SEQUENCE DESIGN AND PHYLOGENETIC INFERENCE WITH GENERATIVE FLOW NETWORKS

Anonymous authors
Paper under double-blind review

ABSTRACT

Phylogenetic inference remains computationally challenging due to the exponentially growing tree topology search space, and current methods rely heavily on multiple sequence alignments (MSAs) which are expensive and error-prone. We propose AncestorGFN, a novel approach leveraging Generative Flow Networks (GFlowNets) for simultaneous sequence generation and phylogenetic inference without requiring MSAs. Our method learns to generate sequences matching a target distribution while the flow trajectories implicitly encode evolutionary relationships. We demonstrate that greedy traceback on maximum-flow trajectories recovers shared ancestral states, and evaluate on the let-7 microRNA family where the learned flow structure captures phylogenetic branching patterns. Furthermore, beam search at inference time discovers novel sequences clustering near known targets, suggesting applications in *de novo* sequence design. This work establishes a foundation for MSA-free phylogenetic inference using generative models.

1 INTRODUCTION

Phylogenetic inference, which reconstructs evolutionary relationships from molecular sequences, remains computationally challenging The number of possible tree topologies grows exponentially with the number of taxa: there are $(2n-5)!!$ unique unrooted bifurcating tree topologies on n species (Zhou et al., 2024). Standard methods including parsimony, maximum-likelihood, and Bayesian approaches have been developed to explore this vast space (Haber & Velasco, 2024), but these rely on multiple sequence alignments (MSAs), which are computationally expensive and can introduce errors that propagate to the inferred trees.

Generative Flow Networks (GFlowNets) offer a promising alternative for exploring large discrete spaces (Bengio et al., 2021; 2023). GFlowNets learn to sample objects with probability $P(x) \propto R(x)$, where $R(x)$ is a reward function, with theoretical connections to variational inference (Malkin et al., 2023) and demonstrated effectiveness in discrete optimization (Zhang et al., 2023; Pan et al., 2022). Structurally, GFlowNets can be viewed as directed acyclic graphs (DAGs) where nodes represent states, edges denote transitions, and flow corresponds to trajectory probabilities (Figure 1).

Generative models have been applied to both sequence generation and phylogenetic inference. Jain et al. (2022) demonstrated GFlowNets for *de novo* protein and DNA sequence generation. PhyloGFN (Zhou et al., 2024) applies GFlowNets to tree topology generation, while PhyloGAN (Smith & Hahn, 2023) uses GANs for generating tree topologies and branch lengths. Phylo-Diffusion (Khurana et al., 2024) conditions diffusion models with phylogenetic knowledge for species image generation. However, using generative models to simultaneously generate sequences and infer phylogenetic relationships from the generation trajectories remains unexplored.

Here, we propose AncestorGFN, which leverages GFlowNets for both sequence generation and phylogenetic inference without requiring MSAs. We demonstrate that the learned flow structure implicitly captures evolutionary relationships, and that beam search at inference time can discover novel sequences near known targets—suggesting applications in *de novo* sequence design.

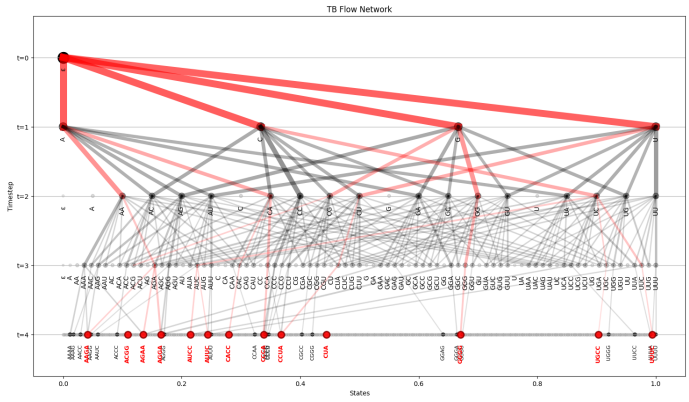

Figure 1: Schematic overview of a GFlowNet trained with the Trajectory Balance (TB) objective. Target sequences (red) represent the desired distribution; black nodes are off-target candidates. Trajectories show sampled paths through the flow network, with the inferred maximum-flow phylogeny highlighted in red.

2 METHOD

AncestorGFN adapts GFlowNets to both sequence generation and phylogenetic inference. We generate RNA sequences matching a target distribution and infer phylogenetic relationships from the learned flow network.

2.1 GFLOWNET TRAINING OBJECTIVES

We compare three GFlowNet training objectives: Trajectory Balance (TB) (Malkin et al., 2022), Detailed Balance (DB) (Bengio et al., 2023), and Forward-Looking Detailed Balance (FL-DB) (Pan et al., 2023). Our implementation primarily uses FL-DB, which addresses challenges of long trajectories and sparse rewards by incorporating intermediate reward signals (see Appendix A for TB and DB formulations).

State and Action Space: Each state represents a partial sequence, with the initial state being an empty sequence ϵ and terminal states being complete sequences. Actions include insertions, substitutions, and deletions of nucleotides (A, U, G, C). We implement several intermediate reward strategies based on sequence similarity to targets (Appendix C).

Forward-Looking Detailed Balance (FL-DB): The key insight of FL-DB is to reparameterize the flow function to incorporate intermediate energies. Given an energy function $E(s)$ defined on all states, we define the forward-looking flow $\tilde{F}(s) \triangleq e^{E(s)}F(s)$, which factors out the energy already accrued at state s. The FL-DB constraint becomes:

$$\tilde{F}(s)P_F(s' \mid s) = \tilde{F}(s')P_B(s \mid s')e^{-E(s \to s')} \tag{1}$$

where $E(s \to s') = E(s') - E(s)$ is the transition energy. In our implementation, we define $E(s)$ using sequence similarity to targets, enabling partial reward signals at each step for more efficient credit assignment.

2.2 FLOW TRACEBACK FOR PHYLOGENY INFERENCE

To infer phylogeny from the trained GFlowNet, we leverage the flow structure of the DAG. We first compute edge flows by forward-propagating from the source: for each edge $(s \to s')$, the flow is defined as:

$$f(s \to s') = F(s) \cdot P_F(s' \mid s) \tag{2}$$

where $F(s)$ is the cumulative flow at node s.

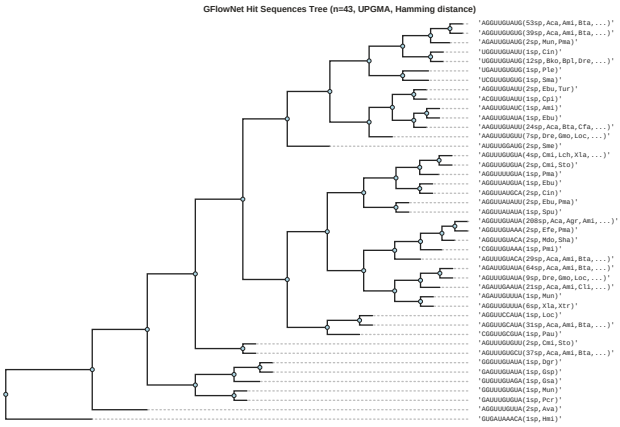

Figure 2: Traditional phylogenetic reconstruction: UPGMA tree of 43 generated let-7 sequences, clustered by Hamming distance. Labels show the sequence and representative species abbreviation. While the tree captures terminal sequence relationships, it does not reveal the generative process or ancestral intermediates.

We then apply **greedy backtracking** to reconstruct evolutionary trajectories: starting from each target terminal state x, we iteratively select the parent node with maximum incoming flow:

$$s^*_{t-1} = \arg \max_{s:(s \to s_t) \in E} f(s \to s_t) \tag{3}$$

This process continues until reaching the root (empty sequence). The intersection points of these trajectories correspond to inferred common ancestors in our model.

3 EXPERIMENTS

We first validate AncestorGFN on short 4bp RNA sequences, where FL-DB with partial rewards significantly outperforms TB and DB in both convergence speed and target hit rate (54.2% vs 7.4% and 6.0%; see Appendix B for details). The learned flow networks reveal phylogenetic structure, with greedy traceback recovering shared ancestral states across different training objectives. We now focus on biologically relevant 10bp sequences from the let-7 microRNA family.

Let-7 MicroRNA Family (10bp). The state space for 10bp sequences grows to $4^{10} = 1,048,576$ possible sequences—over 4,000$\times$ larger than the 4bp case. On random 10bp targets, FL-DB discovers 5$\times$ more unique targets than TB (10/100 vs 2/100), demonstrating that partial reward signals become increasingly critical as search space grows.

Dataset and Setup. We evaluate on the let-7 microRNA family, one of the most conserved miRNA families across species (Roush & Slack, 2008). We extract 58 unique 10bp target sequences from 612 sequences across 107 species, selecting positions 10–19 (a variable region with highest entropy; see Appendix E for details). We use ConservationWeightedHammingReward (Appendix C.4) and ProgressiveHammingReward (Appendix C.5) to weight rewards by evolutionary conservation and provide intermediate feedback. FL-DB achieves coverage of **43/58 (74.1%)** unique let-7 sequences, with sampling frequency positively correlated with species conservation (Spearman $\rho = 0.509$, $p < 0.001$; see Appendix E).

Phylogenetic Structure Comparison. To evaluate whether GFlowNet captures evolutionary relationships, we compare traditional phylogenetic reconstruction with our flow-based approach. Figure 2 shows a UPGMA tree constructed from the 43 generated sequences using Hamming distance—a standard approach in molecular phylogenetics. The clustering groups sequences by similarity, with closely related variants sharing evolutionary branches.

In contrast, Figure 3 visualizes the state-space DAG constructed from trajectories collected during training, where node size indicates visit frequency and edge width represents flow magnitude. Un-

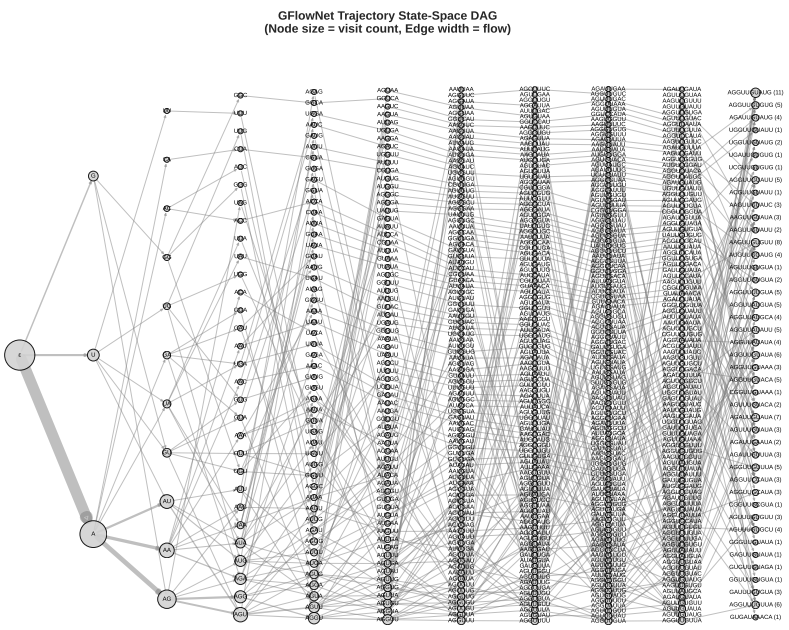

Figure 3: GFlowNet-based evolutionary inference: State-space DAG of generation trajectories for let-7 sequences. Node size indicates visit frequency during sampling; edge width represents flow magnitude. Unlike traditional trees, the DAG reveals shared intermediate states (ancestral sequences) learned by the model, providing a generative view of evolutionary relationships.

like traditional trees that only capture terminal relationships, the DAG reveals shared intermediate states among related sequences—corresponding to inferred common ancestral paths. This demonstrates that even without explicit phylogenetic supervision, the GFlowNet's learned flow structure implicitly captures evolutionary relationships, providing richer information than pairwise distance-based methods.

Novel Sequence Design. While Figure 3 shows trajectories from training-time stochastic sampling, we can also extract sequences deterministically at inference time. We apply beam search with $k = 20$ to the trained model to find the top-k most likely sequences. Figure 4 shows the resulting trajectory DAG, revealing that 5/20 top sequences are known let-7 targets while 15/20 are novel sequences not present in the training set. Notably, these novel sequences cluster near known targets (typically 1–2 Hamming distance), suggesting the model has learned meaningful sequence neighborhoods rather than arbitrary patterns. The shared intermediate states in the beam search DAG provide an alternative view of ancestral relationships: sequences sharing early trajectory prefixes are inferred to have common evolutionary origins. This demonstrates potential for *de novo* sequence design, where trained GFlowNets can propose novel candidates similar to known functional sequences at inference time.

4 Discussion

Our experiments reveal two key insights. First, partial reward signals become increasingly critical as sequence length grows: FL-DB achieves 5× better target coverage than TB on 10bp sequences, suggesting intermediate credit assignment is essential in sparse reward landscapes. Second, GFlowNets implicitly learn evolutionary relationships through their flow structure without explicit phylogenetic supervision, with shared intermediate states corresponding to inferred common ancestors. Beam search at inference time discovers novel sequences near known targets, demonstrating potential for *de novo* sequence design.

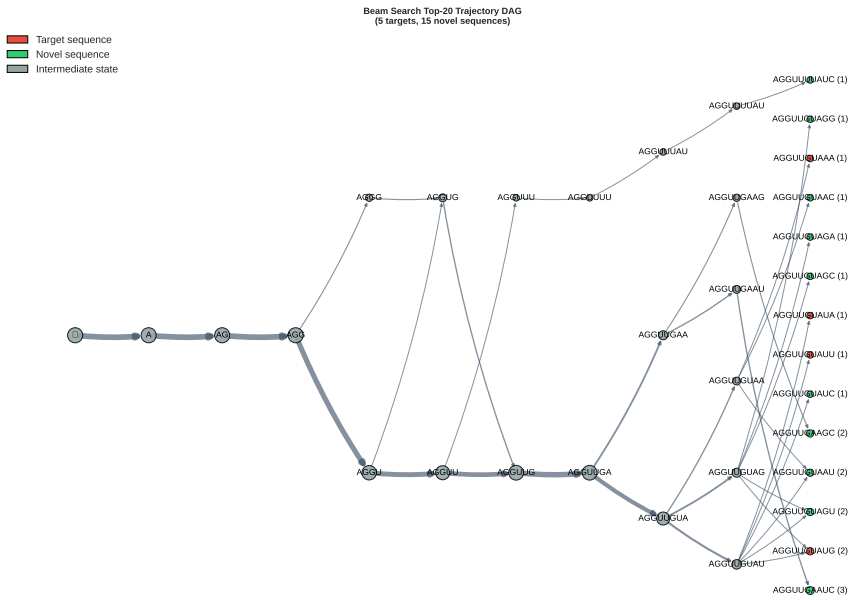

Figure 4: Post-training beam search trajectory DAG for let-7 sequences. Top-20 most likely sequences are extracted via beam search at inference time, with trajectories traced from the root (empty sequence) to terminal states. Red nodes indicate target sequences (5/20), green nodes represent novel sequences (15/20), and gray nodes show shared intermediate states. Novel sequences cluster near known targets (typically 1–2 Hamming distance), suggesting the model learns meaningful sequence neighborhoods.

Limitations and Future Work. Current limitations include: scaling to full-length miRNAs (22bp+), lack of quantitative comparison with ground-truth phylogenies (e.g., Robinson-Foulds distance), and no direct comparison with tools like RAxML or MrBayes. Future directions include hierarchical GFlowNets for longer sequences, incorporating phylogenetic likelihood models as rewards, and extending to protein sequences.

Reproducibility Statement

Code for AncestorGFN will be made available upon publication. All the data from the experiments is provided, ensuring reproducibility. The miRNA LET-7 family sequences used in this study were obtained from MirGeneDB (`https://mirgenedb.org/browse/ALL?family=LET-7&seed=`).

Acknowledgments

[To be added in camera-ready version]

References

Emmanuel Bengio, Moksh Jain, Maksym Korablyov, Doina Precup, and Yoshua Bengio. Flow Network based Generative Models for Non-Iterative Diverse Candidate Generation. In *Advances in Neural Information Processing Systems*, volume 34, pp. 27381–27394. Curran Associates, Inc., 2021. URL `https://proceedings.neurips.cc/paper/2021/hash/e614f646836aaed9f89ce58e837e2310-Abstract.html`.

Yoshua Bengio, Salem Lahlou, Tristan Deleu, Edward J. Hu, Mo Tiwari, and Emmanuel Bengio. GFlowNet Foundations, July 2023. URL `http://arxiv.org/abs/2111.09266`. arXiv:2111.09266 [cs, stat].

Matt Haber and Joel Velasco. Phylogenetic Inference. In Edward N. Zalta and Uri Nodelman (eds.), *The Stanford Encyclopedia of Philosophy*. Metaphysics Research Lab, Stanford University, Summer 2024 edition, 2024.

Moksh Jain, Emmanuel Bengio, Alex Hernandez-Garcia, Jarrid Rector-Brooks, Bonaventure F. P. Dossou, Chanakya Ajit Ekbote, Jie Fu, Tianyu Zhang, Michael Kilgour, Dinghuai Zhang, Lena Simine, Payel Das, and Yoshua Bengio. Biological Sequence Design with GFlowNets. In *Proceedings of the 39th International Conference on Machine Learning*, pp. 9786–9801. PMLR, June 2022. URL https://proceedings.mlr.press/v162/jain22a.html.

Mridul Khurana, Arka Daw, M. Maruf, Josef C. Uyeda, Wasila Dahdul, Caleb Charpentier, Yasin Bakış, Henry L. Bart Jr, Paula M. Mabee, Hilmar Lapp, James P. Balhoff, Wei-Lun Chao, Charles Stewart, Tanya Berger-Wolf, and Anuj Karpatne. Hierarchical Conditioning of Diffusion Models Using Tree-of-Life for Studying Species Evolution, July 2024. URL http://arxiv.org/abs/2408.00160. arXiv:2408.00160.

Nikolay Malkin, Moksh Jain, Emmanuel Bengio, Chen Sun, and Yoshua Bengio. Trajectory balance: Improved credit assignment in GFlowNets. *Advances in Neural Information Processing Systems*, 35:5955–5967, December 2022. URL https://proceedings.neurips.cc/paper_files/paper/2022/hash/27b51baca8377a0cf109f6ecc15a0f70-Abstract-Conference.html.

Nikolay Malkin, Salem Lahlou, Tristan Deleu, Xu Ji, Edward Hu, Katie Everett, Dinghuai Zhang, and Yoshua Bengio. GFlowNets and variational inference, March 2023. URL http://arxiv.org/abs/2210.00580. arXiv:2210.00580.

Ling Pan, Dinghuai Zhang, Aaron Courville, Longbo Huang, and Yoshua Bengio. Generative Augmented Flow Networks, October 2022. URL http://arxiv.org/abs/2210.03308. arXiv:2210.03308.

Ling Pan, Nikolay Malkin, Dinghuai Zhang, and Yoshua Bengio. Better training of GFlowNets with local credit and incomplete trajectories. In *International Conference on Machine Learning*, pp. 26878–26890. PMLR, 2023. URL https://proceedings.mlr.press/v202/pan23c.html.

Sarah Roush and Frank J Slack. The let-7 family of micrornas. *Trends in Cell Biology*, 18(10): 505–516, 2008. doi: 10.1016/j.tcb.2008.07.007.

Megan L Smith and Matthew W Hahn. Phylogenetic inference using generative adversarial networks. *Bioinformatics*, 39:btad543, September 2023. ISSN 1367-4811. doi: 10.1093/bioinformatics/btad543. URL https://doi.org/10.1093/bioinformatics/btad543.

Dinghuai Zhang, Ricky T. Q. Chen, Nikolay Malkin, and Yoshua Bengio. Unifying Generative Models with GFlowNets and Beyond, January 2023. URL http://arxiv.org/abs/2209.02606. arXiv:2209.02606.

Mingyang Zhou, Zichao Yan, Elliot Layne, Nikolay Malkin, Dinghuai Zhang, Moksh Jain, Mathieu Blanchette, and Yoshua Bengio. PhyloGFN: Phylogenetic inference with generative flow networks, March 2024. URL http://arxiv.org/abs/2310.08774. arXiv:2310.08774 [cs, q-bio, stat].

A GFlowNet Training Objectives

We compare three GFlowNet training objectives for sequence generation.

Trajectory Balance (TB) (Malkin et al., 2022): For a complete trajectory $\tau = (s_0 \to s_1 \to \cdots \to s_n = x)$:

$$Z_\theta \prod_{i=1}^{n} P_F(s_i \mid s_{i-1}; \theta) = R(x) \prod_{i=1}^{n} P_B(s_{i-1} \mid s_i; \theta) \tag{4}$$

where Z_θ is a learnable partition function estimate. TB only receives reward at terminal states, leading to poor credit assignment on longer trajectories.

Detailed Balance (DB) (Bengio et al., 2023): For each transition $s \to s'$:

$$F(s)P_F(s' \mid s) = F(s')P_B(s \mid s') \tag{5}$$

where $F(s)$ is the learned state flow function, with boundary condition $F(x) = R(x)$ at terminal states. DB converges faster than TB by learning per-state flow estimates.

Forward-Looking Detailed Balance (FL-DB) (Pan et al., 2023) is detailed in the main text. FL-DB is recommended when meaningful intermediate rewards are available, achieving significantly higher target hit rates.

B Short RNA Sequence Experiments (4bp)

We validate AncestorGFN on short RNA sequences using an alphabet of four nucleotides $\{A, U, G, C\}$ with maximum sequence length of 4 and 5 discrete time steps. We define 13 target RNA motifs including AUUC, AUCC, CACC, CUA, CCUA, CCCA, GGGG, and others.

Training Objective Comparison. For TB and DB, we use a sparse reward signal (1.0 for exact matches, 0.1 otherwise). For FL-DB, we employ AlignmentReward (Appendix C.1) which provides partial credit. All models are trained for 20,000 episodes with learning rate 3×10^{-3}.

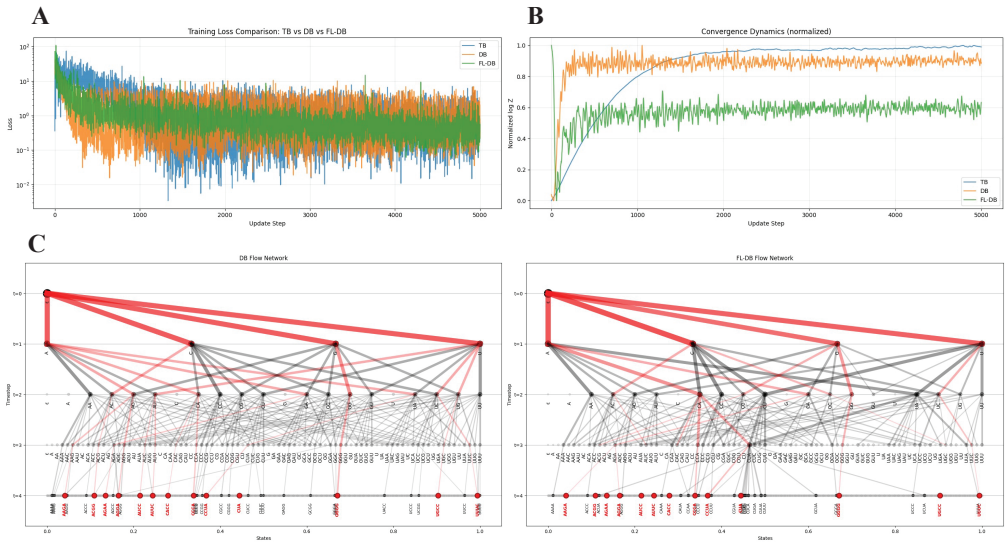

Figure 5: Training comparison of GFlowNet objectives on short RNA sequences. (A) Training loss curves for TB, DB, and FL-DB objectives. (B) Normalized partition function estimates showing convergence dynamics. (C) Flow network visualization for DB (left) and FL-DB (right) models, with target sequences highlighted in red and maximum-flow trajectories shown.

Figure 5A shows training loss comparison: TB suffers from poor credit assignment, while DB and FL-DB converge faster with localized gradient signals. Figure 5B shows convergence dynamics, and Figure 5C displays flow network visualizations.

Flow Network and Phylogeny Inference. Applying greedy traceback, we find that some sequences share common ancestors across different training objectives. For example, both CCCA and GGGG pass through a shared ancestral state, while AAGA and AGGA both trace back to the ancestral sequence AGA. These branching patterns demonstrate that the GFlowNet naturally captures evolutionary relationships.

C Reward Function Designs

We implement several reward functions for sequence generation, each addressing different challenges in training GFlowNets.

C.1 Alignment Reward

The **AlignmentReward** uses Needleman-Wunsch global alignment to provide partial credit:

$$R_{\text{align}}(x) = \max_{y \in \mathcal{T}} \frac{\text{NW}(x, y)}{\text{NW}(y, y)} \tag{6}$$

where $\text{NW}(\cdot, \cdot)$ is the alignment score (match $= +1$, mismatch $= -1$, gap $= -1$). This is useful when the action space includes insertions and deletions.

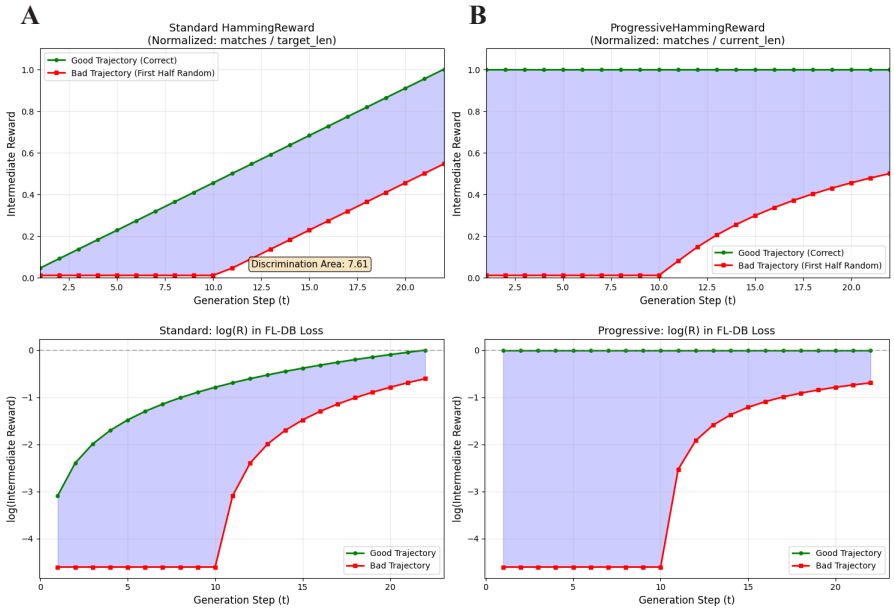

Figure 6: Standard vs. progressive Hamming reward. Progressive normalization by current length provides stronger gradient signals for partial sequences.

C.2 Entropy-Weighted Hamming Reward

The **EntropyWeightedHammingReward** addresses mode collapse on repetitive sequences by boosting complex (high-entropy) targets:

$$R_{\text{entropy}}(x) = R_{\text{hamming}}(x, y^*) \cdot (1 + \alpha \cdot H_{\text{norm}}(y^*)) \tag{7}$$

where $H_{\text{norm}}(y) = -\sum_c p_c \log p_c / \log |\mathcal{A}|$ is the normalized Shannon entropy.

C.3 Adaptive Hamming Reward

The **AdaptiveHammingReward** dynamically decays rewards for frequently hit targets:

$$R_{\text{adaptive}}(x) = R_{\text{hamming}}(x, y^*) \cdot \frac{1}{1 + \beta \log(1 + n_{y^*})} \tag{8}$$

where n_{y^*} is the cumulative hit count for target y^*, encouraging exploration of under-sampled targets.

C.4 Conservation-Weighted Hamming Reward

The **ConservationWeightedHammingReward** weights sequences by evolutionary conservation:

$$R_{\text{conservation}}(x) = R_{\text{hamming}}(x, y^*) \cdot w_{\text{cons}}(y^*) \tag{9}$$

where the conservation weight uses log-scaling:

$$w_{\text{cons}}(y) = \alpha + (1 - \alpha) \cdot \frac{\log(1 + n_{\text{species}}(y))}{\log(1 + \max_{y'} n_{\text{species}}(y'))} \tag{10}$$

Here $n_{\text{species}}(y)$ is the number of species containing sequence y, and α ensures rare sequences still receive some reward. We use $\alpha = 0.1$, giving conserved sequences (found in $\sim$80 species) higher reward than singletons.

C.5 Progressive Hamming Reward

The **ProgressiveHammingReward** improves intermediate reward signals for FL-DB training. Standard Hamming normalization by target length gives weak gradients for partial sequences:

$$R_{\text{standard}}(x_t) = \frac{\text{matches}(x_t, y^*[:t])}{\text{len}(y^*)} \tag{11}$$

Instead, we normalize by current sequence length for intermediate rewards:

$$R_{\text{progressive}}(x_t) = \frac{\text{matches}(x_t, y^*[:t])}{t} \tag{12}$$

For example, generating towards a 10bp target: at $t = 1$, standard reward gives $1/10 = 0.1$ (weak), while progressive reward gives $1/1 = 1.0$ (strong). Terminal rewards still use target length normalization. Figure 6 illustrates this difference.

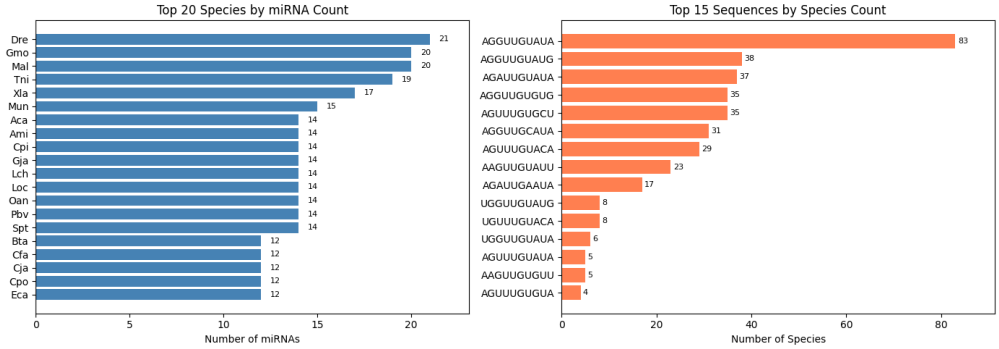

Figure 7: Let-7 dataset statistics. **Left:** Top 20 species by miRNA count. **Right:** Top 15 sequences ranked by conservation (species count).

D Training Objective Comparison

Table 1 compares the three GFlowNet training objectives on the 4bp RNA sequence generation task (Case Study 1).

DB converges faster than TB for simple problems through local flow constraint. **FL-DB** is recommended when meaningful intermediate rewards are available (e.g., partial sequence similarity), achieving significantly higher target hit rates.

Table 1: Comparison of GFlowNet training objectives on 4bp sequences. TB/DB use TargetMatchReward (exact match), FL-DB uses AlignmentReward (partial credit).

Aspect	**TB**	**DB**	**FL-DB**
Credit Assignment	Global	Local	Local + Rewards
Intermediate Rewards	No	No	Yes
Mean Reward	0.166	0.154	0.563
Target Hit Rate	7.4%	6.0%	54.2%

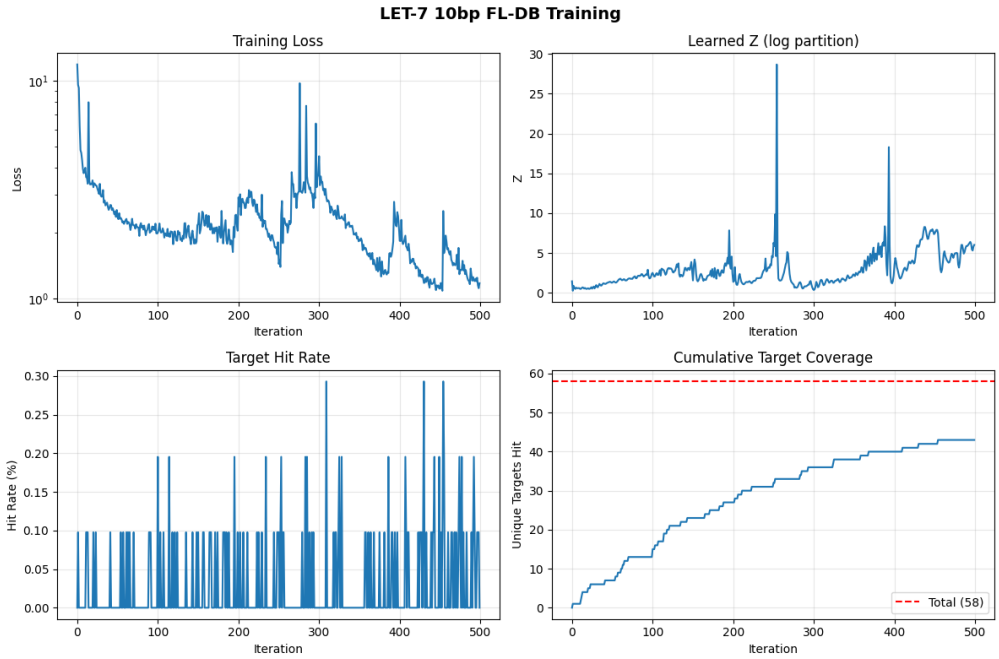

Figure 8: FL-DB training dynamics on let-7 sequences. **Top:** Loss and $\log Z$ convergence. **Bottom:** Target hit rate and cumulative coverage reaching 43/58 (74.1%).

E LET-7 MICRORNA RESULTS

E.1 DATASET

The let-7 dataset comprises 612 miRNA sequences from 107 species. After extracting positions 10–19 (a 10bp variable region with highest entropy), we obtain 58 unique target sequences with conservation ranging from 1 to 83 species per sequence. Figure 7 shows the distribution of miRNA counts across species (left) and the top conserved sequences ranked by species count (right).

E.2 TRAINING CONFIGURATION

- **Objective:** FL-DB with ConservationWeightedHammingReward + ProgressiveHammingReward
- **Action space:** Insertion-only for efficiency
- **Training:** 500 iterations, batch size 1024, learning rate 3×10^{-3}
- **Network:** MLP with hidden layers [32, 16, 8]

Figure 8 shows the training dynamics, including loss convergence, $\log Z$ estimation, target hit rate progression, and cumulative coverage reaching 43/58 (74.1%).

E.3 CONSERVATION CORRELATION

The model preferentially samples conserved sequences (Spearman $\rho = 0.509$, $p < 0.001$), demonstrating that conservation-weighted rewards successfully guide sampling toward evolutionarily important sequences. Figure 9 visualizes this positive correlation between species count and GFlowNet sampling frequency.

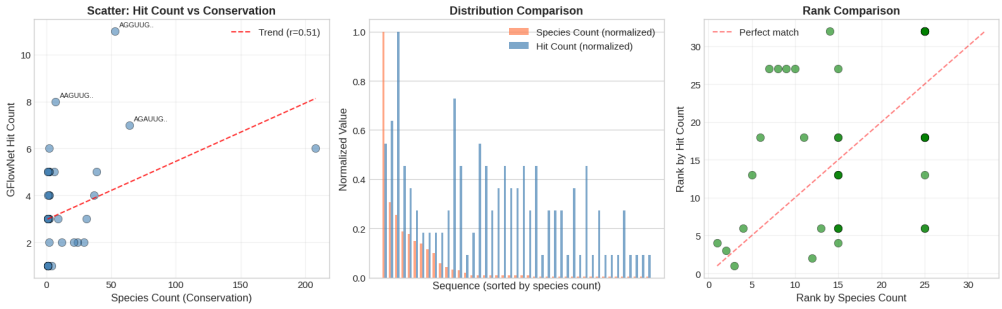

Figure 9: Correlation between species count and GFlowNet sampling frequency.