FINDING PLAUSIBLE REACTION MECHANISM PATHWAYS WITH LEARNED HEURISTIC FUNCTIONS AND POST-HOC ENERGY SCREENING

Anonymous authors
Paper under double-blind review

ABSTRACT

A chemical reaction is composed of a pathway of elementary reaction mechanism steps. Knowledge of the reaction mechanism pathway in a chemical reaction reveal intermediate molecules, allow predictions of similar reactions under different conditions, and validate the feasibility of a reaction. Many existing methods for finding reaction mechanism pathways rely on expert-annotated reaction mechanism data, which are expensive to obtain. We introduce a method that does not rely on any expert data by leveraging an "expansion function" that returns all possible elementary steps for a given set of molecules and the ability to screen implausible elementary steps using a free energy analysis. Using the expansion function, we train a heuristic function that estimates the smallest number of elementary steps needed to reach any given chemical products from any given set of molecules. The expansion function and trained heuristic function are then combined with heuristic search to find reaction mechanism pathways given the reactants and products in a chemical reaction. We repeat the heuristic search with different parameters to obtain multiple possible pathways per reaction. Finally, we use the post-hoc energy screening to screen implausible pathways. Our approach finds pathways for 79.62% of USPTO reactions, 91.71% of Human Benchmark reactions, and 45.11% of ORD reactions. This is a 24.89% and 71.39% improvement in coverage on the Human Benchmark and ORD datasets, respectively, compared to a pathway finding algorithm trained on expert data. While, after post-hoc energy screening, our approach finds between 16.67% and 26.83% fewer plausible pathways compared to the algorithm trained on expert data, it does so without being trained on any expert data, at all. However, it exactly recovers 31 Human Benchmark pathways that were assessed as plausible by experts. In our approach, the energy screen also removes high energy candidates while retaining lower energy alternatives. Together, these results show that the combination of an expansion function and post-hoc energy screening provides a promising alternative to training on expert data for finding reaction mechanism pathways.

1 INTRODUCTION

A chemical reaction mechanism consists of a sequence of elementary reaction mechanism steps. These steps involve processes such as bond formation, bond cleavage, or electron transfer. Together, they describe how reactants are transformed into products, as illustrated in Figure 1. Knowledge of reaction mechanisms allows practitioners to assess reaction feasibility, identify intermediate molecules and impurities (Qiu & Norwood, 2007), and improve reaction efficiency. Mechanistic understanding supports drug discovery (Holdgate et al., 2018), while reaction pathway analysis has applications in materials synthesis (McDermott et al., 2021).

Existing reaction mechanism prediction methods commonly require expert supervision. PMechRP (Miller et al., 2025) is trained on manually curated elementary steps from PMechDB (Tavakoli et al., 2024a), while other methods assign reaction classes and apply expert-curated templates to construct multi-step pathways (Joung et al., 2024; Chen et al., 2024; Joung et al., 2025). Curating such supervision is expensive. We therefore seek to learn search guidance without expert-annotated elementary steps or reaction mechanism pathways.

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

To remove dependence on expert data, we formulate reaction mechanism pathway prediction as a pathfinding problem in which a state represents a set of molecules and an action represents an elementary step. Next, using OrbChain (Tavakoli et al., 2025; 2024b), we implement an expansion function that enumerates all applicable actions and returns their successor states. While this expansion function can then be used to search for reaction mechanism pathways, its application to a single state can take minutes, making brute-force search infeasible. Therefore, we build on supervised learning from random walks (Chervov et al., 2025) and hindsight experience replay (Andrychowicz et al., 2017) to train a heuristic function that generalizes over both states and goals to guide heuristic search (Agostinelli et al., 2024a). By sampling actions using OrbChain instead of computing the entire expansion function, we perform relatively quick random walks to generate start state and goal pairs and use the length of the random walk as the supervision signal. To find pathways, perform weighted A* search with multiple different weight values with the same trained heuristic function. It then records all pathways found to increase reaction mechanism pathway coverage (i.e. the number of pathways found) and increase the number of plausible pathways found and can also recover pathways with substantially different energy profiles. To reduce implausible pathways, we use a post hoc energy screening to calculate the free energy profiles for pathways found by heuristic search and filter out pathways containing implausibly large uphill free energy changes. Moreover, without using expert annotated mechanisms for training, our method exactly recovers 31 pathways that were manually assessed as plausible.

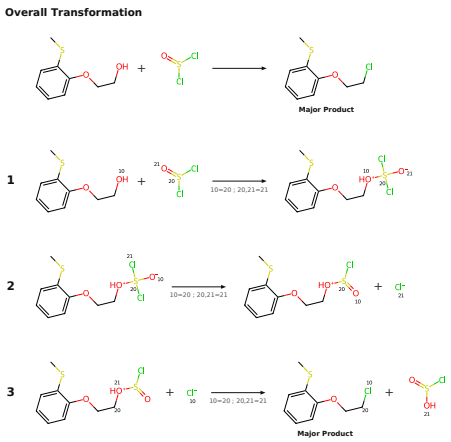


Figure 1: An overall transformation is broken down into a sequence of three elementary steps where all the intermediates are preserved.

On the United States Patent and Trademark Office (USPTO) dataset (Lowe, 2017), our approach finds pathways for 79.62% of reactions. We compare our approach with PMechRP, which we refer to as the Reaction Predictor baseline (Miller et al., 2025). PMechRP is trained on expert-annotated elementary reaction steps, whereas our approach does not require expert-annotated data. Relative to the Reaction Predictor, our approach increases pathway coverage from 73.43% to 91.71% on the Human Benchmark (Miller et al., 2025) and from 26.32% to 45.11% on the Open Reaction Database (ORD) (Kearnes et al., 2021; Miller et al., 2025), corresponding to relative gains of 24.89% and 71.39%, respectively. After energy screening, our approach finds plausible pathways for 13.66%, 20.29%, and 8.27% of reactions for the USPTO, Human Benchmark, and ORD datasets, respectively, and the Reaction Predictor finds plausible pathways for 27.71% and 10.53% for the Human Benchmark, and ORD datasets, respectively. While our approach finds fewer plausible pathways when compared to the Reaction Predictor, unlike the Reaction Predictor, it does so without any expert-annotated data.

2 Preliminaries

2.1 Pathfinding and Expansion Functions

A pathfinding domain is defined as a weighted directed graph (Pohl, 1970), where nodes represent states, edges represent actions that transition between states, and weights on the edges represent transition costs. T denotes the transition function, where $s' = T(s, a)$ if and only if there exists an edge connecting states s and s' for some action, a. c denotes the transition cost function, where $c(s, a)$ is the transition cost when taking action a in state s. The set of all outgoing edges from state s (equivalently, the set of all possible actions one can take in s) is denoted $\mathcal{A}(s)$. A pathfinding problem instance is defined by a tuple (D, s_0, g), where D is the pathfinding domain, s_0 is the start state, and g is the set of states considered goal states. Given a pathfinding problem, the objective is to find a path, which is a sequence of actions, that transforms the start state into a goal state while

attempting to minimize the path cost, where the path cost is the sum of transition costs. The cost-to-go is the cost of a shortest path (i.e. an optimal path), which is a path from a given start state to a given goal that has the lowest possible path cost.

An expansion function for a pathfinding domain is given a state s, enumerates the applicable actions $a_i \in \mathcal{A}(s)$, and returns a list of tuples $(a_i, c(s, a_i), T(s, a_i))$, one for each applicable action. For random walk generation, we use a sampling variant of the expansion function. Instead of enumerating all applicable actions, it samples one action $a \in \mathcal{A}(s)$ and returns the single tuple $(a, c(s, a), T(s, a))$. Repeatedly applying this sampling procedure generates a random walk.

2.2 Heuristic Function Learning

A heuristic function $h(s, g)$ maps a state s and goal g to an estimate of the cost-to-go, which is minimum path cost from s to g. We denote the cost-to-go by $h^*(s, g)$. A learned heuristic $h_\theta(s, g)$, parameterized by θ, estimates this cost. Because $h_\theta(s, g)$ takes both the current state s and goal g as inputs, it is a goal conditioned heuristic. Learned heuristic functions have been used successfully to guide search. For example, DeepCubeA (Agostinelli et al., 2019) learns a heuristic using deep approximate value iteration (DAVI) (Puterman & Shin, 1978). However, since DAVI relies on an expansion function, its computation cost grows linearly with the number of applicable actions. On the other hand, Chervov et al. (2025) construct supervised heuristic targets using random walks. Each random walk step requires only sampling an applicable action and applying the transition function, which can be substantially faster than applying an expansion function. The cost of any random walk from s to a state in g is an upper bound on $h^*(s, g)$ because it represents one valid path, but not necessarily a shortest path. To learn a heuristic that generalizes across goals, the training data must include different goals. Hindsight Experience Replay (HER) (Andrychowicz et al., 2017) achieves this through goal relabelling, in which states reached along a trajectory are treated as goals. Agostinelli et al. (2024a) apply this principle to heuristic learning with random walks.

2.3 Heuristic Search

A* search (Hart et al., 1968) maintains an OPEN priority queue of search nodes. A search node n is composed of its associated state s_n, its path cost $g(n)$ from the start state, a parent pointer, the action that generated it, and its depth, where depth is the number of actions from the start node to n. A* assigns node n the priority $f(n) = g(n) + h(s_n, g)$, where $h(s_n, g)$ estimates the cost-to-go from s_n to a goal state in g. The start node contains s_0, while a goal node contains a goal state. In our method, $h(s_n, g)$ is instantiated by the learned heuristic $h_\theta(s_n, g)$.

A* initializes OPEN with the start node and records the start state and its path cost in CLOSED. At each iteration, A* removes a node n with the smallest priority from OPEN. If its associated state s_n satisfies the goal condition, the path is reconstructed by following parent pointers from n to the start node. Otherwise, the expansion function is applied to s_n. For each resulting successor state, a successor node n' is created with its path cost $g(n')$, heuristic estimate $h(s_{n'}, g)$, priority $f(n')$, parent pointer, and generating action. If $s_{n'}$ has not been recorded in CLOSED, its path cost is recorded and n' is added to OPEN. If it has been recorded but $g(n')$ is lower than the recorded path cost, the record is updated and n' is added to OPEN again. Search terminates when a goal is reached, OPEN becomes empty, or a specified search limit is reached.

3 Approach

3.1 OrbChain Expansion Function for Polar Reactions

For a polar reaction, OrbChain is a mechanistic representation that reduces an elementary reaction to the interaction between two reactive molecular orbitals (MOs): a source orbital and a sink orbital. The source MO contains the electron pair that moves during the reaction and can correspond to a lone pair or a σ/π bond. The sink MO is the orbital that accepts that electron pair, such as an empty orbital or an antibonding/bond-centered orbital involved in bond rearrangement. More complicated electron movement can be represented by chaining adjacent orbitals, which allows the same formalism to capture resonance-assisted processes such as conjugate addition or elimination. This is the central

idea inherited from RMechRP (Tavakoli et al., 2024b; 2025): instead of predicting an entire product directly, the reaction is represented by identifying the pair of orbitals whose interaction defines the elementary arrow-pushing step.

OrbChain connects the orbital representation to explicit molecular graph edits. For polar reactions, given atom-mapped reactants, products, and curved-arrow information, it can identify the reactive source and sink orbitals that explain the observed transformation. For state expansion, given the reactants and a proposed possible source and sink orbital pair, it follows the corresponding two-electron flow and determines which bonds must be broken, formed, or have their bond order changed, thereby generating the possible product structures and the associated arrow-pushing mechanisms. The key consequence is that state expansion (i.e., possible product generation) becomes deterministic once the reactive orbital pair is specified. OrbChain is implemented using OpenEye Scientific Software (OpenEye, Cadence Molecular Sciences, 2024a;b). An example of OrbChain state expansion is provided in Appendix A.1.

3.2 Reaction Mechanism Pathway Finding as a Pathfinding Problem

A state s is a multiset of molecules. The initial state contains the reactant molecules, while states generated during search may contain reaction intermediates and product molecules. A reaction intermediate is a molecule formed by one elementary step and consumed or transformed by a later elementary step. The multiset representation preserves repeated molecular components. An action a is a proposed source and sink orbital pair. Given a state s and action a, OrbChain applies the corresponding electron flow and molecular graph edits to produce the successor state $T(s, a)$. The OrbChain expansion function enumerates the applicable source and sink orbital pairs and returns a list of tuples $(a_i, c(s, a_i), T(s, a_i))$. For random walk generation, we use a sampling variant of the OrbChain expansion function. Instead of enumerating all applicable orbital pairs, it samples one pair and returns its corresponding successor state. Repeatedly applying this procedure generates a random walk.

For a reaction pathway problem, the goal g is the multiset of required product molecules. Let $c_s(m)$ and $c_g(m)$ denote the multiplicity of canonical molecule m in state s and goal g, respectively. A state s satisfies g when $c_s(m) \geq c_g(m)$ for every molecule m. Thus, additional molecules are permitted provided that every required product molecule is present with sufficient multiplicity.

3.3 Heuristic Learning from Random Walks

Building on the random walk sampling and goal relabelling procedures described in Section 2.2, we generate training examples using two consecutive random walks with OrbChain. Starting from a state in a source dataset, the first random walk generates a start state s. The second random walk starts from s and reaches a state whose molecules define the goal g. We use the number of steps in the second random walk, $h_{\mathrm{rw}}(s, g)$, as the supervised cost-to-go target. If s already satisfies g, the target is zero. The learned heuristic takes a state s and goal g as input and predicts the cost-to-go $h_\theta(s, g)$. For a minibatch of B training examples, we optimize θ by minimizing the mean squared error between the predicted costs $h_\theta(s_i, g_i)$ and random walk targets $h_{\mathrm{rw}}(s_i, g_i)$. Figure 2 summarizes the random walk data generation and supervised training procedure.

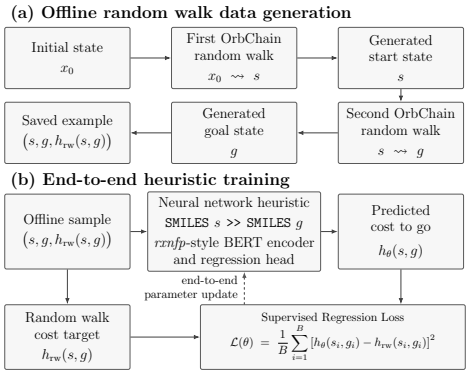


Figure 2: How random walks generate data to train the heuristic function.

To assess the computational advantage of random walk supervision over DAVI, which uses an expansion function, we compare random successor generation and expansion function time. For each dataset, we conduct five trials using 20 states generated by random walks per trial. For the paired timing comparison, we retain a state only when complete expansion finishes within the 30 minute

Table 1: Runtime comparison of complete expansion and random successor generation. Values are the mean ± standard deviation across 5 trial means. Speedup is the expansion time divided by the random successor generation time and are reported for states for which both operations succeeded.

Dataset	Expansion Time (s)	Random Successor Time (s)	Random Successor Speedup	Mean Expansion Outputs
ORD	315.16 ± 79.54	7.66 ± 1.84	40.99 ± 1.13×	8973.34 ± 1777.76
Human Benchmark	57.97 ± 13.51	1.84 ± 0.40	31.47 ± 1.10×	2363.17 ± 445.77
USPTO	33.54 ± 12.52	1.26 ± 0.37	26.13 ± 2.19×	1609.91 ± 455.64

limit and produces at least one successor, and random successor generation returns a valid successor. Timings exclude dataset loading and input state generation. Table 1 shows that random successor generation is $26.1\times$ to $41.0\times$ faster than the expansion function, supporting our use of random walks to construct heuristic learning targets. Table 9, in the Appendix, reports additional outcomes, including timeouts, and expansions producing no successors.

3.4 Heuristic Search

After training we use the learned heuristic function with A* search and we use OrbChain as the expansion function used by the search. For a search node n containing state s_n, we use the weighted priority $f_w(n) = w\, g(n) + h_\theta(s_n, g)$, where $g(n)$ is the path cost from the start state to s_n, $h_\theta(s_n, g)$ is the learned cost-to-go estimate, and w is the search weight. Smaller values of w give the learned heuristic greater relative influence. We run independent searches with different values of w and take the union of the recovered pathways because different weights can recover different pathways.

3.5 Post-Hoc Energy Screening

We apply an automated preprocessing and energy screening pipeline to each recovered pathway without pathway-specific human assessment. Before calculating energies, we check adjacent states for conservation of elemental composition, including implicit hydrogen atoms, and total formal charge. Missing unchanged components are repaired when possible without creating molecular structures or altering connectivity. Pathways with unresolved parsing or balance failures are excluded and reported separately. For each retained pathway, we generate one initial three-dimensional conformer for each distinct molecular species using ETKDGv3 as implemented in RDKit (RDKit; Wang et al., 2020). We optimize each structure using GFN2-xTB with implicit water solvation (Bannwarth et al., 2019) and calculate an approximate Gibbs free energy using zero point and thermal contributions with Grimme's quasi-RRHO treatment Grimme (2012). Molecular free energies are summed to obtain the energy of each pathway state. Additional computational details are provided in Appendix A.5. For a pathway containing states $(s_0, s_1, \ldots, s_L)$, let G_i be the computed free energy of s_i. We compute each signed adjacent-state change as $\Delta G_i = G_{i+1} - G_i$ and define the maximum uphill change as $\Delta G_{\max} = \max_{0 \le i < L} \Delta G_i$. A pathway passes the screen at threshold τ when $\Delta G_{\max} \le \tau$.

We define a reaction pathway as thermodynamically implausible if any uphill free energy change between consecutive pathway states exceeds $\tau = 50$ kcal mol^{-1}. This threshold is conservative: it is roughly twice the energy available from a full ATP hydrolysis cascade (~ 25 kcal mol^{-1}) (Nelson & Cox, 2004) and well above the range of endergonic steps observed in enzyme-catalysed mechanisms (≤ 20 kcal mol^{-1}) (Wolfenden & Snider, 2001). It is permissive enough to retain high energy but chemically meaningful intermediates and avoid penalising pathways where GFN2-xTB may systematically overestimate reaction energies by several kcal mol^{-1} (Bannwarth et al., 2019). Because no comprehensive survey establishes a universal cutoff, we additionally report results using the stricter threshold $\tau = 20$ kcal mol^{-1} and compare both thresholds with available manual plausibility assessments in Appendix A.6. Pathways with missing energy values are marked incomplete and excluded from threshold counts. Because the pipeline does not locate transition states, it does not estimate activation free energy barriers.

4 EXPERIMENTS

4.1 EXPERIMENTAL SETUP

Training data. Training data are generated from 482 chemist verified USPTO reactions (Lowe, 2017), treated as closed shell organic reactions. OrbChain is therefore configured to generate only polar steps involving electron pair movement. We omit the reagent field and use the reactant states to initialize random walk data generation, but do not explicitly designate the corresponding USPTO products as training goals. Each training goal is instead determined by the endpoint of a sampled random walk. The first and second random walks have sampled lengths between 0 and 14 and between 0 and 15, respectively. This procedure generated approximately 30 million supervised examples. Additional training and compute details are provided in Appendix A.2.

State representation and heuristic architecture. Fixed-length hashed molecular fingerprints can map distinct substructures to the same vector position, and taking their bitwise union for a multi-molecule state does not preserve which molecule contributed each feature or how many times it occurs (Rogers & Hahn, 2010). We therefore represent each state s and goal g as $s >> g$ using their SMILES representations, with periods separating multiple molecules. We tokenize this sequence using the *rxnfp* SMILES tokenizer (Schwaller et al., 2021), construct a vocabulary of 359 tokens from the training split, and pad or truncate sequences to 512 tokens. The sequence is encoded by an *rxnfp*-style BERT model with 12 Transformer layers, four attention heads, a hidden dimension of 256, and an intermediate dimension of 512. The final 256 dimensional `[CLS]` representation is passed to a fully connected regression network with residual connections (He et al., 2016), which predicts $h_\theta(s, g)$. Unlike the original *rxnfp* workflow, which uses masked language model pretraining followed by supervised classification (Devlin et al., 2019; Schwaller et al., 2021), we initialize the complete model from scratch and train it end to end using the number of steps in the second random walk as the regression target. The loss updates both the BERT encoder and regression head; additional training details are provided in Appendix A.2.

Evaluation data and search configuration. We evaluate the trained heuristic using reactant and product pairs from three data sources. The USPTO evaluation set contains 476 of the 482 reactions used to initialize training data generation; we exclude six reactions whose initial states already contain all required product molecules because they do not require search. We also evaluate 350 reactions from the Human Benchmark and 133 from ORD (Miller et al., 2025). For each reaction, the reactants define the start state s_0 and the products define the goal g. The USPTO evaluation measures performance on the reaction source used to initialize training data generation, while the Human Benchmark and ORD evaluations measure transfer to reaction sources not used for training data generation.

For each reaction, we independently run weighted A* with $w \in \{0.1, 0.3, 0.5, 0.6, 0.7, 1.0\}$ and a two-hour limit per search. During evaluation, we canonicalize the current state and goal as SMILES (Weininger, 1988; Weininger et al., 1989), represent them as $s >> g$, and tokenize the sequence using the *rxnfp* tokenizer. Negative heuristic predictions are clipped to zero, and successor states are evaluated in inference batches of at most 2,000. A reaction is considered solved when the current state contains every required product molecule with sufficient multiplicity. We record solution status, path cost, generated node count, and runtime, and reconstruct each recovered pathway by following parent pointers from the solution node to the start node.

4.2 RESULTS

Pathway search. We compare A* guided by our learned heuristic with uniform cost search, A* guided by a Tanimoto similarity heuristic, and PMechRP (Miller et al., 2025), which we refer to as the Reaction Predictor baseline. Uniform cost search ranks nodes using only the accumulated path cost and therefore does not use a heuristic. The Tanimoto baseline represents each molecule using a Morgan fingerprint of length 4096 and radius 3 (Rogers & Hahn, 2010; Flower, 1998; Godden et al., 2000). Each occurrence of a goal molecule is matched to at most one molecule in the current state by maximizing the total Tanimoto similarity. Unmatched goal molecules contribute zero similarity, while additional molecules in the current state are not penalized. We define the heuristic by subtracting the mean Tanimoto similarity from one. Consequently, the heuristic is zero when the goal

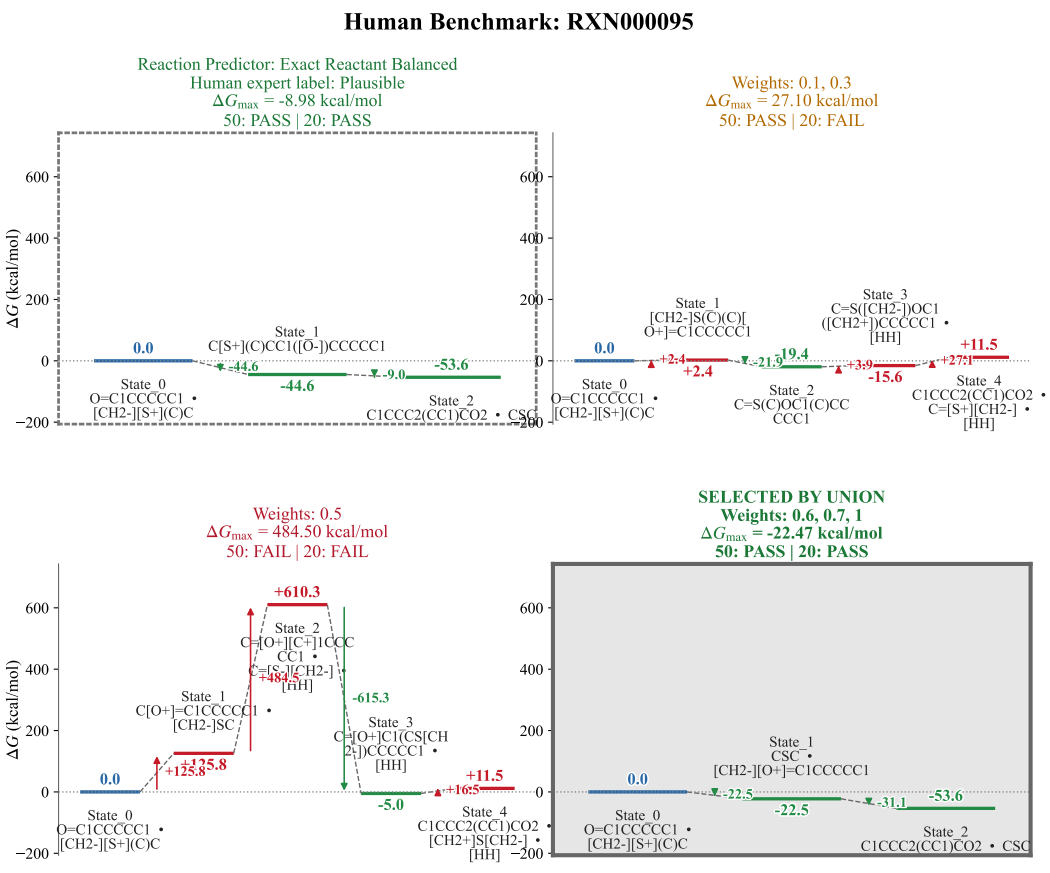


Figure 3: Energy profiles for RXN000095 from the Reaction Predictor and learned heuristic search. The union selects the pathway recovered with $w \in \{0.6, 0.7, 1.0\}$, which passes both screening thresholds.

molecular multiset is contained in the current state, consistent with our goal condition. Each method receives a time limit of two hours per reaction.

Table 2 reports the number and percentage of reactions solved. Path Cost gives the mean for pathway with complete energy profile. A* guided by the learned heuristic achieves substantially greater coverage than uniform cost search and the Tanimoto heuristic. Uniform cost search does not solve any ORD reaction within the time limit of two hours, demonstrating the importance of heuristic guidance in this setting. Additional results by search weight, path cost, and runtime are provided in Appendix A.4.

Energy screening. We apply the automated post-hoc energy screening procedure described in Section 3.5 to the recovered pathways. A pathway passes the primary screen when its maximum signed free energy increase between adjacent pathway states does not exceed 50 kcal mol^{-1}. Table 2 reports the results. At the threshold of 50 kcal mol^{-1}, our approach retains 11 ORD, 71 Human Benchmark, and 65 USPTO pathways, compared with 14 ORD and 97 Human Benchmark pathways for the Reaction Predictor. At the stricter 20 kcal mol^{-1} threshold, our approach retains 10, 60, and 52 pathways, respectively, compared with 12 ORD and 82 Human Benchmark pathways for the Reaction Predictor. Thus, our approach achieves greater coverage before screening but retains fewer energy-passing pathways on the comparable datasets. However, unlike the Reaction Predictor, it requires no expert annotated elementary steps or reaction mechanism pathways for training. The free energy profiles are computed after search and are not used as transition costs or heuristic inputs. Counts of adjacent state changes, agreement with manual assessments, and representative pathways are provided in Appendices A.5, A.6, and A.7.

Table 2: Energy screening of solved pathways. Complete Pathways and Mean Path Cost are computed over pathways with complete energy profiles; a pathway passes threshold τ when $\Delta G_{\max} \leq \tau$, with thresholds in kcal mol^{-1}.

Dataset	Source	Total	Solved	Solved (%)	Complete Pathways	Path Cost	Complete Pathways $\Delta G_{\max} \leq 50$	Complete Pathways $\Delta G_{\max} \leq 20$
ORD	Reaction Predictor	133	35	26.32	31	2.23	**14**	**12**
ORD	Learned Heuristic (best, w=0.5)	133	56	42.11	47	2.00	9	8
ORD	Learned Heuristic (w union)	133	**60**	**45.11**	**53**	2.04	11	10
ORD	Uniform Cost Search	133	0	0.00	–	–	–	–
ORD	Tanimoto Similarity	133	19	14.29	17	1.06	1	0
Human	Reaction Predictor	350	257	73.43	241	2.65	**97**	**82**
Human	Learned Heuristic (best, w=0.6)	350	312	89.14	271	1.85	66	55
Human	Learned Heuristic (w union)	350	**321**	**91.71**	**283**	1.94	71	60
Human	Uniform Cost Search	350	29	8.29	24	1.17	7	7
Human	Tanimoto Similarity	350	136	38.86	125	1.26	40	32
USPTO	Learned Heuristic (best, w=0.7)	476	362	76.05	286	2.15	59	48
USPTO	Learned Heuristic (w union)	476	**379**	**79.62**	**317**	2.48	**65**	**52**
USPTO	Uniform Cost Search	476	16	3.36	15	1.00	11	10
USPTO	Tanimoto Similarity	476	100	21.01	87	1.20	22	20

Table 3: Exact recovery of manually plausible Reaction Predictor pathways on the Human Benchmark with complete energy profile. Exact recovery requires the learned search to produce the same ordered sequence of molecular states. The final columns report exact matches passing each energy threshold. ORD is omitted because no exact matches were found.

Dataset	Supervised source	Complete	Exact pathway matches	Both paths $\Delta G_{\max} \leq 50$	Both paths $\Delta G_{\max} \leq 20$
Human	Best individual ($w = 0.6$)	164	**31**	**15**	**14**
Human	Weight union	169	**31**	**15**	**14**

We compare learned search pathways with Human Benchmark Reaction Predictor pathways that were assessed as plausible by human expert. We define exact recovery as producing the same ordered sequence of molecular states. Table 3 shows that the learned search exactly recovers 31 manually plausible pathways with complete energy profile, despite receiving no manual plausibility labels during training. Among these exact matches, 15 pass the 50 kcal mol^{-1} screen and 14 pass the 20 kcal mol^{-1} screen. Additional results provided in Appendix A.8. Figure 3 illustrates the effect of search weight for RXN000095. The manually plausible Reaction Predictor pathway passes both thresholds, while screening rejects high energy pathways from some weights and retains an alternative from $w \in \{0.6, 0.7, 1.0\}$ that passes both thresholds. Figure 10 provides an additional example of how the search weight affects pathway found and its energy screening outcome.

5 Discussion

Learned heuristic search achieves 45.11%, 91.71%, and 79.62% coverage on ORD, the Human Benchmark, and USPTO, substantially outperforming uniform cost search and the Tanimoto heuristic. Relative to the Reaction Predictor, coverage improves by 71.39% on ORD and 24.89% on the Human Benchmark. The benefit of combining weights and the failure of uniform cost search on ORD demonstrate the importance of complementary, goal directed search. Despite using no expert annotated mechanisms for training, the method exactly recovers 31 pathways manually assessed as plausible (Table 3). Figure 3 further shows that energy screening can reject high energy candidates while retaining alternatives recovered at other weights. Therefore, post-hoc energy screening provides an additional assessment beyond successful goal recovery. Pathways with incomplete energy profiles are reported separately and excluded from the threshold counts.

Complete state expansion remains a computational bottleneck, with generation rates on the order of 10^1 nodes per second. One potential method to avoid complete state expansion would be the use of Q* search (Agostinelli et al., 2024b) instead of A* search, which uses deep Q-networks (Mnih et al., 2015) as heuristic functions to ensure the number of nodes generated during search is independent of the size of the action space.

6 Related Work

Miller et al. (2025) introduce PMechRP, which ranks candidate elementary steps and chains them using depth first search. PMechRP is trained on manually curated elementary steps from PMechDB and generated proton transfer examples. Unlike our approach, it relies on labeled elementary steps. Our approach uses OrbChain to generate possible successor states and learns search guidance from random walks. We use PMechRP as the Reaction Predictor baseline on ORD and the Human Benchmark. Tavakoli et al. (2024b) introduce RMechRP, which uses learned plausibility scores to rank candidate elementary steps for radical reactions and searches a reaction tree to construct mechanistic pathways. In contrast, our learned heuristic estimates the remaining path cost to the specified molecular products and guides weighted A*. In future work, learned step plausibility scores could be used to define transition costs in our search.

In retrosynthetic planning (Corey, 1988), Retro* (Chen et al., 2020) uses a learned synthesis cost to guide search over an AND-OR tree. Like our method, it combines learned guidance with search, but operates on retrosynthetic transformations rather than elementary reaction mechanism steps. Several mechanistic datasets and models generate pathway supervision by assigning reaction classes and applying expert curated templates (Joung et al., 2024; Chen et al., 2024; Joung et al., 2025). Our approach instead generates heuristic training targets through random walks using the OrbChain expansion function, without reaction class assignments or pathways imputed from templates. Its coverage therefore depends on the elementary transformations represented by OrbChain.

7 Future Work

Future work could replace or supplement OrbChain with an expansion function based on a curated library of reaction templates, which have previously been applied recursively to generate reaction intermediates and mechanistic datasets (Chen et al., 2024; Joung et al., 2024; 2025). Restricting expansion to template-matched transformations could reduce successor generation time, but its effect on expansion time and pathway coverage must be evaluated.

Our method assigns unit transition costs and evaluates pathway energies only after search, so energetic differences cannot guide pathway selection. The current energy calculations also use a single ETKDGv3 generated conformer for each molecular species because CREST conformer sampling was computationally prohibitive. Future work could use CREST (Pracht et al., 2020) to sample low energy conformers and compute ensemble free energies, thereby accounting for conformational variability. A surrogate model could then be trained on these free energy changes and used to define nonnegative transition costs. Following McDermott et al. (2021), downhill transitions could receive small positive costs while uphill transitions receive progressively larger costs. This would allow weighted A* to favor thermodynamically preferred pathways without running GFN2-xTB during every state expansion.

8 Conclusion

We introduce a goal conditioned heuristic for reaction mechanism pathway search trained by supervised regression on random walks generated using an elementary step expansion function. This trains a heuristic function without using expert-annotated data. The learned heuristic guides weighted A* search and achieves higher pathway coverage compared to all other methods. Finally, the post-hoc energy screening provides a coarse energetic screen to filter out implausible pathways. These results demonstrate that potentially plausible pathways can be found without expert-annotated data using supervised learning from random walks and post hoc energy screening.

AI USE STATEMENT

We used OpenAI Codex to assist with polishing the writing and drafting portions of the manuscript. We also used it to retrieve and summarize potentially relevant literature. For research execution, OpenAI Codex assisted in implementing training pipeline and writing code to aggregate experimental outputs, compute reported metrics, and generate LaTeX tables. Its role in research execution was limited to supporting implementation and analysis code. It was not used to originate the research question or design the core learning and search algorithms. The authors conducted the primary literature review, verified all cited sources, reviewed and tested AI generated code, verified the reported results against the source data, interpreted the experimental results, and reviewed all AI assisted manuscript revisions. The authors take responsibility for the final content of the paper.

ETHICS STATEMENT

This work uses existing reaction datasets and computational methods and does not involve human subjects or personally identifiable information. Automated reaction mechanism prediction could be applied to hazardous chemical transformations. The recovered pathways are computational candidates and should not be interpreted as experimentally validated procedures. We encourage responsible use and independent chemical evaluation of generated pathways.

REPRODUCIBILITY STATEMENT

To support reproducibility, the main paper defines the state representation, expansion function, random walk data generation procedure, heuristic training objective, search algorithm, and energy screening method. Appendix A.1 provides an OrbChain expansion example. Appendix A.2 reports the training data construction, model settings, hardware, and compute details, while Appendix A.3 reports the evaluation datasets, search weights, time limits, and testing configuration. Appendix A.4 provides detailed search results, and Appendix A.5 documents pathway preprocessing, free energy calculation settings, and energy screening statistics. Appendix A.6 compares the energy screening results with manual plausibility assessments. Appendices A.7 and provide qualitative pathway comparisons and exact recovery results, respectively. Appendix A.9 reports operation outcomes for the dataset generation procedure. To preserve anonymity during review, we will provide reviewers and area chairs with access to an anonymized code repository through a restricted OpenReview comment during the discussion period. We intend to release the code publicly upon acceptance.

REFERENCES

Forest Agostinelli, Stephen McAleer, Alexander Shmakov, and Pierre Baldi. Solving the rubik's cube with deep reinforcement learning and search. *Nature Machine Intelligence*, 1(8):356–363, 2019.

Forest Agostinelli, Rojina Panta, and Vedant Khandelwal. Specifying goals to deep neural networks with answer set programming. In *34th International Conference on Automated Planning and Scheduling*, 2024a. URL https://openreview.net/forum?id=x88vQjoZhK.

Forest Agostinelli, Shahaf S Shperberg, Alexander Shmakov, Stephen McAleer, Roy Fox, and Pierre Baldi. Q* search: Heuristic search with deep q-networks. In *ICAPS Workshop on Bridging the Gap between AI Planning and Reinforcement Learning*, 2024b.

Marcin Andrychowicz, Filip Wolski, Alex Ray, Jonas Schneider, Rachel Fong, Peter Welinder, Bob McGrew, Josh Tobin, OpenAI Pieter Abbeel, and Wojciech Zaremba. Hindsight experience replay. *Advances in neural information processing systems*, 30, 2017.

Christoph Bannwarth, Sebastian Ehlert, and Stefan Grimme. Gfn2-xtb—an accurate and broadly parametrized self-consistent tight-binding quantum chemical method with multipole electrostatics and density-dependent dispersion contributions. *Journal of chemical theory and computation*, 15 (3):1652–1671, 2019.

Binghong Chen, Chengtao Li, Hanjun Dai, and Le Song. Retro*: learning retrosynthetic planning with neural guided a* search. In *International conference on machine learning*, pp. 1608–1616. PMLR, 2020.

Shuan Chen, Ramil Babazade, Taewan Kim, Sunkyu Han, and Yousung Jung. A large-scale reaction dataset of mechanistic pathways of organic reactions. *Scientific Data*, 11(1):863, 2024.

Alexander Chervov, Kirill Khoruzhii, Nikita Bukhal, Jalal Naghiyev, Vladislav Zamkovoy, Ivan Koltsov, Lyudmila Cheldieva, Arsenii Sychev, Arsenii Lenin, Mark Obozov, Egor Urvanov, and Alexey Romanov. A machine learning approach that beats rubik's cubes. In D. Belgrave, C. Zhang, H. Lin, R. Pascanu, P. Koniusz, M. Ghassemi, and N. Chen (eds.), *Advances in Neural Information Processing Systems*, volume 38, Main Conference, pp. 172509–172531. Curran Associates, Inc., 2025. doi: 10.52202/085713-5741. URL https://proceedings.neurips.cc/paper_files/paper/2025/file/fb82011040977c7712409fbdb5456647-Paper-Conference.pdf.

EJ Corey. Robert robinson lecture. retrosynthetic thinking—essentials and examples. *Chemical society reviews*, 17:111–133, 1988.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 conference of the North American chapter of the association for computational linguistics: human language technologies, volume 1 (long and short papers)*, pp. 4171–4186, 2019.

Sebastian Ehlert, Marcel Stahn, Sebastian Spicher, and Stefan Grimme. Robust and efficient implicit solvation model for fast semiempirical methods. *Journal of Chemical Theory and Computation*, 17(7):4250–4261, 2021.

Darren R Flower. On the properties of bit string-based measures of chemical similarity. *Journal of chemical information and computer sciences*, 38(3):379–386, 1998.

Jeffrey W Godden, Ling Xue, and Jürgen Bajorath. Combinatorial preferences affect molecular similarity/diversity calculations using binary fingerprints and tanimoto coefficients. *Journal of Chemical Information and Computer Sciences*, 40(1):163–166, 2000.

Stefan Grimme. Supramolecular binding thermodynamics by dispersion-corrected density functional theory. *Chemistry–A European Journal*, 18(32):9955–9964, 2012. doi: 10.1002/chem.201200497.

Peter E Hart, Nils J Nilsson, and Bertram Raphael. A formal basis for the heuristic determination of minimum cost paths. *IEEE transactions on Systems Science and Cybernetics*, 4(2):100–107, 1968.

Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 770–778, 2016.

Ask Hjorth Larsen, Jens Jørgen Mortensen, Jakob Blomqvist, Ivano E Castelli, Rune Christensen, Marcin Dułak, Jesper Friis, Michael N Groves, Bjørk Hammer, Cory Hargus, et al. The atomic simulation environment—a python library for working with atoms. *Journal of Physics: Condensed Matter*, 29(27):273002, 2017.

Geoffrey A Holdgate, Thomas D Meek, and Rachel L Grimley. Mechanistic enzymology in drug discovery: a fresh perspective. *Nature reviews Drug discovery*, 17(2):115–132, 2018.

Joonyoung F Joung, Mun Hong Fong, Jihye Roh, Zhengkai Tu, John Bradshaw, and Connor W Coley. Reproducing reaction mechanisms with machine-learning models trained on a large-scale mechanistic dataset. *Angewandte Chemie International Edition*, 63(43):e202411296, 2024.

Joonyoung F Joung, Mun Hong Fong, Nicholas Casetti, Jordan P Liles, Ne S Dassanayake, and Connor W Coley. Electron flow matching for generative reaction mechanism prediction. *Nature*, 645(8079):115–123, 2025.

Steven M Kearnes, Michael R Maser, Michael Wleklinski, Anton Kast, Abigail G Doyle, Spencer D Dreher, Joel M Hawkins, Klavs F Jensen, and Connor W Coley. The open reaction database. *Journal of the American Chemical Society*, 143(45):18820–18826, 2021.

Daniel Lowe. Chemical reactions from us patents (1976-sep2016), 2017. URL https://doi.org/10.6084/m9.figshare.5104873.v1.

Matthew J McDermott, Shyam S Dwaraknath, and Kristin A Persson. A graph-based network for predicting chemical reaction pathways in solid-state materials synthesis. *Nature communications*, 12(1):3097, 2021.

Ryan J Miller, Alexander E Dashuta, Brayden Rudisill, David Van Vranken, and Pierre Baldi. Mechanism-aware deep learning for polar reaction prediction. *Journal of the American Chemical Society*, 147(44):41168, 2025.

Volodymyr Mnih, Koray Kavukcuoglu, David Silver, Andrei A Rusu, Joel Veness, Marc G Bellemare, Alex Graves, Martin Riedmiller, Andreas K Fidjeland, Georg Ostrovski, et al. Human-level control through deep reinforcement learning. *nature*, 518(7540):529–533, 2015.

David L. Nelson and Michael M. Cox. *Lehninger Principles of Biochemistry*. W. H. Freeman, New York, 4 edition, 2004.

OpenEye, Cadence Molecular Sciences. OEChem TK 4.1.1.1, 2024a. URL https://www.eyesopen.com.

OpenEye, Cadence Molecular Sciences. OEDepict TK 2.5.5.1, 2024b. URL https://www.eyesopen.com.

Ira Pohl. Heuristic search viewed as path finding in a graph. *Artificial intelligence*, 1(3-4):193–204, 1970.

Philipp Pracht, Fabian Bohle, and Stefan Grimme. Automated exploration of the low-energy chemical space with fast quantum chemical methods. *Physical Chemistry Chemical Physics*, 22(14): 7169–7192, 2020. doi: 10.1039/C9CP06869D.

Martin L Puterman and Moon Chirl Shin. Modified policy iteration algorithms for discounted markov decision problems. *Management Science*, 24(11):1127–1137, 1978.

Fenghe Qiu and Daniel L Norwood. Identification of pharmaceutical impurities. *Journal of liquid chromatography & related technologies*, 30(5-7):877–935, 2007.

RDKit. RDKit: Open-source cheminformatics. https://www.rdkit.org, 2026. Software.

David Rogers and Mathew Hahn. Extended-connectivity fingerprints. *Journal of chemical information and modeling*, 50(5):742–754, 2010.

Philippe Schwaller, Daniel Probst, Alain C Vaucher, Vishnu H Nair, David Kreutter, Teodoro Laino, and Jean-Louis Reymond. Mapping the space of chemical reactions using attention-based neural networks. *Nature machine intelligence*, 3(2):144–152, 2021.

M Tavakoli, RJ Miller, MC Angel, MA Pfeiffer, ES Gutman, AD Mood, D Van Vranken, and P Baldi. Pmechdb: A public database of elementary polar reaction steps. *Journal of Chemical Information and Modeling*, 64(6):1975–1983, 2024a.

Mohammadamin Tavakoli, Pierre Baldi, Ann Marie Carlton, Yin Ting Chiu, Alexander Shmakov, and David Van Vranken. Ai for interpretable chemistry: Predicting radical mechanistic pathways via contrastive learning. *Advances in Neural Information Processing Systems*, 36, 2024b.

Mohammadamin Tavakoli, Yin Ting T Chiu, Ann Marie Carlton, David Van Vranken, and Pierre Baldi. Chemically informed deep learning for interpretable radical reaction prediction. *Journal of Chemical Information and Modeling*, 65(3):1228, 2025.

Shuzhe Wang, Jarosław Witek, Gregory A. Landrum, and Sereina Riniker. Improving conformer generation for small rings and macrocycles based on distance geometry and experimental torsional-angle preferences. *Journal of Chemical Information and Modeling*, 60(4):2044–2058, 2020. doi: 10.1021/acs.jcim.0c00025.

David Weininger. Smiles, a chemical language and information system. 1. introduction to methodology and encoding rules. *Journal of chemical information and computer sciences*, 28(1):31–36, 1988.

David Weininger, Arthur Weininger, and Joseph L Weininger. Smiles. 2. algorithm for generation of unique smiles notation. *Journal of chemical information and computer sciences*, 29(2):97–101, 1989.

Richard Wolfenden and Mark J Snider. The depth of chemical time and the power of enzymes as catalysts. *Accounts of chemical research*, 34(12):938–945, 2001.

A Appendix

A.1 OrbChain State Expansion Example

Each search state is represented as a multiset of molecules, but OrbChain processes only the corresponding set of unique molecules during expansion. OrbChain identifies the applicable elementary steps for these molecules and generates the products associated with each step. Each generated product defines a successor state, while molecular multiplicities remain part of the search state representation and goal test. For polar reactions, we adapt OrbChain by using fully occupied donor orbitals and empty acceptor orbitals in place of the half occupied orbitals used for radical reactions. Figure 4 illustrates this expansion procedure.

The multiset representation preserves repeated molecular components for state bookkeeping and the goal test. The current OrbChain implementation operates on unique molecular structures rather than molecular multiplicities, so we convert the multiset to its corresponding set of unique molecules before expansion. Consequently, multiplicity is retained by the search framework but does not affect successor generation.

A.2 Training and Data Generation Details

Training examples were generated using 20 parallel Slurm jobs distributed across heterogeneous compute nodes. Each job was allocated 48 CPUs and executed 40 worker processes, giving a combined allocation of 960 CPUs. The training process maintained a replay buffer containing at most 300,000 examples. The buffer was initialized with 300,000 examples, and at each subsequent epoch, 50,000 newly generated examples replaced the oldest examples. Excluding interruptions caused by cancelled jobs, generating this dataset required about 15 days of active computation using up to 20 parallel Slurm jobs on heterogeneous compute nodes. This duration is an empirical system measurement rather than a standardized hardware benchmark.

We trained the heuristic for 600 epochs. At each epoch, the model was trained on the 300,000 examples in the replay buffer using minibatches of 200. We minimized mean squared error using the Adam optimizer with an initial learning rate of 10^{-4}. The learning rate was multiplied by 0.9999993 after each optimizer update. We used a fixed validation set containing 40,000 examples generated from the same USPTO source reactions and OrbChain expansion function as the training examples, but sampled in a separate generation run. We retained the checkpoint with the lowest validation mean squared error. A separate set of 40,000 examples generated using the same procedure was used only for final testing. Training required approximately 2.5 days using one GPU on the `gpu-A100` partition and eight allocated CPU cores.

A.3 Testing Details

For each evaluation reaction, we independently ran weighted A* using each search weight $w \in \{0.1, 0.3, 0.5, 0.6, 0.7, 1.0\}$. During search, negative neural heuristic predictions were clipped to zero. For a node that did not satisfy the goal condition, its priority was

$$f_w(n) = w\, g(n) + h_\theta(s_n, g), \tag{1}$$

where s_n is the state represented by node n, $g(n)$ is its path cost, and $h_\theta(s_n, g)$ is the learned estimate of the remaining cost to goal state g. For a goal node, the heuristic contribution was set

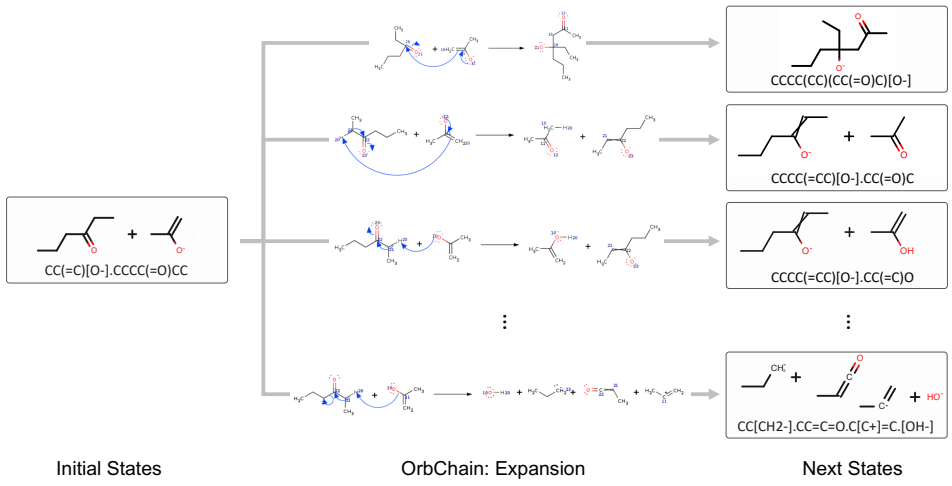


Figure 4: Example of an OrbChain state expansion. OrbChain processes the unique molecules in the current search state, enumerates applicable elementary reaction steps, and generates the corresponding successor states.

to zero, giving $f_w(n) = w\, g(n)$. Thus, smaller values of w gave greater relative influence to the learned heuristic. All transitions had unit cost.

At each iteration, weighted A* removed and expanded one node from OPEN. The heuristic values of its successor states were evaluated by the neural network in inference batches of at most 2000 states. A generated state was added to OPEN if it had not been encountered previously or if it was reached with a lower path cost. Ties in priority were resolved by insertion order. Each search terminated when a goal node was removed from OPEN, OPEN became empty, or the time limit of two hours was reached. For a solution, the pathway was reconstructed using parent pointers and verified by reapplying the recorded actions from the initial state.

A.4 Additional Search Results

Table 4 reports search performance for all tested weights, while Tables 5 and 6 group the solved reactions by solution cost and runtime. In table 4, we report the number and percentage of reactions solved. Path Cost, Nodes, and Secs denote the mean solution cost, number of generated nodes, and runtime, respectively, computed over solved reactions only. Nodes/Sec is the ratio of the mean number of generated nodes to the mean runtime.

Uniform cost search and the Tanimoto baseline primarily recover pathways with one or two steps, whereas the learned heuristic also recovers pathways with three or more steps. This indicates that learned goal directed guidance becomes more useful as the required pathway length and search space increase. On ORD and the Human Benchmark, the union across search weights recovers more pathways than the Reaction Predictor. Their solution cost distributions differ. The Reaction Predictor returns a larger proportion of pathways with three or more steps on the Human Benchmark, while the selected learned heuristic pathways are concentrated at shorter solution costs. This comparison does not establish that one method produces more mechanistically detailed pathways because the methods generate and rank pathways differently.

Most pathways recovered by the learned heuristic are found within the first 30 minutes, while additional pathways are recovered when search continues for up to two hours. Runtime values for the Reaction Predictor and the search methods are not directly comparable because they use different inference and pathway construction procedures. The union across search weights counts each solved reaction once. If multiple weights solve the same reaction, only one result is included in the union tables. We select the result with the lowest solution cost, followed by the fewest generated nodes and then the shortest runtime as tie breakers. The reported cost and runtime for that reaction therefore come from the selected result.

Table 4: Search performance across evaluation datasets and solver configurations. The union counts each reaction solved by any tested search weight once. When multiple weights solve the same reaction, we select the result with the lowest path cost, followed by the fewest generated nodes and then the shortest runtime as tie breaking criteria.

Dataset	Solver / Variant	Total	Solved	% Solved	Path Cost	Nodes	Secs	Nodes/Sec
ORD 133	Reaction Predictor	133	35	26.32%	2.23	–	99.33	–
ORD 133	Learned Heuristic (w=0.1)	133	43	32.33%	2.56	4.60E+04	1440.73	31.91
ORD 133	Learned Heuristic (w=0.3)	133	46	34.59%	2.04	4.51E+04	1412.37	31.93
ORD 133	Learned Heuristic (w=0.5)	133	56	42.11%	2.00	5.54E+04	1753.87	31.60
ORD 133	Learned Heuristic (w=0.6)	133	54	40.60%	1.98	4.30E+04	1489.23	28.87
ORD 133	Learned Heuristic (w=0.7)	133	53	39.85%	1.92	4.42E+04	1498.14	29.52
ORD 133	Learned Heuristic (w=1)	133	45	33.83%	1.76	3.36E+04	1148.07	29.29
ORD 133	Learned Heuristic (w union)	133	**60**	**45.11%**	2.00	4.65E+04	1531.10	30.36
ORD 133	Uniform Cost Search	133	0	0.00%	–	–	–	–
ORD 133	Tanimoto Similarity	133	19	14.29%	1.05	4.75E+04	1461.03	**32.50**
Human 350	Reaction Predictor	350	257	73.43%	2.67	–	500.94	–
Human 350	Learned Heuristic (w=0.1)	350	282	80.57%	2.27	1.59E+04	428.49	37.03
Human 350	Learned Heuristic (w=0.3)	350	293	83.71%	2.04	1.74E+04	472.86	36.82
Human 350	Learned Heuristic (w=0.5)	350	310	88.57%	1.96	2.20E+04	599.20	36.74
Human 350	Learned Heuristic (w=0.6)	350	312	89.14%	1.92	2.33E+04	650.35	35.90
Human 350	Learned Heuristic (w=0.7)	350	310	88.57%	1.89	2.15E+04	596.97	36.04
Human 350	Learned Heuristic (w=1)	350	305	87.14%	1.83	2.45E+04	672.68	36.45
Human 350	Learned Heuristic (w union)	350	**321**	**91.71%**	1.93	2.28E+04	612.90	37.24
Human 350	Uniform Cost Search	350	29	8.29%	1.24	1.73E+05	2764.57	**62.64**
Human 350	Tanimoto Similarity	350	136	38.86%	1.29	5.40E+04	992.43	54.46
USPTO 476	Learned Heuristic (w=0.1)	476	348	73.11%	2.98	1.91E+04	365.20	52.26
USPTO 476	Learned Heuristic (w=0.3)	476	353	74.16%	2.62	1.72E+04	343.82	49.94
USPTO 476	Learned Heuristic (w=0.5)	476	358	75.21%	2.43	1.82E+04	375.77	48.46
USPTO 476	Learned Heuristic (w=0.6)	476	362	76.05%	2.40	1.98E+04	397.30	49.74
USPTO 476	Learned Heuristic (w=0.7)	476	362	76.05%	2.32	2.10E+04	411.80	50.95
USPTO 476	Learned Heuristic (w=1)	476	357	75.00%	2.19	2.16E+04	443.78	48.61
USPTO 476	Learned Heuristic (w union)	476	**379**	**79.62%**	2.40	2.41E+04	496.89	48.45
USPTO 476	Uniform Cost Search	476	16	3.36%	1.00	2.83E+05	3647.05	**77.59**
USPTO 476	Tanimoto Similarity	476	100	21.01%	1.20	6.24E+04	893.55	69.81

A.5 Energy Details

A.5.1 Pathway Balancing

Before energy calculations, each recovered pathway is parsed into its sequence of states. Atom map numbers are removed, each molecular component is canonicalized, and repeated components are preserved. For every component, we compute its elemental composition, including hydrogen atoms represented implicitly in the molecular representation, and its total formal charge. We then compare these quantities between every pair of adjacent pathway states.

OrbChain may omit an unchanged molecular component when constructing a successor state. To address this, we compute the difference in elemental composition and total formal charge between adjacent states. When this difference uniquely matches one or more complete molecular components present in the earlier state, those components are copied into the later state. The repair is accepted only when the matching combination is unique. Components introduced by a repair are propagated forward through subsequent states when necessary to maintain atom and charge balance throughout the pathway. For example, suppose

$$s_0 = \{A, S\}, \qquad s_1 = \{B\}, \qquad s_2 = \{C\},$$

where A, B, and C have equal elemental compositions and formal charges. If the first transition is missing exactly the complete component S, we copy S from s_0 into s_1. Since the same component is then missing from s_2, it is propagated forward:

$$s_0 = \{A, S\}, \qquad s_1 = \{B, S\}, \qquad s_2 = \{C, S\}.$$

Table 5: Reactions solved by each method, grouped by solution path cost. The cost columns include only solved reactions and report the number with path cost 1, 2, 3, or 4 or greater.

Dataset	Solver / Variant	Total	Solved	% Solved	Cost 1	Cost 2	Cost 3	Cost 4+
ORD 133	Reaction Predictor	133	35	26.32%	1	25	9	0
ORD 133	Learned Heuristic (w=0.1)	133	43	32.33%	14	13	10	**6**
ORD 133	Learned Heuristic (w=0.3)	133	46	34.59%	15	17	**12**	2
ORD 133	Learned Heuristic (w=0.5)	133	56	42.11%	15	30	8	3
ORD 133	Learned Heuristic (w=0.6)	133	54	40.60%	14	30	8	2
ORD 133	Learned Heuristic (w=0.7)	133	53	39.85%	14	31	7	1
ORD 133	Learned Heuristic (w=1)	133	45	33.83%	14	28	3	0
ORD 133	Learned Heuristic (w union)	133	**60**	**45.11%**	15	**36**	6	3
ORD 133	Uniform Cost Search	133	0	0.00%	0	0	0	0
ORD 133	Tanimoto Similarity	133	19	14.29%	**18**	1	0	0
Human 350	Reaction Predictor	350	257	73.43%	31	95	**89**	**42**
Human 350	Learned Heuristic (w=0.1)	350	282	80.57%	**97**	110	38	37
Human 350	Learned Heuristic (w=0.3)	350	293	83.71%	**97**	126	46	24
Human 350	Learned Heuristic (w=0.5)	350	310	88.57%	**97**	153	42	18
Human 350	Learned Heuristic (w=0.6)	350	312	89.14%	**97**	164	36	15
Human 350	Learned Heuristic (w=0.7)	350	310	88.57%	**97**	166	36	11
Human 350	Learned Heuristic (w=1)	350	305	87.14%	**97**	171	29	8
Human 350	Learned Heuristic (w union)	350	**321**	**91.71%**	**97**	**177**	33	14
Human 350	Uniform Cost Search	350	29	8.29%	22	7	0	0
Human 350	Tanimoto Similarity	350	136	38.86%	**97**	39	0	0
USPTO 476	Learned Heuristic (w=0.1)	476	348	73.11%	**80**	130	46	**92**
USPTO 476	Learned Heuristic (w=0.3)	476	353	74.16%	**80**	142	53	78
USPTO 476	Learned Heuristic (w=0.5)	476	358	75.21%	**80**	156	63	59
USPTO 476	Learned Heuristic (w=0.6)	476	362	76.05%	**80**	162	64	56
USPTO 476	Learned Heuristic (w=0.7)	476	362	76.05%	**80**	173	64	45
USPTO 476	Learned Heuristic (w=1)	476	357	75.00%	**80**	183	62	32
USPTO 476	Learned Heuristic (w union)	476	**379**	**79.62%**	**80**	**184**	**66**	49
USPTO 476	Uniform Cost Search	476	16	3.36%	16	0	0	0
USPTO 476	Tanimoto Similarity	476	100	21.01%	**80**	20	0	0

If more than one combination of existing components could account for the difference, the repair is marked as ambiguous. A pathway is excluded from energy calculations when the ambiguity remains unresolved after propagation. The balancing procedure only carries complete existing molecular components between states. It does not create new molecules, modify molecular connectivity, or infer missing reaction products. We do not remove components that occur unchanged throughout the pathway. After repair, every adjacent state pair is checked again for equality of elemental composition and total formal charge. Only pathways for which every transition passes this check are passed to the energy calculation pipeline. Pathways with parsing failures or with unresolved atom, charge, or component differences after propagation are excluded and recorded separately. This includes differences only in charge, mixed fragment differences, missing components, and repair ambiguities that remain unresolved. Changes in radical electron counts are recorded as warnings but do not independently cause a pathway to fail the balance check.

A.5.2 Free Energy Profile Computation

We identify each distinct molecular species appearing in the retained pathways and compute its free energy once. The computed value is reused when the same species appears in multiple states or pathways. One initial three-dimensional conformer is generated for each species using ETKDGv3 as implemented in RDKit (RDKit; Wang et al., 2020), with a random seed of 42, followed by force field pre-optimization using MMFF94 or UFF. Our implementation supports CREST conformer sampling (Pracht et al., 2020) and Boltzmann averaging over the resulting conformer ensemble. However, we did not enable these options for the reported experiments because of their computational cost. Each molecular species is therefore represented by a single ETKDGv3 conformer. Each generated structure is optimized using GFN2-xTB (Bannwarth et al., 2019) with analytical lin-

Table 6: Solved reactions grouped by runtime interval.

Dataset	Solver / Variant	Total	Solved	% Solved	$<$ 1800 s	1800–3600 s	3600–7200 s	$\geq$ 7200 s
ORD 133	Reaction Predictor	133	35	26.32%	35	0	0	0
ORD 133	Learned Heuristic (w=0.1)	133	43	32.33%	34	3	6	0
ORD 133	Learned Heuristic (w=0.3)	133	46	34.59%	35	5	6	0
ORD 133	Learned Heuristic (w=0.5)	133	56	42.11%	37	**11**	**8**	0
ORD 133	Learned Heuristic (w=0.6)	133	54	40.60%	40	8	6	0
ORD 133	Learned Heuristic (w=0.7)	133	53	39.85%	39	7	7	0
ORD 133	Learned Heuristic (w=1)	133	45	33.83%	34	9	2	0
ORD 133	Learned Heuristic (w union)	133	**60**	**45.11%**	**44**	9	7	0
ORD 133	Uniform Cost Search	133	0	0.00%	0	0	0	0
ORD 133	Tanimoto Similarity	133	19	14.29%	13	4	2	0
Human 350	Reaction Predictor	350	257	73.43%	239	16	1	**1**
Human 350	Learned Heuristic (w=0.1)	350	282	80.57%	267	9	6	0
Human 350	Learned Heuristic (w=0.3)	350	293	83.71%	274	13	6	0
Human 350	Learned Heuristic (w=0.5)	350	310	88.57%	283	14	13	0
Human 350	Learned Heuristic (w=0.6)	350	312	89.14%	278	**20**	14	0
Human 350	Learned Heuristic (w=0.7)	350	310	88.57%	283	16	11	0
Human 350	Learned Heuristic (w=1)	350	305	87.14%	275	12	**18**	0
Human 350	Learned Heuristic (w union)	350	**321**	**91.71%**	**292**	13	16	0
Human 350	Uniform Cost Search	350	29	8.29%	13	4	12	0
Human 350	Tanimoto Similarity	350	136	38.86%	113	8	15	0
USPTO 476	Learned Heuristic (w=0.1)	476	348	73.11%	335	7	6	0
USPTO 476	Learned Heuristic (w=0.3)	476	353	74.16%	339	11	3	0
USPTO 476	Learned Heuristic (w=0.5)	476	358	75.21%	345	7	6	0
USPTO 476	Learned Heuristic (w=0.6)	476	362	76.05%	342	13	7	0
USPTO 476	Learned Heuristic (w=0.7)	476	362	76.05%	341	13	8	0
USPTO 476	Learned Heuristic (w=1)	476	357	75.00%	339	10	8	0
USPTO 476	Learned Heuristic (w union)	476	**379**	**79.62%**	**353**	**14**	**12**	0
USPTO 476	Uniform Cost Search	476	16	3.36%	5	2	9	0
USPTO 476	Tanimoto Similarity	476	100	21.01%	83	7	10	0

earized Poisson-Boltzmann (ALPB) implicit water solvation (Ehlert et al., 2021) and the L-BFGS optimizer implemented in ASE (Hjorth Larsen et al., 2017). A numerical Hessian is computed using finite differences to obtain vibrational frequencies. The absolute free energy is

$$G = E_{\text{elec}} + \text{ZPVE} + H_{\text{vib}} + H_{\text{trans}} + H_{\text{rot}} + pV - T(S_{\text{vib}} + S_{\text{trans}} + S_{\text{rot}}), \tag{2}$$

where translational and rotational partition functions are evaluated analytically from molecular mass and the inertia tensor. Vibrational entropy is corrected using Grimme's quasi-RRHO scheme Grimme (2012), which interpolates smoothly between harmonic-oscillator and free-rotor entropy for modes below a threshold of $\omega_0 = 100\ \text{cm}^{-1}$, eliminating the well-known singularity of the harmonic approximation at low frequencies:

$$S_{\text{vib}} = w \cdot S_{\text{HO}} + (1 - w) \cdot S_{\text{FR}}, \qquad w = \frac{1}{1 + (\omega_0/\nu)^4}. \tag{3}$$

Because one conformer is used for each molecular species, no conformer ensemble averaging is performed. The free energy of each reaction state is the sum of the molecular free energies of its constituent species. The relative free energy profile is constructed as $\Delta G(\text{state}_n) = G(\text{state}_n) - G(\text{state}_0)$. This additive approximation treats the molecular species within each state as noninteracting.

A.5.3 Energy Threshold Counts

Table 7 compares search coverage with counts of signed free energy changes between adjacent states. An intermediate change refers to the signed free energy difference $\Delta G_{p,i}$ between two adjacent pathway states, rather than the absolute free energy of an individual intermediate. The

Table 7: Search coverage and signed adjacent state free energy changes in complete pathway profiles. Thresholds are in kcal mol^{-1}.

Dataset	Source	Total	Solved	Solved (%)	All intermediate changes in complete pathways	$\Delta G_{p,i} \leq 50$	$\Delta G_{p,i} \leq 20$
ORD	Reaction Predictor	133	35	26.32	69	**50**	**46**
ORD	Learned Heuristic (best, w=0.5)	133	56	42.11	94	41	37
ORD	Learned Heuristic (w union)	133	**60**	**45.11**	**108**	48	43
ORD	Uniform Cost Search	133	0	0.00	–	–	–
ORD	Tanimoto Similarity	133	19	14.29	18	2	1
Human	Reaction Predictor	350	257	73.43	**639**	**462**	**429**
Human	Learned Heuristic (best, w=0.6)	350	312	89.14	500	259	234
Human	Learned Heuristic (w union)	350	**321**	**91.71**	550	300	268
Human	Uniform Cost Search	350	29	8.29	28	11	11
Human	Tanimoto Similarity	350	136	38.86	157	72	64
USPTO	Learned Heuristic (best, w=0.7)	476	362	76.05	614	320	283
USPTO	Learned Heuristic (w union)	476	**379**	**79.62**	**785**	**427**	**372**
USPTO	Uniform Cost Search	476	16	3.36	15	11	10
USPTO	Tanimoto Similarity	476	100	21.01	104	39	37

Solved columns report the number and percentage of reactions for which each method recovered a pathway. Let $\mathcal{P}_{\text{complete}}$ denote the set of pathways with complete energy profiles. For a pathway $p = (s_{p,0}, s_{p,1}, \ldots, s_{p,L_p}) \in \mathcal{P}_{\text{complete}}$, let $\Delta G_{p,i} = G(s_{p,i+1}) - G(s_{p,i})$ denote the signed free energy change between adjacent states, where $G(s)$ is the computed free energy of state s and $i = 0, \ldots, L_p - 1$ indexes the pathway transitions. The number of individual adjacent changes satisfying threshold τ is

$$C_\tau = \sum_{p \in \mathcal{P}_{\text{complete}}} \#\left\{i : \Delta G_{p,i} \leq \tau\right\}. \tag{4}$$

The threshold columns report C_τ for $\tau \in \{20, 50\}$ kcal mol^{-1}. Each pathway can therefore contribute multiple adjacent changes. By contrast, Table 2 reports the number of complete pathways for which every adjacent change satisfies the threshold. For example, consider a pathway with adjacent changes $(-10, 15, 60)$ kcal mol^{-1}. At $\tau = 50$ kcal mol^{-1}, two changes contribute to C_{50}, but the pathway does not pass the pathway level screen because $\Delta G_{\max} = 60$ kcal mol^{-1}.

For the learned heuristic union, each reaction is represented by at most one complete pathway. If several search weights recover complete pathways for the same reaction, we select the pathway with the smallest $\Delta G_{\max}$. The intermediate changes from this selected pathway contribute to the union counts. Reactions for which no search weight produces a complete energy profile are excluded from the energy counts. For rows labeled *best*, we select the individual weight with the highest number of solved reactions. Ties are resolved using the number of complete energy profiles, followed by the numbers passing the 50 and 20 kcal mol^{-1} thresholds and then the smaller weight. Because the energy screening is applied after search, the screening threshold can be adjusted without rerunning the pathway search. We use 50 kcal mol^{-1} as the permissive primary threshold and report 20 kcal mol^{-1} as a stricter sensitivity threshold to illustrate how the screen can be adapted to different requirements.

A.6 Agreement with Manual Plausibility Assessment

For Reaction Predictor pathways (Miller et al., 2025), we compare the energy screen with the available manual plausibility labels. Let $\mathcal{M}$ denote the set of pathways manually assessed as plausible, and let $\mathcal{M}_{\text{complete}} \subseteq \mathcal{M}$ denote the manually plausible pathways with complete energy profiles. Let $\mathcal{E}_\tau$ denote the set of pathways with complete energy profiles satisfying $\Delta G_{\max} \leq \tau$. The plausible overlap is $|\mathcal{M} \cap \mathcal{E}_\tau|$, the number of pathways that pass the energy screen and are manually assessed as plausible. We define

$$\text{Plausibility rate} = \frac{|\mathcal{M} \cap \mathcal{E}_\tau|}{|\mathcal{E}_\tau|} \times 100\%, \tag{5}$$

Table 8: Agreement between manual plausibility assessments and the adjacent state energy screen for Reaction Predictor pathways. Screened counts include only complete energy profiles. Thresholds are in kcal mol^{-1}.

Dataset	Threshold τ	Manually plausible $\lvert\mathcal{M}\rvert$	Plausible with complete energy $\lvert\mathcal{M}_{\text{complete}}\rvert$	Energy filtered $\lvert\mathcal{E}_\tau\rvert$	Plausible overlap $\lvert\mathcal{M}\cap\mathcal{E}_\tau\rvert$	Plausibility rate (%)	Plausible pathway coverage (%)
Human	50	196	185	97	85	87.63%	45.95%
Human	20	196	185	82	74	90.24%	40.00%
ORD	50	21	20	14	9	64.29%	45.00%
ORD	20	21	20	12	7	58.33%	35.00%

which measures the proportion of pathways passing the energy screen that are also manually assessed as plausible. We define

$$\text{Plausible pathway coverage} = \frac{|\mathcal{M}\cap\mathcal{E}_\tau|}{|\mathcal{M}_{\text{complete}}|}\times 100\%, \tag{6}$$

which measures the proportion of manually plausible pathways with complete energy profiles that pass the energy threshold. The two metrics provide complementary information but do not establish pathway feasibility. They quantify agreement between the energy screen and the available manual assessments in two directions. Plausibility rate characterizes agreement among the pathways retained by the screen, whereas plausible pathway coverage characterizes how much of the manually plausible set is preserved. Together, they show how often the energy screen agrees with manual assessment and how many manually plausible pathways it excludes. Results for both thresholds are reported in Table 8.

A.7 Qualitative Pathway Comparisons

Figures 5–9 compare pathways for five reactions selected from the Human Benchmark dataset of 350 reactions. For each reaction, the pathway returned by the Reaction Predictor is compared with the pathway found using the Learned Heuristic with $w = 1$. In each energy profile, the initial pathway state is assigned a relative free energy of 0 kcal mol^{-1} and serves as the reference point. Subsequent state energies are reported relative to this initial state, while the arrows report signed free energy changes between adjacent states. The 50 kcal mol^{-1} screen is applied to each adjacent uphill change rather than to the energy of a state relative to zero.

The five comparisons illustrate several possible outcomes. For RXN000051, both methods return the same pathway and intermediate, and both pathways pass the energy screen with adjacent energy changes of -2.9 and -6.6 kcal mol^{-1}. For RXN000095, the methods reach the same target through different intermediates, but both pathways still pass. The Reaction Predictor pathway has adjacent changes of -44.6 and -9.0 kcal mol^{-1}, while the Learned Heuristic pathway has changes of -22.5 and -31.1 kcal mol^{-1}. By contrast, both methods reach the same target for RXN000046, but only the Reaction Predictor pathway passes the energy screen. Its adjacent energy changes are -7.7 and -1.6 kcal mol^{-1}, whereas the Learned Heuristic pathway contains an uphill change of 210.1 kcal mol^{-1} and therefore fails the 50 kcal mol^{-1} threshold.

The remaining examples highlight differences between manual plausibility assessments and the energy screen. For RXN000171, the Reaction Predictor pathway was manually classified as chemically implausible and contains an uphill change of 145.4 kcal mol^{-1}. The Learned Heuristic instead finds a direct downhill transformation of -26.1 kcal mol^{-1}. For RXN000134, the Reaction Predictor pathway was manually classified as chemically plausible, but its 127.4 kcal mol^{-1} uphill change fails the energy screen. The Learned Heuristic finds an alternative with adjacent changes of -3.5 and -46.5 kcal mol^{-1}, both of which pass the screen.

Figure 10 provides another example of how our approach of varying search weight affects pathway selection. For RXN000153, the Reaction Predictor pathway was manually assessed as plausible by human experts. Our method recovers alternative pathways across the evaluated weights. The pathways recovered with $w \in \{0.1, 0.3, 0.5\}$ fail both energy thresholds, whereas those recovered with

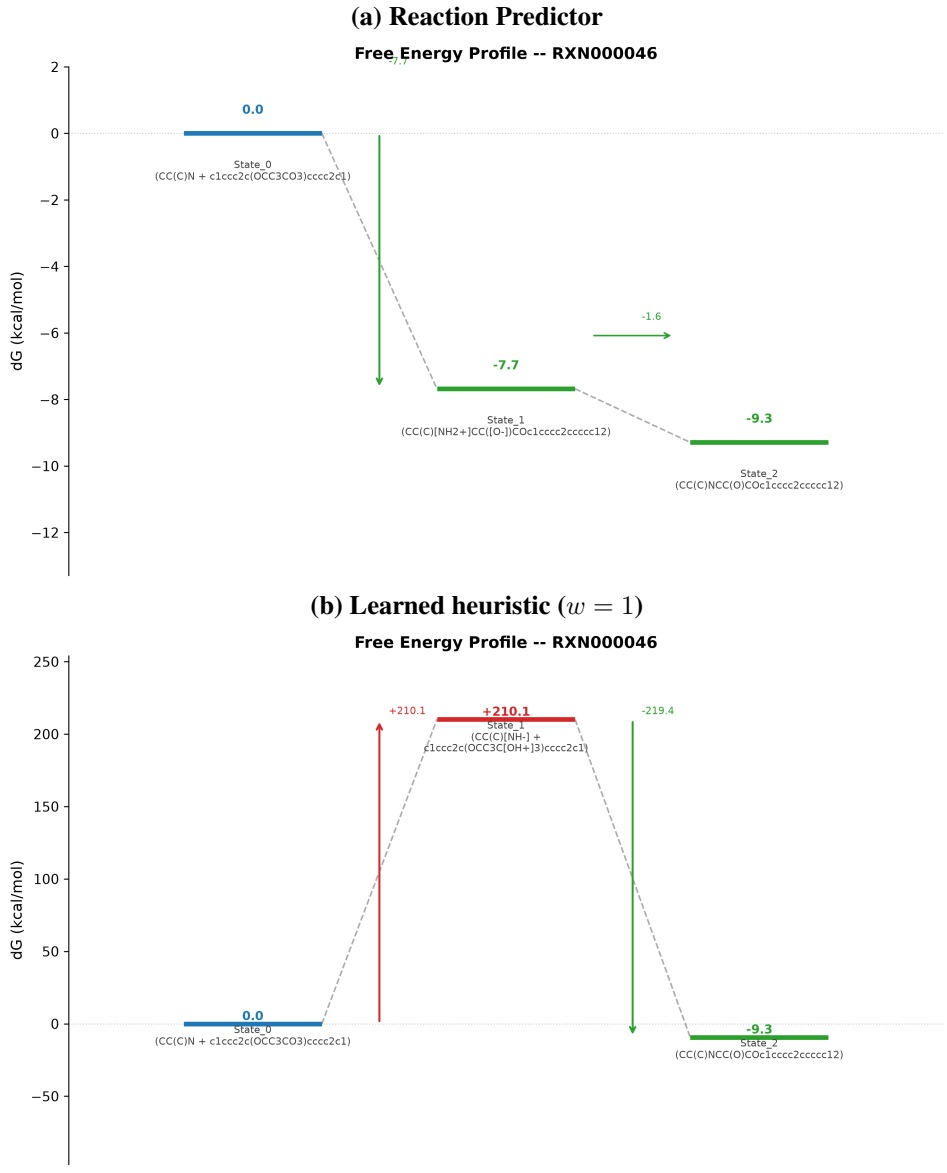


Figure 5: Pathway comparison for RXN000046. Both methods reach the same target using different intermediates. The Reaction Predictor pathway passes the energy screen, whereas the Learned Heuristic pathway contains a 210.1 kcal mol^{-1} uphill step and fails the 50 kcal mol^{-1} threshold.

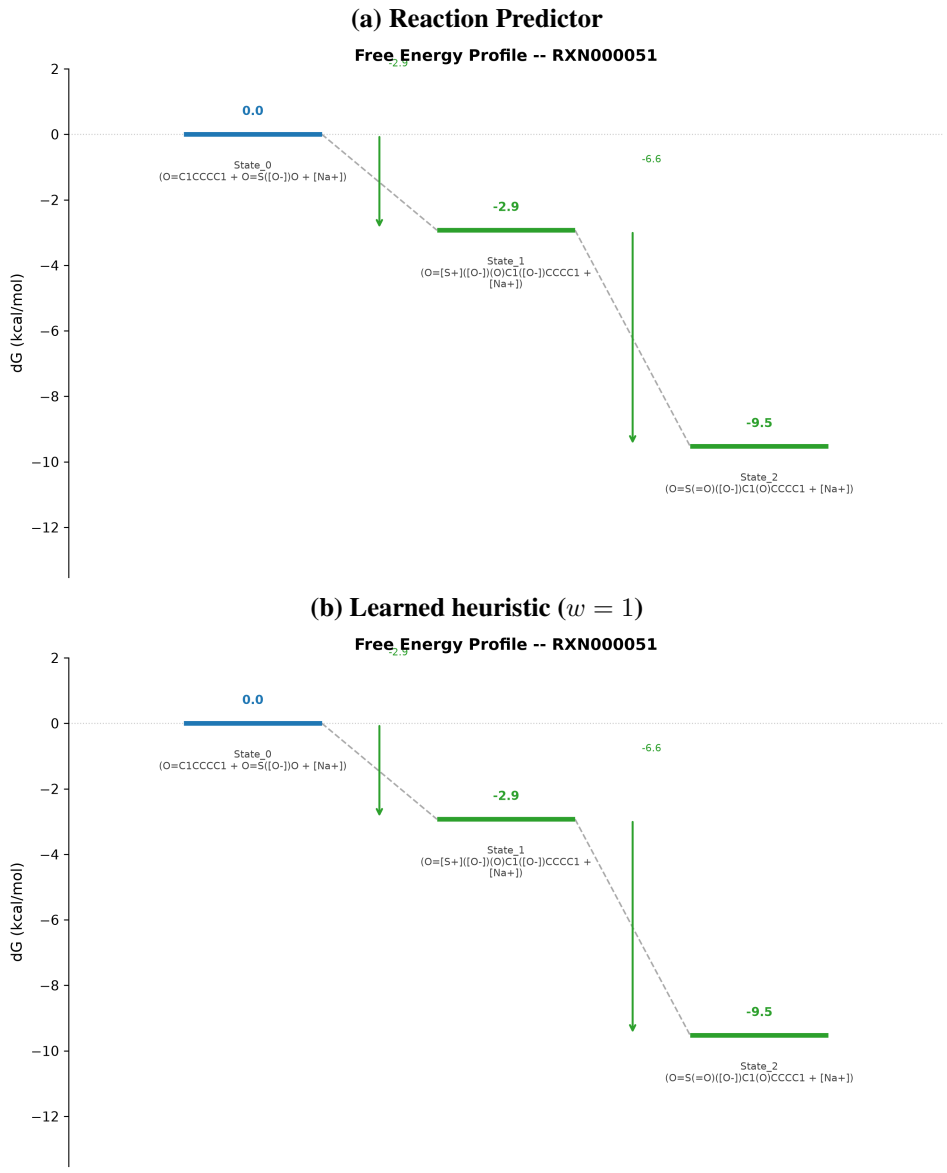


Figure 6: Pathway comparison for RXN000051. Both methods return the same intermediate sequence and both pathways pass the energy screen.

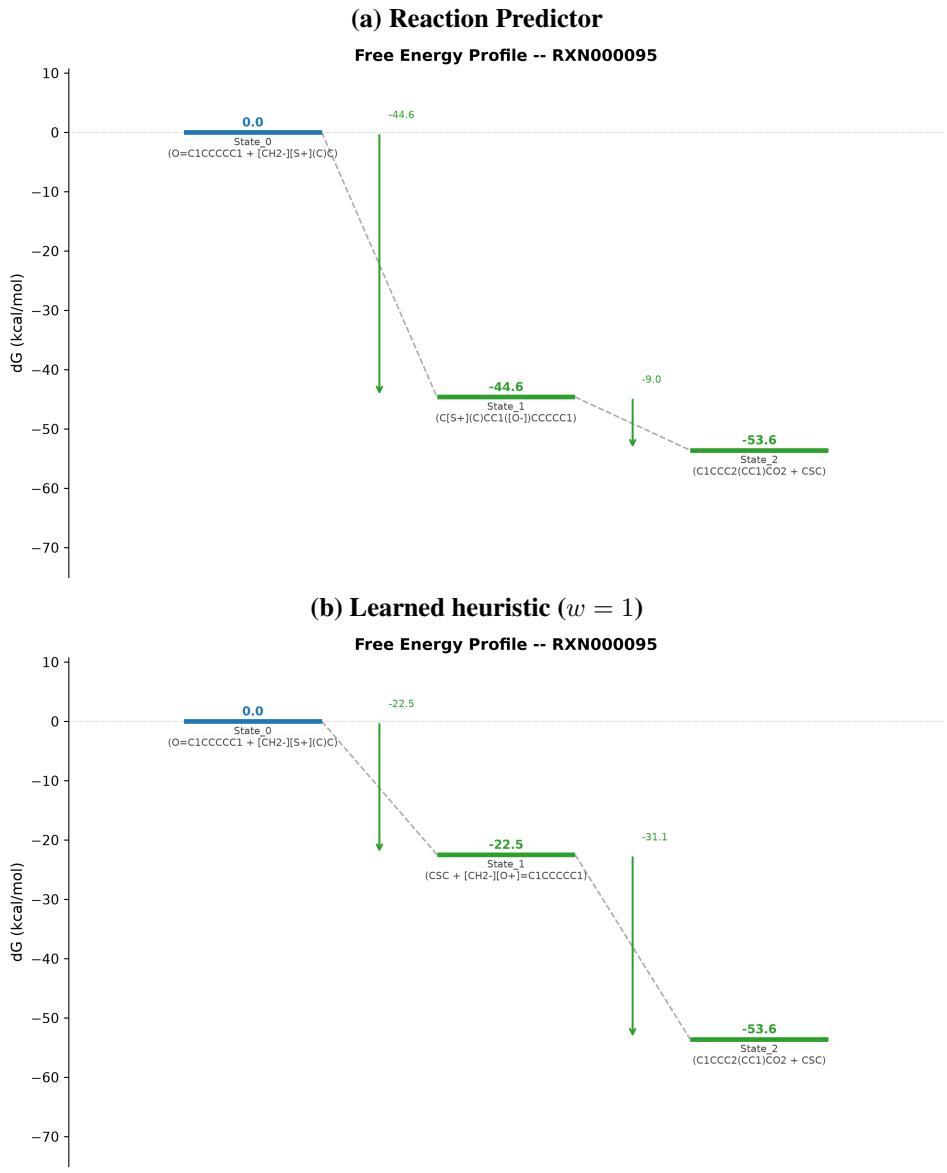
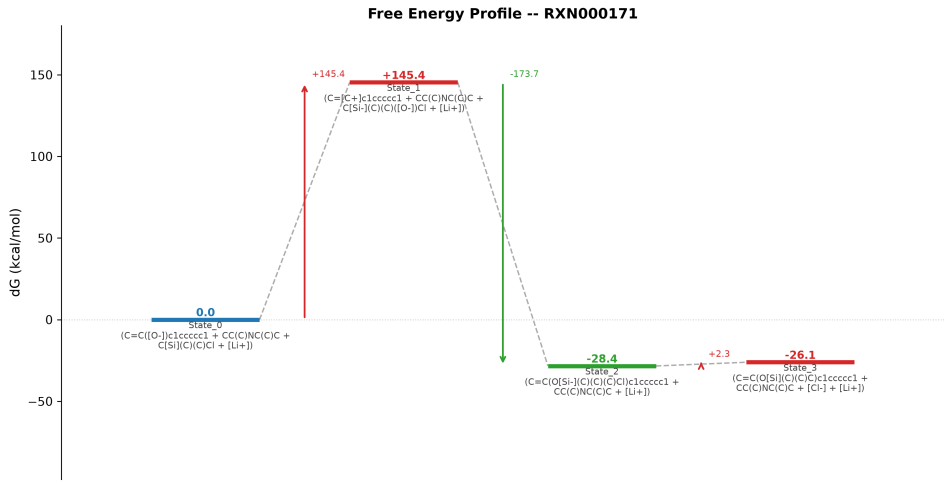


Figure 7: Pathway comparison for RXN000095. Both methods reach the same target through different intermediates, and both pathways pass the energy screen.

(a) Reaction Predictor



(b) Learned heuristic ($w = 1$)

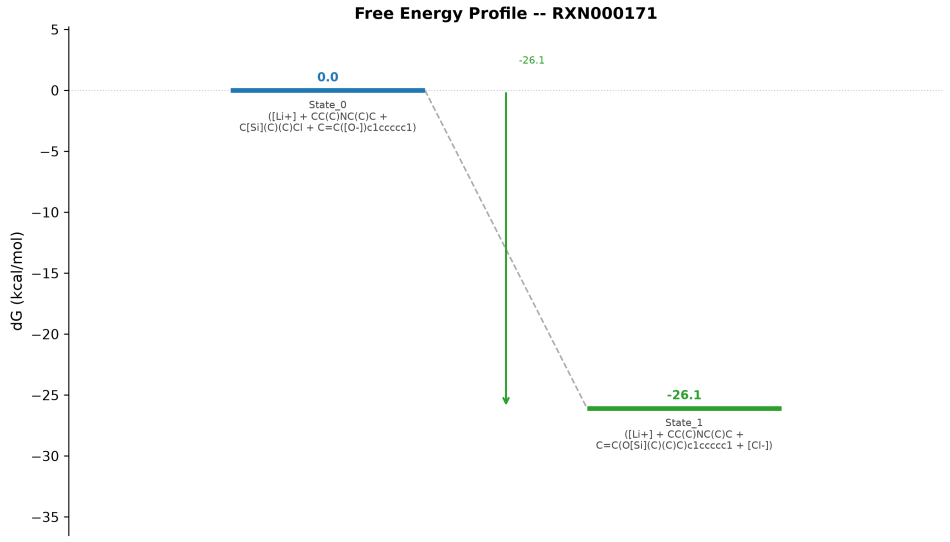


Figure 8: Pathway comparison for RXN000171. The Reaction Predictor introduces a high energy intermediate and fails the $50\ \mathrm{kcal\,mol^{-1}}$ screen with a $145.4\ \mathrm{kcal\,mol^{-1}}$ uphill step. The Learned Heuristic instead reaches the target in a single downhill step of $-26.1\ \mathrm{kcal\,mol^{-1}}$ and passes the energy screen.

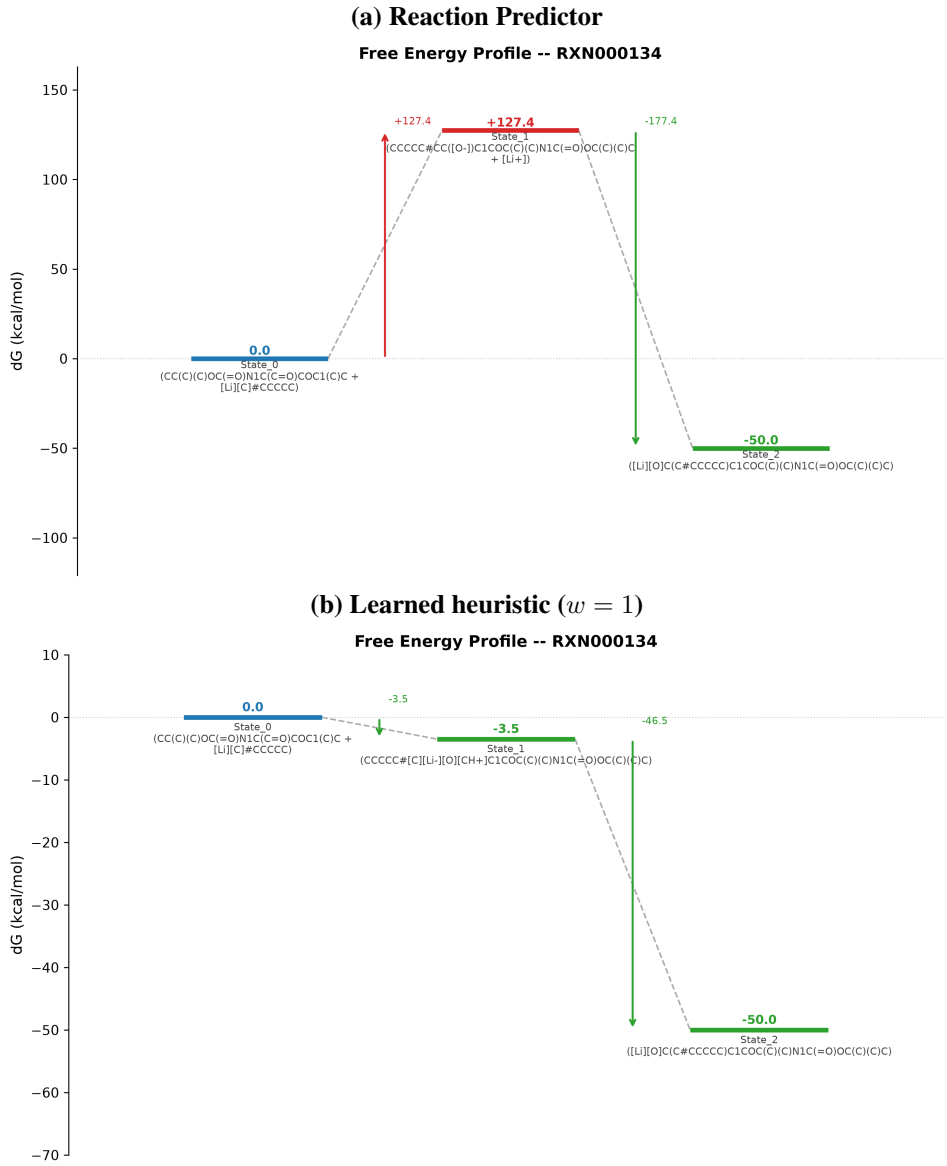


Figure 9: Pathway comparison for RXN000134. The initial manual plausibility assessment classified the Reaction Predictor pathway as chemically plausible, but its energy profile contains a 127.4 $\mathrm{kcal\,mol^{-1}}$ uphill step and fails the 50 $\mathrm{kcal\,mol^{-1}}$ screen. The Learned Heuristic passes the screen, with adjacent energy changes of -3.5 and -46.5 $\mathrm{kcal\,mol^{-1}}$.

Table 9: Operation outcomes for 100 sampled random walk states from each dataset across five trials. Expansion success, expansion timeouts, zero successors, and random successor success are reported as counts out of 100 attempted states.

Dataset	Expansion success	Expansion timeouts	Zero successors	Random successor success
ORD	92	6	2	98
Human Benchmark	100	0	0	100
USPTO	100	0	0	100

$w \in \{0.6, 0.7, 1.0\}$ pass both thresholds. The union therefore retains an energetically feasible alternative recovered by the latter weights. This example demonstrates how searching across multiple weights expands the set of candidate pathways, while post hoc energy screening rejects high energy alternatives and retains a pathway satisfying the specified energy thresholds.

A.8 Automated Screening of Pathways Assessed as Plausible by Human Experts

The Reaction Predictor generated pathways that were subsequently reviewed and assessed as plausible by human experts. We compared these pathways with those generated by the Learned Heuristic, which was trained without expert annotated data. Both the Reaction Predictor and Learned Heuristic pathways used in this comparison are balanced A.5. An exact structure match requires the complete ordered sequence of canonical intermediate states to be identical in both pathways. Using this criterion, the Learned Heuristic recovered 32 pathways that exactly matched Reaction Predictor pathways assessed as plausible by human experts. Of these, 31 had complete energy profiles in both results.

The automated energy screen classified 15 of the 31 complete exact matches as plausible at the $\Delta G_{\max} \leq 50$ kcal mol^{-1} threshold. Fourteen also passed the stricter $\Delta G_{\max} \leq 20$ kcal mol^{-1} threshold. RXN000129 passed the 50 kcal mol^{-1} threshold but not the 20 kcal mol^{-1} threshold. RXN000121 was not screened because its Reaction Predictor energy profile was incomplete.

These results show that the Learned Heuristic can recover complete pathways previously assessed as plausible by human experts without using human supervision during training. The automated energy screen retained 48.4% of the exact matches with complete energy profiles at the 50 kcal mol^{-1} threshold and 45.2% at the 20 kcal mol^{-1} threshold. This agreement supports the use of energy screening to identify plausible pathway candidates among pathways generated by the Learned Heuristic. The results for the pathways where they match the exact structure and are below threshold are shown in 11 to 14.

A.9 Operation Outcomes for Dataset Generation

For each dataset, we evaluated 100 random walk states across five trials of 20 states. We applied complete expansion and random successor generation to each state. Complete expansion was classified as successful when it finished within 30 minutes and generated at least one successor. Calls exceeding this limit were classified as timeouts, while completed calls returning no successors were recorded separately. Random successor generation had no configured timeout and was classified as successful when it returned one valid successor. The results are shown in 9.

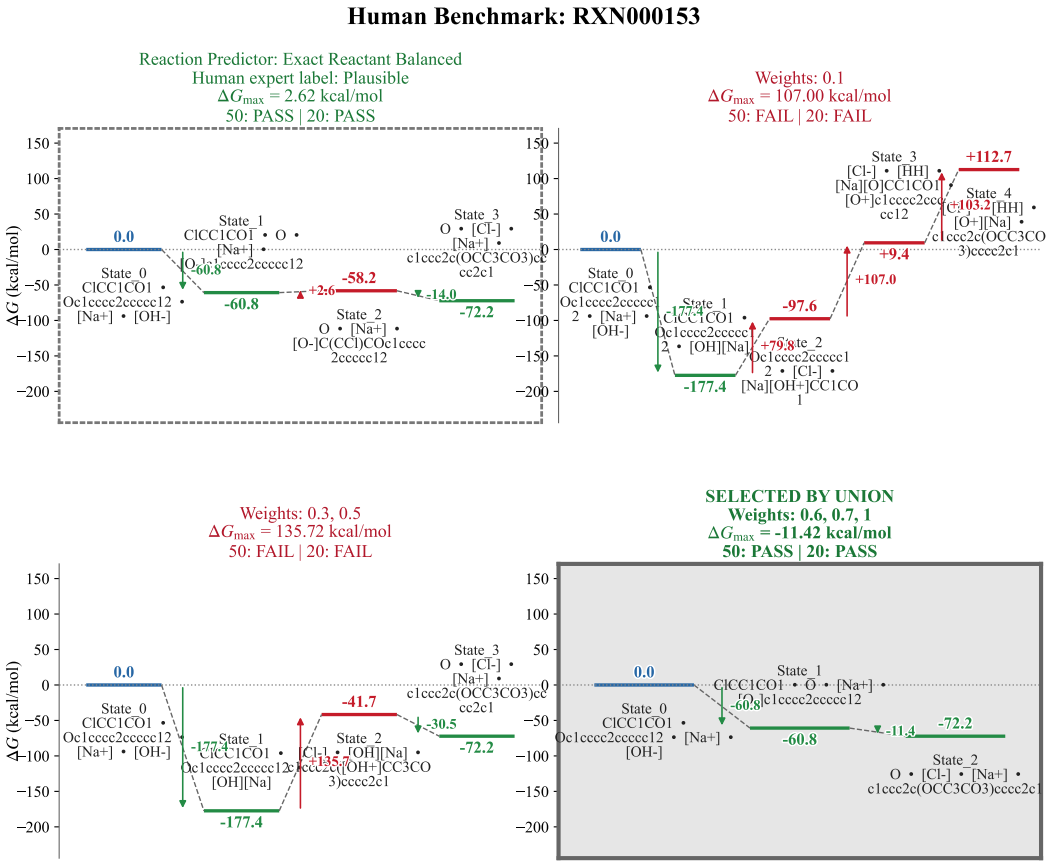


Figure 10: Energy profiles for RXN000153 from the Reaction Predictor and learned heuristic search. The Reaction Predictor pathway was assessed as plausible by human experts. The union selects the pathway recovered with $w \in \{0.6, 0.7, 1.0\}$, which passes both screening thresholds.

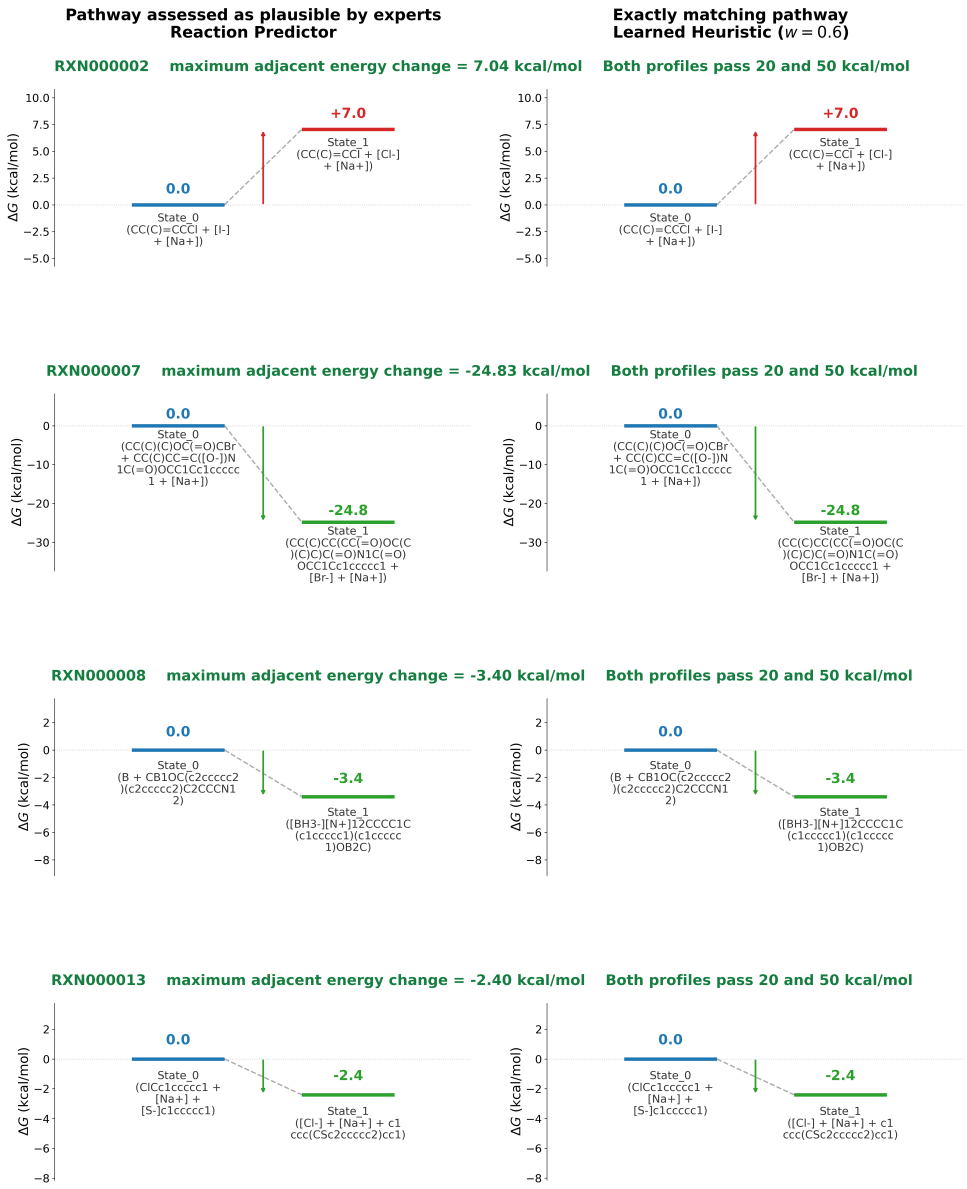


Figure 11: The Reaction Predictor pathway is shown beside the matching learned heuristic pathway obtained with $w = 0.6$.

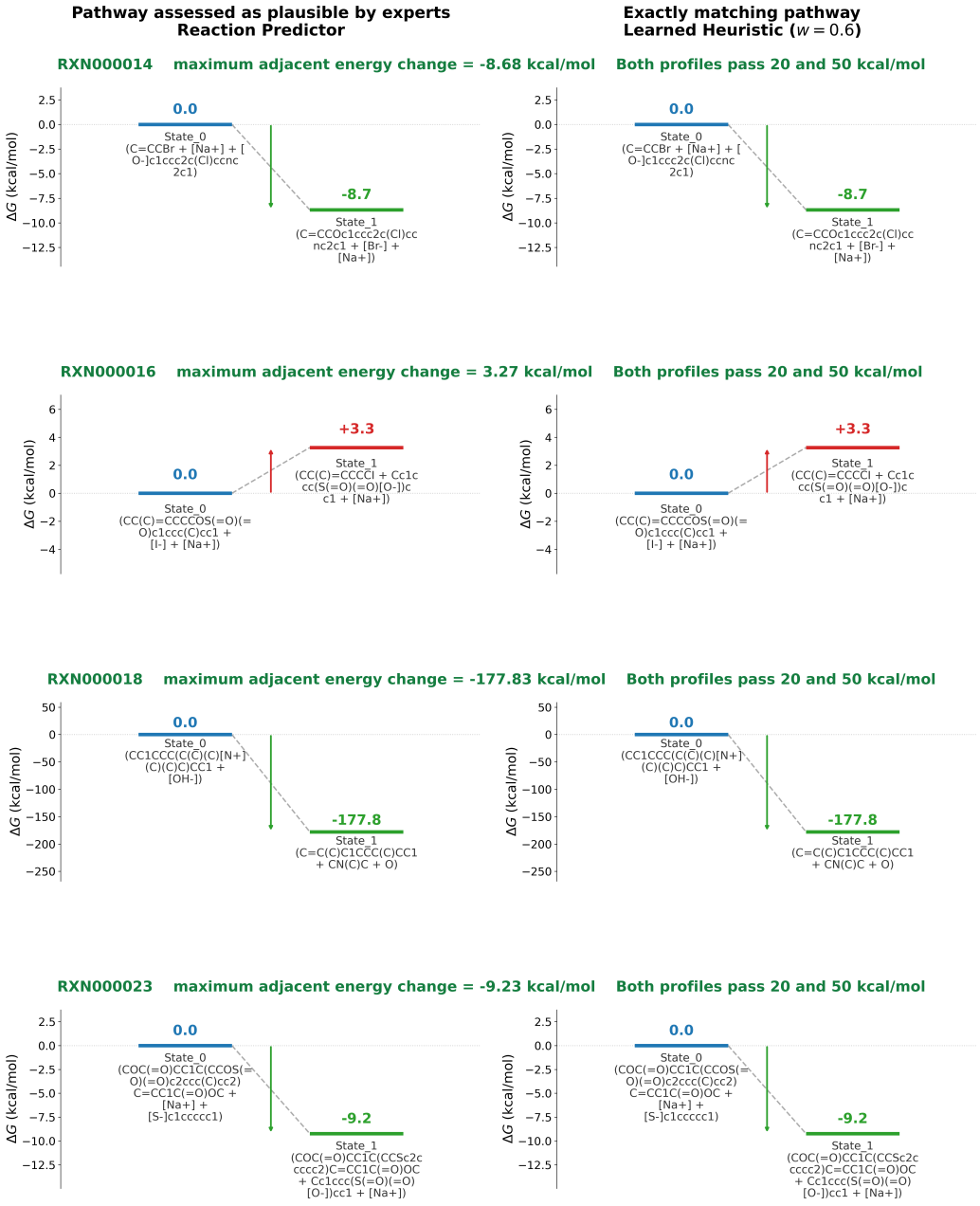


Figure 12: The Reaction Predictor pathway is shown beside the matching learned heuristic pathway obtained with $w = 0.6$.

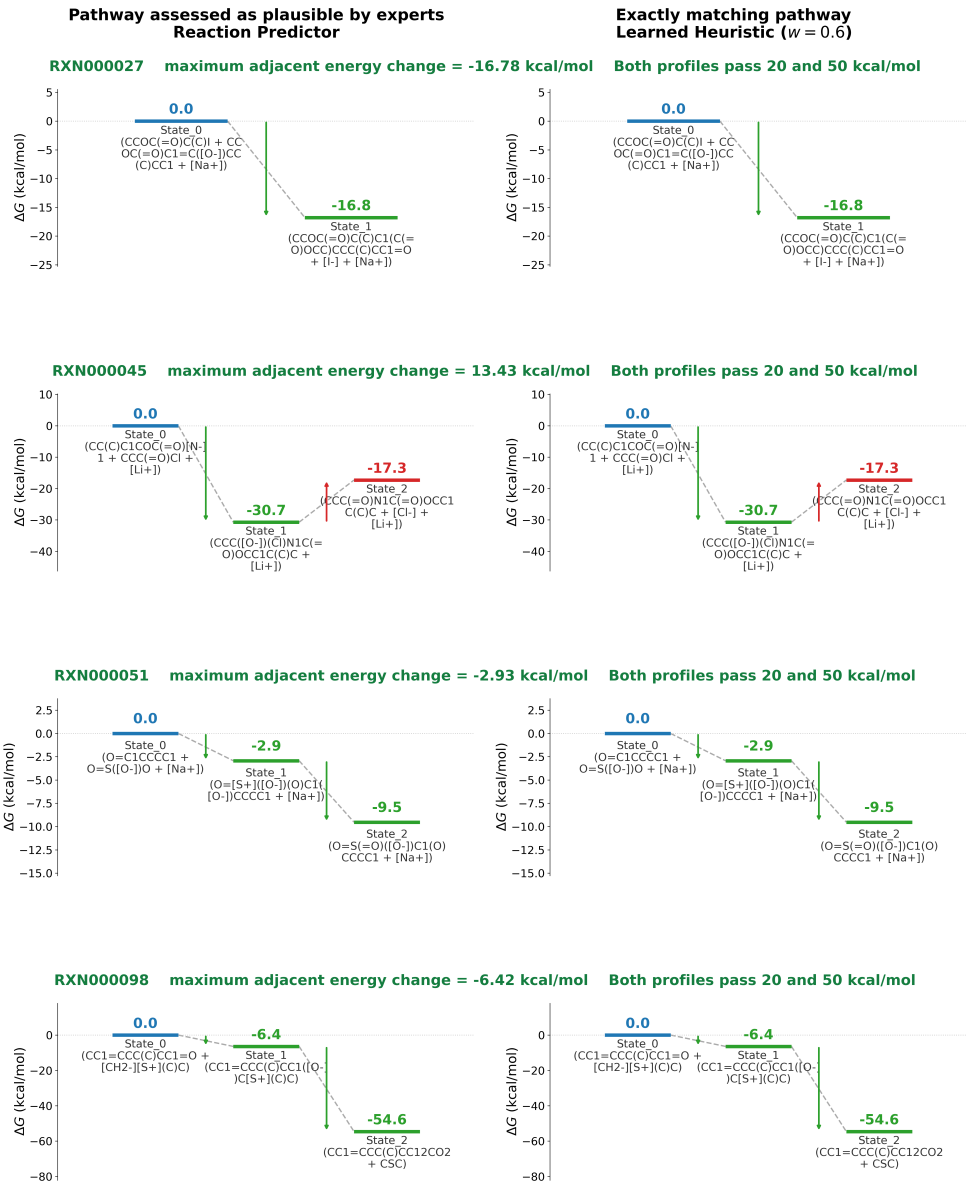


Figure 13: The Reaction Predictor pathway is shown beside the matching learned heuristic pathway obtained with $w = 0.6$.

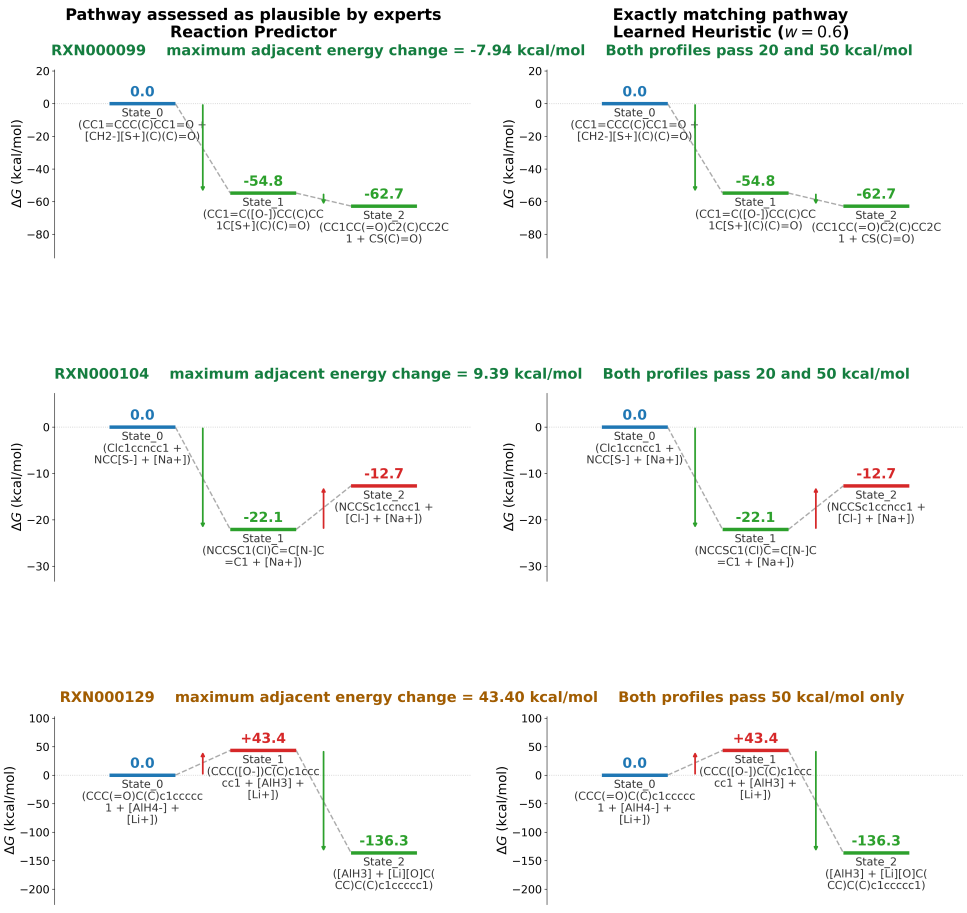


Figure 14: Exact pathway matches that pass energy screening in both results. RXN000129 passes the 50 kcal mol^{-1} threshold but does not pass the 20 kcal mol^{-1} threshold.