Reconstruction of ancestral protein sequences using autoregressive generative models

Matteo De Leonardis
DISAT, Politecnico di Torino, Corso Duca degli Abruzzi 24, 10129, Torino, Italy

Andrea Pagnani
DISAT, Politecnico di Torino, Corso Duca degli Abruzzi 24, 10129, Torino, Italy
Italian Institute for Genomic Medicine,
IRCCS Candiolo, SP-142, 10060, Candiolo, Italy and
INFN, Sezione di Torino, Via Pietro Giuria 1, 10125 Torino, Italy

Pierre Barrat-Charlaix*
DISAT, Politecnico di Torino, Corso Duca degli Abruzzi, 10129, Torino, Italy

Abstract

Ancestral sequence reconstruction (ASR) is an important tool to understand how protein structure and function changed over the course of evolution. It essentially relies on models of sequence evolution that can quantitatively describe changes in a sequence over time. Such models usually consider that sequence positions evolve independently from each other and neglect epistasis: the context-dependence of the effect of mutations. On the other hand, the last years have seen major developments in the field of generative protein models, which learn constraints associated with structure and function from large ensembles of evolutionarily related proteins. Here, we show that it is possible to extend a specific type of generative model to describe the evolution of sequences in time while taking epistasis into account. We apply the developed technique to the problem of Ancestral Sequence Reconstruction (ASR): given a protein family and its evolutionary tree, we try to infer the sequences of extinct ancestors. Using both simulations and data coming from experimental evolution we show that our method outperforms state-of-the-art ones. Moreover, it allows for sampling a greater diversity of potential ancestors, allowing for a less biased characterization of ancestral sequences.

* Correspondance to: PBC, DISAT pierre.barratcharlaix@polito.it

I. INTRODUCTION

Homologous proteins have a common evolutionary origin that can go back to billions of years. Throughout their evolution, they diversify through mutations while selection preserves their biological function. Consequently, many protein families contain thousands of sequences that are highly variable and yet maintain similar structures and functions. On the other hand, even a few mutations can destabilize a protein and destroy its function. A quantitative description how protein sequences change in time is thus a challenging problem, with important consequences for our understanding of the evolution of life.

Many probabilistic models of protein sequence evolution have been developed. Commonly used ones describe the evolution at each sequence position as a Markov chain across amino acid states, taking into account average properties of the substitution process such as more frequent transitions between similar amino acids [1–3]. Variations in evolutionary speed at different sites are often represented by using a set of substitution rates to which sites can be assigned, usually coming from a Gamma distribution [4]. An important and widely accepted assumption is that sequence positions evolve independently. This has the advantage of greatly simplifying sequence evolution models, making them convenient to manipulate analytically and computationally manageable. However, it comes at the cost of ignoring epistasis, that is the fact that the effect of a mutation depends on the rest of the sequence.

Sequence evolution models are used in the general field of phylogenetics which explores the evolutionary relations between proteins. An notable application is that of ancestral sequence reconstruction (ASR): given a set of homologous sequences and their phylogenetic tree, ASR consists in inferring likely sequences for the internal nodes of the tree, which correspond to extinct ancestral proteins. Reconstructed proteins can then be synthesized and tested in the lab. The technique is used to study the sequence-function relationship in proteins, for instance by understanding which mutations cause a change in enzymatic activity or binding specificity of a protein [5–7]. It can also be used to address fundamental evolutionary questions, such as the evolution reaction specificity or thermostability of proteins across the tree of life [8, 9].

The large amount of protein sequence data combined with recent theoretical and computational work has also allowed the development of generative protein sequence models. These models build on the idea that the sequence variability among homologous protein with similar biological functions inform us about the sequence-function relationship. In practice, generative models are trained using large amounts of protein sequences and consist of a probability distribution $P(\mathbf{s})$ over any potential amino acid sequence, with functional ones presumably being more probable. Classes of models include ones inspired from statistical physics such as the Potts model [10] and restricted Boltzmann machines [11], or based on neural networks such as transformers [12, 13]. A major achievement of these models is the possibility of using them to sample new artificial sequences that are distant from any natural protein but still functional [14, 15].

An essential ingredient for the success of generative models is the modeling of *epistasis*: the fact that the effect of a mutation on protein function depends on the rest of the sequence. Epistasis is caused by interaction between amino acids, and is essential to describe the fitness landscape of a protein [16, 17]. Interestingly, it has also been suggested that epistasis may be the cause of variable evolutionary rates across phylogenetic trees [18]. Since common sequence evolution models ignore epistasis, they can only represent a crude approximation of the evolutionary constraints acting on a protein. As the change of a protein sequence in time depends on functional constraints, it is reasonable to expect that an inaccurate representation of the fitness landscape negatively affects the modeling of dynamics.

There has been effort in the phylogenetics community to develop models that take epistasis into account. For instance in [19, 20], authors build an evolutionary model based on a structure-based fitness landscape. The evolutionary models obtained in this way can be used to detect the presence of epistasis and to show that including it leads to better fit of the data, but not to infer a phylogenetic tree or to reconstruct the states at internal nodes. Other approaches that perform phylogenetic inference under the assumption of co-evolution make strong approximations such as the one of non-overlapping pairs of co-evolving sites [21]. Another promising direction is the use of generative models for phyogenetic tasks. However, the non-independence of mutations that characterizes generative models makes it challenging to use them for dynamical purposes. Different studies have proposed using Potts models to describe evolutionary dynamics, but current techniques allow for little analytical treatment and are limited to forward simulation of sequences [22, 23].

In this study, we set out to extend the application of generative models to describe evolutionary dynamics. First, we develop an analytically and numerically tractable sequence evolution model with generative properties, based on the so-called ArDCA generative model and its autoregressive architecture [24]. Our model accounts for epistasis and is generative over long-term evolution, but also allows use of some of the standard techniques used in phylogenetics such as *e.g.* Felsenstein's pruning algorithm or an algorithm for irreversible models that we use here [25, 26]. We then apply our model to ancestral sequence reconstruction (ASR) and demonstrate, using simulated data, that it outperforms state-of-the-art reconstruction techniques that assume independent sites, both when maximizing or sampling from the posterior. We use the program IQ-TREE [27] to compare to state of the art methods, and the list of methods that we use within IQ-TREE is detailed in the Methods section. Finally, we validate our approach with recent experimental data on directed evolution and show that reconstruction of a known ancestor is done more accurately than using a site-independent method. To our knowledge, this is the first use of such data to evaluate reconstruction methods.

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 $P_i(b|a,t) = \left(e^{t\mathbf{Q}^i}\right)_{ab}$.

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 with positive off-diagonal elements and $\mathbf{\Pi}$ is diagonal with positive entries that sum to 1 [28]. The diagonal elements of $\mathbf{H}$ are determined by requiring that the rows of $\mathbf{Q}$ sum to zero. The two matrices have simple interpretations. On the first hand,

$\mathbf{\Pi}$ fixes the long-term equilibrium frequencies, that is $P_i(b|a,t) \xrightarrow[t\to\infty]{} \pi_b$. On the other, $\mathbf{H}$ influences the dynamics of the Markov chain but does not change the equilibrium distribution. Most commonly, both matrices are considered to be independent of the sequence position i, and $\mathbf{H}$ can potentially be scaled in order to represent different rates of evolutionary change [4].

In order to incorporate constraints coming from a protein's structure and function into the evolutionary model, we develop a protein family specific model of protein sequence evolution based on the the autoregressive generative model ArDCA [24]. Autoregressive models *à la* ArDCA build from the the chain rule of conditional probabilities:

$$P(a_1,\ldots,a_L) = P(a_1)P(a_2|a_1)\ldots P(a_L|a_1,\ldots,a_{L-1}) = \prod_{i=1}^{L} P(a_i|a_{<i}) \tag{2}$$

where $a_{<i} = a_1,\ldots,a_{i-1}$ represents the amino acid states before position i and L is the length of the sequence. By construction, Eq. (2) is an exact decomposition of the joint probability distribution of the sequence $a_1,\ldots,a_L$. There are $L!$ such decompositions of P: for any permutation σ of the positions $\{1,\ldots,L\}$, $P(a_1,\ldots,a_L) = \prod_{i=1}^{L} P(a_{\sigma_i}|a_{<\sigma_i})$ is another exact decomposition of P.

ArDCA models the diversity of sequences in a protein family by proposing a specific functional form for conditional probabilities. In other words, the model is defined by L functions p_i depending on parameters $\boldsymbol{\theta}_i$ with the desired property

$$p_i(a_i|a_{<i};\boldsymbol{\theta}_i) \simeq P(a_i|a_{<i}). \tag{3}$$

The precise functional form of $p_i(a_i|a_{<i};\boldsymbol{\theta}_i)$ is given in the Methods section. The model then assigns a probability $P^{AR}(\mathbf{a})$ to any sequence $\mathbf{a} = \{a_1,\ldots,a_L\}$ of L amino acids:

$$P^{AR}(\mathbf{a}) = \prod_{i=1}^{L} p_i(a_i|a_{<i};\boldsymbol{\theta}_i), \tag{4}$$

Note that since the model is trained on aligned sequences, states a_i can include the gap symbol, which is treated as any other amino acid. 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 $a_1,\ldots,a_{i-1}$. The set of parameters $\{\boldsymbol{\theta}_i\}$ is learned by maximum-likelihood using the aligned sequences of members of the family. Note that the autoregressive architecture is also employed in the context of deep-learning methods, to which the model we describe below could potentially be generalized [13, 29]. Deep autoregressive methods differ from ArDCA in that they use a more complex parametrization of p_i and are usually trained on large set of unaligned proteins rather than a single family.

As explained above, the decomposition of Eq. 2 is valid for any ordering of the sequence positions $\{1, \ldots, L\}$. Each decomposition will lead to a different set of parameters $\{\boldsymbol{\theta}_i\}$ and thus to a different generative model. The ordering used in ArDCA is not the natural $\{1, \ldots, L\}$ but rather an order where positions are sorted by increasing variability, which has been shown to give good generative capacities [24]. For simplicity, we keep the notation of Eq. 4: the position we call $i = 1$ is not the first sequence position but rather the most conserved one, and so on until $i = L$ which represents the most variable position.

It has been shown in [24] that the generative capacities of ArDCA are comparable to that of state of the art models such as bmDCA [17]. This means that a set of sequences sampled from the probability in Eq. 4 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. The generative capacities of a protein model comes from its ability to represent epistasis, that is the relation between the effect of a mutation and the 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 at the previous positions $\{1, \ldots i-1\}$.

We take advantage of the autoregressive architecture to define a generative evolutionary model. Given two amino acid sequences $\mathbf{a}$ and $\mathbf{b}$, we propose that the probability of $\mathbf{a}$ evolving into $\mathbf{b}$ in time t take the form

$$P(\mathbf{b}|\mathbf{a}, t) \overset{\text{def}}{=} \prod_{i=1}^{L} q_i(b_i | a_i, b_{<i}, t), \tag{5}$$

where the position specific conditional propagator q_i is defined as

$$q_i(b_i|a_i, b_{<i}, t) = \left(e^{t \cdot Q^i(b_{<i})}\right)_{a_i, b_i}, \quad Q^i(b_{<i}) = \mathbf{H} \cdot \begin{pmatrix} p_i(1|b_{<i}) & 0 & 0 \\ 0 & \ddots & 0 \\ 0 & 0 & p_i(q|b_{<i}) \end{pmatrix}. \tag{6}$$

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. Compared to Eq. 1, matrices $\mathbf{H}$ and $\mathbf{\Pi}$ now depend on the position i but also on the context $b_{<i}$. An important consequence of this choice is that our evolutionary model will converge at long times to the generative distribution P^{AR}:

$$q_i(b_i|a_i, b_{<i}, t) \xrightarrow[t\to\infty]{} p_i(b_i|b_{<i}), \quad P(\mathbf{b}|\mathbf{a}, t) \xrightarrow[t\to\infty]{} P^{AR}(\mathbf{b}). \tag{7}$$

We argue here that such a property is essential to build a realistic protein sequence evolution model, particularly when considering evolution over long periods. 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 [16]. 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.

In the Methods section and in the Supplementary Material , we compute the transition rates associated to the propagator of Eq. 5 and show that it can be seen as an approximation of dynamics in the fitness landscape defined by P^{AR}. It becomes exact at large times, as Eq. 7 points out, and at small times. There are caveats to this approximation: our model has a non-reversible dynamic – although the context-dependent site propagators in Eq. 6 are reversible – and in fact is not even a Markov process. 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 [25]. However, it is definitely out of the ordinary to model evolution with a non Markovian process. Another undesired consequence is that the generative distribution P^{AR} is not stationary at all times in this process. This is in principle worrying, as it means that if dynamics are started from natural sequences, sequences generated at intermediate times could be non-functional according to the generative model.

These caveats are, to some extent, the price to pay to model epistasis on long time scales – see Eq. 7 – while keeping an analytically tractable model. While definitely undesirable, they seem to have limited quantitative consequences: in Figure S1, we show that deviations of the dynamics from the equilibrium P^{AR} are quantitatively small. Another argument in this direction is the fact that reconstruction depends weakly on the placement of the root, indicating that the irreversibility of the model is not too strong (Section B 3 of the

Supplementary Material). Furthermore, the results that we present below show that our propagator improves ASR in different settings and can thus be seen as a useful approximation.

A final remark is that, as the ArDCA model itself, the proposed dynamic depends on the order in which decomposition Eq. 2 is made. Indeed, a consequence of the autoregressive structure of the model is that the first position treated by the model ($i = 1$) "evolves" independently from the context, while the last one depends on all the rest of the sequence. In practice, it is difficult to say whether a given ordering better describes biological evolution: there is an astronomically large number of permutations $L!$, and there is no obvious direct measure of whether one better fits evolutionary dynamics. For this reason, we make the simplifying choice of only considering the ordering by increasing diversity of sites, which has been found in [24] to have good generative capacities.

We underline that this approach has important differences with standard models of evolution used in phylogenetics. 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 evolutionary model is then kept low to reduce the risk of overfitting, for instance by using a predetermined set of evolutionary 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 computationally intensive algorithms [30, 31].

Here instead, parameters of the generative model in Eq. 4 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 [10, 24]. Despite this, it appears that the generative properties of such models are not strongly affected by the phylogeny [32, 33]. This allows us to proceed in two steps: first construct the model from data while ignoring phylogeny, and then use it for phylogenetic inference tasks.

An advantage of this approach is that once the model of Eq. 4 is inferred, the propagator in Eq. 5 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 of more complex models of evolution [18]. However, a constraint is that the inference of the generative model requires the existence of an appropriate training set, that is a protein family with sufficient variability among its members.

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.

The reconstruction with the autoregressive model proceeds as follows.

(i) For position $i = 1$, we use the evolutionary model defined by the equilibrium frequencies p_1 to reconstruct a state a_1^n at each internal node n of the tree. For $i = 1$ the transition rate matrix Q^1 as defined in Eq. 6 depends only on p_1, which in turn does not depend on the context. For a branch of length t, the transition probabilites between two states a and b is $q_1(b|a,t) = \left(e^{tQ^1}\right)_{ab}$.

(ii) Iterating through subsequent positions $i > 1$: we reconstruct state a_i^n at each internal node n using the model defined in Eq. 6, with the context $a_{<i}^n$ having been already reconstructed in the previous iterations. The procedure is the same as the $i = 1$ case, the only difference being that the transition rate matrix Q^i now also depends on the context at postions $1, \ldots, i-1$.

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 evolutionary 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 the cause of the time-irreversibility of the model. For this reason, we compute the probability of reconstructions using an algorithm adapted to irreversible models [26], described in details in Section A of the Supplementary Material .

Using this technique we obtain, for any internal node n and any alignment position i, the posterior probability $P(a_i^n|\mathcal{T}, \mathcal{D})$ of the amino acid state a_i^n given the tree $\mathcal{T}$ and the sequences at the leaves $\mathcal{D}$. This probability is computed by marginalizing over other the states of other internal nodes. We call *maximum a posteriori* reconstruction (MAP) the state obtained by maximizing $P(a_i^n|\mathcal{T}, \mathcal{D})$. In this case, each iteration reconstructs the most probable residue at position i for all internal nodes of the tree. Alternatively, states of internal nodes can be sampled from $P(a_i^n|\mathcal{T}, \mathcal{D})$ to obtain a *posterior sampling* reconstruction. In any case, our reconstruction is marginal: the posterior at a node is obtained by marginalizing over the states of other nodes. While it is in principle possible to extend it to joint reconstruction, as explained in [26], we have not implemented it and do not consider it in this work.

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 a 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 S4.

A consequence of the irreversibility of the evolutionary model is that the reconstruction potentially depends on the placement of the root of the tree. This is not an issue in the results that follow since we work with simulated trees for which the root is known exactly. However, it may be a concern when applying this to biological datasets. In Section B 3 of the Supplementary Material , we explore the effect of root placement on the reconstruction. Results are overall reassuring, with the difference between reconstructions remaining below a Hamming distance of 0.5% even for large errors in root placement.

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 for nodes that are remote from the leaves. 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 internal nodes depths (defined as the distance to the closest leaf), allowing us to better evaluate reconstruction quality, see Supplementary Material and Figure S5. 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 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 (see Table I for details on these three other families). It is important to note that the approach that we propose only makes sense when considering the evolution a protein family on which the model in Eq. 4 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. 7. 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 [34].

For reconstruction, we compare our autoregressive approach to the commonly used IQ-TREE program [27] with the flag `-m MFP` to use the ModelFinder [35]. In this mode, when supplied with a protein sequence alignment and a tree, IQ-TREE infers a joint substitution rate matrix for all sequence positions. Because the best evolutionary model found may differ when using two different alignments, we pick for each family the model most commonly found by IQ-TREE across a reduced range of simulations (Methods). The list of models found

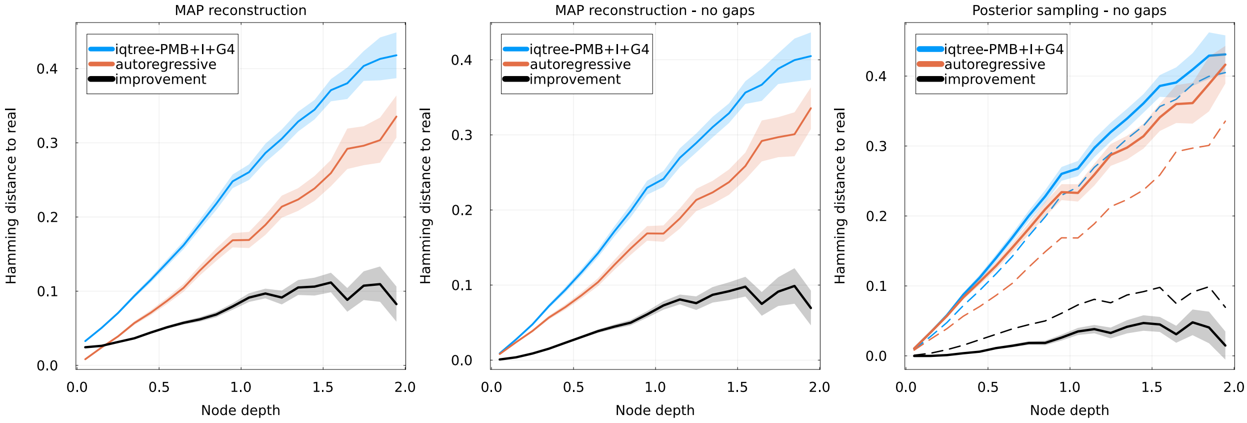

FIG. 1. Hamming distance (normalized by sequence length) between reconstructed and real sequences as a function of node depth, defined as the distance from the node to the closest leaf in the "ground-truth" tree used to simulate the data. Reconstruction if performed using IQ-TREE and our autoregressive approach, with the evolutionary model used by IQ-TREE reported in the legend. The difference between the two methods ("improvement") is shown as a black curve. Estimation of the uncertainty 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 a posteriori reconstruction. **Center**: Hamming distance ignoring gapped positions, using MAP reconstruction. **Right**: comparison of posterior sampling (solid lines) and MAP (dashed lines) reconstructions, ignoring gaps.

and used in our analysis is reported in Methods (section IV F): in most cases, the PMB matrix was used [36], with different options for across-sites rate variability (`+I+G4` or `+I+R3`). Ancestral states are then reconstructed using an empirical Bayesian method [37]. We either selected the state corresponding to the maximum of the posterior (MAP) or sampled from the posterior. In the extra analysis of the Supplementary Material , we also use the flag `+C60` to perform reconstruction using profile mixture models [38]. As for the autoregressive model, we provide the topology of the real tree to IQ-TREE and let it re-compute the branch lengths.

Autoregressive evolver. 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 (Figure S10 for additional families) show this Hamming distance as a function of the node depth, that is the distance separating the node from the leaves along the branches of the tree on which evolution was simulated, and for a MAP reconstruction. Hamming distance is computed including gap characters in the aligned sequences on the right panel, while they are ignored on the central one, and is normalized by the length of the sequence: a distance of 1 would thus indicate entirely different sequences. We see that the autoregressive reconstruction clearly outperforms the state of the art method: the improvement in Hamming distance increases with node depths, and the distance to the real ancestor drops from ~ 0.4 to ~ 0.3 when using the autoregressive approach. The increase in reconstruction quality with node depths is consistent with recent findings that epistasis only becomes important at relatively large sequence divergences [39, 40].

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 [27, 41]. On the contrary, our autoregressive approach, like many generative models, treats gaps as if they were an additional amino acid and will reconstruct ancestral sequences that can contain gaps. This effect is particularly visible at low node depths and benefits the autoregressive approach as aligned ancestral sequences can in fact contain gaps. Considering gaps as an additional amino acid is an advantage in our setup, as both evolvers use this convention. However, it is not clear that this advantage extends to real biological data, as the insertion-deletions processes during evolution may not be accurately captured by our model. For this reason, we also show the performance of reconstruction when ignoring the effects of gaps in the Hamming distance. This also leads to a smaller but 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 when reconstructing by sampling the posterior. In this case, an ensemble of sequences is reconstructed for each internal node, and the metric is the average Hamming distance between this ensemble and the real ancestor. 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 MAP case.

To understand how these results depend on the complexity of the evolutionary model used

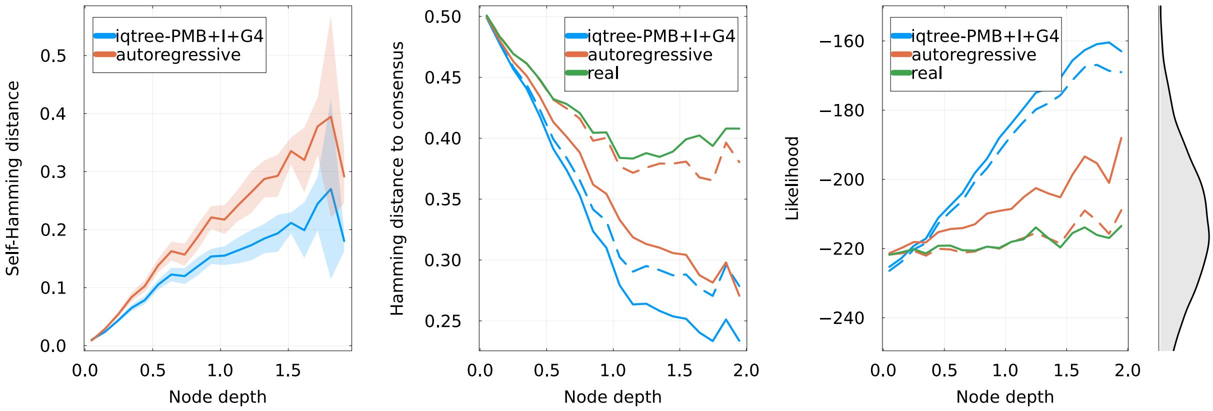

FIG. 2. **Left**: for posterior sampling reconstruction, average pairwise normalized Hamming distance among sequences reconstructed for each internal node. This quantifies the diversity of possible ancestral reconstructions. **Center**: Normalized Hamming distance between reconstructed sequences and the consensus sequence of the alignment. Solid lines represent MAP reconstruction or the real internal sequences, and dashed lines posterior sampling. IQ-TREE appears more biased towards the consensus sequence. **Right**: Log-likelihood of reconstructed and real sequences in the autoregressive model, *i.e.* using the logarithm of Eq. 4. MAP methods (orange and blue solid lines) are biased towards more probable sequences. Posterior sampling autoregressive reconstruction gives sequences that are at the same likelihood level than the real ancestors. The equilibrium distribution of likelihood of sequences generated by Eq. 4 is shown on the right.

by IQ-TREE, we extend the comparison to reconstruction using the profile mixture models proposed by IQ-TREE [38]. In our case, we use the `C60` flag to have IQ-TREE infer 60 different site specific profiles, with the likelihood at each site being averaged over these profiles. Results are shown in Supplementary Figure S7 (Figure S11 for additional families). It is clear that the profile model improves IQ-TREE's reconstruction, as the improvement now peaks at a Hamming distance of approximately 0.06 instead of 0.1 in Figure 1. However, the performance of the autoregressive reconstruction remains consistently above the independent model.

Properties of reconstructed sequences. To further analyze the reconstructed sequences, we first look at the diversity of generated ancestors when sampling the posterior. The left panel of Figure 2 (Figure S12 for additional families) shows the average normalized Hamming distance between sequences reconstructed at the same internal node, as a function of depth.

For deeper nodes (depth $\gtrsim 1$), the autoregressive approach reconstructs a significantly more diverse set of sequences than IQ-TREE: Hamming distance between reconstructions saturates at 0.2 for the latter, while it steadily increases for the former. Higher diversity can be interpreted as a greater uncertainty concerning the ancestral sequence. However, this must be put in the context of Figure 1: sequences obtained by autoregressive reconstruction are more varied but also on average closer to the real ancestor.

The difference in sequence diversity for the two methods is in part explained by the central panel of Figure 2, which shows the Hamming distance between reconstructed ancestors and the consensus sequence of the multiple sequence alignment at the leaves. It appears there that for deep nodes, IQ-TREE reconstructs sequences that are relatively similar to the consensus, with an average distance between the posterior sampling reconstruction and the consensus of about 0.3. Contrasting with that, results of the autoregressive method shows less bias towards the consensus with an average distance of 0.4 for deep nodes, in line with the real ancestors. We also note that MAP sequences for both method are always closer to the consensus than sampled ones, a bias that had already been observed [42].

The bias induced by ignoring the equilibrium distribution of the sequences is also visible in the right panel of Figure 2: it shows the log-likelihood of reconstructed and real ancestral sequences according to the generative model. Note that the log-likelihood here comes from the log-probability of Eq. 4 and can be interpreted as the "quality" of a sequence according to the generative model. It is unrelated to the likelihood computed in the phylogenetic reconstruction algorithm. Reconstructions with IQ-TREE increase in likelihood when going deeper in the tree, eventually resulting in "too good" sequences that are very uncharacteristic of the equilibrium generative distribution as can be seen from the histogram on the right. This effect also happens with the MAP reconstruction of the autoregressive model, although to a lesser extent. The autoregressive reconstruction obtained from sampling the posterior does not suffer from this bias and reconstructs sequences with a log-likelihood that is similar to that of the real ancestors. Interestingly, IQ-TREE's reconstruction using a profile model suffers less from these biases, as can be seen in Figures S8& S13. This suggests that having a more precise evolutionary model tends to reduce biases in the reconstruction.

Potts evolver. We assess the performance of our reconstruction method in the case where the evolver is a Potts model. Potts models are a simple type of generative model and have been used extensively to model protein sequences. They can be used to predict contact in three dimensional structures, effects of mutations, protein-protein interaction partners [10]. They can be sampled to generate novel sequences which are statistically similar to natural ones and often functional [15, 23]. Additionally, it has recently been shown that they can be used to describe the evolution of protein sequences both qualitatively and quantitatively [22].

Potts and autoregressive models both accurately reproduce the statistical properties of protein families. In this sense, they correspond to similar long-term generative distributions in the sense of Eq. 7. However, the dynamics of a Potts model are fundamentally different from the ones of usual evolutionary models, including our autoregressive one. Indeed, they are described by a *discrete* time Markov chain, instead of the continuous time used in models based on substitution rate matrices such as in Eq. 1 [23]. For Metropolis steps which we use here, the discrete time corresponds to attempts at mutation which can be either accepted or rejected depending on the effect of the mutation according to the model. These dynamics naturally give rise to different evolutionary timescales for various sequence positions, as well as interesting qualitative behavior such as the entrenchment of mutations [40].

To see how this change in dynamics affects our results, we *(i)* sample a large and varied ensemble of sequences from the Potts model and use it to train an autoregressive model, in a way to guarantee consistent long-term distributions between the Potts and autoregressive, and *(ii)* evolve the Potts model along random phylogenies, generating alignments for the leaves and the internal nodes in the same way as above. We then attempt reconstruction of internal nodes using the inferred autoregressive model and IQ-TREE. Figure 3 shows the results of reconstruction, with panels directly comparable to Figure 1. We again see a consistent improvement when using the autoregressive model over IQ-TREE, although of a much smaller amplitude, with an absolute improvement gain in Hamming distance of about 2% for deep internal nodes.

D. Results on experimental evolution data.

We take advantage of recent developments in directed evolution experiments to test our method in a controlled setting. We use the data published in [43]: in this work, authors evolved the antibiotic resistant proteins β-lactamase PSE-1 and acetyltransferase AAC6 by submitting them to cycles of mutagenesis and selection for function. Starting from a wild-type

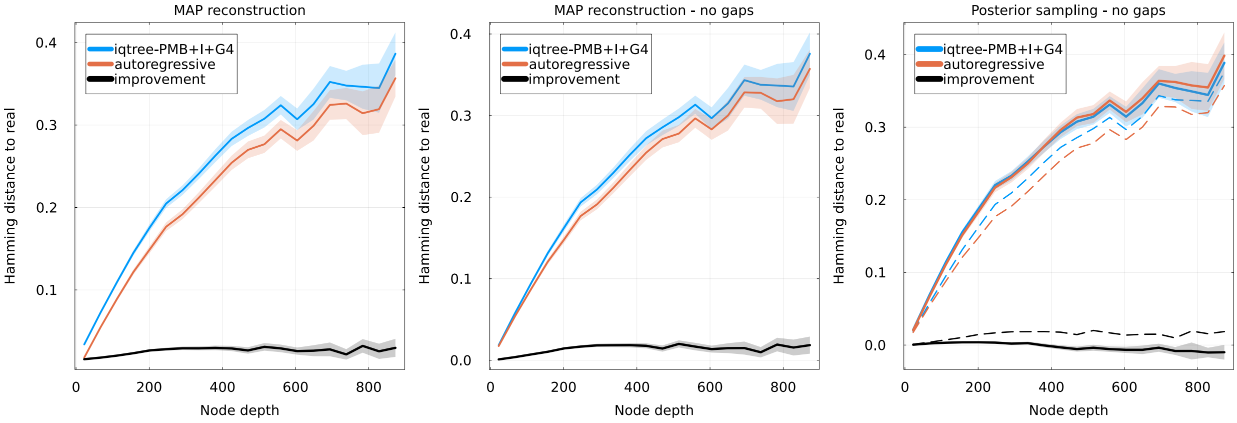

FIG. 3. Analogous to Figure 1, but using a Potts model as the evolver. Normalized Hamming distance between reconstructed and real sequences as a function of node depth, using IQ-TREE and our autoregressive approach. The difference between the two methods is shown as a black curve. The evolver and reconstruction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

protein, they obtained thousands of diverse functional sequences after the directed evolution. An interesting result of this work is that it is possible to recover structural information about the wild-type from the set of evolved sequences.

Here, we use this data as a test setting for ASR: the sequences obtained after directed evolution all derive from a common ancestor, the wild-type, of which we know the amino acid sequence. We can thus reconstruct the wild-type sequence using different ASR methods and compare it to the ground truth. The phylogeny is not known, but given the large population size during the experiment and the relatively low number of selection rounds, it is reasonable to approximate it using a star-tree, *i.e.* a tree with a single coalescent event taking place at the root (see Methods). Since the reconstruction task is most interesting when using relatively varied sequences, we decide to use data for the PSE-1 wild-type where 20 cycles of mutagenesis & selection have been performed, resulting in a mean Hamming distance of 12% to the wild-type.

Our ASR procedure is as follows. We randomly pick the amino acid sequences of M

proteins among the ones evolved from PSE-1 after 20 cycles of mutagenesis & selection, with $3 \leq M \leq 640$. The total number of sequences at round 20 of directed evolution is much larger, making it computationally hard to use all of them. We then construct a star-like phylongeny and place the M selected sequences at the leaves, and perform ASR using either IQ-TREE or our autoregressive method which we have trained on an alignment of PSE-1 homologs. We obtain the reconstructed amino acid sequence of the root, which we can then compare to the actual wild-type. As a comparison, and because our approximation of the phylogeny is very simple, we also attempt to reconstruct the root by taking the consensus sequence of the M leaves. We repeat this procedure 100 times for each value of M for a statistical assessment of the different methods.

The results are shown in Figure 4. The left panel shows the average non-normalized Hamming distance to the wild-type as a function of the number of leaves used M. For a low M, all methods understandably make a large number of errors, with a mean Hamming distance larger than 10 for $M = 3$. For a higher M, IQ-TREE and the autoregressive method stabilize to a fixed number of errors: we find a Hamming distance of ~ 4.3 for IQ-TREE and ~ 2.9 for the autoregressive. The consensus curiously reaches a minimum at intermediate M, a fact discussed in the Supplementary Material , and saturates at a Hamming distance of 6 when considering all sequences of the round 20. The reconstruction errors are overwhelmingly located at six sequence positions. In the central panel, the fraction of mistakes made at these six positions over the 100 repetitions of $M = 640$ leaves is shown for each method. We observe that there are two positions (169 and 193) where IQ-TREE systematically fails at recovering the wild-type state while the autoregressive model's reconstruction is correct. Interestingly, the corresponding mutations are considered beneficial by the ArDCA model, see Figure S6. Inversely, IQ-TREE recovers the wild-type state more often at position 107. The right panel shows the logo of the set of reconstructed sequences at these 6 positions and for each method.

Overall, we see that the reconstruction of the autoregressive model is more accurate. This gain in accuracy comes from the representation of the functional constraints acting on the PSE-1 protein by the generative model, which are inferred separately using an alignment of homologs. The improvement in reconstruction errors is modest, going from an average Hamming distance of 4.3 to 2.9. However, the gain is intrinsically limited by the data itself: the evolved sequences have an average Hamming distance of about 12% to the ancestor,

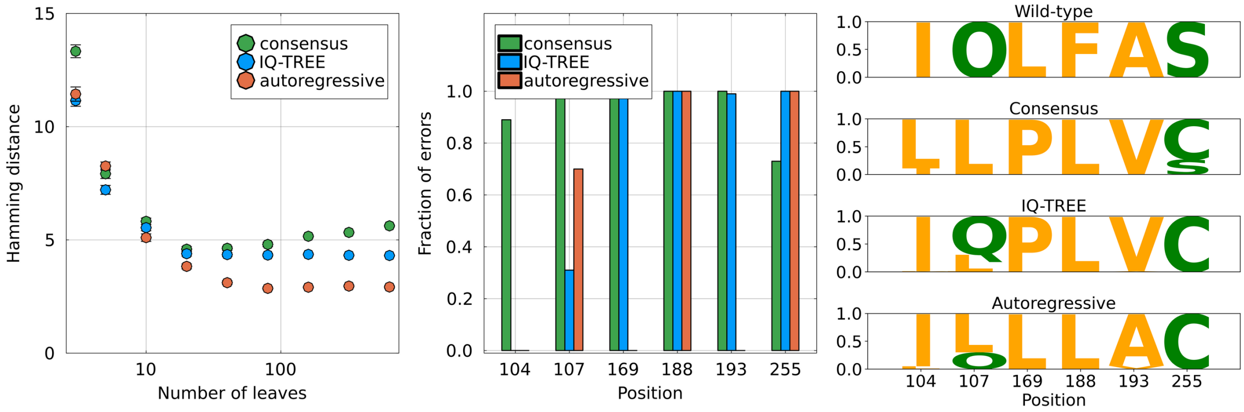

FIG. 4. Reconstruction of the wild-type PSE1 sequence used in [43] using sequences from round 20 of the directed evolution. **Left.** Non normalized Hamming distance to the wild-type PSE1 sequence as a function of the number of sequences M used for reconstruction. The fact that the consensus method has a local minimum is discussed in the Supplementary Material . For comparison, the average distance between a leaf sequence and the wild-type is 25. The error bars are computed using the standard deviation obtained from the 100 choices of sequences. **Middle.** For the six sequence positions where most of the reconstruction errors are located, fraction of errors of each method out of 100 independent reconstructions using different sets of $M = 640$ leaves. **Right.** Sequence logo of the reconstructed sequence for the three methods, obtained using 100 independent reconstructions with different sets of $M = 640$ leaves. The logo is only shown for the six positions where most errors are located. For example, all three methods fail 100 times at position 147, reconstructing a leucine L instead of a phenylalanine F.

which is experimentally challenging but remains small compared to the divergence found in the homologs of PSE-1. For instance, the root-to-tip distance estimated by IQ-TREE and the autoregressive model are respectively 0.13 and 0.15, corresponding to the regime of shallow trees when comparing with Figure 1.

III. DISCUSSION

The reconstruction of ancestral protein sequences has long been a cornerstone of evolutionary biology, helping to elucidate the mechanisms of protein function and evolution over billions of years. The accuracy of ASR has profound implications not only for our understanding of evolution but also for practical applications in synthetic biology and proteins engineering. However, the widely used models in phylogenetics often rely on the assumption of independent sequence evolution at different positions, neglecting epistatic interactions that play a crucial role in determining protein function. This simplification limits their ability to accurately capture the full complexity of evolutionary dynamics.

In this study, we addressed this limitation by developing a novel generative model based on the ArDCA autoregressive framework, which explicitly accounts for epistasis, an essential factor in protein evolution. By incorporating the dependencies between amino acids within sequences, our model offers a more realistic description of protein evolution, capturing the non-independence of mutations over time. A significant contribution of this work is extending the application of generative models to cope with phylogenetic constraints. Our model not only preserves the generative capacity over long-term evolution but it also enables the use of classical phylogenetic techniques normally restricted to independent-site models. The ability to integrate generative context-aware models into these established algorithms represents a substantial advance, allowing for more accurate inference of evolutionary relationships and ancestral states. This, besides the theoretical interest in ASR, is a powerful tool to help us understanding how phylogenetic constraints impact the structure and/or the function of the protein of interest.

Our evaluation of the model using simulated data demonstrated that it outperforms IQ-TREE, a state-of-the-art tool for ASR, in reconstructing ancestral sequences. This improvement highlights the importance of incorporating epistasis into evolutionary models, as ignoring these interactions likely leads to less accurate reconstructions. Furthermore, we validated our approach using experimental data from directed evolution experiments. These data offer a unique opportunity to test the accuracy of ASR methods, and our model achieved more accurate reconstructions of known ancestors compared to IQ-TREE, underscoring the robustness of our approach.

Using the generative nature of our model we can sample sequences at internal nodes that should in principle remain functional despite being distant from any naturally occurring protein. Most ASR studies have used maximum a posteriori or maximum likelihood reconstructions, as Bayesian reconstructions are more often found to accumulate deleterious mutations and can be non-functional [9, 44]. At the same time, the most likely solution can be biased and may be unrepresentative of the phenotype of the real ancestor, leading to incorrect biological conclusions [42, 45]. We ourselves observe these biases in our simulations, in the form of a convergence to the consensus sequence and an unnaturally high likelihood according to the generative model. Being able to propose an ensemble of sequences sampled from a generative model at each internal node could thus lead to more robust biological conclusions about ancestral life.

Another feature of our model is the way it models gaps. IQ-TREE, as well as many other phylogenetic reconstruction methods, treats alignment gaps as missing information, and will reconstruct amino-acid states at these positions [27, 41]. In contrast, most alignment based generative models such as ArDCA treat gaps as a particular state that a position can be in, on equal footing with other amino acids [12, 17, 24]. This can have drawbacks when modeling evolution, as the dynamics of insertions-deletions and of point mutations can be quite different [46]. However, being able to model gaps during ancestral reconstruction likely increases accuracy, as there is no reason to think that ancestral sequences would align to extant ones without any gaps.

Despite its good performance, our model comes with several caveats. First, our *ad hoc* way to infer branch lengths is not ideal and differs from standards used in the field. The method would clearly benefit from improvements in this direction. More importantly, the nature of our approximation has unsatisfying consequences, as the dynamic is non Markovian, irreversible, and does not remain at equilibrium with the generative model at all times. As evolution in an epistatic landscape is particularly challenging to model and requires some kind of approximation. We think our method should be considered as such: a useful approximation that allows incorporating context-dependence in phylogenetic models while remaining analytically and numerically tractable. The quantitative consequences of its undesirable properties are limited, as shown in the supplementary analysis on root placement and on the out-of-equilibrium dynamics. Overall, our results show that the benefits of the method outweigh its disadvantages.

The success of our model in both simulations and experimental validation suggests that generative models with autoregressive architectures are powerful tools for studying the dynamics of protein sequence evolution. By capturing the intricacies of epistatic interactions, our model not only improves the accuracy of ancestral sequence reconstruction but also provides new insights into the underlying evolutionary processes. Future work could explore the application of this model to other protein families and further refine the methodology to enhance its applicability in broader phylogenetic contexts.

In conclusion, the integration of epistasis into evolutionary models represents a necessary and timely advancement for the field. Our generative model provides a more nuanced understanding of protein evolution, paving the way for more accurate reconstructions of ancestral sequences and a deeper exploration of the evolutionary dynamics that shape the diversity of life.

IV. METHODS

A. ArDCA

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

$$P^{AR}(\mathbf{a}) = \prod_{i \in \sigma(L)} p_i(a_i | a_{<i}), \tag{8}$$

where $\sigma(L)$ is a permutation of the L first integers and $a_{<i}$ stands for $a_1, \ldots, a_{i-1}$. 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.

Following [24], we model the conditional probabilities p_i as:

$$p_i(b|a_{<i}) = \frac{1}{Z_i} \exp\left(\sum_{j<i} J_{ij}(b, a_j) + h_i(b) \right), \tag{9}$$

with the i q-dimensional vectors $J_{i\cdot}$ and h_i are learned parameters. It is worth observing that the proposed parametrization of the conditional probabilities p_i enables an efficient parameters learning by likelihood maximization. In the machine learning community, this particular parametrization is known as the soft-max regression [47], which is the generalization to multi-class class regression of the standard logistic regression. The model is normally trained using a multiple sequence alignment of homologous proteins, *i.e.* a protein family, by finding the parameters J and h that maximize the likelihood of the sequences. It was shown in [24] that this specific parametrization captures essential features of the variability of members of a protein family.

By definition, homologous proteins share a joint evolutionary history and cannot be considered as statistically independent. To avoid biases, a reweighting is applied to sequences based on their vicinity to other sequences. This scheme has been showed to substantially increase the performance of such models [10].

B. Approximative nature of the propagator

The autoregressive propagator defined in Eq. 5 is practical because it allows computation of the transition probability between any two sequences and for any time. However, it is only an approximation of the dynamics, as we will show below. The full derivation of these results can be found in Section B 2 of the Supplementary Material .

The propagator that we would ideally like to use would *(i)* be Markovian and time reversible and *(ii)* have the generative model P^{AR} as its stationary distribution. It is possible to derive a transition rate matrix $\mathbf{Q}$ that has these properties (Supplementary Material):

$$Q_{\mathbf{ab}} = \mu \begin{cases} 0 \quad \text{if } \mathbf{a} \text{ and } \mathbf{b} \text{ differ at more than two sites,} \\ p_i(b_i|a_{<i}) \quad \text{if } \mathbf{a} \text{ and } \mathbf{b} \text{ differ only at site } i, \\ \sum_{i=1}^{L}(p_i(a_i|a_{<i}) - 1) \quad \text{if } \mathbf{a} = \mathbf{b}, \end{cases} \tag{10}$$

where $\mathbf{a}$ and $\mathbf{b}$ are any two sequences and μ is a scalar rate. Note that the transition rate here is from sequence to sequence, and $\mathbf{Q}$ is of dimensions $q^L \times q^L$ with $q = 21$ the number of amino-acid states plus the gap symbol. The corresponding transition probability matrix P' would be defined by

$$P'(\mathbf{b}|\mathbf{a}, t) = (e^{t\mathbf{Q}})_{\mathbf{ab}}. \tag{11}$$

The main issue is that because of the dimensions of $\mathbf{Q}$ and because we are incapable of calculating its eigenvectors and eigenvalues, P' cannot be used in practice. There exist workarounds if the goal is to sample from P' [19, 48]. However, they are not applicable to the task of ASR.

Our autoregressive propagator P has two properties that make it an attractive approximation. First,

$$P(\mathbf{b}|\mathbf{a}, t) \xrightarrow[t\to\infty]{} P^{AR}(\mathbf{b}), \tag{12}$$

meaning that it has the right stationary distribution at long times. Informally, we can write $P \simeq P'$ for $t \to \infty$. Secondly, in the case where matrix $\mathbf{H}$ of Eq. 1 has uniform off-diagonal terms equal to μ, the derivative of P with respect to time at $t = 0$ happens to be the $\mathbf{Q}$ of Eq. 10. Therefore,

$$P(\mathbf{b}|\mathbf{a}, t) \underset{t \to 0}{\sim} (\mathbb{1} + t\mathbf{Q})_{\mathbf{ba}}\,, \tag{13}$$

where $\mathbb{1}$ is the identity. This means that for small times, P and P' are equal up to order one in t. Our P is therefore an approximation of the desired P', which becomes exact at small and large times.

Even though we have shown in the text that it gives good results, there are caveats to this approximation. The first is that our propagator does not define a Markovian dynamic, and is also time irreversible. The second is that it does not remain in equilibrium with the generative P^{AR} at intermediate times. However, the approximation can still be useful if deviations from equilibrium are not too large. In the Supplementary Material , we show that sequences generated from $P(\mathbf{b}|\mathbf{a}, t)$ when starting from an equilibrium sample have a lower likelihood than expected, but which remains well under the intrinsic variations of likelihood of a sample of P^{AR}. We therefore conclude that even if our propagator has the undesirable property of going out of equilibrium at intermediate times, these deviations remain quite small.

C. Branch length inference

To perform ancestral sequence reconstruction, not only the topology of the tree but also the branch lengths are needed. When comparing the autoregressive method to IQ-TREE, it would be unfair to use the branch lengths of the real tree since they do not correspond to the dynamical models used in IQ-TREE. For the same reason, using the branch lengths reconstructed by IQ-TREE would also be problematic. We thus perform reconstruction with the autoregressive by taking the tree inferred by IQ-TREE as an input and by re-optimizing its branches.

While optimizing branch lengths of a fixed topology is possible using site independent models, it is more challenging with the autoregressive evolver as it requires an explicit summation over all states at given internal nodes. For this reason, we resort to using a profile model with a shared substitution rate for this task. The algorithm used to re-infer branches is described in section A 3 of the Supplementary Material . In short, it attempts to scale the branches of IQ-TREE's tree using a profile model. Figure S4 shows the good quality of the reconstruction using this technique.

D. 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 evolver 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 evolver model's equilibrium distribution, and evolution is simulated along each branch independently starting from the root. In the case of the autoregressive evolve, the dynamics is the one of Eq. 5. In the case of the Potts model, we use a Markov chain with the Metropolis update rule. 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.

E. Experimental evolution data

To validate the proposed method, we use data from Directed Evolution experiment on Beta-lactamase PSE-1 published in [43]. Beta-lactamase is an enzyme produced by bacteria that provides them resistance to the beta-lactam antibiotic class. Its activity relies on the ability to hydrolize the beta-lactam ring, inhibiting the effect of these antibiotics. In [43], the PSE-1 wild type (WT) undergoes 20 rounds of controlled *in vivo* evolution with an average target mutation rate of approximately 3%-4% per round while being selected for its inhibition effect on ampicillin. The bacterial population in the experiment is approximately 5×10^4, and the fraction of bacteria surviving each selection round is around 1%. At round 20, the last one of the experiment, the library of mutated variants has accumulated an average Hamming distance from WT of 12.9% and an average pairwise distance of 19.8%.

A family of 42k homologous sequences is available from PFAM with code PF13354. For this family, an Hidden Markov Model (HMM) of length 214, built on 66 seed sequences, is contextually available. We aligned the experimental sequences to the family HMM according to the following procedure:

1. the WT sequence (length 266) is aligned to the HMM using HMMER [49];

2. insertion sites in the aligned WT sequence are removed from the aligned WT sequence and from all the other sequences of the experimental library;

3. at positions where the aligned WT has a gap, a gap is also inserted in sequences of the experimental library.

This method ensures that all sequences from the experiment are aligned in the same manner.

It has been noticed in [22] that taking into account the transition possibilities between amino acids allowed by the genetic code is important when describing short term evolutionary dynamics with generative models. In our framework, a natural way to include these is by using the symmetric matrix $\mathbf{H}$ in the decomposition of Eq. 1. Terms of the $\mathbf{H}$ matrix do not affect the equilibrium distribution of the model, which thus remains generative, but influences the short term dynamics. Here, we simply counted the number of possibilities to transition from any amino acid to any other based on the genetic code, and we constructed the corresponding $\mathbf{H}$ matrix. The diagonal matrix remains given by the equilibrium probabilities of amino acids in the context of the sequence, as given by Eq. 6. We found that this substantially improves the results of the autoregressive reconstruction for the experimental evolution data.

F. Reconstruction with IQ-TREE

We run IQ-TREE using the `-asr` flag to generate states at internal nodes of the tree. By default, IQ-TREE reconstructs the maximum a posteriori (MAP) sequence at internal nodes [37]. It also generates a “state” file containing the posterior probabilities of amino acids at each internal node that we use to sample internal sequences.

On simulated data, we ran IQ-TREE using the model finder routine to select the evolutionary model [35]. For each simulated data set, *i.e.* a protein family and an evolver, we ran the model finder on a reduced set of trees. Since running the model finder is time consuming, we used these test runs to select a best model for each family/evolver and performed more extensive simulations using this one. The selected best models are reported in Table I.

The model most frequently found was based on the PMB matrix [36], with different options for rates depending on the family and evolver, *e.g.* `+G4`, `+I+G4` or `+R4` On the directed evolution data, the two most frequently found models were JTT [3] and a between patient

			IQ-TREE model	
Family		Alignment length	ArDCA	Potts
PF00014	Trypsin inhibitor Kunitz domain	53	`PMB+R3`	`PMB+I+G4`
PF00072	Response regulator receiver domain	112	`PMB+I+G4`	`PMB+I+G4`
PF00076	RNA recognition motif	70	`PMB+R3`	`PMB+I+G4`
PF00595	PDZ domain	82	`PMB+R3`	`PMB+R5`
PF13354	Beta-lactamase enzyme	214	`JTT+G4`	

TABLE I. Protein families used in this work. The last two columns give the best hit models found by IQ-TREE, for the two different evolvers (autoregressive and Potts).

HIV model [50]. Since the latter is clearly unrelated to the protein that is considered here, we used the `JTT+G4` model for reconstruction.

In addition, we used IQ-TREE to perform reconstruction with profile mixture models, using the `+C60` flag. Experiments with less complex models, *e.g.* `+C10` and `+C20`, did not lead to an improvement as large as the `+C60` flag: for this reason, we only show results for the latter. For each family, reconstruction was then performed using the model in Table I and appending the profile flag (*e.g.* legend of Figure S7).

G. Code & data availability

The code used in this work is accessible at the following links:

- the implementation of the reconstruction algorithm described here is available at `https://github.com/PierreBarrat/AncestralSequenceReconstruction.jl`
- the code used in simulations and data analysis is available at `https://github.com/PierreBarrat/AutoRegressiveASR`.

Acknowledgments

We thank Juan Rodriguez-Rivas for useful discussions.

[1] M.O. Dayhoff, R.M. Schwartz, and B.C. Orcutt. A model of evolutionary change in proteins. *Atlas of Protein Sequence and Structure.*, 1978.

[2] S Henikoff and J G Henikoff. Amino acid substitution matrices from protein blocks. *Proceedings of the National Academy of Sciences*, 89(22):10915–10919, November 1992. doi: 10.1073/pnas.89.22.10915.

[3] David T. Jones, William R. Taylor, and Janet M. Thornton. The rapid generation of mutation data matrices from protein sequences. *Bioinformatics*, 8(3):275–282, June 1992. ISSN 1367-4803. doi:10.1093/bioinformatics/8.3.275.

[4] 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.

[5] Merridee A Wouters, Ke Liu, Peter Riek, and Ahsan Husain. A Despecialization Step Underlying Evolution of a Family of Serine Proteases. *Molecular Cell*, 12(2):343–354, August 2003. ISSN 1097-2765. doi:10.1016/S1097-2765(03)00308-3.

[6] Karin Voordeckers, Chris A. Brown, Kevin Vanneste, Elisa van der Zande, Arnout Voet, Steven Maere, and Kevin J. Verstrepen. Reconstruction of Ancestral Metabolic Enzymes Reveals Molecular Mechanisms Underlying Evolutionary Innovation through Gene Duplication. *PLOS Biology*, 10(12):e1001446, December 2012. ISSN 1545-7885. doi:10.1371/journal.pbio.1001446.

[7] Georg K. A. Hochberg and Joseph W. Thornton. Reconstructing Ancient Proteins to Understand the Causes of Structure and Function. *Annual Review of Biophysics*, 46(1):247–269, 2017. doi:10.1146/annurev-biophys-070816-033631.

[8] Satoshi Akanuma, Yoshiki Nakajima, Shin-ichi Yokobori, Mitsuo Kimura, Naoki Nemoto, Tomoko Mase, Ken-ichi Miyazono, Masaru Tanokura, and Akihiko Yamagishi. Experimental evidence for the thermophilicity of ancestral life. *Proceedings of the National Academy of Sciences*, 110(27):11067–11072, July 2013. doi:10.1073/pnas.1308215110.

[9] J. K. Hobbs, C. Shepherd, D. J. Saul, N. J. Demetras, S. Haaning, C. R. Monk, R. M. Daniel, and V. L. Arcus. On the Origin and Evolution of Thermophily: Reconstruction of Functional Precambrian Enzymes from Ancestors of Bacillus. *Molecular Biology and Evolution*, 29(2): 825–835, February 2012. ISSN 0737-4038, 1537-1719. doi:10.1093/molbev/msr253.

[10] 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.

[11] Jérôme Tubiana, Simona Cocco, and Rémi Monasson. Learning protein constitutive motifs from sequence data. *eLife*, 8:e39397, March 2019. ISSN 2050-084X. doi:10.7554/eLife.39397.

[12] Roshan M. Rao, Jason Liu, Robert Verkuil, Joshua Meier, John Canny, Pieter Abbeel, Tom Sercu, and Alexander Rives. MSA Transformer. In *Proceedings of the 38th International Conference on Machine Learning*, pages 8844–8856. PMLR, July 2021.

[13] Noelia Ferruz, Steffen Schmidt, and Birte Höcker. ProtGPT2 is a deep unsupervised language model for protein design. *Nature Communications*, 13(1):4348, July 2022. ISSN 2041-1723. doi:10.1038/s41467-022-32007-7.

[14] Pengfei Tian, John M. Louis, James L. Baber, Annie Aniana, and Robert B. Best. Co-evolutionary fitness landscapes for sequence design. *Angewandte Chemie (International ed. in English)*, 57(20):5674–5678, May 2018. ISSN 1433-7851. doi:10.1002/anie.201713220.

[15] William P. Russ, Matteo Figliuzzi, Christian Stocker, Pierre Barrat-Charlaix, Michael Socolich, Peter Kast, Donald Hilvert, Remi Monasson, Simona Cocco, Martin Weigt, and Rama Ranganathan. An evolution-based model for designing chorismate mutase enzymes. *Science*, 369(6502):440–445, July 2020. ISSN 0036-8075, 1095-9203. doi:10.1126/science.aba3304.

[16] 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.

[17] 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.

[18] 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.

[19] Douglas M. Robinson, David T. Jones, Hirohisa Kishino, Nick Goldman, and Jeffrey L. Thorne. Protein Evolution with Dependence Among Codons Due to Tertiary Structure.

Molecular Biology and Evolution, 20(10):1692–1704, October 2003. ISSN 0737-4038. doi:10.1093/molbev/msg184.

[20] Nicolas Rodrigue, Hervé Philippe, and Nicolas Lartillot. Assessing Site-Interdependent Phylogenetic Models of Sequence Evolution. *Molecular Biology and Evolution*, 23(9):1762–1775, September 2006. ISSN 0737-4038. doi:10.1093/molbev/msl041.

[21] Xavier Meyer, Linda Dib, Daniele Silvestro, and Nicolas Salamin. Simultaneous Bayesian inference of phylogeny and molecular coevolution. *Proceedings of the National Academy of Sciences*, 116(11):5027–5036, March 2019. doi:10.1073/pnas.1813836116.

[22] Matteo Bisardi, Juan Rodriguez-Rivas, Francesco Zamponi, and Martin Weigt. Modeling Sequence-Space Exploration and Emergence of Epistatic Signals in Protein Evolution. *Molecular Biology and Evolution*, page msab321, November 2021. ISSN 1537-1719. doi:10.1093/molbev/msab321.

[23] Sophia Alvarez, Charisse M. Nartey, Nicholas Mercado, Jose Alberto de la Paz, Tea Huseinbegovic, and Faruck Morcos. In vivo functional phenotypes from a computational epistatic model of evolution. *Proceedings of the National Academy of Sciences*, 121(6):e2308895121, February 2024. doi:10.1073/pnas.2308895121.

[24] 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.

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

[26] 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.

[27] 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.

[28] 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.

[29] Ali Madani, Ben Krause, Eric R. Greene, Subu Subramanian, Benjamin P. Mohr, James M. Holton, Jose Luis Olmos, Caiming Xiong, Zachary Z. Sun, Richard Socher, James S. Fraser, and Nikhil Naik. Large language models generate functional protein sequences across diverse families. *Nature Biotechnology*, 41(8):1099–1106, August 2023. ISSN 1546-1696. doi:10.1038/s41587-022-01618-2.

[30] 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.

[31] 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.

[32] 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.

[33] 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.

[34] 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.

[35] Subha Kalyaanamoorthy, Bui Quang Minh, Thomas K. F. Wong, Arndt von Haeseler, and Lars S. Jermiin. ModelFinder: Fast model selection for accurate phylogenetic estimates. *Nature Methods*, 14(6):587–589, June 2017. ISSN 1548-7105. doi:10.1038/nmeth.4285.

[36] Shalini Veerassamy, Andrew Smith, and Elisabeth R. M. Tillier. A Transition Probability Model for Amino Acid Substitutions from Blocks. *Journal of Computational Biology*, 10(6): 997–1010, December 2003. doi:10.1089/106652703322756195.

[37] Z Yang, S Kumar, and M Nei. A new method of inference of ancestral nucleotide and amino acid sequences. *Genetics*, 141(4):1641–1650, December 1995. ISSN 1943-2631. doi: 10.1093/genetics/141.4.1641.

[38] Le Si Quang, Olivier Gascuel, and Nicolas Lartillot. Empirical profile mixture models for phylogenetic reconstruction. *Bioinformatics*, 24(20):2317–2323, October 2008. ISSN 1367-4803.

doi:10.1093/bioinformatics/btn445.

[39] Yeonwoo Park, Brian P. H. Metzger, and Joseph W. Thornton. Epistatic drift causes gradual decay of predictability in protein evolution. *Science*, 376(6595):823–830, May 2022. doi:10.1126/science.abn6895.

[40] Leonardo Di Bari, Matteo Bisardi, Sabrina Cotogno, Martin Weigt, and Francesco Zamponi. Emergent time scales of epistasis in protein evolution. *Proceedings of the National Academy of Sciences*, 121(40):e2406807121, October 2024. doi:10.1073/pnas.2406807121.

[41] 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.

[42] Paul D. Williams, David D. Pollock, Benjamin P. Blackburne, and Richard A. Goldstein. Assessing the Accuracy of Ancestral Protein Reconstruction Methods. *PLOS Computational Biology*, 2(6):e69, June 2006. ISSN 1553-7358. doi:10.1371/journal.pcbi.0020069.

[43] Michael A. Stiffler, Frank J. Poelwijk, Kelly P. Brock, Richard R. Stein, Adam Riesselman, Joan Teyra, Sachdev S. Sidhu, Debora S. Marks, Nicholas P. Gauthier, and Chris Sander. Protein Structure from Experimental Evolution. *Cell Systems*, 10(1):15–24.e5, January 2020. ISSN 24054712. doi:10.1016/j.cels.2019.11.008.

[44] Geeta N. Eick, Jamie T. Bridgham, Douglas P. Anderson, Michael J. Harms, and Joseph W. Thornton. Robustness of Reconstructed Ancestral Protein Functions to Statistical Uncertainty. *Molecular Biology and Evolution*, 34(2):247–261, February 2017. ISSN 0737-4038. doi:10.1093/molbev/msw223.

[45] Lucas C Wheeler, Shion A Lim, Susan Marqusee, and Michael J Harms. The thermostability and specificity of ancient proteins. *Current opinion in structural biology*, 38:37–43, June 2016. ISSN 0959-440X. doi:10.1016/j.sbi.2016.05.015.

[46] Christoph Feinauer, Marcin J. Skwark, Andrea Pagnani, and Erik Aurell. Improving Contact Prediction along Three Dimensions. *PLOS Computational Biology*, 10(10):e1003847, October 2014. ISSN 1553-7358. doi:10.1371/journal.pcbi.1003847.

[47] Trevor Hastie. The elements of statistical learning: data mining, inference, and prediction, 2009.

[48] Chris A. Nasrallah, David H. Mathews, and John P. Huelsenbeck. Quantifying the Impact of Dependent Evolution among Sites in Phylogenetic Inference. *Systematic Biology*, 60(1):60–73, January 2011. ISSN 1063-5157. doi:10.1093/sysbio/syq074.

[49] Sean Eddy. Hmmer: biosequence analysis using profile hidden markov models. `hmmer.org`, 2023. version 3.4.

[50] David C. Nickle, Laura Heath, Mark A. Jensen, Peter B. Gilbert, James I. Mullins, and Sergei L. Kosakovsky Pond. HIV-Specific Probabilistic Models of Protein Evolution. *PLOS ONE*, 2 (6):e503, June 2007. ISSN 1932-6203. doi:10.1371/journal.pone.0000503.

[51] Joseph Felsenstein. Evolutionary trees from DNA sequences: A maximum likelihood approach. *Journal of Molecular Evolution*, 17(6):368–376, November 1981. ISSN 0022-2844, 1432-1432. doi:10.1007/BF01734359.

[52] Thomas Harvey Rowan. *Functional stability analysis of numerical algorithms.* PhD thesis, Department of Computer Science, University of Texas at Austin, Austin, TX, 1990.

[53] Steven G. Johnson. The NLopt nonlinear-optimization package. `https://github.com/stevengj/nlopt`, 2007.

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

Matteo De Leonardis, Andrea Pagnani, Pierre Barrat-Charlaix

Appendix A: Reconstruction algorithm

The classical pruning algorithm described in [51] allows one to compute, for each sequence position, the likelihood of the data at the leaves of a tree given an amino acid state at its root. It is then possible to infer marginal ancestral state by iteratively re-rooting the tree at all internal nodes and *e.g.* maximizing the corresponding posterior distribution of the root state. This technique is only possible if the model of evolution is reversible, in which case the position of the root is purely conventional.

Because the autoregressive model of evolution is irreversible, we cannot change the root of the tree and need to adapt the above algorithm. Our method is essentially an adaptation of the algorithm described in [26]. We first describe a general version of the algorithm, which could be used for any evolutionary model. We then explain how we apply it to our specific autoregressive evolver

1. General description of the algorithm

Our aim is to obtain, for each sequence position, a *marginal* reconstruction at each internal node. Given a node n in a rooted tree $\mathcal{T}$, calling x_n its amino acid state and $\mathcal{D}$ the amino acid states at the leaves, we want to compute the probability

$$\mathcal{L}_n(x) \overset{\text{def}}{\equiv} P(\mathcal{D}|\mathcal{T}, x_n = x), \tag{A1}$$

that is the probability of the data knowing that n is in state x. We will see below that our way to compute $\mathcal{L}_n$ involves a prior distribution of internal states coming from the root node, and $\mathcal{L}_n$ is thus not strictly speaking a likelihood. However, we will abusively refer to it as likelihood in what follows. We define the maximum a posteriori (MAP) reconstruction as

$\arg\max_x \mathcal{L}_n(x)$, and a "posterior sampling" reconstruction as a sample from a normalized $\mathcal{L}_n(x)$. Note that since we consider a known and fixed tree and to lighten notation, we ignore the dependence on $\mathcal{T}$ in the following equations.

To compute $\mathcal{L}_n$, we introduce the following notation: let a be the ancestral and $\mathcal{C}_n$ the children nodes of n. Then, let $T_n(y, x)$ be the transition probability from amino acid state y to x for the branch $a \to n$. Importantly, T_n is a "directed" quantity: it describes the evolution from a to n. This is irrelevant for reversible models, but is important in the autoregressive case. Finally, we call q the number of different amino acid states that a site can be in: expressions of the form $\sum_{y=1}^{q}$ refer to sum over all amino acid states. In the autoregressive model, $q = 21$ for the 20 natural amino acids and the gap symbol.

First, we use the fact that if n is known to be in some state x, leaf-data on either sides of the branch $a \to n$ are independent. We call $\mathcal{D}_{below}$ the data at the leaves of the clade below n, and $\mathcal{D}_{above}$ the data at the leaves on the other side of the $a \to n$ branch. We can then write

$$\mathcal{L}_n(x) = P(\mathcal{D}_{below}|x_n = x)P(\mathcal{D}_{above}|x_n = x). \tag{A2}$$

To simplify notation, we define the following quantities:

$$\begin{aligned} v_n(x) &= P(\mathcal{D}_{below}|x_n = x) \\ u_n(y) &= P(\mathcal{D}_{above}|y_a = y) \text{ where } a = \text{ancestor}(n) \end{aligned} \tag{A3}$$

Note that $u_n(y)$ stands for the likelihood of $\mathcal{D}_{above}$ given that the *ancestor* a of n is in a given state y. This allows us to simplify Eq. A2 to obtain

$$\mathcal{L}_n(x) = v_n(x) \cdot \sum_{y=1}^{q} u_n(y) T_n(y, x). \tag{A4}$$

In other words, we split the likelihood into a "below" term v depending on the state x of n, and an "above" term u depending on the state y of the ancestor a. The two are linked by the transition probability $T_n(y, x)$ along the branch $a \to n$. Summing over all states y then yields $\mathcal{L}_n(x)$.

To compute $v_n(x)$ and $u_n(x)$, we use the following set of recursive relations:

$$\begin{aligned} v_n(x) &= \prod_{c\in\mathcal{C}_n} \sum_{y=1}^{q} T_c(x,y)v_c(y) \\ &= \prod_{c\in\mathcal{C}_n} \left(\mathbf{T}_c\mathbf{v}_c\right)_x , \\ u_n(x) &= \sum_{y=1}^{q} u_a(y)T_a(y,x) \cdot \prod_{c\in\mathcal{C}_a\backslash n} \sum_{y=1}^{q} T_c(x,y)v_c(y) \\ &= \left(\mathbf{u}_a^T\mathbf{T}_a\right)_x \cdot \prod_{c\in\mathcal{C}_a\backslash n} \left(\mathbf{T}_c\mathbf{v}_c\right)_x , \end{aligned} \tag{A5}$$

where we used bold-font symbols – *e.g.* $\mathbf{v}_n$ or $\mathbf{T}_n$ – to represent vector $[v_n(1),\ldots,v_n(q)]$ and the $q\times q$ transition probability matrix $T(x,y)$.

The expression for $v_n(x)$ essentially says that the likelihood of data at the tips of the clade below n is a product of likelihoods coming from subclades of the children of n, each weighted by the transition matrix T_c of branch $n \to c$. On the other hand, the expression for $u_n(x)$ takes into account information coming from above the ancestor a – the term $\mathbf{u}_a^T\mathbf{T}_a$ – and from the children of a at the exception of n – the term $\prod_{c\in\mathcal{C}_a\backslash n}\mathbf{T}_c\mathbf{v}_c$. It is clear that fixing n, this set of recursive relations involves all leaves, and also all branches at the exception of the $a \to n$ one. This last branch is taken into account when combining $\mathbf{v}_n$ and $\mathbf{u}_n$ in Eq. A4. Finally, the set of relations is closed by the following conditions:

- if n is a leaf, $v_n(x) = \delta_{x,x_n}$ where δ is the Kronecker function and x_n the observed state at n.

- if n is the root, $u_n(x) = \pi(x)$ with $\pi = [\pi(1)\ldots\pi(q)]$ being the equilibrium frequencies of amino acids according to the sequence evolution model.

Computing $\mathcal{L}_n(x)$ is done by applying the following steps.

- Traverse the tree in post-order and compute $\mathbf{v}_n$ for each node encountered. Since the traversal is post-order, $\mathbf{v}_c$ for $c\in\mathcal{C}_n$ is always available.

- Traverse the tree in pre-order and compute $\mathbf{u}_n$. Since the traversal is pre-order, $\mathbf{u}_a$ for $a = \text{ancestor}(n)$ is always known and $\mathbf{v}_n$ is known from the previous step.

- For each node n, compute $\mathcal{L}_n$ applying Eq. A4.

2. Application to the autoregressive model

Our autoregressive evolution model has the following unusual properties: *(i)* evolution depends on the relevant context, *e.g.* sites $1, \ldots, i-1$ for position i; *(ii)* as a corollary, the transition rate matrix Q defining evolution depends on the sequence *towards* which evolution is happening, as in Eq. 5; *(iii)* evolution is not reversible, meaning that the orientation of the branches of the tree matters.

We show below that the algorithm described above adapts without problems to these particularies. Reconstruction with the autoregressive model proceeds iteratively from the first to the last sequence position. Assume that we are reconstructing internal states at position i, and that positions $1, \ldots, i-1$ are already reconstructed for all internal nodes. We then apply the following steps.

- For all nodes n, compute the profile $\pi_{n,i}(x) = p_i(x|x_n^1, \ldots, x_n^{i-1})$, where p_i is a parameter of the autoregressive model defined in Eq. 4 and $x_n^1, \ldots, x_n^{i-1}$ is the context at node n.

- For all nodes n and given the equilibrium frequencies $\pi_{n,i}$ at this node and position, compute the transition probability matrix $\mathbf{T}_n$ for the branch ancestor(n) → n. This matrix is defined as

  $$\mathbf{T}_n = e^{t_n Q},$$

  with Q defined in Eq. 1 and t_n the length of the branch.

- When all transition matrices and node-specific equilibrium frequencies are known, apply the algorithm of the previous section to reconstruct state x_n^i at all nodes n.

3. Branch length inference

To reconstruct the branch length, we start from expressions of the likelihood Eq. A1 & Eq. A4. We first note that this expression is specific to a given sequence position $i \in \{1 \ldots L\}$, and thus rename quantities such as $\mathcal{L}_n$ to $\mathcal{L}_n^i$. Then, by summing over all possible states of internal node n, we obtain an expression for the probability of the data $\mathcal{D}_i$ at position i

knowing the tree:

$$\begin{aligned} P(\mathcal{D}_i|\mathcal{T}) &= \sum_{x=1}^{q} P(\mathcal{D}_i|\mathcal{T}, x_n = x) \\ &= \sum_{x=1}^{q} \mathcal{L}_n^i(x) \\ &= \sum_{x,y} u_n^i(y) T_n^i(y,x) v_n^i(x) \\ &= \left\langle \mathbf{u}_n^i | \mathbf{T}_n^i | \mathbf{v}_n^i \right\rangle . \end{aligned} \tag{A6}$$

Finally, the likelihood of the all the leaf sequences is obtained by multiplying over sequence positions:

$$P(\mathcal{D}|\mathcal{T}) = \prod_{i=1}^{L} \left\langle \mathbf{u}_n^i | \mathbf{T}_n^i | \mathbf{v}_n^i \right\rangle . \tag{A7}$$

Starting from this last expression, we use two techniques to infer MAP branch lengths. In practice, due to computational time considerations, we use the second one (branch scaling). Importantly, since Eq. A7 involves a product over all sequence positions, it is not possible to apply it to the autoregressive evolution model. Indeed, the only way to compute *e.g.* $\mathbf{v}_n^i$ for the autoregressive model is to have *fixed* the internal node states at positions $1, \ldots, i-1$, making $\mathbf{v}_n^1, \ldots, \mathbf{v}_n^{i-1}$ irrelevant. To avoid this difficulty, we apply the two methods below using a profile model with site specific frequencies instead of the autoregressive one.

a. Single branch length optimization

Expression Eq. A7 is practical because it allows one to compute the probability of the data as an explicit function of the transition matrices $\mathbf{T}_n^i$ of branch above node n ($\mathbf{v}_n$ and $\mathbf{u}_n$ do not depend on the branch above n). Note that since $\mathbf{T}_n^i = e^{t_n \mathbf{Q}_n^i}$, the dependence on the branch length t_n is also explicit. We use this to find the t_n that maximizes $P(\mathcal{D}|\mathcal{T})$:

$$t_n = \arg\max \sum_{i=1}^{L} \log \left\langle \mathbf{u}_n^i | e^{t_n \mathbf{Q}_n^i} | \mathbf{v}_n^i \right\rangle , \tag{A8}$$

where we take the logarithm for computational reasons.

It is straightforward to obtain an analytical expression for the gradient of the above expression with respect to t_n, making optimization reasonably fast. We then optimize all branch lengths starting from the IQ-TREE inferred tree and cycling over the following steps until convergence is reached:

- Compute messages $\mathbf{u}_n$ and $\mathbf{v}_n$ for all internal nodes n.
- Pick a non-root internal node n, and optimize its branch length t_n.

This algorithm is guaranteed to converge since the likelihood increases at each step. However, it is also computationally expensive: optimizing a single branch n requires computing the quantities $\mathbf{u}_n$ and $\mathbf{v}_n$, which in turn requires using the recursive relations in Eq. A5 over the whole tree. Since we assess the quality of ancestral reconstruction by applying it to many trees, we use in practice the quicker method described below

b. Scaling branch lengths

In order to make the branch length inference faster, we adopt a scaling strategy. We start from the tree inferred by IQ-TREE, using the settings described in the Methods section: for each node n, let t_n^0 be the branch length inferred by IQ-TREE. We construct the scaled tree $\mathbf{T}_\mu$ by multiplying the branches by a factor μ: the branch above any node n is $t_n = \mu t_n^0$. We then find the scaling factor μ that maximizes the likelihood:

$$\mu^\star = \arg\max_\mu P(\mathcal{D}|\mathcal{T}_\mu), \tag{A9}$$

where the right-hand side can be numerically evaluated using the expression A7 at any internal node n (in our case, we use the root node). In contrast with the individual branch optimization, it is not possible to write the gradient of the likelihood with respect to μ, and we must use a derivative free optimization technique [52, 53]. However, since only one parameter must be optimized, this technique turns out to be much quicker for the trees of a hundred leaves that we use in the main text. The results can be seen in Figure 4.

Appendix B: Autoregressive evolution model

1. Simplified expression for a homogeneous H

For each site i, the main difference between our model and a traiditional GTR is that the equilibrium frequencies of the Markov chain are computed using the context at the previous sites $1, \dots, i-1$. Considering Eq. 1 and Eq. 6, this means that the diagonal matrix is determined using the generative model. On the other hand, the symetric matrix $\mathbf{H}$ can be given any value without changing the long term generative properties of the dynamical model, *i.e.* Eq. 7. Here, we show that if the transitions defined by $\mathbf{H}$ are uniform, *i.e.* $H_{ab} = \mu$ for any $a \neq b$, the propagator takes a simplified form:

$$\begin{aligned} q_i(b_i|a_i, b_{<i}, t) &= e^{-\mu t}\delta_{a_i,b_i} + (1 - e^{-\mu t})p_i(b_i|b_{<i}), \\ P(\mathbf{b}|\mathbf{a}, t) &= \prod_{i=1}^{L} q_i(b_i|a_i, b_{<i}, t). \end{aligned} \tag{B1}$$

The interpretation of the site propagator $q_i(b_i|a_i, b_{<i}, t)$ is straightforward: if no mutation occurs with probability $e^{-\mu t}$, site i remains in its original state a_i; otherwise, with probability $(1 - e^{-\mu t})$, it is resampled using the equilibrium probability given by the generative model and the context of the sequence $p_i(b_i|b_{<i})$. Note that the assumption of a scalar matrix is reasonable if one wishes to ignore the different transition rates between amino-acids. Interestingly, this form is analogous to the F81 model of DNA evolution [51], which also parametrizes the transition rate matrix $\mathbf{Q}$ using only the long term equilibrium frequencies $(\pi_A, \pi_C, \pi_G, \pi_T)$.

To lighten notation, we drop the explicit dependence on the position i and the sequence context $b_{<i}$ by defining $p_b = p_i(b_i|b_{<i})$. We will then compute the n eigenvectors and eigenvalues of $\mathbf{Q}$, where $n = 21$ for the amino acids and gap symbol. First, note that for the continuous time Markov chain to be well defined, we need the rows of $\mathbf{Q}$ to sum to 0. We thus have the following expression for the elements of $\mathbf{Q}$:

$$\mathbf{Q} = \mu \begin{pmatrix} p_1 - 1 & p_2 & \dots & p_n \\ p_1 & p_2 - 1 & \dots & p_n \\ \dots & \dots & \dots & \dots \\ p_1 & p_2 & \dots & p_n - 1 \end{pmatrix} = \mu \left(\mathbf{1}\mathbf{p}^\dagger - I \right)$$

where $\mathbf{1}$ is the n-dimensional vector whose entries are all 1s, I is the identity matrix, and $\mathbf{p} = (p_1, \dots, p_q)$. In particular we note that the outer product $\mathbf{1}\mathbf{p}^\dagger$ is a rank-one projector onto the state $\mathbf{p}$, and thus it has a left eigenvector equal to $\mathbf{p}^\dagger$ (associated to the eigenvalue 1) and $n-1$ eigenvalues equal to 0. Indeed:

$$\mathbf{p}^\dagger \mathbf{Q} = \mu \mathbf{p}^\dagger \left(\mathbf{1}\mathbf{p}^\dagger - I \right) = 0$$

As $p(b|a,t) = [\exp(\mathbf{Q}t)]_{ab}$, we need to compute the exponential of $\mathbf{Q}$. To do so, we first note that:

$$\begin{aligned}
\mathbf{Q}^2 &= \mu^2 \left(\mathbf{1}\mathbf{p}^\dagger - I\right)\left(\mathbf{1}\mathbf{p}^\dagger - I\right) \\
&= \mu^2 \left(\mathbf{1}\underbrace{\mathbf{p}^\dagger\mathbf{1}}_{=1}\mathbf{p}^\dagger - 2\mathbf{1}\mathbf{p}^\dagger + I\right) \\
&= \mu^2\left(-\mathbf{1}\mathbf{p}^\dagger + I\right) \\
&= -\mu\mathbf{Q}
\end{aligned}$$

which in turn implies that $\mathbf{Q}^k = (-1)^{k-1}\mu^{k-1}\mathbf{Q}$. From this simple relation for all integer powers of $\mathbf{Q}$ we can explicitly compute the exponential of the $\mathbf{Q}$ matrix from following power series:

$$\begin{aligned}
e^{t\mathbf{Q}} &= \sum_{k=0}^{\infty} \frac{t^k\mathbf{Q}^k}{k!} \\
&= I + \sum_{k=1}^{\infty} \frac{t^k\mathbf{Q}^k}{k!} \\
&= I - \frac{1}{\mu}\mathbf{Q}\sum_{k=1}^{\infty}\frac{t^k\mu^k\left(-1\right)^k}{k!} \\
&= I - \frac{1}{\mu}\mathbf{Q}\left(e^{-\mu t} - 1\right) \\
&= Ie^{-\mu t} + \mathbf{1}\mathbf{p}^\dagger\left(1 - e^{-\mu t}\right)
\end{aligned}$$

We thus obtain the desired result:

$$q(b|a,t) = e^{-\mu t}\delta_{ab} + (1 - e^{-\mu t})p_b. \tag{B2}$$

2. Non-Markovianity and approximative nature of the propagator

The propagator of the main text is useful because it allows calculation of the transition *probability* between any two sequences and for any time. However, it is only an approximation, in a way that we show below. The structure of the next four paragraphs is as follows.

a. Our propagator does not respect global balance. The consequences are that *(i)* our dynamics is not Markovian and *(ii)* the generative model distribution P^{AR} is not stationary.

b. A consequence of the first point is that our propagator is irreversible.

c. Our propagator can be seen as an approximation of a continuous Markovian dynamic with P^{AR} as a stationary distribution. The approximation is exact at large times and at order one for small times.

d. The deviations between our approximate dynamics and the "correct" ones remain small for intermediate times.

The calculations below are valid for the simplified expression of the propagator in Eq. B1, that is for a uniform $\mathbf{H}$ in Eq. 1 of the main text. However, there is little doubt that the results are also valid for a more general $\mathbf{H}$. To simplify notation, we also consider the case $\mu = 1$: the case of a generic μ is easily re-derived.

a. Non-markovianity

A Markov chain that has a stationary distribution $\pi(\mathbf{a})$ and a transition probability matrix $q(\mathbf{b}|\mathbf{a})$ will verify *global balance*:

$$\pi(\mathbf{a}) = \sum_{\mathbf{b}} \pi(\mathbf{b}) q(\mathbf{a}|\mathbf{b}). \tag{B3}$$

Here, we design a small toy example to show that our propagator does not in general satisfy global balance.

Consider a sequence of length $L = 2$ where each position can be in two states, 0 or 1. Assume that the "fitness landscape" of this protein is such that sequences $\{0, 0\}$ and $\{1, 1\}$ are equally functional, while $\{0, 1\}$ and $\{1, 0\}$ are not functional. Since an organism possessing sequences $\{0, 1\}$ or $\{1, 0\}$ would suffer a fitness loss, they would appear less frequently in nature. The sequence alignment of this "family" could then have the following statistics:

$$P(\{0, 0\}) = P(\{1, 1\}) = \frac{1}{2}(1 - \varepsilon) \quad \text{and} \quad P(\{0, 1\}) = P(\{1, 0\}) = \frac{\varepsilon}{2}, \tag{B4}$$

with $\varepsilon \ll 1$. A well trained autoregressive model would consequently have the following properties:

$$\begin{aligned} &p_1(0) = p_1(1) = \frac{1}{2}, \\ &p_2(0|0) = p_2(1|1) = 1 - \varepsilon \quad \text{and} \quad p_2(0|1) = p_2(1|0) = \varepsilon. \end{aligned}$$

Indeed, state 0 or 1 are equally likely at position one, and given state a at position one the state at position two must also be a with probability $1-\varepsilon$. The corresponding autoregressive distribution P^{AR} is exactly equal to the natural one in Eq. B4.

We now set out to show that global balance does not hold in this case. Consider sequence $\{1,0\}$, which has probability $\varepsilon/2$. Then for any given time t we expect

$$P^{AR}(\{1,0\}) = \frac{\varepsilon}{2} = \sum_{\mathbf{a}} P^{AR}(\mathbf{a}) P(\{1,0\}|\mathbf{a},t),$$

where P is the propagator of Eq. B1.

To show the inequality, it is enough to consider one term of the sum on the right-hand side: the one with $\mathbf{a} = \{0,0\}$. Indeed, using Eq. B1 we immediately obtain

$$\begin{aligned} P^{AR}(\{0,0\}) P\left(\{1,0\}|\{0,0\},t\right) &= \frac{1-\varepsilon}{2} \cdot (1-e^{-t}) \frac{1}{2} \cdot \left(e^{-t} + (1-e^{-t})\varepsilon\right) \\ &\sim \mathcal{O}(1). \end{aligned}$$

Since at least one term in the sum is of order one and the terms are all positive, the sum itself is $\mathcal{O}(1)$. Since the left-hand side has order ε and ε can be chosen arbitrarily small, global balance cannot hold. Therefore, the target distribution P^{AR} of the autoregressive model, defined in Eq. 4 of the main text, is *not* the equilibrium of the propagator $P(\mathbf{b}|\mathbf{a},t)$ defined in Eq. 5.

Another important consequence is that the process is not Markovian. We know from the main text that at long times, $P(\mathbf{b}|\mathbf{a},t)$ converges to $P^{AR}(\mathbf{b})$. Injecting this in Eq. B3, we see that that global balance holds for $t \to \infty$. If $P(\mathbf{b}|\mathbf{a},t)$ was a Markov process, this would mean that P^{AR} is its stationary distribution and that global balance should hold at all times t. As the example above shows, this is not the case. Therefore, our process is not Markovian.

b. Irreversibility

For a stochastic model with stationary distribution π and transition probability $q(\mathbf{b}|\mathbf{a},t)$, time reversibility is equivalent to respecting *detailed balance*: for any two sequences $\mathbf{a}$ and $\mathbf{b}$ and any time t, one should have

$$\pi(\mathbf{a}) q(\mathbf{b}|\mathbf{a},t) = \pi(\mathbf{b}) q(\mathbf{a}|\mathbf{b},t). \tag{B5}$$

Detailed balance implies global balance, as summing over either $\mathbf{a}$ or $\mathbf{b}$ in Eq. B5 directly gives Eq. B3. As the previous section showed, the autoregressive propagator does not satisfy global balance. Therefore, it cannot be time reversible. We stress that the cause of irreversibility here is not epistasis in itself, but rather the structure of the autoregressive propagator. In fact, it is perfectly possible to design dynamical epistatic models that are time reversible, either with discrete time [23] or with continuous time (Section B 2 c).

Note that irreversibility only happens at the sequence level, and not for individual positions. Indeed for each position i and given a sequence context, the autoregressive model has the same structure as classical sequence evolution models. In particular, it is time reversible: given a context and any two amino acid states a_i and b_i, there is no objective way of determining whether a_i evolved in to b_i or the reverse.

c. *Instantaneous transition rates*

If the autoregressive propagator was Markovian, it would be defined by its transition rate matrix $\mathbf{Q}$:

$$P(\mathbf{b}|\mathbf{a}, t) \sim \left(e^{t\mathbf{Q}}\right)_{\mathbf{ab}}, \tag{B6}$$

where we use the $\sim$ symbol to remind that the above equation does not actually hold. Note that the $\mathbf{Q}$ here is a sequence-to-sequence transition rate matrix of dimension $q^L \times q^L$ where $q = 21$ is the number of amino-acid plus the gap symbol. It is different from the position specific Q^i of the main text.

As we have seen, the process is not Markovian. However, we can still calculate the instantaneous transition rate by defining

$$Q_{\mathbf{ab}} \overset{\text{def}}{\equiv} \frac{\mathrm{d}P(\mathbf{b}|\mathbf{a}, t)}{\mathrm{d}t}\Big|_{t=0}. \tag{B7}$$

Doing so in the case where $\mathbf{H}$ is uniform and using Eq. B1 for the transition probabilities yields the following $\mathbf{Q}$:

$$Q_{\mathbf{ab}} = \begin{cases} 0 \quad \text{if } \mathbf{a} \text{ and } \mathbf{b} \text{ differ at more than two sites,} \\ p_i(b_i|a_{<i}) \quad \text{if } \mathbf{a} \text{ and } \mathbf{b} \text{ differ only at site } i, \\ \sum_{i=1}^{L}(p_i(a_i|a_{<i}) - 1) \quad \text{if } \mathbf{a} = \mathbf{b}, \end{cases} \tag{B8}$$

where the p_i are the conditional probabilities defined by the autoregressive model. This form is very similar to the one used in other works dealing with epistatic model in phylogenetics [19, 20, 48]. It is quite straightforward to interpret: the transition rate for sequences at distance strictly higher than one vanishes, meaning that at most one substitution can occur in an infinitesimal amount of time; if two sequences differ at site i, then the transition rate is the probability of observing the new amino acid b_i in the context of the starting sequence $\mathbf{a}$. The diagonal elements ensure that lines of $\mathbf{Q}$ sum to 0.

It is interesting to note that the stationary distribution for $\mathbf{Q}$ is the generative distribution $P^{AR}(\mathbf{a}) = \prod_{i=1}^{L} p_i(a_i|a_{<i})$, that is:

$$\sum_{\mathbf{a}} P^{AR}(\mathbf{a}) Q_{\mathbf{ab}} = 0 \quad \text{for all sequences } \mathbf{b}. \tag{B9}$$

To demonstrate this, we first note $\mathcal{N}_i(\mathbf{b})$ the ensemble of sequences that differ from $\mathbf{b}$ at position i only. Using Eq. B8, we can write

$$\begin{aligned}\sum_{\mathbf{a}} P^{AR}(\mathbf{a}) Q_{\mathbf{ab}} =& \sum_{i=1}^{L} \sum_{\mathbf{a}\in\mathcal{N}_i(\mathbf{b})} P^{AR}(\mathbf{a}) p_i(b_i|a_{<i}) + P^{AR}(\mathbf{a}) \sum_{i=1}^{L} (p_i(a_i|a_{<i}) - 1) \\ =& \sum_{i=1}^{L} \sum_{\mathbf{a}\in\mathcal{N}_i(\mathbf{b})} p_i(b_i|a_{<i}) \prod_{j=1}^{L} p_j(a_j|a_{<j}) + P^{AR}(\mathbf{b}) \sum_{i=1}^{L} (p_i(b_i|b_{<i}) - 1),\end{aligned}$$

where the first term involves all sequences at distance one from $\mathbf{b}$ and the second handles the case $\mathbf{a} = \mathbf{b}$. To make progress, we note that the sum over $\mathcal{N}_i(\mathbf{b})$ can be simplified as follows (for a generic function f):

$$\begin{aligned}\sum_{\mathbf{a}\in\mathcal{N}_i(\mathbf{b})} f(\mathbf{a}) =& \sum_{\mathbf{a}} \left(f(\mathbf{a}) \prod_{\substack{j=1 \\ j\neq i}}^{L} \delta_{a_j,b_j} \right) \\ =& \sum_{\substack{a_i=1 \\ a_i\neq b_i}}^{q} f(b_1, \ldots, b_{i-1}, a_i, b_{i+1}, \ldots, b_L).\end{aligned}$$

This essentially means that inside the sum the symbol a_j can be transformed into b_j if $j \neq i$, and that the remaining symbol a_i is traced over with the condition $a_i \neq b_i$. Using this, our calculation yields

$$\begin{aligned}\sum_{\mathbf{a}} P^{AR}(\mathbf{a}) Q_{\mathbf{ab}} &= \sum_{i=1}^{L} p_i(b_i|b_{<i}) \prod_{\substack{j=1\\ j\neq i}}^{L} p_j(b_j|b_{<j}) \sum_{\substack{a_i=1\\ a_i\neq b_i}}^{q} p(a_i|b_{<i}) \\ &+ P^{AR}(\mathbf{b}) \sum_{i=1}^{L} (p_i(b_i|b_{<i}) - 1) \\ &= P^{AR}(\mathbf{b}) \sum_{i=1}^{L} (1 - p(b_i|b_{<i})) + P^{AR}(\mathbf{b}) \sum_{i=1}^{L} (p_i(b_i|b_{<i}) - 1) \\ &= 0. \end{aligned}$$

What this means is that the $\mathbf{Q}$ of Equations B7 and B8 is the one that we would like to use: it defines a time reversible Markov process with a stationary distribution P^{AR} that is generative. We call P' this "correct" Markov process, which is defined by

$$P'(\mathbf{b}|\mathbf{a}, t) = (e^{t\mathbf{Q}})_{\mathbf{ab}}. \tag{B10}$$

However, since matrix $\mathbf{Q}$ is of dimensions $q^L \times q^L$ and we do not know how to compute its eigenvectors, we cannot actually compute $P'(\mathbf{b}|\mathbf{a}, t)$.

Instead we use the process P introduced in the main text, which has two properties: *(i)* its derivative at $t = 0$ is $\mathbf{Q}$ (Eq. B7) and *(ii)* it has P^{AR} as a stationary state for $t \to \infty$. In other words, P verifies the following:

$$\begin{aligned} &P(\mathbf{b}|\mathbf{a}, t) \simeq (\mathbb{1} + t\mathbf{Q})_{\mathbf{ab}} \simeq P'(\mathbf{b}|\mathbf{a}, t) \quad \text{for } t \ll 1, \\ &P(\mathbf{b}|\mathbf{a}, t) - P'(\mathbf{b}|\mathbf{a}, t) \xrightarrow[t\to\infty]{} 0, \end{aligned} \tag{B11}$$

where $\mathbb{1}$ is the identity matrix. In other words, the propagator of the main text is an approximation of the continuous time dynamics associated with P^{AR}, which becomes exact at small and large times.

d. Deviations at intermediate times

An undesired consequence of our approximation is that when starting with sequences sampled from the target distribution P^{AR}, the propagator P of the main text generates out-of-equilibrium sequences at intermediate times. On the contrary, equilibrium would be maintained if using the exact propagator P' of Eq. B10. In mathematical terms, and using the notation of the previous section, we would have

$$\begin{aligned}\sum_{\mathbf{a}} P^{AR}(\mathbf{a})P'(\mathbf{b}|\mathbf{a},t) = P^{AR}(\mathbf{b})\\ \sum_{\mathbf{a}} P^{AR}(\mathbf{a})P(\mathbf{b}|\mathbf{a},t) = \pi_t(\mathbf{b}),\end{aligned} \tag{B12}$$

where π_t is a distribution over sequences that becomes equal to P^{AR} for $t \to 0$ and $t \to \infty$. In order to quantify how far from equilibrium the model goes, we need to compare P^{AR} and π_t at intermediate times. We do this by performing two numerical experiments.

First, starting from an initial sequence **a** sampled from P^{AR}, we compute the average log-likelihood of sequences sampled from $P(\mathbf{b}|\mathbf{a},t)$. We then average this process over **a** to define

$$\mathcal{L}(t) = \sum_{\mathbf{a},\mathbf{b}} P^{AR}(\mathbf{a})P(\mathbf{b}|\mathbf{a},t)\log\left(P^{AR}(\mathbf{b})\right) = \sum_{\mathbf{b}} \pi_t(\mathbf{b})\log\left(P^{AR}(\mathbf{b})\right). \tag{B13}$$

For a perfect approximation, $\mathcal{L}(t)$ should remain equal to the average log-likelihood of sequences sampled from the generative model at all times. The right panel of Figure S1 shows that $\mathcal{L}(t)$ drops at intermediate times, which means that our propagator generated sequences that are “worse” than the generative model. However, the magnitude of this drop (about 5 at its minimum) is small when compared to the distribution of log-likelihoods sampled from P^{AR}. It is also small compared to the biases in the likelihood of reconstructed sequences shown in Figure 2 of the main text.

Our second test consists in using a tree generated in the same way as the ones used in the main text, and to simulate evolution using our autoregressive model by starting from an equilibrated root sequence. We then compute the distribution of log-likelihood of the leaves sequences. Again, for a process that is always at equilibrium, the distribution at the leaves should be the same as the one used to generate the root. The left panel of Figure S1 shows that this is not the case, with the log-likelihood of the leaves being on average lower. However, the two distributions are still quite close, in particular for their left tail.

We conclude from these experiments that even if our propagator has the undesirable property of going out of equilibrium at intermediate times, these deviations remain quite small. The autoregressive propagator can thus be seen as a useful *approximation*, allowing reconstruction at internal nodes without sacrificing much of the generative properties of the original model.

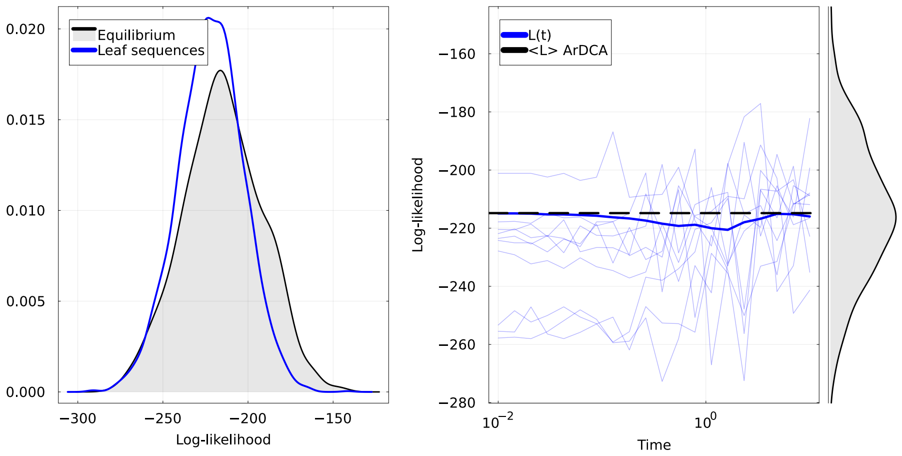

Figure S 1. Because it does not respect global balance, the propagator generates "out of equilibrium" sequences at intermediate times. **Left**: Distribution of log-likelihood of sequences at the tips of a tree (blue curve), when simulated using the autoregressive propagator and with a root sampled from the ArDCA model. For a dynamics that remains in equilibrium, the distribution should match the one of the ArDCA model (black curve). The shift indicates a slight out of equilibrium behavior. The tree used is generated in the same way as those used in the main text. **Right**: Log-likelihood of trajectories obtained by sampling the auto-regressive propagator at different times. Thin blue curves are example individual trajectories, with the initial sequence taken randomly from the equilibrium distribtion of the ArDCA model. The thick blue curve is the average of many individual trajectories. The black curve is the average log-likelihood of sequences sampled from the ArDCA equilibrium distribution. The drop in average likelihood around $t = 1$ is indicative of the out of equilibrium behavior. However its amplitude remains small with respect to the width of the equilibrium distribution

3. Position of the root

Because the autoregressive model is irreversible, the probability of a reconstruction depends on the orientation of the branches of the tree, and thus on the placement of the root. To quantify this dependence, we perform the following numerical experiment.

1. *Original tree and reconstruction.* We first generate a tree at random and simulate evolution on it using the autoregressive model, using the same procedure as in the main text. Note that by construction, the placement of the root for this original tree is

known exactly. We then perform ancestral reconstruction using the same autoregressive model, and refer to these ancestral sequences as the *original reconstruction.*

2. *Reconstruction on re-rooted trees.* We then iteratively root the original tree at each internal node, and perform reconstruction again using the same leaf sequences as before. In this case, the placement of the root is wrong in the sense that it does not correspond to the evolutionary process that generated the leaf sequences.

We use original trees of $n = 100$ leaves, and make 10 repetitions of this experiment. For a given repetition, the sequence at each internal node is reconstructed $n - 1 = 99$ times. Since there are 99 internal nodes and 10 repetitions, we obtain a set of $\sim 10^5$ reconstructed sequences. For each of these, we can compute:

- the amplitude of the re-rooting event, that is the branch-length distance between the original root of the tree and the one for which the reconstruction was performed;
- the variation with respect to the original reconstruction, measured in Hamming distance;
- the loss in performance, that is the increase in Hamming distance to the real ancestor with respect to the original reconstruction.

Figure S2 shows the results of this experiment. On its top-left panel, we see that there are indeed variations in the reconstructed sequences when changing the position of the root. However, the amplitude of these variations are quite limited, as they are on average smaller than 0.4%. We find the loss in performance to be one order of magnitude lower, typically around 0.05%. This suggests that the variations mostly occur at sites where the reconstruction was unreliable to begin with.

The top-right panel shows the same quantities but only for nodes that are close to the original root of the tree (distance < 0.1). These are nodes where we can expect more variation, as they are located far from the leaves. We indeed see that there reconstruction varies much more when the root is changed, with a difference of up to 0.08 Hamming distance extreme root misplacements. On the other hand, the loss in performance of the reconstruction remains very small, on the order of 0.1%. Again, this suggests that the change in reconstructed sequence when misplacing the root mostly occurs in parts of the sequence that were unreliable to begin with.

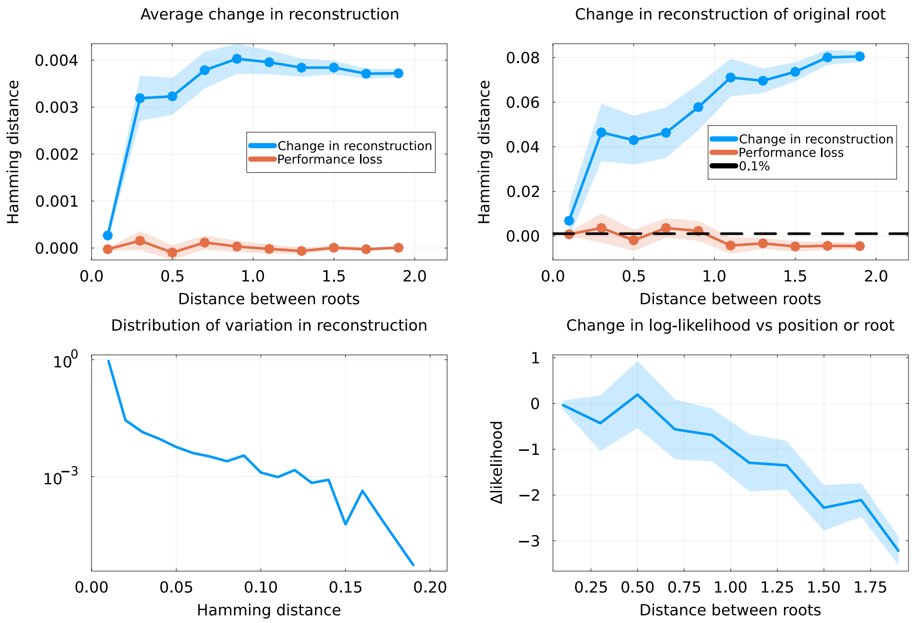

Figure S 2. Dependence of ancestral sequence reconstruction on the position of the root. **Top-left**: Variation in sequence reconstruction and loss of performance as a function of the amplitude of the re-rooting. The blue curve shows the average Hamming distance between MAP ancestral sequences when using the original tree (*i.e.* correct root placement) or a re-rooted tree, as a function of the amplitude of the re-rooting. The orange curve shows the degradation in reconstruction performance when changing the root position. **Top-right**: Same as top-left, but showing only nodes that are close (distance < 0.1) to the original root of the tree. These nodes are the farthest away from the leaves. Variation in the reconstruction is clearly larger, but the loss in performance remains very small. **Bottom-left**: Distribution of the variation in reconstruction for re-rooting of large amplitude (*i.e.* distance > 1.5): most reconstructions vary very little. In rare cases, the reconstruction varies significantly: in 0.2% of cases, the Hamming distance between two reconstructions is greater than 10%. **Bottom-right**: Average change in log-likelihood of the reconstruction of the root as a function of the amplitude of the re-rooting.

The bottom-left panel shows the distribution of variation in reconstruction for the larger root displacements (about 70 000 reconstructions). As expected, the variation is small in the vast majority of cases. Interestingly however, we observe that changing the root of the tree leads to large fluctuations in reconstruction in rare cases. For instance, in about 0.2% of cases, the Hamming distance between two reconstructions is greater than 10%.

Finally, the bottom-right shows that the likelihood of the reconstruction of the new root sequence decreases with how far it is placed from the original root. This means that if the position of the root was unknown, it could still be guessed with reasonable accuracy based on the likelihood.

Appendix C: Directed evolution data

1. Minimum reconstruction error of the consensus

In the left panel of Figure 4, the Hamming distance of the consensus of M sequences to the wild-type sequence shows a minimum for an intermediate value of M. This is at first counter-intuitive, and we present here a minimalistic example to illustrate this phenomenon.

We consider the simplified case with binary sequences of length L and a star-like tree with M leaves at equal distance from the root. The root sequence is $\mathbf{r} = (0, \ldots, 0)$, and the sequence of leaf m is $\mathbf{x}^m = (x_1, \ldots, x_L)$ with $x_i \in \{0, 1\}$. We now assume that the first site in the sequence is much more variable than the others, so that it is frequent for sequence x^m to have a 1 at position $i = 1$, but rare at positions $i > 1$. The probability of observing state 1 at a site i in a leaf sequence is

$$P(x_i^m = 1) = \begin{cases} \frac{1}{2} + \varepsilon & \text{if } i = 1, \\ \varepsilon & \text{if } i \neq 1, \end{cases} \tag{C1}$$

where $\varepsilon > 0$ is a parameter that in principle depends on the root-to-tip distance of the tree.

We now consider the consensus of the leaf-sequences and how close it is to the root $\mathbf{r} = (0, \ldots, 0)$. For the first position $i = 1$, the probability that the consensus differs from the root is the probability that more than $M/2$ leaves have mutated at this position. This is the probability that a binomial variable of parameters $\left(\frac{1}{2} + \varepsilon, M\right)$ takes a value larger than $M/2$: we call $\alpha\left(\frac{1}{2} + \varepsilon, M\right)$ this probability. Likewise, for a position $i > 1$, the probability that the consensus differs from the root is the probability $\alpha(\varepsilon, M)$ that a binomial variable of parameters (ε, M) takes a value larger than $M/2$.

It is then immediate that the average Hamming distance $H(M)$ between the consensus and the root if there are M leaves is

$$\langle H(M)\rangle = (L-1)\alpha(\varepsilon, M) + \alpha\left(\frac{1}{2}+\varepsilon, M\right). \tag{C2}$$

Ideally, we would like to show that for certain values of ε, $\langle H(M)\rangle$ has a minimum at intermediate M. Unfortunately, we are unable to give analytical expressions for $\alpha(p, M)$ for generic p and M. Before exploring this with a numerical simulation, we show that in our setup the consensus of $M = 1$ sequence can be better than the consensus of an infinite number of sequences. The limits of α for large and small M are easily obtained:

$$\alpha(p, M=1) = p \quad \text{and} \quad \alpha(p, M\to\infty) = \begin{cases} 1 & \text{if } p > \frac{1}{2} \\ 0 & \text{if } p < \frac{1}{2} \end{cases}.$$

Here, with $0 < \varepsilon < 1/2$, we have $\alpha\left(\frac{1}{2}+\varepsilon, M\to\infty\right) = 1$ and $\alpha(\varepsilon, M\to\infty) = 0$. In other words, for $M\to\infty$, the consensus at the first site will always differ from the root (as expected because it mutates "fast") while the consensus at other slow-evolving sites will be equal to the root state. We therefore obtain

$$\langle H(M\to\infty)\rangle = 1 \quad \text{and} \quad \langle H(M=0)\rangle = L\varepsilon + \frac{1}{2}. \tag{C3}$$

If $L\varepsilon < 1/2$, we observe that on average, the consensus of one sequence is closer to the root than the consensus of an infinite number of sequences.

The general case is explored in Figure S3: we show the numerical values of these $\alpha\left(\frac{1}{2}+\varepsilon, M\right)$ and $\alpha(\varepsilon, M)$ for $\varepsilon = 0.05$ and $L = 10$. The first term $\alpha\left(\frac{1}{2}+\varepsilon, M\right)$ increases monotonically from $\frac{1}{2}+\varepsilon$ to 1, while the second decreases from ε to 0. Combining the two with Eq. C2, we see that $\langle H(M)\rangle$ has a minimum at an intermediate M.

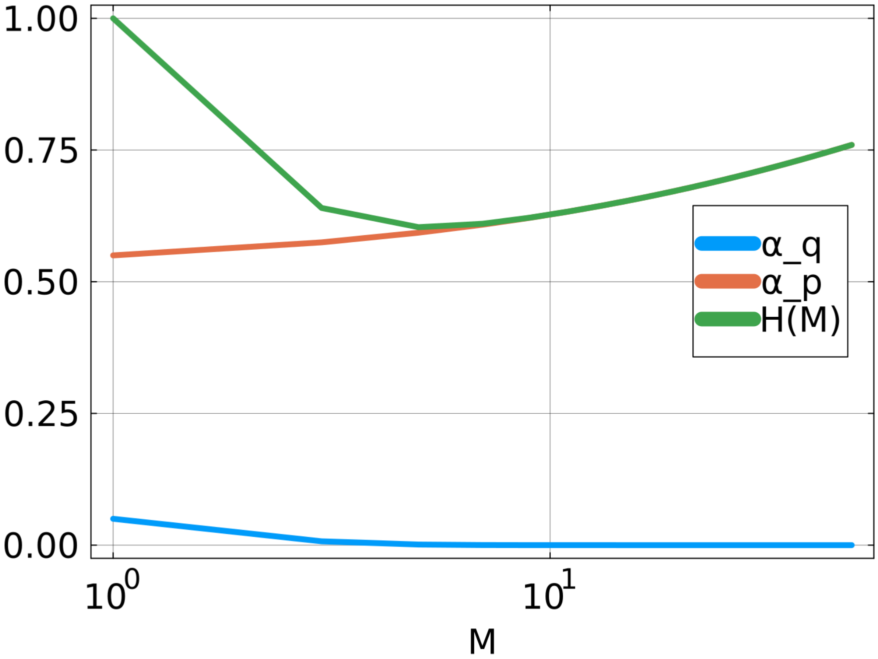

Figure S 3. Quantities $\alpha\left(\frac{1}{2}+\varepsilon, M\right)$, $\alpha\left(\varepsilon, M\right)$ and $\langle H\rangle$ as a function of the number of leaves M (odd values only). $\alpha(p, M)$ is defined to be the probability that a binomial variable of parameters (p, M) takes a value below $M/2$. $\alpha\left(\frac{1}{2}+\varepsilon, M\right)$ is increasing from $1/2+\varepsilon$ to 1 while $\alpha\left(\varepsilon, M\right)$ is decreasing from ε to 0. The average Hamming distance reaches a minimum for an intermediate number of leaves. Values of parameters: $\varepsilon = 0.05$, $L = 10$.

Appendix D: Supplementary figures

1. Extra figures

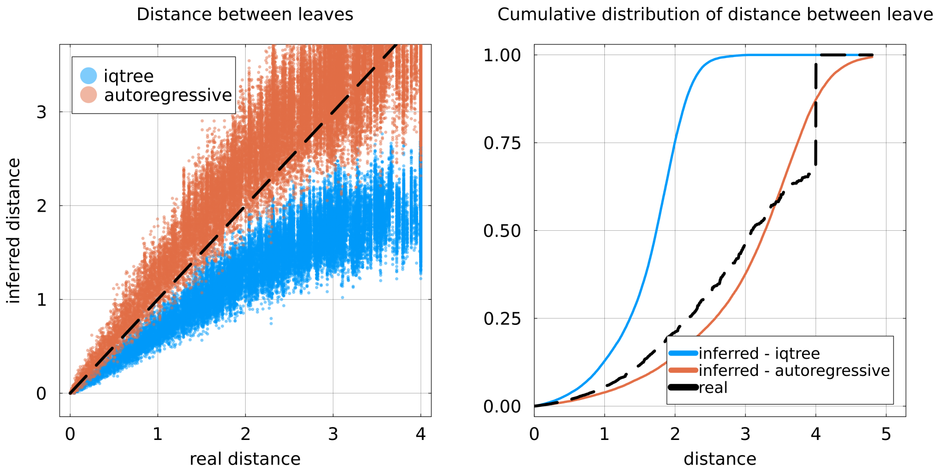

Figure S 4. Quality of branch length inference with the single-branch technique of section A 3 b, using data simulated with the autoregressive evolver and a tree with fixed topology. This is the technique used in the reconstructions of the main text. The original branch lengths inferred by IQ-TREE are displayed for comparison. **Left**: inferred distance vs distance in the real trees for every pair of leaves. **Right**: Cumulative distribution of pairwise distance along the tree between leaves for the two inference methods and for the real tree. The discontinuity in the curve for the real tree is caused by the ultrametricity and fixed total height of the generated trees.

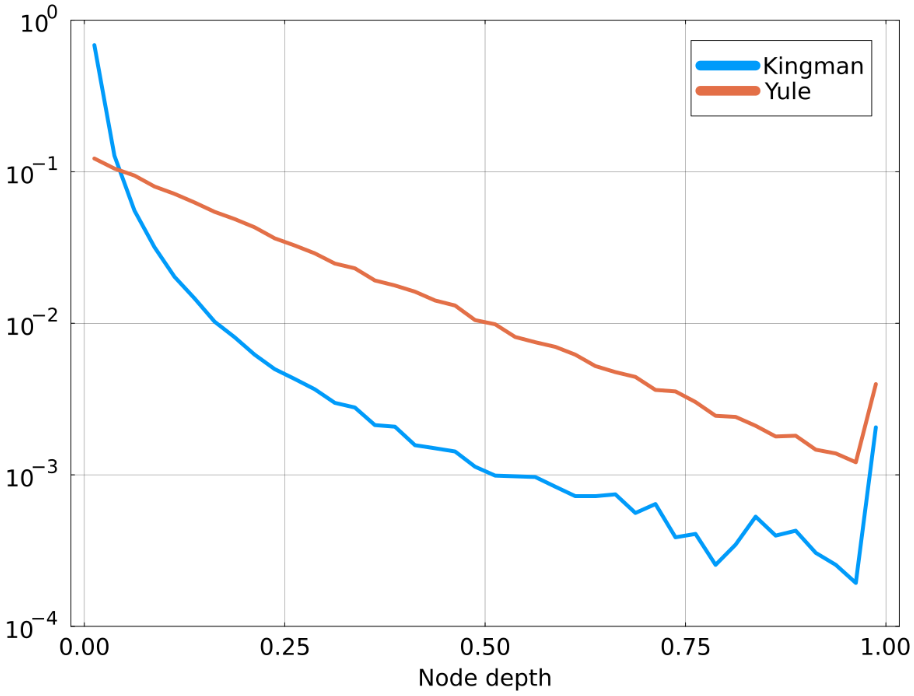

Figure S 5. Distribution of node depth for trees coming from the Kingman and Yule coalescents. Node depth is defined as the distance from a node to the closest leaf. Data is obtained by sampling several trees from each coalescent. Heights of trees are normalized to one. The Kingman process concentrates most of the nodes in close vicinity to the leaves, while the Yule process spreads them more evenly.

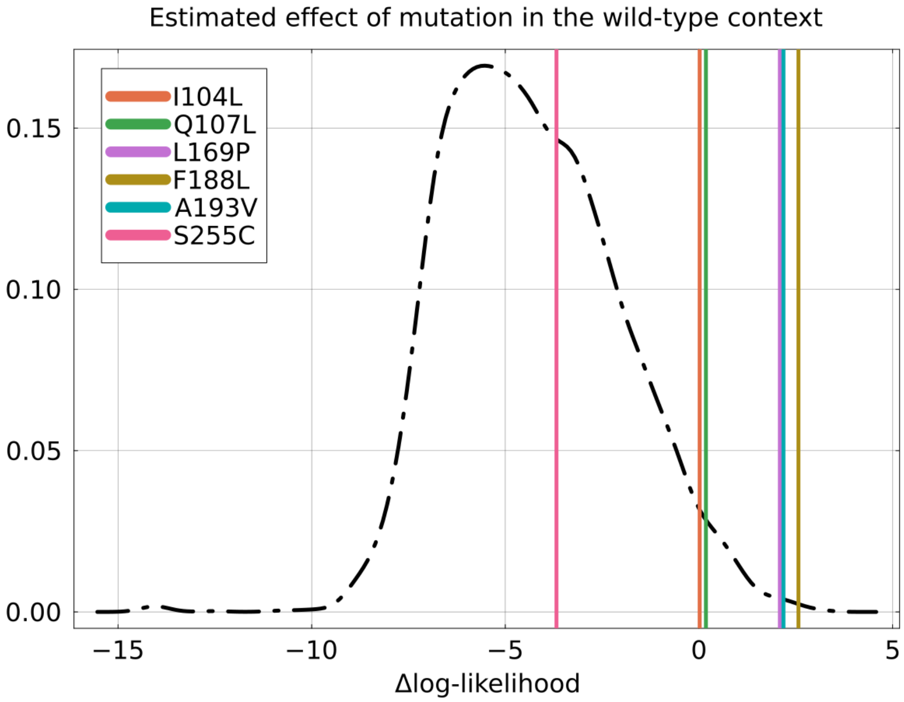

Figure S 6. Distribution of estimated effect of single mutations by ArDCA in the PSE1 sequence (black curve). The effect of a mutations is estimated by computing the difference in log-likelihood between the mutant sequence and the wild-type: negative values are detrimental and 0 represents a neutral mutation. As expected, most mutations are estimated to be detrimental but mutations found in the consensus of round 20 are mostly beneficial or neutral. The six reconstruction errors in Figure 4 are displayed as vertical bars. The two positions 169 and 193 where ArDCA outperforms IQ-TREE correspond to beneficial mutations.

2. Reconstruction of PF00072 using profile models

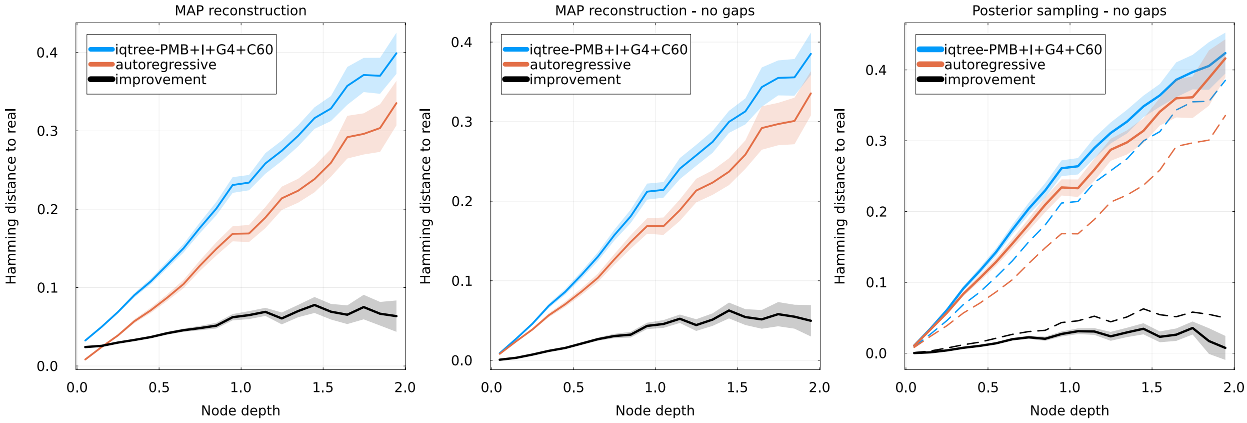

Figure S 7. Equivalent to Figure 1 of the main text, but using the `+C60` flag in IQ-TREE's reconstruction (profile model).

Hamming distance between reconstructed and real sequences as a function of node depth, using IQ-TREE and our autoregressive approach. The evolution model used by IQ-TREE is reported in the legend. The difference between the two methods ("improvement") is shown as a black curve. Estimation of the uncertainty 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 a posteriori reconstruction. **Center**: Hamming distance ignoring gapped positions, using MAP reconstruction. **Right**: comparison of posterior sampling (solid lines) and MAP (dashed lines) reconstructions, ignoring gaps.

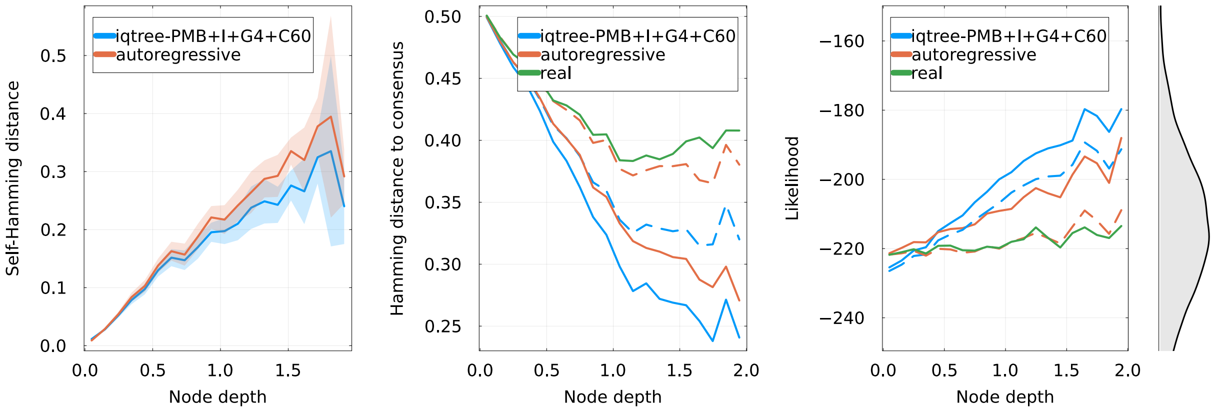

Figure S 8. Equivalent to Figure 2 of the main text, but using the `+C60` flag in IQ-TREE's reconstruction (profile model).

Left: for posterior sampling reconstruction, average pairwise Hamming distance among sequences reconstructed for each internal node. This quantifies the diversity of possible ancestral reconstructions. **Center**: Hamming distance between reconstructed sequences and the consensus sequence of the alignment. Solid lines represent MAP reconstruction or the real internal sequences, and dashed lines posterior sampling. IQ-TREE appears more biased towards the consensus sequence. **Right**: Log-likelihood of reconstructed and real sequences in the autoregressive model, *i.e.* using the logarithm of Eq. 4. MAP methods (orange and blue solid lines) are biased towards more probable sequences. Posterior sampling autoregressive reconstruction gives sequences that are at the same likelihood level than the real ancestors. The equilibrium distribution of likelihood of sequences generated by Eq. 4 is shown on the right.

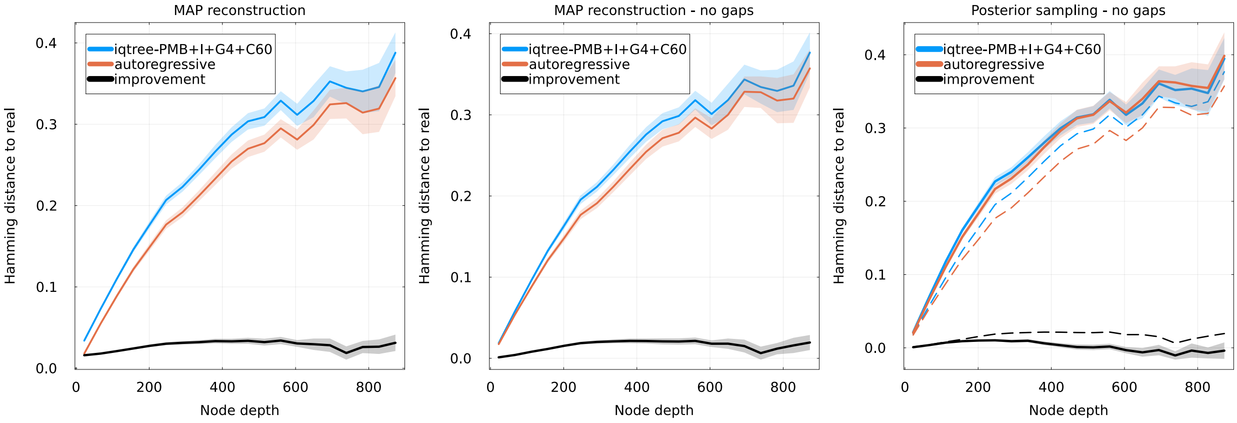

Figure S 9. Equivalent to Figure 3 of the main text. Analogous to Figure 7, but using a Potts model as the evolver. Hamming distance between reconstructed and real sequences as a function of node depth, using IQ-TREE and our autoregressive approach. The difference between the two methods is shown as 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 MAP reconstruction. **Center**: Hamming distance ignoring gapped positions, using MAP reconstruction. **Right**: comparison of posterior sampling (solid lines) and MAP (dashed lines) reconstructions, ignoring gaps.

3. Results for other protein families

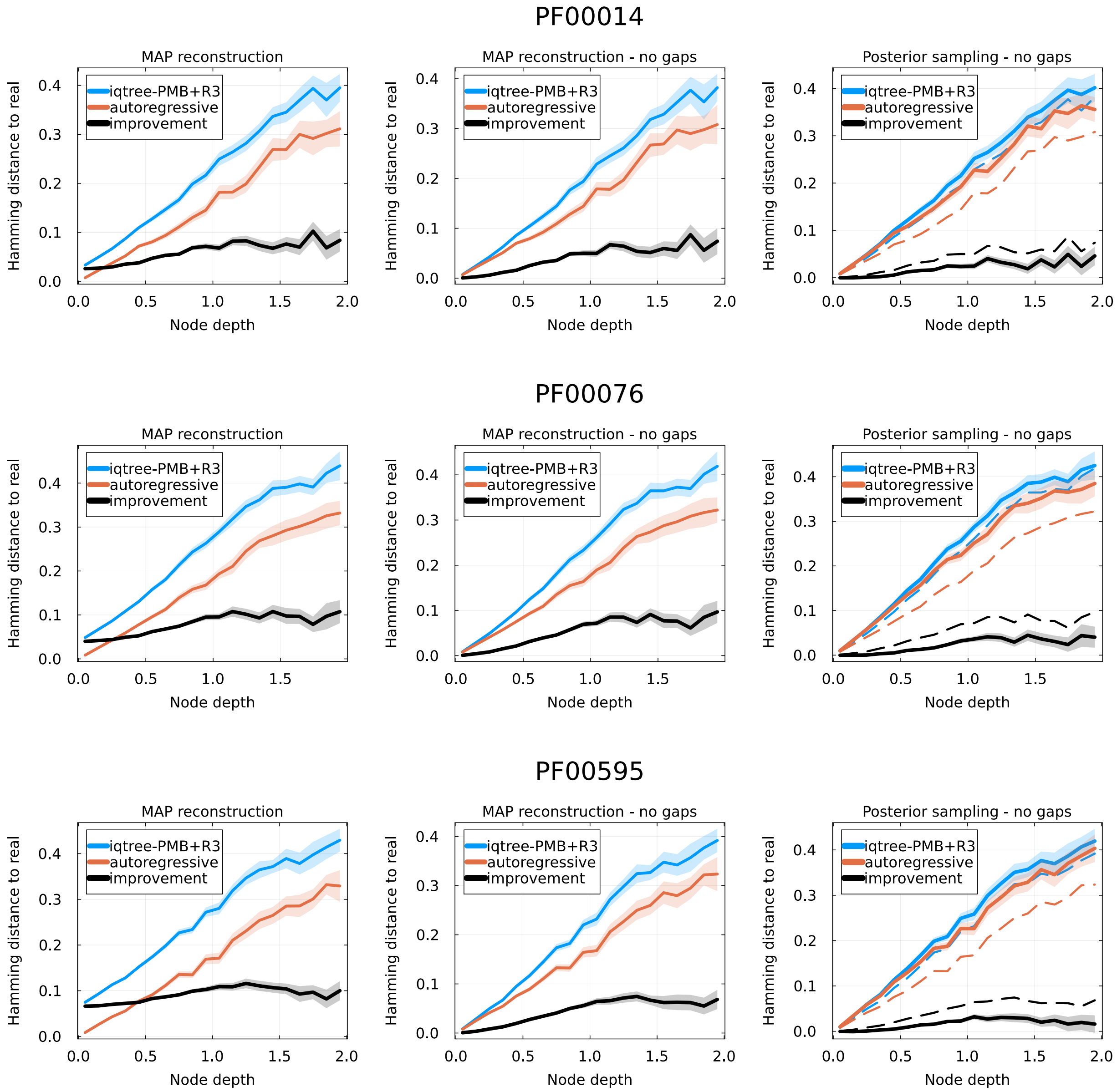

Figure S 10. Equivalent to Figure 1 of the main text using three other protein families. Hamming distance between reconstructed and real sequences as a function of node depth, using IQ-TREE and our autoregressive approach. The evolution model used by IQ-TREE is reported in the legend. The difference between the two methods ("improvement") is shown as a black curve. Estimation of the uncertainty 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 a posteriori reconstruction. **Center**: Hamming distance ignoring gapped positions, using MAP reconstruction. **Right**: comparison of posterior sampling (solid lines) and MAP (dashed lines) reconstructions, ignoring gaps.

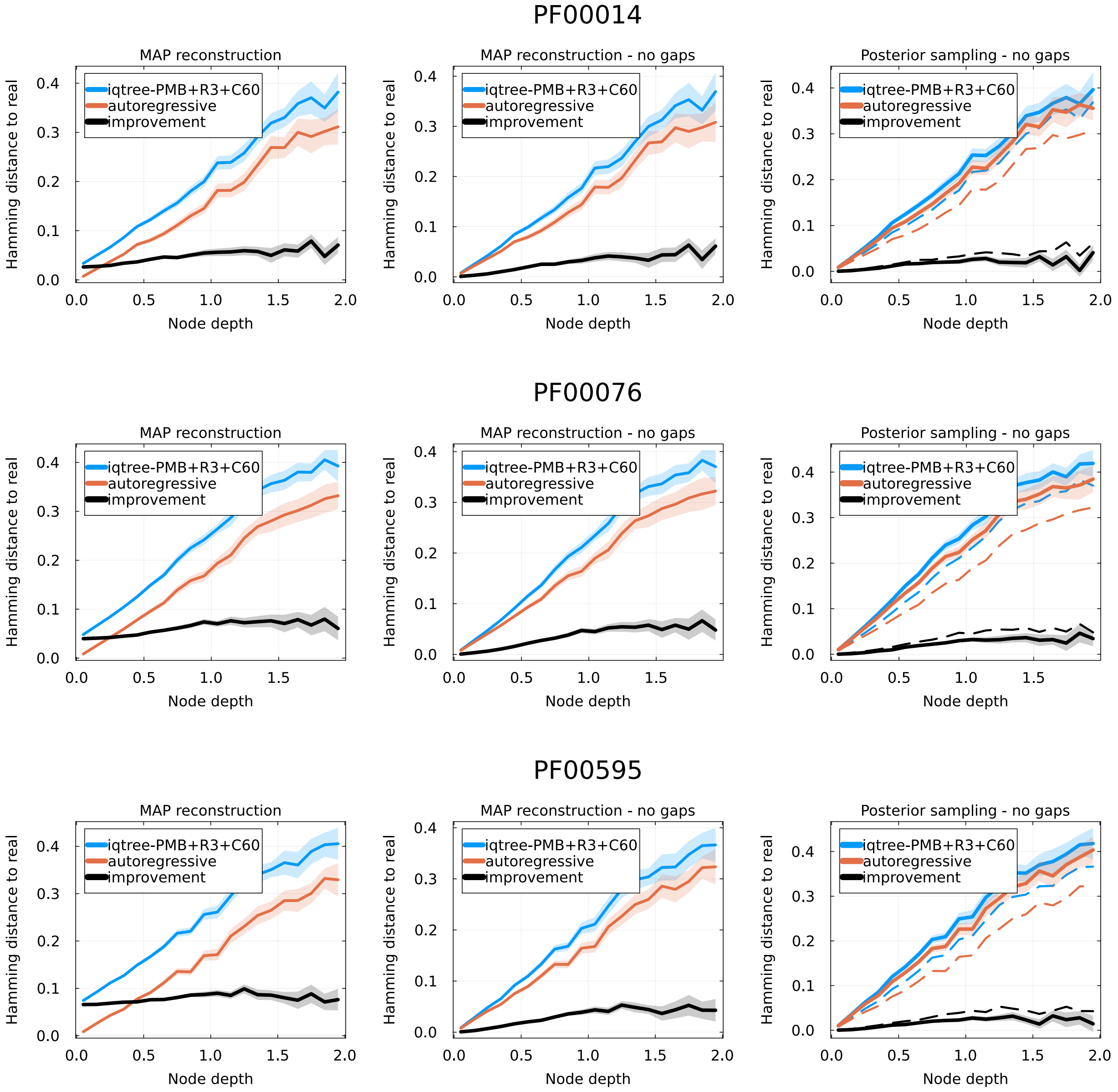

Figure S 11. Equivalent to Figure 1 of the main text using three other protein families, and using the +C60 flag in IQ-TREE's reconstruction (profile model).

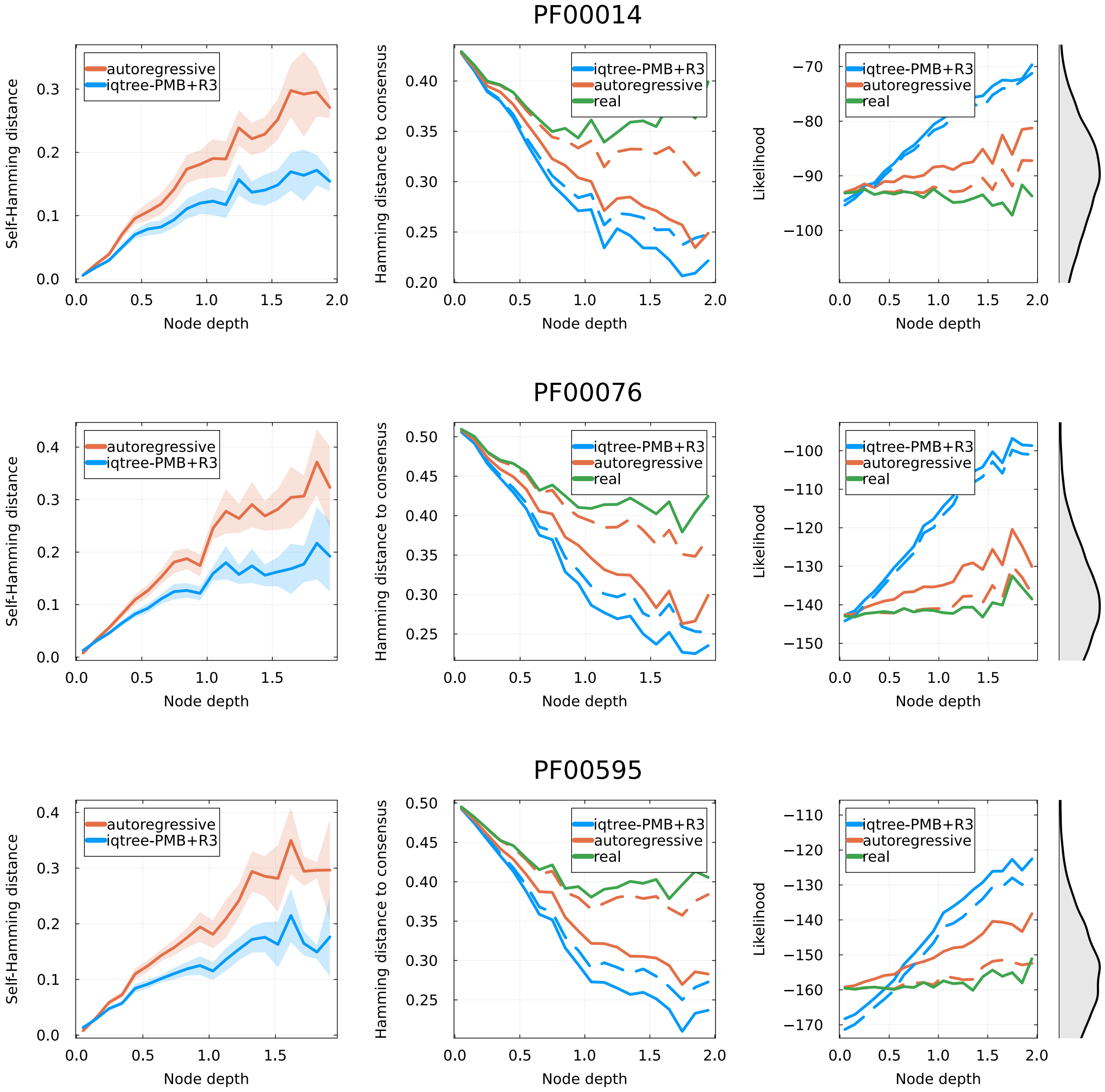

Figure S 12. Equivalent to Figure 2 of the main text using three other protein families. **Left**: for posterior sampling reconstruction, average pairwise Hamming distance among sequences reconstructed for each internal node. This quantifies the diversity of possible ancestral reconstructions. **Center**: Hamming distance between reconstructed sequences and the consensus sequence of the alignment. Solid lines represent MAP reconstruction or the real internal sequences, and dashed lines posterior sampling. IQ-TREE appears more biased towards the consensus sequence. **Right**: Log-likelihood of reconstructed and real sequences in the autoregressive model, *i.e.* using the logarithm of Eq. 4. MAP methods (orange and blue solid lines) are biased towards more probable sequences. Posterior sampling autoregressive reconstruction gives sequences that are at the same likelihood level than the real ancestors. The equilibrium distribution of likelihood of sequences generated by Eq. 4 is shown on the right.

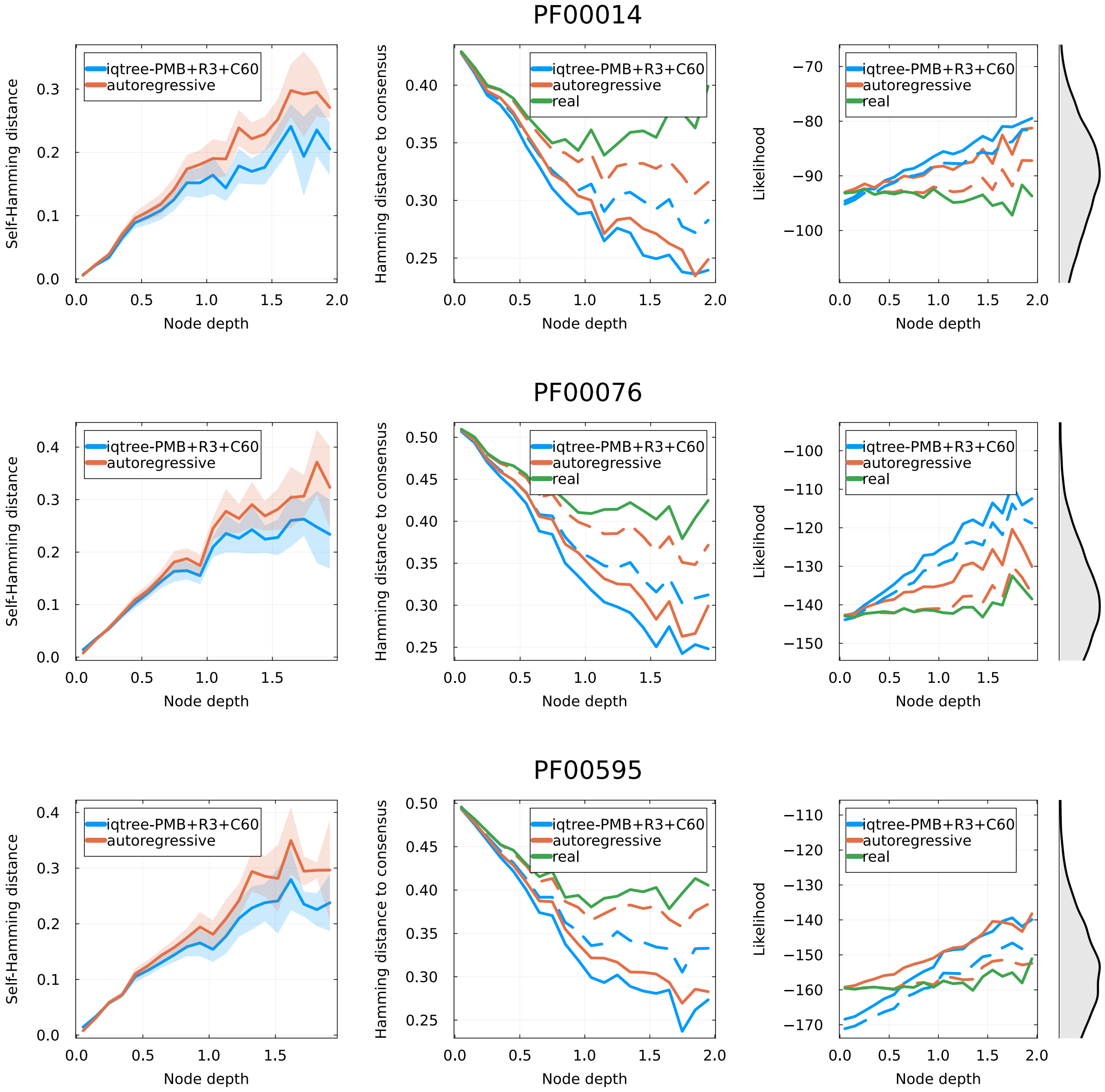

Figure S 13. Equivalent to Figure 2 of the main text using three other protein families, and using the `+C60` flag in IQ-TREE's reconstruction (profile model).