Differentiable Search of Evolutionary Trees from Leaves

Ramith Hettiarachchi [1] **Avi Swartz** [2] **Sergey Ovchinnikov** [3]

Abstract

Inferring the most probable evolutionary tree given leaf nodes is an important problem in computational biology that reveals the evolutionary relationships between species. Due to the exponential growth of possible tree topologies, finding the best tree in polynomial time becomes computationally infeasible. In this work, we propose a novel differentiable approach as an alternative to traditional heuristic-based combinatorial tree search methods in phylogeny. The optimization objective of interest in this work is to find the most parsimonious tree (i.e., to minimize the total number of evolutionary changes in the tree). We empirically evaluate our method using randomly generated trees of up to 128 leaves, with each node represented by a 256-length protein sequence. Our method exhibits promising convergence ($< 1\%$ error for trees up to 32 leaves, $< 8\%$ error up to 128 leaves, given only leaf node information), illustrating its potential in much broader phylogenetic inference problems and possible integration with end-to-end differentiable models. The code to reproduce the experiments in this paper can be found at `https://github.ramith.io/diff-evol-tree-search`.

1. Introduction

Evolutionary trees (or phylogenetic trees) provide biologists with a structured, hierarchical representation of how current species are related through hypothetical ancestors that are probably extinct at present. Beyond theoretical constructs, they have practical applications in various fields of biology and medicine. For instance, phylogenetic techniques are crucial in the decision-making process when responding to emerging viruses (Attwood et al., 2022).

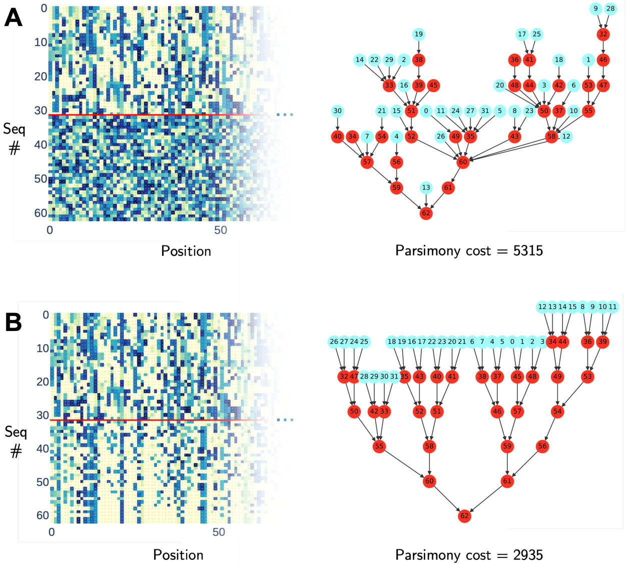

Figure 1. Search for evolutionary tree ($N = 32$, $l = 256$) left : sequences (leaves and ancestors), right : tree topology. A) random initialization, B) converged solution. optimal solution cost = 2913.

Parsimony methods are one of many methods (such as distance methods, maximum likelihood, maximum compatibility) for constructing evolutionary trees. The parsimony principle states that the most acceptable explanation of an occurrence is the one that requires the minimum number of assumptions or explanations (Sober, 1981). Thus, inferring the most parsimonious tree given leaf nodes requires finding the tree that explains the data with minimum number of evolutionary steps. This combinatorial problem was shown to be NP-Complete (Foulds & Graham, 1982; Steel, 1992).

Due to the complexity of the problem, existing methods consider heuristic search techniques by limiting the search space. Although this does not guarantee that the algorithm will find the optimal solution, they do facilitate exploration of a vast number of tree topologies, starting from an initial guess and iteratively refining it. These methods can be broadly categorized into 1) tree rearrangement methods 2) branch and bound methods 3) neighbor joining methods (Saitou & Nei, 1987; Giribet, 2007; Felsenstein, 2004).

With the success of deep learning methods, there have been several new directions in constructing evolutionary trees. (Zhu & Cai, 2021) et al. propose an alignment-free method in which an attention model (Vaswani et al., 2017) is trained

[1] FAS Division of Science, Harvard University [2] Molecular and Cellular Biology Program, University of Washington [3] JHDSF Program, Harvard University. Correspondence to: Sergey Ovchinnikov <so@fas.harvard.edu>.

Published at the Differentiable Almost Everything Workshop of the 40^{th} International Conference on Machine Learning, Honolulu, Hawaii, USA. July 2023. Copyright 2023 by the author(s).

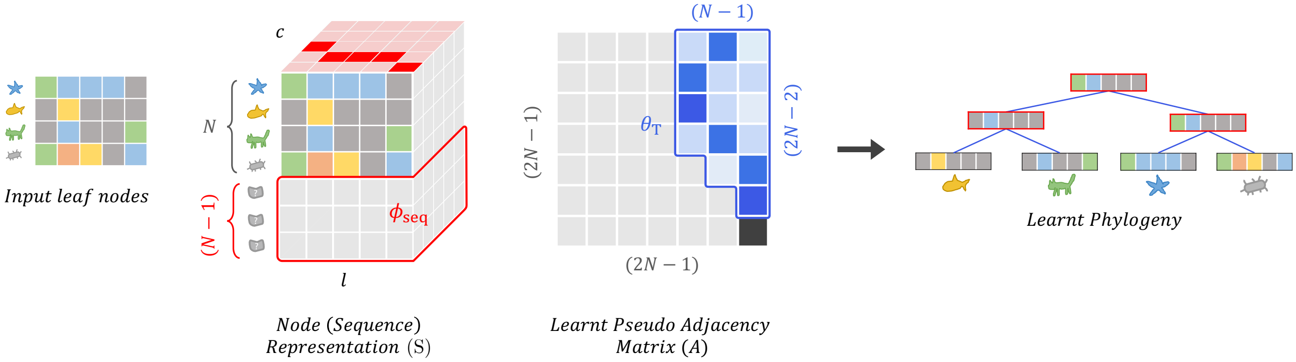

Figure 2. Visual depiction of the constructed adjacency matrix and the node representation using learnable parameters θ_T and ϕ_{seq}.

through reinforcement learning to reconstruct evolutionary trees. However, it requires algorithmic post-processing to produce the final tree, preventing it from being end-to-end differentiable. (Azouri et al., 2023) demonstrate a deep-Q-learning agent on empirical data consisting up to 20 leaves.

With the success of using hyperbolic geometry for hierarchical data, there has been work on obtaining continuous embeddings for trees (Monath et al., 2019; Chami et al., 2020). Subsequently, optimization in the hyperbolic space for phylogeny (Wilson, 2021), developing new metrics (Matsumoto et al., 2021) and addressing the phylogenetic placement problem (Jiang et al., 2022) have been explored. Given the challenge of scalability in traditional Bayesian phylogenetics, methods based on variational inference have been proposed (Zhang & Matsen, 2019; Dang & Kishino, 2019). Recently, (Zhang, 2023) proposed a topological feature learning framework for phylogenetic inference using graph neural networks.

In contrast to these work, our approach circumvents the discreteness of the raw tree and sequence representations in the first place and model their relationship in a differentiable manner. By doing so, we obtain a soft-parsimony score that can be optimized in an end-to-end differentiable manner, without the need for any prior training data.

We perform experiments for tree topologies up to 128 leaves and analyze our method for 3 tasks. 1) Learn the tree given all sequences, 2) learn the ancestors given the tree topology and leaves (small parsimony), and 3) learn the tree and the ancestors given leaves (maximum parsimony). For the small parsimony problem we achieve the $\approx 0\%$ mean error for all tree complexities, meaning that our approach can find the optimal ancestral sequences if the tree topology is known. For the maximum parsimony problem, we achieve $< 1\%$ error for trees up to 32 leaves, $< 8\%$ error up to 128 leaves.

Our work opens up new realms for integration with models with more complex cost functions that go beyond site-independence assumption. For example, the cost function can integrate pseudo-likelihood between nodes using protein language models or those conditioned on protein structure.

2. Notation and background

2.1. Maximum parsimony problem (learn tree and ancestors)

A rooted phylogenetic tree is a directed acyclic graph (DAG) $\mathcal{G} = (V, E)$. Given a set of N leaves, the maximum parsimony problem intends to find the phylogenetic tree ($\mathcal{T}$) and ancestor nodes that describe the given data with minimum number of evolutionary steps (Carmel et al., 2014; Kannan & Wheeler, 2012). For a fixed alphabet $\mathcal{A} = \{1, ..., c\}$, each node in this DAG can be represented by an l dimensional vector $\mathbf{s} = (s_1, ..., s_l)$ such that $\mathbf{s} \in \mathcal{A}^l$. We consider the Hamming distance d on the node representation $d((s_1, ..., s_l), (s_1^{'}, ..., s_l^{'}))$ that describes the number of indices i such that $s_i^{'} \neq s_i$. Let $\delta(\mathcal{T}) = \sum_{(\mathbf{u},\mathbf{v}) \in E(\mathcal{T})} d(\mathbf{u}, \mathbf{v})$ represent the total number of evolutionary changes in the tree $\mathcal{T}$. The maximum parsimony problem is then to find the tree that minimizes $\delta(\mathcal{T})$.

2.2. Complexity

The maximum parsimony problem has been comprehensivey studied in the literature as a special case of the Steiner tree problem (Hwang & Richards, 1992). Further (Foulds & Graham, 1982) showed that even when $|\mathcal{A}| = 2$, the problem of finding the tree with the minimum number of evolutionary steps is NP-Complete. Given N leaves, the number of rooted bifurcating tree topologies that exists can be calculated as $(2N - 3)!!$ (Cavalli-Sforza & Edwards, 1967). Thus, even for a tree with only 12 leaves, there are more than 13 billion tree topologies in total.

2.3. Small parsimony problem (learn ancestors)

The small parsimony problem (Carmel et al., 2014) is a much simplified version of the problem in which the phylogenetic tree topology $\mathcal{T}$ is already given. Therefore, the task is to find the best possible ancestors. There are a number of dynamic programming (DP) algorithms proposed to solve this problem in polynomial such as the Fitch's algorithm (Fitch, 1971) and the Sankoff's algorithm (Sankoff, 1975).

3. Methodology

In the following subsections, we consider how the discrete aspects of the problem are relaxed and how gradient-based optimization can be performed.

3.1. Relaxations

There are two aspects to this problem that make it inherently non-differentiable. First, each element of the sequence is combinatorial. Second, considering the adjacency matrix, the space of valid and meaningful tree topologies is sparse.

Sequence representation In both the small and maximum parsimony problems, the ancestor sequences are unknown. We denote learnable parameters $\phi_{seq} \in \mathbb{R}^{(2N-1)\times l\times c}$ to represent the ancestors to be optimized. To obtain a continuous relaxation of the categorical nature of amino acid types, we transform the real tensor into a tensor $\hat{\phi}_{seq}$, where each element represents a probability distribution over the character space for each position of the sequence. This transformation is done using the softmax function (Jang et al., 2017), with the sharpness of the distribution controlled by the temperature parameter τ_2.

$$\hat{\phi}_{seq_{ijk}} = \frac{e^{\phi_{seq_{ijk}}/\tau_2}}{\sum_{m=1}^{c} e^{\phi_{seq_{ijm}}/\tau_2}} \tag{1}$$

This probability tensor $\hat{\phi}_{seq_{ijk}}$ representing ancestors, combined with the one-hot encoding of known leaves results in the node representation tensor S.

Tree representation Since the adjacency matrix of any DAG can be permuted to be a strictly upper triangular matrix (Nicholson, 1975; Li et al., 2022; Charpentier et al., 2022), we ensure the acyclicity of the graphs represented by the adjacency matrix (A) by enforcing it to be strictly upper triangular. Furthermore, since leaves cannot be connected to each other, we ensure that the first N columns of A are zero. The remaining positions are parameterized as θ_T.

Due to the irregular structure of the adjacency matrix representation and to speed up the implementation, we first set the non-parameterized region of the adjacency matrix to $-\inf$, and then apply the softmax function for each row. This ensures that the parameterized positions of the adjacency matrix will be represented by the correct probability. Thus, the i^{th} row of A represents the probabilities of node i being a child of all other nodes, respectively.

$$A_{ij} = \frac{e^{\theta_{T_{ij}}/\tau_1}}{\sum_{k=1}^{N-1} e^{\theta_{T_{ik}}/\tau_1}} \tag{2}$$

3.2. Tree enforcing loss function

With the relaxation of the adjacency matrix, optimizing parameters to reduce L_{cost} does not explicitly guide the optimization towards a bifurcating tree. Therefore, we enforce the following regularization constraint to maintain the bifurcating property. This regularization forces tree nodes to have exactly two child node connections.

$$L_b = \sum_{j=1}^{N-1} abs((\sum_{i=1}^{2N-2} A_{ij}) - 2)^2 \tag{3}$$

3.3. Differentiable soft parsimony score calculation

To calculate the number of evolutionary steps that have occurred in the DAG, we formulate the evolutionary cost calculation as follows. The transposed node representation tensor $S_p = S^{\mathsf{T}}_{(2,0,1)} \in \mathbb{R}^{c\times(2N-1)\times l}$ is constructed by representing the characters of the alphabet as its first dimension.

$$\mathcal{L}_{cost}(\theta_T, \phi_{seq}, \tau_1, \tau_2) = \frac{1}{2}\sum_{i=1}^{2N-1}\sum_{j=1}^{l}\sum_{k=1}^{|\mathcal{A}_c|} |S_{p_k} - A \times S_{p_k}|_{ij} \tag{4}$$

In this equation, the adjacency matrix A describes the connection of each node to its parent in the tree structure. The matrix S_{pk}, considers k^{th} character of the alphabet at a time. Thus, in the matrix S_{pk}, $(S_{pk})_{ij} = 1$ iff i^{th} sequence has j^{th} position equal to the k^{th} character in the alphabet $\mathcal{A}$. The matrix multiplication $A \times S_{p_k}$ serves as a lookup of the sequence table (S_{p_k}) and returns the parent corresponding to each node in the tree. Thus, the difference between the matrices $|S_{p_k} - A \times S_{pk}|$ represents the distance between each child and its parent in the k^{th} character space. By summing these differences over all dimensions, the cost function captures the overall evolutionary cost. For a visual depiction of this calculation, refer to the Appendix Figure 6.

3.4. Bi-level optimization

In the maximum parsimony problem, we need to traverse both the tree and sequence spaces, and there is a dependency between these two. Note that for each tree topology, there is a best set of sequences that define how good each topology is. And once the topology changes, these sequences are no longer valid. Thus, for this task, we formulate the optimization procedure as a bi-level optimization problem. Experimentally, we obtain better results with this formulation than optimizing both of the parameters independently (see the appendix A.2 for the ablation study).

$$\theta_T^* = \arg\min_{\theta_T} \mathcal{L}_{\text{cost}}\left(\phi^*_{seq}(\theta_T), \theta_T\right) + \lambda L_b \tag{5}$$

$$\text{s.t. } \phi^*_{seq}(\theta_T) = \arg\min_{\phi_{seq}} \mathcal{L}_{\text{cost}}\left(\phi_{seq}, \theta_T\right) + \lambda L_b \tag{6}$$

During implementation, we perform k gradient descent iterations for the inner objective (Eq. 6). Gradient descent is performed for both objectives using the Adam optimizer (Kingma & Ba, 2015). Furthermore, we use the JAXopt library to optimize multiple initialization seeds as a batch and obtain the best result (Blondel et al., 2021).

Table 1. Evaluation on the Maximum Parsimony and Small Parsimony Problems

Tree Complexity		Maximum Parsimony (learn both tree & seqs)			Small Parsimony (learn seqs given tree)		
N	Mean optimal solution	Mean solution	Mean error	Mean error as a % w.r.t optimal solution	Mean solution	Mean error	Mean error as a % w.r.t optimal solution
4	277.2	277.2	0.0 ± 0.0	0.000%	277.2	0.0 ± 0.0	0.000%
8	653.1	653.1	0.0 ± 0.0	0.000%	653.1	0.0 ± 0.0	0.000%
16	1407.6	1407.7	0.1 ± 0.3	0.007%	1407.6	0.0 ± 0.0	0.000%
32	2915.4	2936.3	20.9 ± 7.4	0.717%	2915.4	0.0 ± 0.0	0.000%
64	5929.3	6188.6	259.3 ± 27.4	4.373%	5929.3	0.0 ± 0.0	0.000%
128	11971.1	12885.5	914.4 ± 99.6	7.638%	11971.3	0.2 ± 0.4	0.001%

3.5. Evaluation

In order to evaluate our method, we first generate known evolutionary trees (each consisting of $4 - 128$ leaves, 256 sequence length, alphabet size $c = 20$). Therefore, we generate complete binary trees starting with a random sequence as the root, make two copies of the sequence at each node, and generate two random sets of indices each with $m = 50$ elements which are mutated to a different character. For each tree complexity, we perform 10 random initializations for the leaf sequences to generate examples. Further details on ground truth generation are included in Appendix A.

4. Results

We first analyze a simpler task in which all nodes S (leaves and ancestors) are known, yet the tree topology is unknown. For all tree complexities, the adjacency matrix converges to the groundtruth tree. This task is similar to a hierarchical clustering task, where all the nodes are known, and we need to establish a hierarchical dependency between them.

4.1. Small Parsimony (learn ancestors given tree)

The ground-truth ancestor solutions for the known tree topology is obtained by the Sankoff algorithm. Table 1 shows the results for varying number of tree sizes. For all cases, the error between ours and the optimal parsimony score is ≈ 0.

4.2. Maximum Parsimony (learn ancestors and tree)

In this task, only the leaf sequences are known, and we need to optimize towards both the best tree topology and ancestors. As shown in Figure 3 and Table 1, the mean error increases as the complexity of the tree increases. Note that for up to $N = 8$ the possible tree topologies are even enumerable, as they result in only 135,135 combinations. Thus, our method also converges to the optimal solutions. However, from $N = 16$ to 128 leaves, tree topologies grows from $\approx 10^{15}$ to 10^{250} possibilities and our method converges to local optima. We intend to explore methods to simplify the loss landscape and gradually increase the complexity in order to discover better solutions.

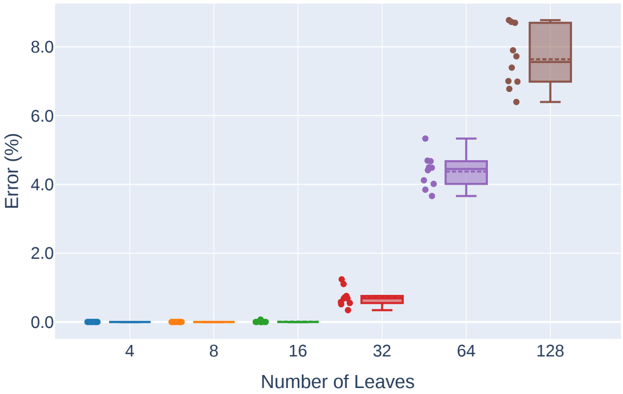

Figure 3. **Evaluation on the maximum parsimony problem (learn both the tree and ancestors, given leaves)**. Error is calculated w.r.t Sankoff algorithm solutions on the groundtruth tree topology.

5. Discussion and Future Work

Our work establishes a new direction for generating evolutionary trees by traversing a soft tree and sequence space. Although here we focused on minimizing the parsimony cost as the objective (with unit cost for any change), our general optimization method can be coupled with various loss functions. For instance, the parsimony cost assumes site-independence, which means that any position wise evolutionary dependence in amino acids is ignored. Thus, if the tree is known, this independence property can be utilized to develop a dynamic programming algorithm that can derive the most optimal ancestral sequences (i.e. Sankoff algorithm (Sankoff, 1975)). Therefore, our method can be most beneficial when this condition is lifted (e.g. integration of pseudo-likelihood between nodes using protein language models or those conditioned on protein structure).

Even though the error increases with tree complexity, future work could explore an iterative procedure to combine subtrees gradually (e.g. for 128 leaves, breaking down the problem to two sets of 64 leaves or 4 sets of 32 leaves and running gradient descent, then using these answers as a better initialization for the original problem).

References

Attwood, S. W., Hill, S. C., Aanensen, D. M., Connor, T. R., and Pybus, O. G. Phylogenetic and phylodynamic approaches to understanding and combating the early SARS-CoV-2 pandemic, 2022. ISSN 14710064.

Azouri, D., Granit, O., Alburquerque, M., Mansour, Y., Pupko, T., and Mayrose, I. The tree reconstruction game: phylogenetic reconstruction using reinforcement learning. *arXiv preprint arXiv:2303.06695*, 2023.

Blondel, M., Berthet, Q., Cuturi, M., Frostig, R., Hoyer, S., Llinares-López, F., Pedregosa, F., and Vert, J.-P. Efficient and modular implicit differentiation. *arXiv preprint arXiv:2105.15183*, 2021.

Carmel, A., Musa-Lempel, N., Tsur, D., and Ziv-Ukelson, M. The worst case complexity of maximum parsimony. *Journal of computational biology : a journal of computational molecular cell biology*, 21(11), 2014. ISSN 15578666. doi: 10.1089/cmb.2014.0128.

Cavalli-Sforza, L. L. and Edwards, A. W. Phylogenetic analysis. Models and estimation procedures. *American Journal of Human Genetics*, 19(3), 1967. ISSN 00029297. doi: 10.2307/2406616.

Chami, I., Gu, A., Chatziafratis, V., and Ré, C. From trees to continuous embeddings and back: Hyperbolic hierarchical clustering. In *Advances in Neural Information Processing Systems*, volume 2020-December, 2020.

Charpentier, B., Kibler, S., and Günnemann, S. Differentiable DAG Sampling. In *ICLR 2022 - 10th International Conference on Learning Representations*, 2022.

Dang, T. and Kishino, H. Stochastic variational inference for Bayesian phylogenetics: A case of CAT model. *Molecular Biology and Evolution*, 36(4), 2019. ISSN 15371719. doi: 10.1093/molbev/msz020.

Felsenstein, J. *Inferring phylogenies*, volume 2. Sinauer associates Sunderland, MA, 2004.

Fitch, W. M. Toward defining the course of evolution: Minimum change for a specific tree topology. *Systematic Biology*, 20(4), 1971. ISSN 1076836X. doi: 10.1093/sysbio/20.4.406.

Foulds, L. R. and Graham, R. L. The steiner problem in phylogeny is NP-complete. *Advances in Applied Mathematics*, 3(1), 1982. ISSN 10902074. doi: 10.1016/S0196-8858(82)80004-3.

Giribet, G. Efficient tree searches with available algorithms. *Evolutionary Bioinformatics*, 3:117693430700300014, 2007.

Hwang, F. K. and Richards, D. S. Steiner tree problems. *Networks*, 22(1), 1992. ISSN 10970037. doi: 10.1002/net.3230220105.

Jang, E., Gu, S., and Poole, B. Categorical reparameterization with gumbel-softmax. In *5th International Conference on Learning Representations, ICLR 2017 - Conference Track Proceedings*, 2017.

Jiang, Y., Tabaghi, P., and Mirarab, S. Learning Hyperbolic Embedding for Phylogenetic Tree Placement and Updates. *Biology*, 11(9), 2022. ISSN 20797737. doi: 10.3390/biology11091256.

Kannan, L. and Wheeler, W. C. Maximum Parsimony on Phylogenetic networks. *Algorithms for Molecular Biology*, 7(1), 2012. ISSN 17487188. doi: 10.1186/1748-7188-7-9.

Kingma, D. P. and Ba, J. L. Adam: A method for stochastic optimization. In *3rd International Conference on Learning Representations, ICLR 2015 - Conference Track Proceedings*, 2015.

Li, X., CAI, Y., Sun, M., and Li, P. Causal discovery via cholesky factorization. 2022.

Matsumoto, H., Mimori, T., and Fukunaga, T. Novel metric for hyperbolic phylogenetic tree embeddings. *Biology Methods and Protocols*, 6(1), 2021. ISSN 23968923. doi: 10.1093/biomethods/bpab006.

Monath, N., Zaheer, M., Silva, D., McCallum, A., and Ahmed, A. Gradient-based hierarchical clustering using continuous representations of trees in hyperbolic space. In *Proceedings of the ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 2019. doi: 10.1145/3292500.3330997.

Nicholson, V. A. Matrices with permanent equal to one. *Linear Algebra and Its Applications*, 12(2), 1975. ISSN 00243795. doi: 10.1016/0024-3795(75)90067-1.

Saitou, N. and Nei, M. The neighbor-joining method: a new method for reconstructing phylogenetic trees. *Molecular biology and evolution*, 4(4), 1987. ISSN 07374038. doi: 10.1093/oxfordjournals.molbev.a040454.

Sankoff, D. Minimal Mutation Trees of Sequences. *SIAM Journal on Applied Mathematics*, 28(1), 1975. ISSN 00361399. doi: 10.1137/0128004.

Sober, E. The principle of parsimony. *British Journal for the Philosophy of Science*, 32(2), 1981. ISSN 00070882. doi: 10.1093/bjps/32.2.145.

Steel, M. The complexity of reconstructing trees from qualitative characters and subtrees. *Journal of Classification*, 9(1), 1992. ISSN 01764268. doi: 10.1007/BF02618470.

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, Ł., and Polosukhin, I. Attention is all you need. In *Advances in Neural Information Processing Systems*, volume 2017-December, 2017.

Wilson, B. Learning phylogenetic trees as hyperbolic point configurations. *arXiv preprint arXiv:2104.11430*, 2021.

Zhang, C. Learnable topological features for phylogenetic inference via graph neural networks. In *The Eleventh International Conference on Learning Representations*, 2023. URL https://openreview.net/forum?id=hVVUY7p64WL.

Zhang, C. and Matsen, F. A. Variational Bayesian phylogenetic inference. In *7th International Conference on Learning Representations, ICLR 2019*, 2019.

Zhu, T. and Cai, Y. Applying Neural Network to Reconstruction of Phylogenetic Tree. In *ACM International Conference Proceeding Series*, 2021. doi: 10.1145/3457682.3457704.

A. Appendix

A.1. Groundtruth data generation

We generate complete binary trees starting with a random sequence as root, make two copies of the sequence at each node, and generate two random sets of indices each with $m = 50$. These two sets of random indices are mutated into two copies, so that a new random amino acid is introduced at each index. It should be noted that this process does not necessarily mean that the generated complete binary trees are the trees with minimum number of evolutionary steps for the reached leaves (for these leaves there could be a better tree topology and ancestors). However, since mutations are introduced only at 50/256 $\approx 20\%$ of the sequence length, the probability of the existence of better topologies is low, yet we find the best ancestors for this topology by applying the Sankoff algorithm, and this serves us as the groundtruth. Therefore, these serve as test samples to assess whether the optimization procedure converges to tree and its corresponding ancestors with evolutionary steps that are sufficiently close to the generated groundtruth.

A.2. Ablation study of optimization methods

As mentioned in 3.4, we compare the bi-level optimization procedure with an independent alternative optimization scheme. In this formulation, we treat the tree parameters and sequence parameters independently and alternatively perform gradient descent using Adam optimizer. We find that, even though this procedure works similarly well for trees with fewer leaves ($N \leq 16$), it accumulates high error as the tree grows. The results are shown in Table 2.

$$\theta_T^* = \arg\min_{\theta_T} \mathcal{L}_{\text{cost}} \left(\theta_T\right) + \lambda L_b \tag{7}$$

$$\phi_{seq}^* = \arg\min_{\phi_{seq}} \mathcal{L}_{\text{cost}} \left(\phi_{seq}\right) + \lambda L_b \tag{8}$$

Table 2. Ablation study on two different methods for optimization on the maximum parsimony problem

Tree Complexity		Bi-level optimization			Alternative Optimization		
N	Mean optimal solution	Mean solution	Mean error	Mean error as a % w.r.t optimal solution	Mean solution	Mean error	Mean error as a % w.r.t optimal solution
4	277.2	277.2	0.0 ± 0.0	0.000%	277.2	0.0 ± 0.0	0.000%
8	653.1	653.1	0.0 ± 0.0	0.000%	653.1	0.0 ± 0.0	0.000%
16	1407.6	1407.7	0.1 ± 0.3	0.007%	1411.1	3.5 ± 5.6	0.249%
32	2915.4	2936.3	20.9 ± 7.4	0.717%	2977.9	62.5 ± 18.1	2.144%
64	5929.3	6188.6	259.3 ± 27.4	4.373%	6341.0	411.7 ± 64.3	6.943%
128	11971.1	12885.5	914.4 ± 99.6	7.638%	13597.29	1632.0 ± 129.0	13.639%

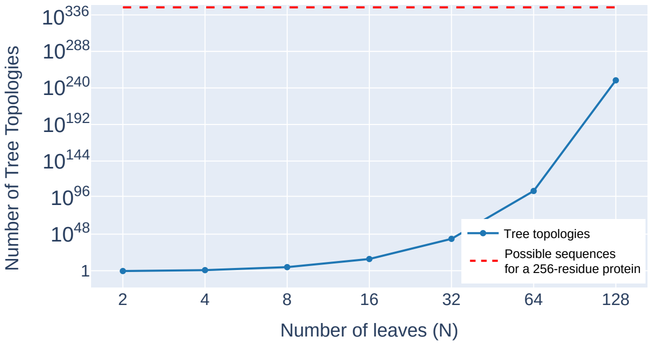

Figure 4. Number of rooted bifurcating tree topologies for N leaves (blue) and number of possible amino acid sequences (red) for a single 256-residue protein (i.e. $20^{256} \approx 1.16 \times 10^{333}$)

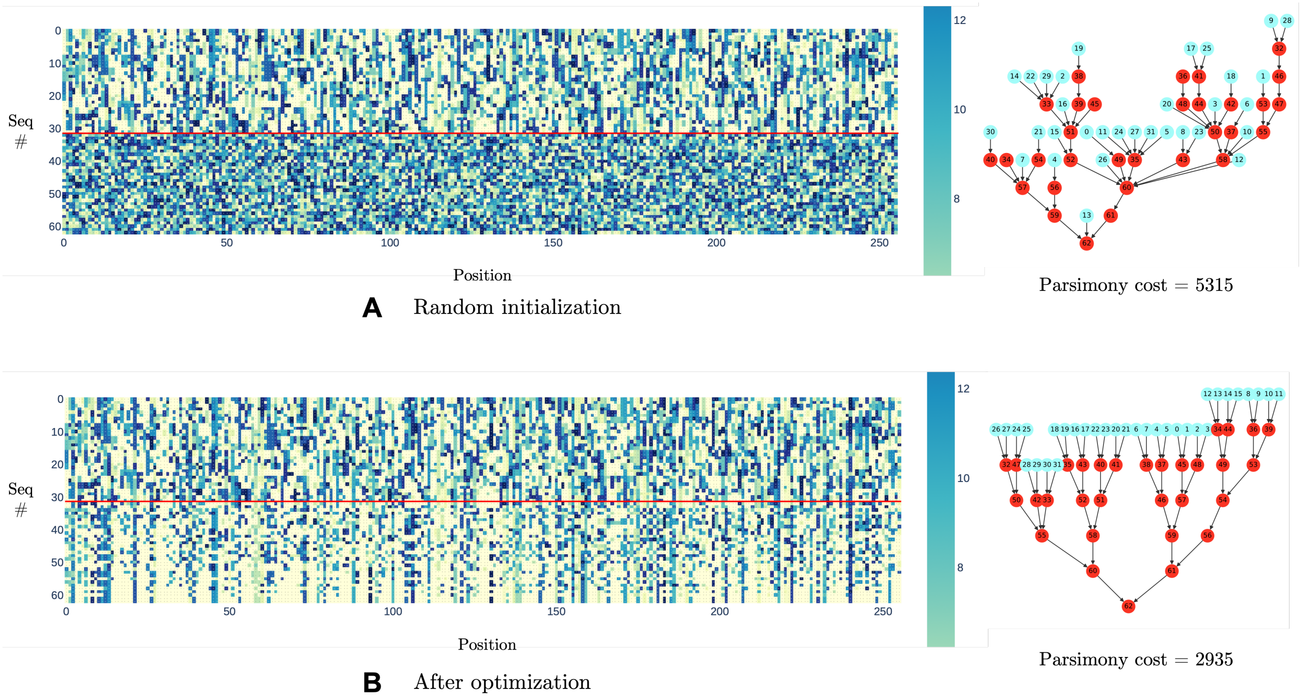

Figure 5. Search for evolutionary tree ($N = 32$, $l = 256$) left : sequences, right : tree topology. A) random initialization, B) converged solution. optimal solution cost = 2913.

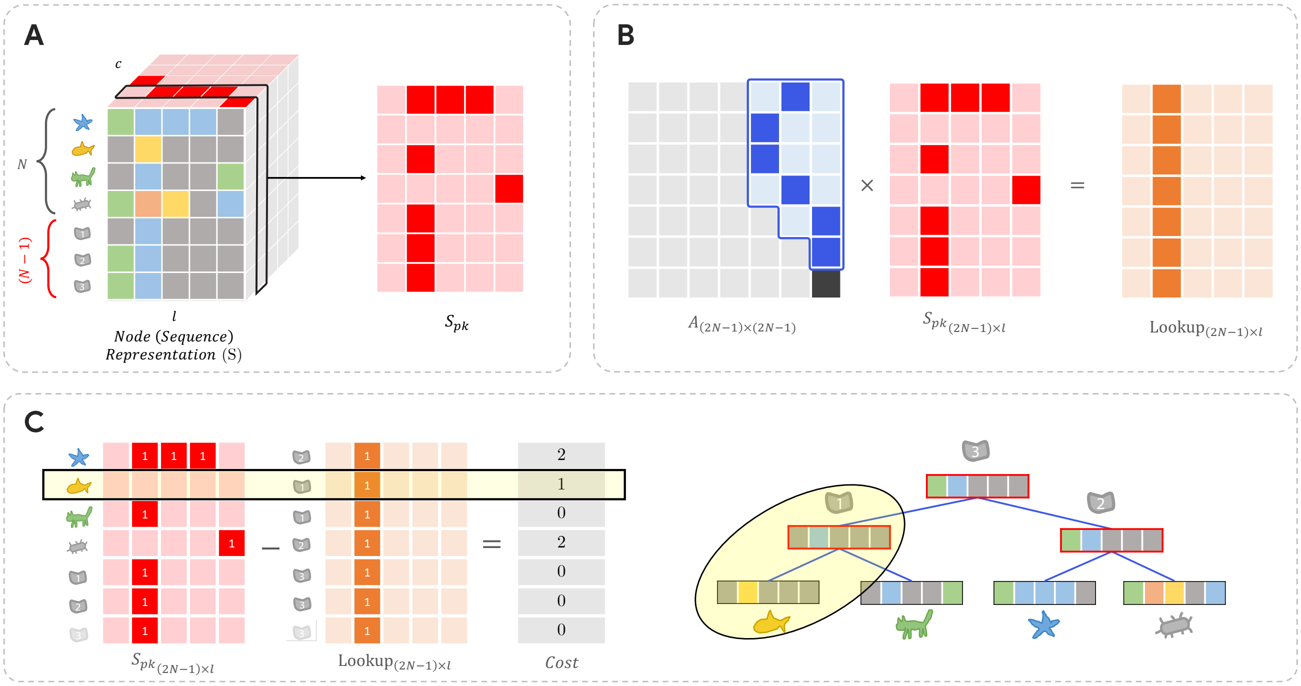

Figure 6. Explanation of parsimony cost calculation. A) considering one character dimension at a time. B) The matrix multiplication $A \times S_{p_k}$ serves as a look-up of the sequence table (S_{p_k}). C) Difference between the child and parent sequence in the k^{th} character space.