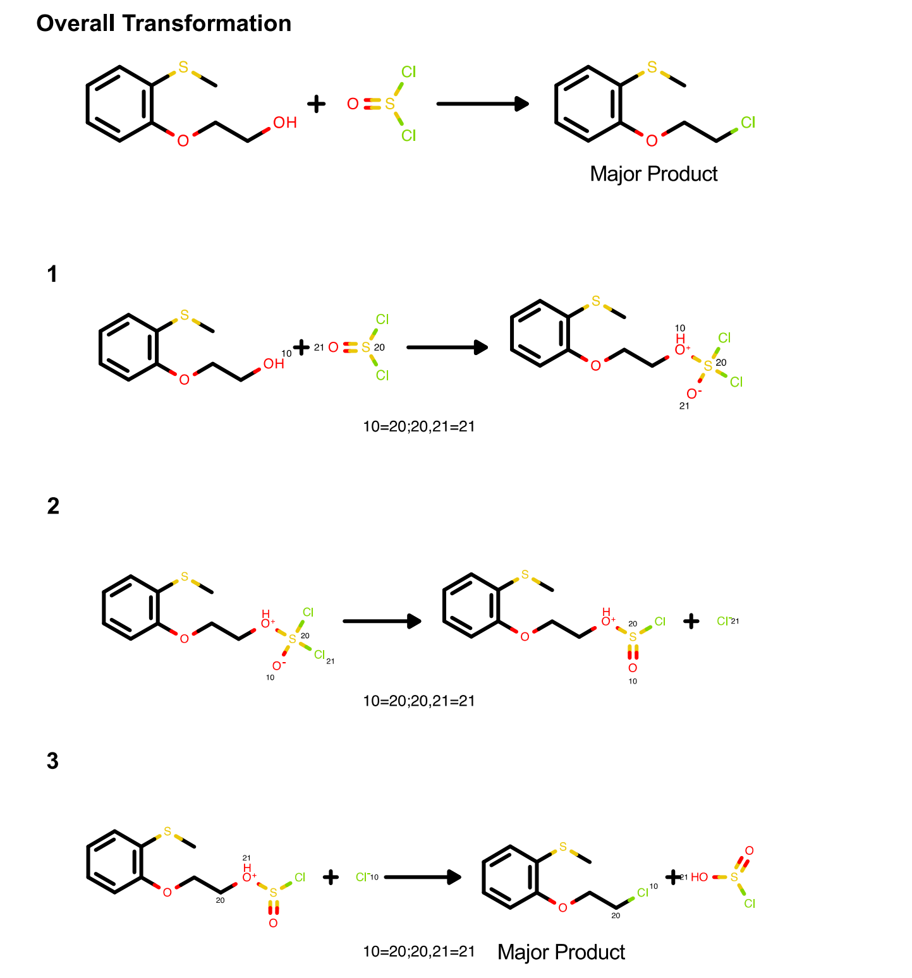
Paper

Rojina Panta[1,2], Mohammadamin Tavakoli[3], Christian Geils[1,2], Pierre Baldi[4], Forest Agostinelli[1,2]

1. AI Institute, University of South Carolina, Columbia, South Carolina, USA
2. Department of Computer Science and Engineering, University of South Carolina, Columbia, South Carolina, USA
3. Department of Computing and Mathematical Sciences, California Institute of Technology, Pasadena, California, USA
4. Department of Computer Science, University of California, Irvine, California, USA

Reaction Mechanism and Motivation

- ❖Chemical reactions are composed of reaction mechanisms.
- ❖Reaction mechanisms are elementary reaction steps, such as making bonds or breaking bonds and donating electrons, that transform the chemical reactants into chemical products.
- ❖Knowing reaction mechanisms in a given chemical reaction allows practitioners to validate the feasibility of the reaction, identify intermediate molecules, and improve reaction efficiency.



Challenges with Reaction Mechanism Prediction

- ❖Existing chemical reaction prediction methods focus on predicting the products provided by the reactants and skip the intermediate reaction steps, which makes it harder for users to understand how the predictions were made.
- ❖There are also existing models for elementary steps, like OrbChain [1], which can predict reaction mechanisms using a beam search algorithm, an incomplete search algorithm that does not explicitly consider path cost.
- ❖To address this, we learn a heuristic function and perform heuristic search that is complete and explicitly takes path cost into account.

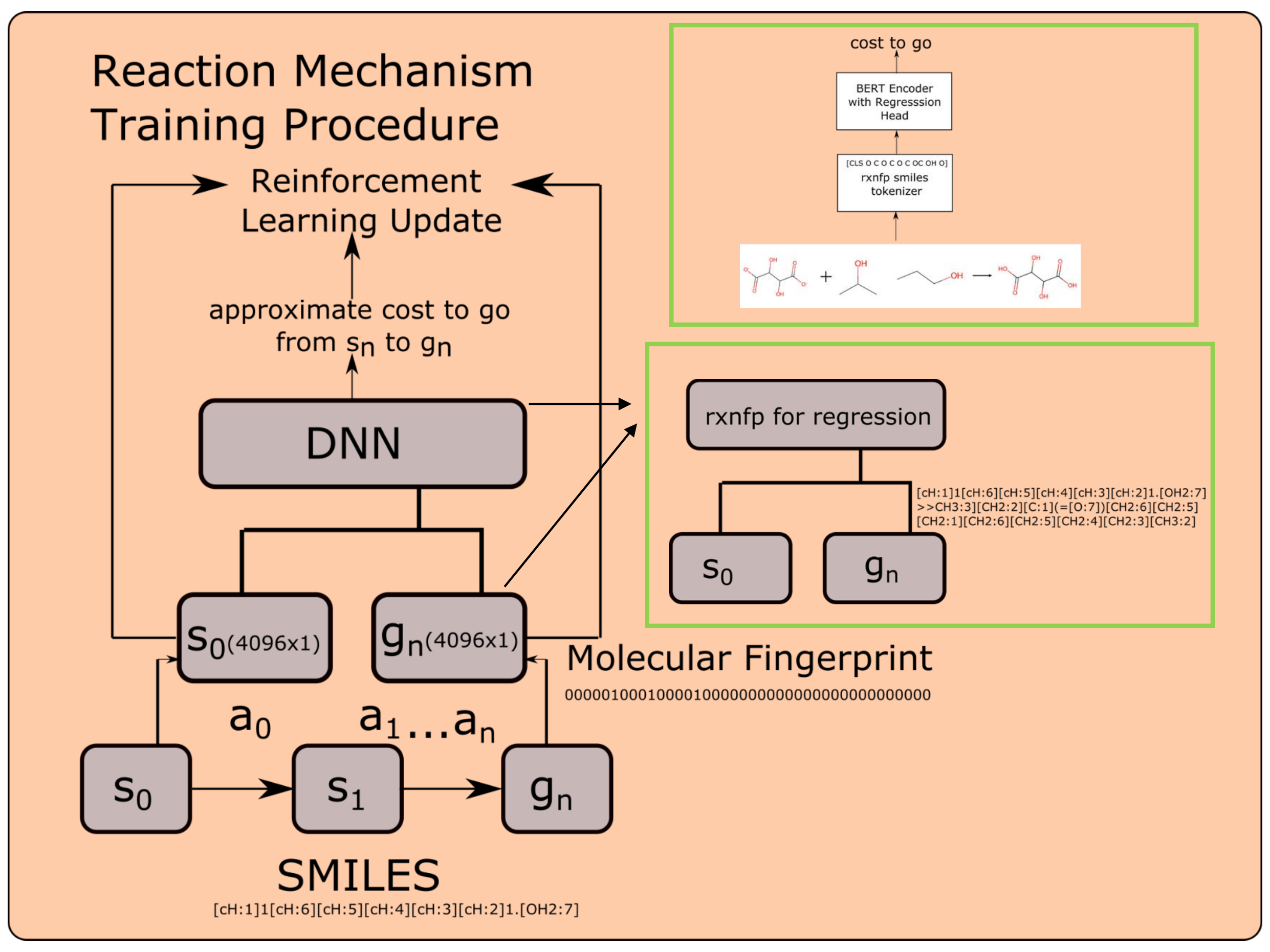
Our Approach

We pose finding a sequence of reaction mechanisms as a pathfinding problem in which the start states are reactants, and the goal states are products.

Offline Data Generation

- ❖We use United States Patent and Trademark Office (USPTO) dataset for training.
- ❖We create an offline dataset of precomputed start states, goal states, and expanded start states.
- ❖It takes about 4-5 days to generate 1.7 million datasets with parallel processing that utilizes 48 CPUs.

We built on the DeepCubeA [2], an existing algorithm, to learn a heuristic function with deep reinforcement learning and then use the learned heuristic function with a heuristic search algorithm.


Reaction Mechanism Training Procedure

Reinforcement Learning Update

approximate cost to go from s_n to g_n

DNN

$s_{0(4096x1)}$ $g_{n(4096x1)}$ Molecular Fingerprint 00000100010000100000000000000000000000000

s_0 $\xrightarrow{a_0}$ s_1 $\xrightarrow{a_1 \ldots a_n}$ g_n

SMILES [cH:1]1[cH:6][cH:5][cH:4][cH:3][cH:2]1.[OH2:7]

cost to go

BERT Encoder with Regresssion Head

[CLS O C O C O C OC OH O] rxnfp smiles tokenizer

rxnfp for regression

s_0 g_n

[cH:1]1[cH:6][cH:5][cH:4][cH:3][cH:2]1.[OH2:7]>>CH3:3][CH2:2][C:1](=[O:7])[CH2:6][CH2:5][CH2:1][CH2:6][CH2:5][CH2:4][CH2:3][CH3:2]

Learning a heuristic function with hindsight experience replay

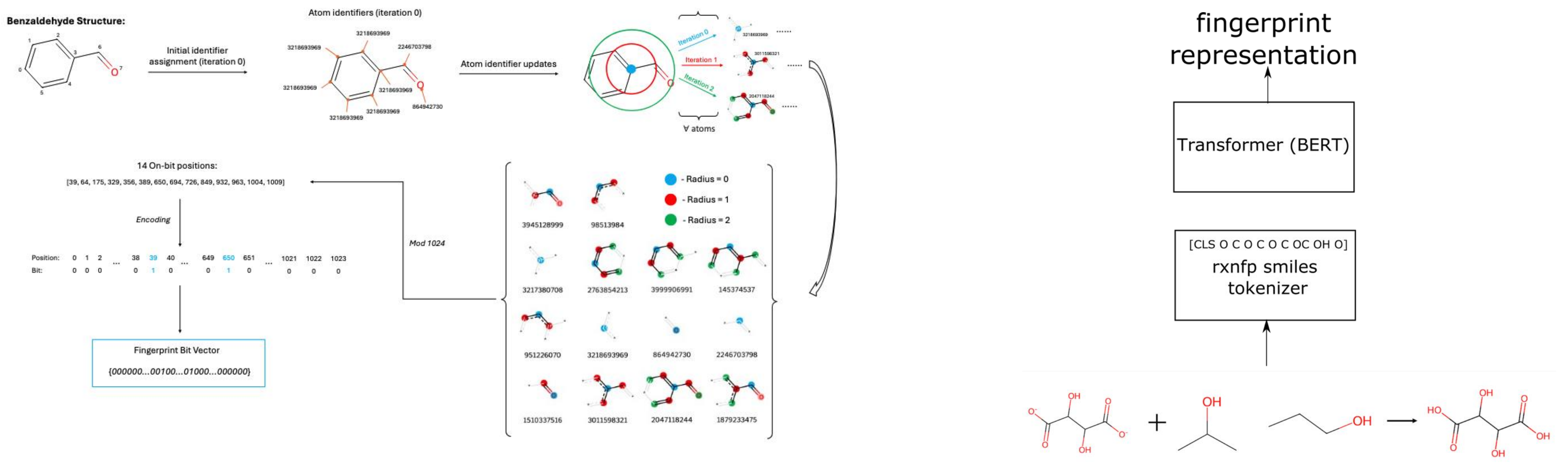
- ❖DeepCubeA is trained to solve puzzles for predefined goals.
- ❖To handle this, we build on hindsight experience replay to train a heuristic function that generalizes over any start and goal pair [3].

Heuristic Search

- ❖We then use the learned heuristic function with the A* search algorithm to find paths from start states to goal states.

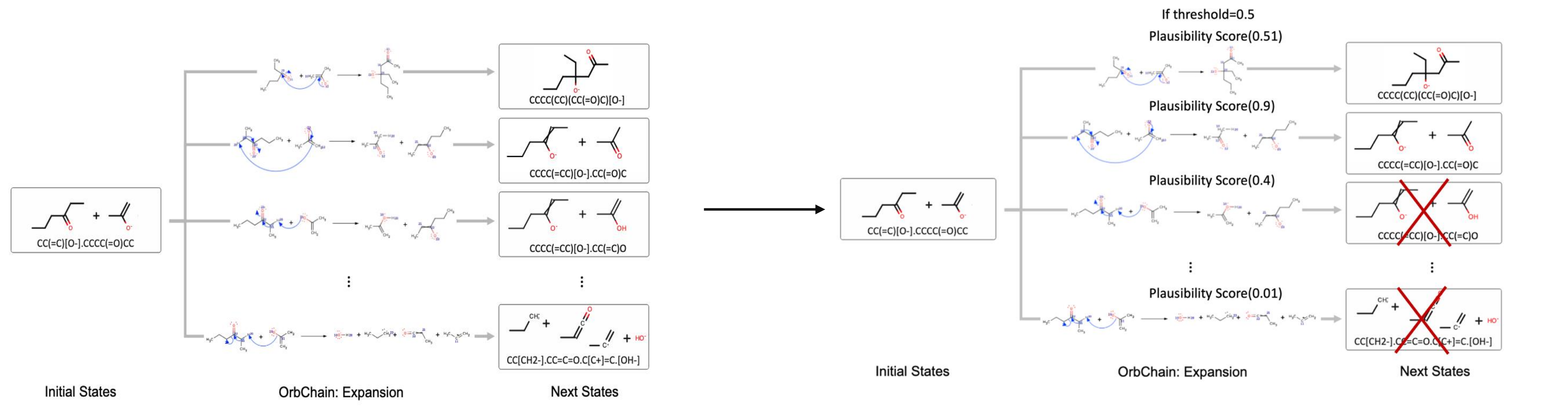
Limitations of Molecular Fingerprint Representation

- ❖Molecular fingerprints, such as ECFPs[4], are static representations of molecules that indicate the presence or absence of a structure, represented as fixed-length bit vectors.
- ❖Reactants and products comprise a set of molecules, and existing libraries like RDKit and OEChem generate fingerprints by applying the union function on bit vectors of individual molecules, leading to a lossy representation of molecules.



Fingerprint Representation: (Left) Molecular Fingerprint Representation, (Right) Learnable Fingerprint Representation using Transformer

- ❖To address this, we use transformer architecture to generate fingerprints, which allows us to learn a representation of reactions instead of individual molecules.
- ❖Our model is inspired by the transformer architecture introduced in rxnfp[5]; however, rather than relying on the original pretrained rxnfp network, we use only its SMILES tokenizer and architectural hyperparameters while implementing our own BERT encoder and a custom regression head.



Elementary Steps: (Left) OrbChain Expansion without Plausibility Score, (Right) OrbChain Expansion with Plausibility Score

Results

- ❖Start/goal pairs are generated by taking 0-6 random steps from a given start state.
- ❖We also test with Tanimoto similarity to guide search.
- ❖DeepCubeA outperforms uniform cost search and the Tanimoto similarity.
- ❖DeepCubeA with transformer outperforms one trained with ECFP.

Step/s	Solver	Path Cost	% Solved	Nodes	Secs	Nodes/Sec
Steps=0	DeepCubeA (rxnfp)	0.00	**100.00%**	3.09E+2	4.07	76.03
	DeepCubeA (ecfp)	0.00	**100.00%**	3.09E+2	3.87	79.97
	Uniform Cost Search	0.00	**100.00%**	3.09E+2	4.61	67.13
	Tanimoto Similarity	0.00	**100.00%**	3.09E+2	3.71	83.42
Steps=1	DeepCubeA (rxnfp)	1.00	**100.00%**	6.82E+2	10.07	67.81
	DeepCubeA (ecfp)	1.00	**100.00%**	7.49E+2	9.70	77.26
	Uniform Cost Search	1.00	**100.00%**	4.26E+4	553.33	76.95
	Tanimoto Similarity	1.00	**100.00%**	3.13E+4	429.29	72.97
Steps=2	DeepCubeA (rxnfp)	1.93	**100.00%**	1.14E+3	16.81	68.05
	DeepCubeA (ecfp)	2.07	**100.00%**	1.63E+4	267.16	60.87
	Uniform Cost Search	1.67	20.00%	1.32E+5	1497.77	87.96
	Tanimoto Similarity	1.75	26.67%	1.10E+5	1229.10	89.13
Steps=3	DeepCubeA (rxnfp)	2.79	93.33%	9.29E+3	127.02	73.18
	DeepCubeA (ecfp)	2.77	86.67%	4.14E+4	578.88	71.54
	Uniform Cost Search	-	0.00%	-	-	-
	Tanimoto Similarity	-	0.00%	-	-	-
Steps=4	DeepCubeA (rxnfp)	3.60	**100.00%**	2.63E+4	350.06	75.33
	DeepCubeA (ecfp)	3.33	60.00%	6.36E+4	821.64	77.36
	Uniform Cost Search	3.00	6.67%	1.43E+5	1962.28	73.01
	Tanimoto Similarity	3.00	6.67%	2.47E+4	272.15	90.64
Steps=5	DeepCubeA (rxnfp)	3.92	86.67%	6.37E+4	744.31	85.60
	DeepCubeA (ecfp)	3.40	33.33%	8.40E+4	968.49	86.69
	Uniform Cost Search	-	0.00%	-	-	-
	Tanimoto Similarity	-	0.00%	-	-	-
Steps=6	DeepCubeA (rxnfp)	3.55	73.33%	4.70E+4	678.27	69.29
	DeepCubeA (ecfp)	3.20	33.33%	6.14E+4	933.86	65.73
	Uniform Cost Search	-	0.00%	-	-	-
	Tanimoto Similarity	-	0.00%	-	-	-

Future Work

- ❖Even though we have better representation with improved test accuracy on the simulated dataset, we have yet to perform well while testing on the original USPTO dataset, so we first want to see if the goal is reachable or not from the start state with our current architecture that uses OrbChain, so we plan to do a reachability test using a supervised learning approach.
- ❖To speed up the process, we will use a deep Q-network (DQN) to learn an action-value function that predicts the cost-to-go for actions in given states based on molecular orbital interactions and use Q* search [6] to find paths without requiring full state expansion.
- ❖USPTO dataset we use has only a single product called major product, which does not contain by-products of chemical reactions. So, the product can contain a set of goal states instead of a single goal. We plan to train a heuristic function that generalizes over states and a set of goal states.
- ❖The transition cost that we currently use does not account for the feasibility of the reaction mechanism path. If transitions are independent, the most feasible path would be the product of the probabilities of each transition. So, we plan to modify our transition cost by using a negative log probability of transition cost.

References

1. Tavakoli, Mohammadamin, et al. "AI for Interpretable Chemistry: Predicting Radical Mechanistic Pathways via Contrastive Learning." Advances in Neural Information Processing Systems 36 (2024).
2. Agostinelli, Forest, et al. "Solving the Rubik's cube with deep reinforcement learning and search." Nature Machine Intelligence 1.8 (2019): 356-363.
3. Andrychowicz, Marcin, et al. "Hindsight experience replay." Advances in neural information processing systems 30 (2017).
4. Rogers, David, and Mathew Hahn. "Extended-connectivity fingerprints." *Journal of chemical information and modeling* 50.5 (2010): 742-754.
5. Schwaller, Philippe, et al. "Mapping the space of chemical reactions using attention-based neural networks." *Nature machine intelligence* 3.2 (2021): 144-152.
6. Agostinelli, Forest, et al. "A* search without expansions: Learning heuristic functions with deep q-networks." arXiv preprint arXiv:2102.04518 (2021).