Reconstruction of ancestral protein sequences using autoregressive generative models

Pierre Barrat-Charlaix* and Andrea Pagnani

DISAT, Politecnico di Torino, Italy

(Dated:)

Abstract

abstract

I. INTRODUCTION

II. RESULTS

A. Autoregressive model of sequence evolution

Models of evolution commonly used in phylogenetics rely on the assumptions that sequence positions evolve independently and that evolution at each position i follows a continuous time Markov chain (CTMC) parametrized by a substitution rate matrix $\mathbf{Q}^i$. Matrix $\mathbf{Q}^i$ is of dimensions $q \times q$ where $q = 4$ for DNA, 20 for amino acids or 64 for codon models. The probability of observing a change from state a to state b during evolutionary time t is then given by $P(b|a,t) = \left(e^{t\mathbf{Q}^i}\right)_{ab}$.

If the model is time-reversible, it is a general property of CTMCs that the substitution rate matrix can be written as

$$\mathbf{Q} = \mathbf{H} \cdot \mathbf{\Pi} = \mathbf{H} \cdot \begin{pmatrix} \pi_1 & 0 & 0 \\ 0 & \ddots & 0 \\ 0 & 0 & \pi_q \end{pmatrix}, \tag{1}$$

where $\mathbf{H}$ is symmetric and $\mathbf{\Pi}$ is diagonal with entries that sum to 1 [1]. The two matrices have simple interpretations. On the first hand, $\mathbf{\Pi}$ fixes the long-term equilibrium frequencies, that is $P(b|a,t) \xrightarrow[t\to\infty]{} \pi_b$. On the other, $\mathbf{H}$ influences the dynamics of the Markov chain but does not change the equilibrium distribution. Most commonly, $\mathbf{\Pi}$ is considered to be independent of the sequence position i, while $\mathbf{H}$ can be multiplied by position-dependent rates in order to model the different variability of different sites [2–4].

In order to incorporate constraints coming from a protein's structure and function into the evolutionary model, we develop a family-specific model of protein sequence evolution based on the the autoregressive generative model ArDCA [5]. ArDCA models the diversity

* Correspondance to: PBC, DISATpierre.barratcharlaix@polito.it

of sequences in a protein family using a set of learned conditional probabilities. In practice, the model assigns a probability to any sequence $\mathbf{a} = \{a_1, \dots, a_L\}$ of L amino acids:

$$P^{AR}(\mathbf{a}) = \prod_{i=1}^{L} p_i(a_i|a_{<i}), \tag{2}$$

where the product runs over positions in the sequence and $a_{<i} = a_1, \dots, a_{i-1}$ represents all amino acids before position i. Functions p_i represent the probability according to the model to observe state a_i in position i, given that the previous amino acids were $a_1, \dots, a_{i-1}$. Their precise functional form is given in the methods section. They are learned using the aligned sequences of members of the family. In actual implementations, the order in which the product in Eq. 2 is performed is not the natural $(1, \dots, L)$ but rather an order where positions are sorted by increasing variability. This does not significantly effect the model we present below, and we keep the notation of Eq. 2 for simplicity.

It has been shown in [5] that the generative capacities of ArDCA are comparable to that of state of the art models such as bmDCA [6]. This means that a set of sequences sampled from the probability in Eq. 2 is statistically hard to distinguish from the natural sequences used in training or, in other words, that the model can be used to sample new artificial homologs of a protein family. Generative capacities of a protein model come from its ability to represent epistasis, that is the relation between the effect of a mutation and sequence context in which it occurs. Here, epistasis is modeled through the conditional probabilities p_i: the distribution of amino acids at position i depends on the states of the previous positions $1, \dots i-1$.

We take advantage of the autoregressive architecture to define the following evolution model. Given two amino acid sequences $\mathbf{a}$ and $\mathbf{b}$, we propose

$$P(\mathbf{b}|\mathbf{a}, t) = \prod_{i=1}^{L} q_i(b_i|a_i, b_{<i}, t), \tag{3}$$

where the conditional propagator q_i is defined as

$$q_i(b_i|a_i, b_{<i}, t) = \left(e^{t \cdot Q^i(b_{<i})}\right)_{a_i, b_i}, \quad Q^i(b_{<i}) = \mathbf{H} \cdot \begin{pmatrix} p_i(1|b_{<i}) & 0 & 0 \\ 0 & \ddots & 0 \\ 0 & 0 & p_i(q|b_{<i}) \end{pmatrix}. \tag{4}$$

According to these equations, evolution for each position i follows a standard CTMC. However, we use the decomposition of Eq. 1 to set the equilibrium frequency at i to $p_i(b|b_{<i})$. In other words, we consider that position i evolves in the context of $b_1, \ldots, b_{i-1}$, and that its dynamics are constrained by its long term frequency given by the autoregressive model. An important consequence of this choice is that our evolution model will converge at long times to the generative distribution P^{AR}:

$$q_i(b_i|a_i, b_{<i}, t) \xrightarrow[t\to\infty]{} p_i(b_i|b_{<i}), \quad P(\mathbf{b}|\mathbf{a}, t) \xrightarrow[t\to\infty]{} P^{AR}(\mathbf{b}). \tag{5}$$

We argue here that such a property is essential to build a realistic protein sequence evolution model, particularly when considering evolution over a relatively long time frame. Note that to converge to a generative distribution, accurate modeling of epistasis is required. Using site-specific frequencies would not be sufficient, as the effect of mutations in a protein sequence typically depends on the context [7]. The technique proposed here allows us to represent epistasis through the context dependent probabilities p_i, while still considering each sequence position one at a time.

Interestingly, we note that the model in Eq. 3 is not time reversible, although context dependent site propagators in Eq. 4 are reversible. We show in the SM that this is mainly an artifact of the autoregressive nature of the model coupled with epistasis. Using non-time reversible evolutionary models is uncommon in the field, but this is mainly due to practical considerations and there are no fundamental reasons for evolution itself to be reversible [8]. In practice, this means that algorithms using this model have to be adapted accordingly.

We underline that this approach has important differences with standard models of evolution used in phylogenenetics. In phylogenetic reconstruction, the tree and the sequence evolution model are usually inferred at the same time and from the same data. The number of parameters of the evolution model is then kept low to reduce the risk of overfitting, for instance by using site specific rates to account for variable and conserved sites. Methods that introduce more complex models such as site specific frequencies do so by jointly inferring the parameters and the tree, leading to relatively complex algorithms [9, 10].

Here instead, parameters of the generative model in Eq. 2 are learned from a protein family, *i.e.* a set of diverged homologous protein sequences. While it is true that these sequences share a common evolutionary history and cannot be considered as independent samples, common learning procedures only account for this in a very crude way [5, 11]. Despite this, it appears that the generative properties of such models are not strongly affected by ignoring the phylogeny [12, 13]. This allows us to proceed in two steps: first construct the model from data while ignoring phylogeny, and then only use it for phylogenetic inference tasks.

An advantage of this approach is that once the model of Eq. 2 is inferred, the propagator in Eq. 3 comes “for free” as no additional parameters are required. Importantly, our model does not use site specific substitution rates. Indeed, it has been shown that these can be seen as emergent properties when using more complex evolution models such as the one presented here [14]. However, a disadvantage is that the technique is only applicable to a given protein family at a time, and requires the existence of an appropriate training set for the model.

B. Ancestral sequence reconstruction

We apply our evolutionary model to the task of ancestral sequence reconstruction (ASR). The goal of ASR is the following: given a set of extant sequences with a shared evolutionary history and the corresponding phylogenetic tree, is it possible to reconstruct the sequences of extinct ancestors at the internal nodes of the tree? Along with the autoregressive evolutionary model described above, we thus need two inputs to perform ASR: a known phylogenetic tree, and the multiple sequence alignment of the leaf sequences. The length of the aligned sequences has to exactly correspond to that of the autoregressive model.

To reconstruct ancestral sequences using the autoregressive model, we proceed as follows:

i for sequence position $i = 1$, use the evolution model defined by the equilibrium frequencies p_1 to reconstruct a state a_1^n at each internal node n of the tree;

ii iterating through subsequent positions $i > 1$: reconstruct state a_i^n at each internal node n using the model defined in Eq. 4, with the context $a_{<i}^n$ having been already reconstructed in the previous iterations.

It is important to note that when any position $i > 1$ is reconstructed, the context at different internal nodes of the tree may differ. For a branch joining two nodes (n, m) of the tree, the evolution model will thus differ if we go down or up the branch: in one case the context at node n must be used, in the other case the context at node m. This is a consequence of the time-irreversibility of the model. For this reason, we use a variant of Felsenstein's pruning algorithm that is adapted to irreversible models [15]. This comes at no computational cost.

Note that this method is adapted to both maximum likelihood and Bayesian inference. In the ML case, each iteration reconstructs the most probable residue is at a position i given the already reconstructed context. In the Bayesian case, residue a_i^n is instead sampled from the posterior distribution.

In any realistic application, the phylogenetic tree has to be reconstructed from the aligned sequences. In principle, a consistent approach would use the same evolutionary model for tree inference and ASR. However, our model does not allow us to reconstruct the tree. Therefore, in any realistic application, the tree is reconstructed using an evolutionary model that typically will differ from ours.

To reduce issues related to this evolutionary model discrepancy, we adopt the following strategy: our ASR method blindly trusts the topology of the input tree, but recomputes the branch length using the sequences. As explained in the Methods, there is no direct way to optimize branch length with the autoregressive model. For simplicity, we use profile model with position-specific amino acid frequencies for this task. This provides a relatively accurate estimate of the branch lengths, as shown in Figure S1.

C. Results on simulated data

There are two difficulties when evaluating the capacity of a model to perform ASR. The first is that in the case of biological data, the real phylogeny and ancestral sequences are usually not known. As a consequence, one must rely on simulated data to measure the quality of reconstruction. The second is that the reconstruction of an ancestral sequence is always uncertain, as evolutionary models are typically stochastic. The uncertainty becomes higher as nodes that are more remote from the leaves are considered. This means that it is only possible to make a statistical assessment about the quality of a reconstruction.

To test our approach, we adopt the following setup. We first generate phylogenetic trees by sampling from a coalescent process. We decide to use Yule's coalescent instead of the more common Kingman. The latter tends to produce a large majority of internal nodes in close vicinity to the leaves with the others separated by very long branches, resulting in a trivial reconstruction for most nodes and a very hard one for the deep nodes. Yule's coalescent generates a more even distribution of node depths, allowing us to better evaluate reconstruction quality, see Supplementary Materialand Figure S2. For each tree, we simulate the evolution of sequences using a model that we refer to as "evolver" to obtain two multiple sequence alignments, one for the leaves and one for the internal nodes of the tree. We then reconstruct internal nodes using the desired approach by using the leaf alignment and the tree topology as input data.

We will consider two kinds of evolver models: *(i)* the same autoregressive model that we will then use for reconstruction, which is obviously an ideal case and *(ii)* an evolutionary model based on a Metropolis sampling of a Potts model. These two evolvers come from models trained on actual protein families: we use evolvers based on the PF00072 response regulator family for results of the main text, and show results for three other families (PF00014, PF00076 and PF00595) in the Supplementary Material. It is important to note that the approach that we propose only makes sense when considering the evolution of a precise protein family, on which the model in Eq. 2 is trained. Hence, any evolver model used in our simulations should reproduce at long times the statistics of the considered protein family, *i.e.* it should satisfy Eq. 5. For this reason, we only consider the two evolvers above and do not use more traditional evolutionary models such as an arbitrary GTR on amino-acids [16].

For reconstruction, we compare our autoregressive approach to the commonly used IQ-TREE program [17]. When supplied with a protein sequence alignment and a tree, IQ-TREE infers a joint substitution rate matrix for all sequence positions, with rate that can differ accross positions. Both methods run on a fixed tree topology, with branch lengths being re-inferred using maximum likelihood (see Methods).

We first investigate the case of the autoregressive evolver. This setting is of course ideal for our method, as there is perfect coincidence between the model used to generate the data and to perform ASR. We first evaluate the quality of reconstruction by computing the Hamming distance of the real and inferred sequences for each internal node of the simulated phylogenies. The left and central panels of Figure 1 show this Hamming distance as a function of the node depth, that is the distance separating the node from the leaves, and for a maximum likelihood reconstruction. On the left panel distance is computed including gap characters in the aligned sequences, while they are ignored on the central one. We see that the autoregressive reconstruction clearly outperforms the state of the art method, with the

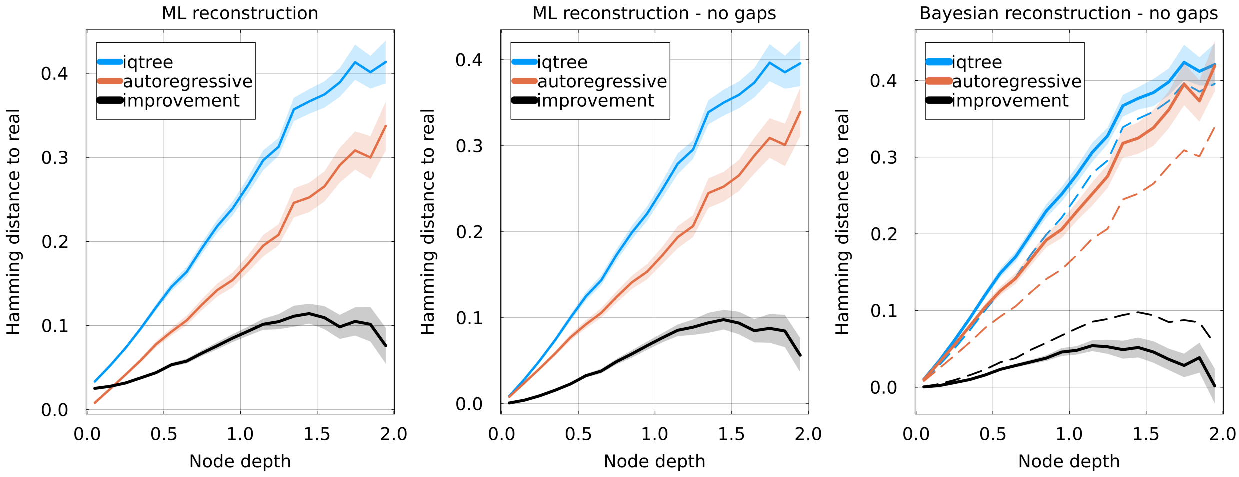

FIG. 1. Hamming distance between reconstructed and real sequences as a function of node depth, using IQ-TREE and our autoregressive approach. The difference between the two methods ("improvement") is shown as a black curve. Estimation of the incertitude is shown as a ribbon. The evolver and reconstruction autoregressive models are learned on the PF00072 family. **Left**: Hamming distance between the full aligned sequences, gaps included, using maximum likelihood reconstruction. **Center**: Hamming distance ignoring gapped positions, using maximum likelihood reconstruction. **Right**: comparison of Bayesian (solid lines) and maximum likelihood (dashed lines) reconstructions, ignoring gaps.

improvement in Hamming distance increasing with node depths and reaching 0.1.

Interestingly, the performance of IQ-TREE degrades if Hamming distance is computed including gaps, as in the left panel. This is because like other popular methods, IQ-TREE treats gaps in input sequences as unknown amino acids, and reconstructs an ancestral amino acid for gapped positions [17, 18]. On the contrary, our autoregressive approach treats gaps as if they were an additional amino acid and will reconstruct ancestral sequences that can contain gaps. This benefits the autoregressive approach as aligned ancetral sequences can in fact contain gaps. This effect is particularly visible at low node depths. However, ignoring the effects of gaps in the Hamming distance also leads to a clear improvement when using the autoregressive approach as shown in the central panel.

The right panel of Figure 1 shows the quality of the reconstruction for Bayesian reconstruction. In this case, a ensemble of sequences is reconstructed for each internal node, and the metric is the average Hamming distance between this ensemble and the real ancestor.

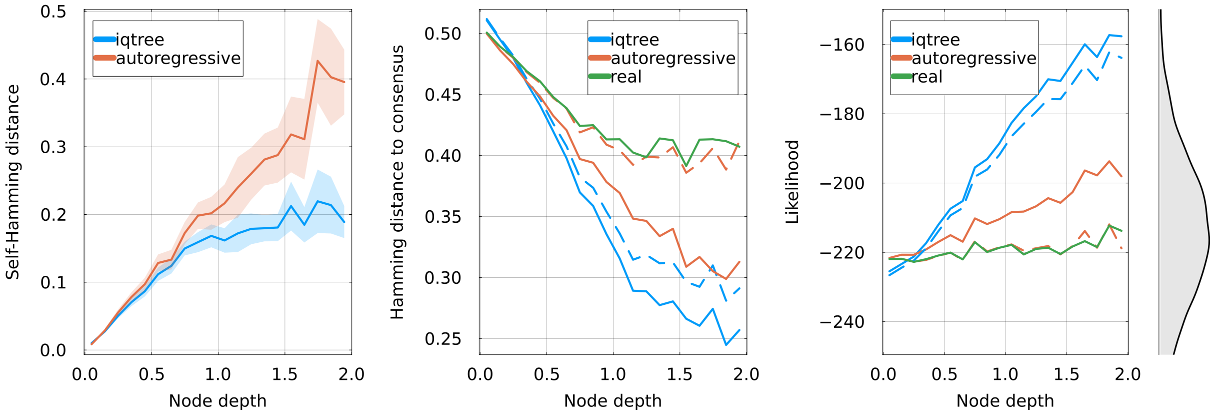

FIG. 2. **Left**: for Bayesian reconstruction, average pairwise Hamming distance among sequences reconstructed for each internal node. This quantifies the diversity of sequences obtained using Bayesian reconstruction. **Center**: Hamming distance between reconstructed sequences and the consensus sequence of the alignment. Solid lines represent maximum likelihood reconstruction or the real internal sequences, and dashed lines Bayesian reconstruction. IQ-TREE appears more biased towards the consensus sequence. **Right**: Likelihood of reconstructed and real sequences in the autoregressive model, *i.e.* using Eq. 2. Maximum likelihood methods (orange and blue solid lines) are biased towards more probable sequences. Bayesian autoregressive reconstruction gives sequences that are at the same likelihood level than the real ancestors.

Gaps are again ignored when computing the Hamming distance. We again observe an improvement when using the autoregressive method, of slightly lesser magnitude than in the maximum likelihood case.

III. DISCUSSION

IV. METHODS

A. ArDCA

The ArDCA model assigns a probability to any sequence of amino acids of length L given by

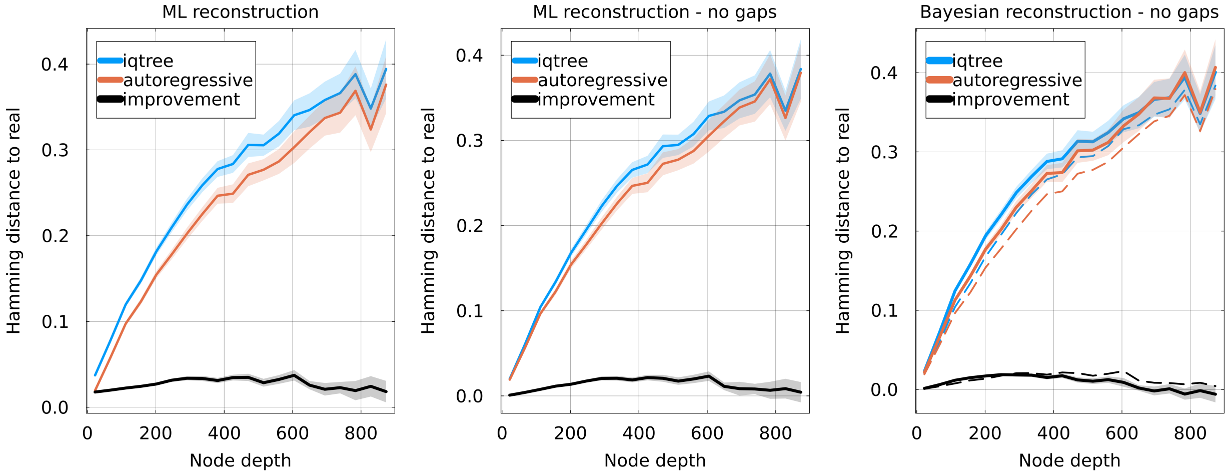

FIG. 3. Analogous to Figure 1, but using a Potts model as the evolver. Hamming distance between reconstructed and real sequences as a function of node depth, using IQ-TREE and our autoregressive approach. The difference between the two methods is shown as a black curve. The evolver and reconstruction autoregressive models are learned on the PF00072 family. **Left**: Hamming distance between the full aligned sequences, gaps included, using maximum likelihood reconstruction. **Center**: Hamming distance ignoring gapped positions, using maximum likelihood reconstruction. **Right**: comparison of Bayesian (solid lines) and maximum likelihood (dashed lines) reconstructions, ignoring gaps.

$$P^{AR}(\mathbf{a}) = \prod_{i\in\sigma(L)} p_i(a_i|a_{<i}), \tag{6}$$

where $\sigma(L)$ is a permutation of the L first integers and $a_{<i}$ stands for $a_1, \dots, a_{i-1}$. This means that the order in which the conditional probabilities p_i are applied is not necessarily the sequence order. The permutation σ is fixed at model inference.

Conditional probabilities p_i are defined as

$$p_i(b|a_{<i}) = \frac{1}{Z_i}\exp\left(\sum_{j<i} J_{ij}(b,a_j) + h_i(b)\right), \tag{7}$$

with the i q-dimensional vectors $J_{i\cdot}$ and h_i are learned parameters. It was shown in [5] that such a parametrization captures essential features of the variability of members of a proteins family.

B. Branch length inference

- how it works generally
- why I can't use AR directly
- how we do it with the profile model
- refer to the SI figure for results

C. Simulations

A simulation is performed as follows. First, a random tree of $n = 100$ leaves is generated from Yule's coalescent. We then normalize its height to a fixed value H that depends on the forward model used: for the autoregressive model we use $H = 2.0$, while for the Potts model combined with Metropolis steps, we use $H = 8$ sweeps, *i.e.* $H = 8L$ Metropolis steps where L is the length of the sequences.

A root sequence is sampled from the forward model's equilibrium distribution, and evolution is simulated along each branch independently starting from the root. In this way, we obtain for each repetition a tree and the alignments for internal and leaf nodes. Results presented in this work are obtained by averaging over $M = 100$ such simulations for each protein family.

- proper presentation of ardca (sequence ordering, functional form of cond probs)
- details of simulations: number of leaves, size of trees, etc...
- how I run iqtree + results of model finder
- how branch lengths are reconstructed

[1] Ziheng Yang. *Computational Molecular Evolution.* Oxford Series in Ecology and Evolution. Oxford University Press, Oxford, New York, October 2006. ISBN 978-0-19-856702-8.

[2] Ziheng Yang. Maximum likelihood phylogenetic estimation from DNA sequences with variable rates over sites: Approximate methods. *Journal of Molecular Evolution*, 39(3):306–314, September 1994. ISSN 1432-1432. doi:10.1007/BF00160154.

[3] Alexandros Stamatakis. RAxML version 8: A tool for phylogenetic analysis and post-analysis of large phylogenies. *Bioinformatics (Oxford, England)*, 30(9):1312–1313, May 2014. ISSN 1367-4811. doi:10.1093/bioinformatics/btu033.

[4] Lam-Tung Nguyen, Heiko A. Schmidt, Arndt von Haeseler, and Bui Quang Minh. IQ-TREE: A Fast and Effective Stochastic Algorithm for Estimating Maximum-Likelihood Phylogenies. *Molecular Biology and Evolution*, 32(1):268–274, January 2015. ISSN 0737-4038. doi:10.1093/molbev/msu300.

[5] Jeanne Trinquier, Guido Uguzzoni, Andrea Pagnani, Francesco Zamponi, and Martin Weigt. Efficient generative modeling of protein sequences using simple autoregressive models. *Nature Communications*, 12(1):5800, October 2021. ISSN 2041-1723. doi:10.1038/s41467-021-25756-4.

[6] Francisco McGee, Sandro Hauri, Quentin Novinger, Slobodan Vucetic, Ronald M. Levy, Vincenzo Carnevale, and Allan Haldane. The generative capacity of probabilistic protein sequence models. *Nature Communications*, 12(1):6302, November 2021. ISSN 2041-1723. doi:10.1038/s41467-021-26529-9.

[7] Michael Socolich, Steve W. Lockless, William P. Russ, Heather Lee, Kevin H. Gardner, and Rama Ranganathan. Evolutionary information for specifying a protein fold. *Nature*, 437(7058): 512–518, September 2005. ISSN 1476-4687. doi:10.1038/nature03991.

[8] Felsenstein, Joseph. *Inferring Phylogenies.* Oxford university press edition, September 2003. ISBN 978-0-87893-177-4.

[9] A L Halpern and W J Bruno. Evolutionary distances for protein-coding sequences: Modeling site-specific residue frequencies. *Molecular Biology and Evolution*, 15(7):910–917, July 1998. ISSN 0737-4038. doi:10.1093/oxfordjournals.molbev.a025995.

[10] Vadim Puller, Pavel Sagulenko, and Richard A Neher. Efficient inference, potential, and limitations of site-specific substitution models. *Virus Evolution*, 6(2), August 2020. ISSN

2057-1577. doi:10.1093/ve/veaa066.

[11] Simona Cocco, Christoph Feinauer, Matteo Figliuzzi, Remi Monasson, and Martin Weigt. Inverse Statistical Physics of Protein Sequences: A Key Issues Review. *Reports on Progress in Physics*, 81(3):032601, March 2018. ISSN 0034-4885, 1361-6633. doi:10.1088/1361-6633/aa9965.

[12] Adam J. Hockenberry and Claus O. Wilke. Phylogenetic Weighting Does Little to Improve the Accuracy of Evolutionary Coupling Analyses. *Entropy*, 21(10):1000, October 2019. ISSN 1099-4300. doi:10.3390/e21101000.

[13] Edwin Rodriguez Horta and Martin Weigt. On the effect of phylogenetic correlations in coevolution-based contact prediction in proteins. *PLoS computational biology*, 17(5):e1008957, May 2021. ISSN 1553-7358. doi:10.1371/journal.pcbi.1008957.

[14] Jose Alberto de la Paz, Charisse M. Nartey, Monisha Yuvaraj, and Faruck Morcos. Epistatic contributions promote the unification of incompatible models of neutral molecular evolution. *Proceedings of the National Academy of Sciences*, page 201913071, March 2020. ISSN 0027-8424, 1091-6490. doi:10.1073/pnas.1913071117.

[15] Bastien Boussau and Manolo Gouy. Efficient Likelihood Computations with Nonreversible Models of Evolution. *Systematic Biology*, 55(5):756–768, October 2006. ISSN 1063-5157. doi:10.1080/10635150600975218.

[16] Andrew Rambaut and Nicholas C. Grass. Seq-Gen: An application for the Monte Carlo simulation of DNA sequence evolution along phylogenetic trees. *Bioinformatics*, 13(3):235–238, June 1997. ISSN 1367-4803. doi:10.1093/bioinformatics/13.3.235.

[17] Bui Quang Minh, Heiko A Schmidt, Olga Chernomor, Dominik Schrempf, Michael D Woodhams, Arndt von Haeseler, and Robert Lanfear. IQ-TREE 2: New Models and Efficient Methods for Phylogenetic Inference in the Genomic Era. *Molecular Biology and Evolution*, 37(5):1530–1534, May 2020. ISSN 0737-4038. doi:10.1093/molbev/msaa015.

[18] Ziheng Yang. PAML 4: Phylogenetic Analysis by Maximum Likelihood. *Molecular Biology and Evolution*, 24(8):1586–1591, August 2007. ISSN 0737-4038. doi:10.1093/molbev/msm088.

Supplementary Material: Reconstruction of ancestral protein sequences using autoregressive generative models

Appendix A: Autoregressive evolution model

- Show the simplified expression for H=1
- discuss irreversibility, with example

Appendix B: Supplementary figures

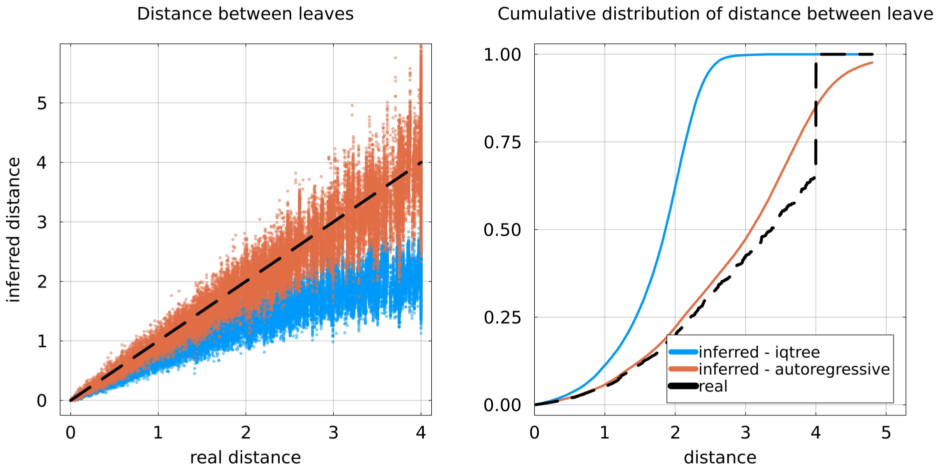

Figure S 1. Quality of branch length inference using data simulated with the autoregressive evolver. Inference is performed using the topology of a tree and leaf sequences generated using the autoregressive evolver. Two techniques are compared: IQ-TREE and the profile model corresponding to the autoregressive evolver. **Left**: inferred distance vs distance in the real trees for every pair of leaves. **Right**: Cumulative distribution of pairwise distance between leaves for the two inference methods and for the real trees. The discontinuity in the curve for the real tree is caused by the ultrametricity and fixed total height of the generated trees.

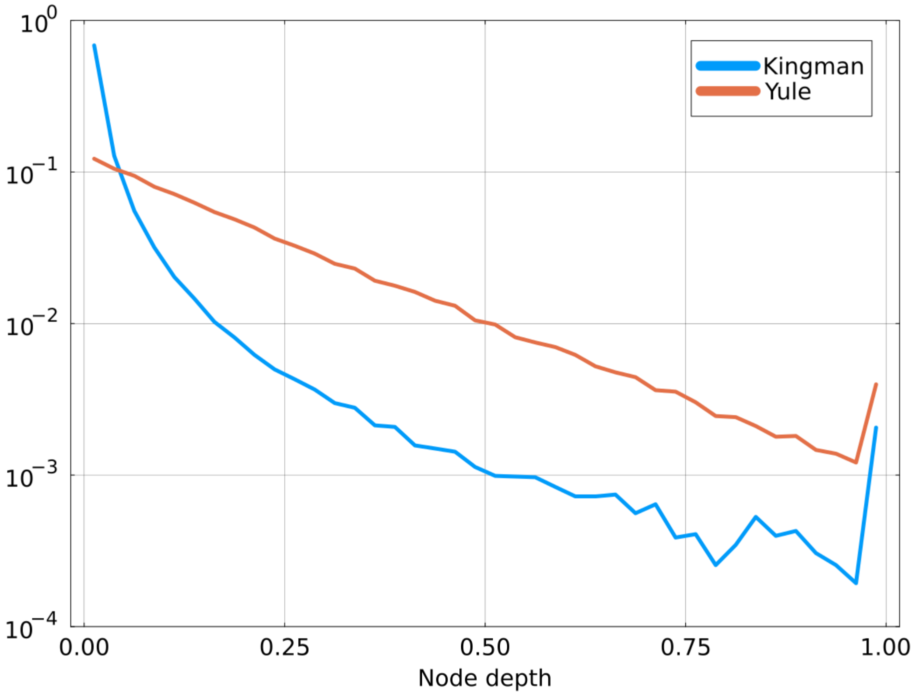

Figure S 2. Distribution of node depth for trees coming from the Kingman and Yule coalescents. Node depth is defined as the distance from a node to the closest leaf. Data is obtained by sampling several trees from each coalescent. Heights of trees are normalized to one. The Kingman process concentrates most of the nodes in close vicinity to the leaves, while the Yule process spreads them more evenly.