BIOINFORMATICS
Vol. 00 no. 00 2015
Pages 1–8

Prediction of Missing Sequences and Branch Lengths in Phylogenomic Data

Diego Darriba [1], Michael Weiß [2,3] and Alexandros Stamatakis [1,4,*]

[1]Scientific Computing Group, Heidelberg Institute for Theoretical Studies, Schloss-Wolfsbrunnenweg 35, 69118 Heidelberg, Germany
[2]Department of Biology, University of Tübingen, Auf der Morgenstelle 1, 72076, Tübingen, Germany
[3]Steinbeis Innovation Center for Organismal Mycology and Microbiology, Vor dem Kreuzberg 17, 72070 Tübingen, Germany
[4]Institute of Theoretical Informatics, Karlsruhe Institute of Technology, Karlsruhe, 76131, Germany

Received on XXXXX; revised on XXXXX; accepted on XXXXX

Associate Editor: XXXXXXX

ABSTRACT

Motivation: The presence of missing data in large-scale phylogenomic datasets has negative effects on the phylogenetic inference process. One effect that is caused by alignments with missing per-gene or per-partition sequences is that the inferred phylogenies may exhibit extremely long branch lengths. We investigate if statistically predicting missing sequences for organisms by using information from genes/partitions that *have* data for these organisms alleviates the problem and improves phylogenetic accuracy.

Results: We present several algorithms for correcting excessively long branch lengths induced by missing data. We also present methods for predicting/imputing missing sequence data. We evaluate our algorithms by systematically removing sequence data from three empirical and 100 simulated alignments. We then compare the Maximum Likelihood trees inferred from the gappy alignments and on the alignments with predicted sequence data to the trees inferred from the original, complete datasets. The datasets with predicted sequences showed one to two orders of magnitude more accurate branch lengths compared to the branch lengths of the trees inferred from the alignments with missing data. However, prediction did not affect the RF distances between the trees.

Availability: https://github.com/ddarriba/ForeSeqs

Contact: diego.darriba@h-its.org

Supplementary information: Supplementary data are available at Bioinformatics online.

1 INTRODUCTION

At present, typical large-scale phylogenomic datasets are assembled by concatenating several genes of the organisms under study. Such phylogenomic datasets often contain a high proportion of systematically missing data. This is because, specific gene sequences might not be available for certain taxa (e.g., specimens are unavailable or taxa do not contain the specific gene). Such phylogenomic datasets are also called 'patchy' or 'gappy' alignments. In likelihood-based models, the missing per-gene sequences are typically represented by undetermined characters. Throughout this manuscript we refer to user-defined subsets of alignment sites (e.g., genes) as 'partitions'. We further assume that all relevant likelihood model parameters (GTR rates, α, branch lengths) are estimated/optimized independently (also known as unlinked parameters) for each partition.

The presence of missing data has two notable effects on the phylogenetic inference process. Firstly, depending on the structure of the missing data blocks *and* under certain model parameter configurations (most importantly unlinked branch lengths), gappy datasets can give rise to so-called terraces in tree space (Sanderson *et al.*, 2011). A terrace in tree space contains a set of distinct tree topologies that have exactly the same likelihood score. Secondly, entirely missing data for a subset of taxa in a partition can generate extremely long branch lengths. This effect is more pronounced when data is systematically (instead of randomly throughout the tree) missing for an entire subtree.

Here, we address the latter problem. That is, we introduce and evaluate methods for correcting these artificially long branches, given a partitioned alignment and a fixed tree (e.g., the best-scoring Maximum Likelihood tree). We present, assess, and make available two algorithms for this purpose: branch length stealing and sequence prediction/imputation.

Since there is no data available for inferring branch lengths in a subtree that only comprises missing data for a specific partition, we first need to estimate or approximate these branch lengths. We can do this by using information present in other partitions that *have* data for the specific subtree. We call this process branch length stealing (Stamatakis, 2014).

Once we have stolen the branch lengths for a missing data subtree in a partition, we deploy a stochastic approach to map mutations to branches. Thereby, we can predict the missing sequence data by using the phylogenetic likelihood model as a predictive/generating process based on marginal ancestral probability vectors (MAPV) (Yang, 2006).

*to whom correspondence should be addressed

Note that given an ancestral sequence AACTCG and a simple Jukes-Cantor model of nucleotide substitution (Jukes and Cantor, 1965), a descendant sequence ATTCCG has exactly the same distance to the ancestral sequence as TACTAA. Since sequence imputation is a randomized stochastic procedure, the simulation should ideally be carried out several times to obtain a sufficiently large sample of possible outcomes. We can determine the stability of the prediction and also detect potential outliers by comparing the trees inferred from the different predicted replicates to each other.

1.1 Terminology

We define the '*reference alignment*' or '*reference data*' as a multiple sequence alignment (MSA) without missing data. Throughout this paper we refer to the best-known Maximum Likelihood (ML) tree inferred from the reference data as the '*reference tree*' and to the branches of that tree as '*reference branches*'.

Each branch defines a split/bipartition of the tree. For each partition i, if there is only missing data on one side of a split, we define the corresponding subtree as '*i-undetermined subtree*', and the corresponding branch as '*i-undetermined branch*'. We denote a subtree that does contain data as '*i-determined subtree*'. Note that an i-determined subtree *can* contain missing data in some, but not all taxa. In other words, an i-determined subtree can contain one or several i-undetermined subtrees. We call a branch that connects two i-determined subtrees an '*i-determined branch*'.

We further define an undetermined branch that does not have adjacent undetermined branches or tip nodes in exactly one of the two subtrees it roots as 'i-rooting branch'. This rooting branch splits the tree into an i-undetermined and an i-determined subtree. The nodes defining this rooting branch are the 'root nodes' of the i-undetermined and i-determined subtrees respectively.

In the example presented in Figure 1, branch b_0 splits the tree into subtrees (τ_1, τ_2) and (τ_3, τ_4, τ_5). For partition 2, branch $\{A_0, A_1\}$, denoted as b_0, *is* a 2-rooting branch, since the subtree (τ_3, τ_4, τ_5) does not contain any data for partition 2 and because there are no adjacent 2-undetermined branches in the subtree (τ_1, τ_2). Note that b_1 to b_4 are 2-undetermined branches, but *not* 2-rooting branches. This is because they are adjacent to either a 2-undetermined branch or a tip node. Using the rooting branch b_0, we can determine the root nodes of the 2-undetermined and 2-determined subtrees. In our example, A_1 is the root node of the 2-undetermined subtree, and A_0 is the root node of the determined subtree. We denote A_0 and A_1 as '*complementary 2-root nodes*'.

1.2 Test Case: Species Delimitation

A general approach to investigating the evolutionary and genetic structure of a group of related organisms using multilocus genetic sequence data typically involves, among other tasks, determining whether the populations are genetically isolated on an evolutionary timescale and are thus putative species. This task, denoted as *species delimitation* is usually carried out in a separate way, prior to the phylogenetic inference (Yang and Rannala, 2014).

Species delimitation consists in clustering molecular sequences into entities that correspond to species. use sequence similarity to associate reads with taxonomic ranks. The Poisson tree process (PTP) method (Zhang *et al.*, 2013) can delimit species using non-ultrametric phylogenies, based in the phylogenetic species concept (PSC) introduced by Eldredge and Cracraft (1980) and refined later by Baum and Donoghue (1995). In general, phylogenetic species are the smallest units for which phylogenetic relationships can be reliably inferred. The PSC states that species reside at the transition point between evolutionary relationships that are best represented phylogenetically and relationships that are best reflected by reticulating genealogical connections (Goldstein and Desalle, 2000). These differences are reflected by branch lengths that represent the mean expected number of substitutions per site between two branching events. PTP assumes that the number of substitutions between species is significantly higher than the number of substitutions within species.

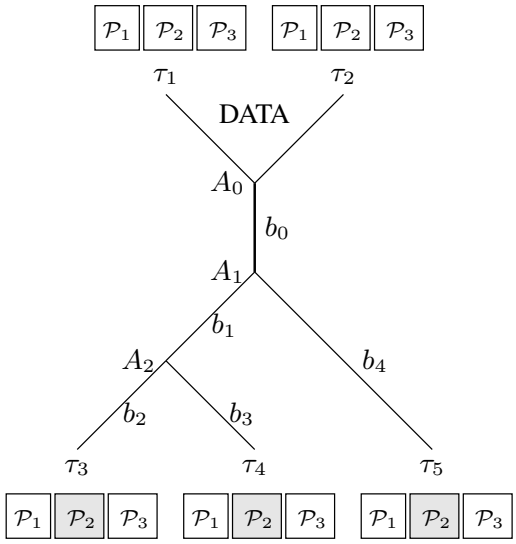

Figure 1: Example of a phylogenetic tree with missing data. τ_3, τ_4 and τ_5 are tip nodes with missing data, A_i are the inner nodes (A_0 is the ancestral node of the subtree containing data), $\mathcal{P}_i$ is the partition i for the taxa above and shown in gray if there is only missing data, b_0 is the rooting branch of partition 2.

2 BRANCH LENGTH STEALING

A prerequisite for conducting sequence prediction is to obtain a 'good' estimate of the branch lengths in the missing data subtree. Since there is no data available for estimating branches in the missing data subtree(s) of a partition, we need to obtain a branch length from elsewhere to conduct reasonable simulations/predictions. The underlying idea is to 'steal' branch length information from other partitions of the phylogenomic dataset that have data on both sides of the split/bipartition that is defined by a missing branch in the current partition. We call this approach 'branch length stealing'. In the following we describe the two distinct branch length stealing methods we assessed.

Averaging Among Partitions

Let b be an arbitrary branch. Further, let $\mathcal{C}$ be a coincidence matrix $\mathcal{C}(b, i)$ where $\mathcal{C}(b, i) = 1$ means that branch b is i-determined. We define a set $\theta_b := \{i \mid \mathcal{C}(b, i) = 1\}$, that contains the partition indices for which b is a i-determined branch.

Let $l_i(b) \in \mathcal{R}^+$ be the length of b for partition $i \in \theta_b$. Further, let m be the partition with missing data for which we want to steal the branch length $l_m(b)$. We can simply compute $l_m(b)$ as the weighted average over the branch length values in the set θ_b:

$$l_m(b) = \frac{\sum\limits_{i \in \theta_b} l_i(b)\omega_i}{\sum\limits_{i \in \theta_b} \omega_i}, \quad (1)$$

where ω_i is the weight assigned to partition i, summing to unity.

The weights ω_i are the partition length (number of sites) to alignment length ratios. Note that it *can* happen that there does not exist any partition i that has data on both sides of the split/branch b under consideration (i.e., $\theta_b = \emptyset$). In this case, branch length stealing can not be applied.

The above approach is expected to work well on datasets with homogeneous per-partition tree *and* branch lengths. If this is not the case, heterogeneity among partitions can bias results. Thus, we need to incorporate additional information from the determined branches *within* the partition for which we are trying to steal branches. The rationale for this is that the tree length in the partition m under consideration can deviate substantially from tree lengths in other partitions. In this case, an averaged stolen branch length as used above, will not fit the branch length distribution in partition m well.

Computing a Tree-Wide Branch Length Scaler

In order to address this problem, we can decrease the sensitivity of our approach to heterogeneous per-partition tree lengths by multiplying stolen branch length with a partition-specific branch-length scaler σ_m. To this end, we modify the stealing approach as follows. We initially compute a branch-length scaler by comparing the lengths of determined branches in partition m with corresponding branch lengths in other partitions. We define a set $\delta_m := \{b \mid \mathcal{C}(b,m) = 1 \wedge \exists i \neq m\, (\mathcal{C}(b,i) = 1)\}$ that contains all m-determined branches that are also i-determined for some $i \neq m$. For each branch in δ_m, we compute the average ratio between the branch length in partition m and the branch lengths in all other partitions where that branch is determined. We again scale this quantity by the relative partition size in terms of number of sites. The branch-length scaler σ_m is then computed as follows:

$$\sigma_m = \frac{1}{|\delta_m|} \sum_{c \in \delta_m} r_i c, \quad (2)$$

$$r_c = \frac{\sum\limits_{\substack{i \in \theta_c \\ i \neq m}} \frac{l_m(c)\omega_i}{l_i(c)}}{\sum\limits_{\substack{i \in \theta_c \\ i \neq m}} \omega_i} \forall c \in \delta_m, \quad (3)$$

where r_c is the ratio computed for the m-determined branch c and ω_i is the weight assigned to partition i, summing to unity.

3 PREDICTION ALGORITHM

Our initial approach for predicting missing sequences simply consisted in selecting the state that maximizes the per-site log likelihood score at each site. However, if the branches are long enough, the transition probabilities will converge to the equilibrium frequencies. In this case, the states of the predicted sites will converge to the most frequent equilibrium state. If the branches are shorter, the predicted ML states for the missing sequence are, in almost all cases, identical to the states with the highest marginal ancestral probability in the corresponding MAPV (or the corresponding ancestral sequence). Thus, we need to implement an explicit stochastic approach for predicting missing sequences.

We propose two alternative methods based on either directly simulating a MAPV or on using the most likely ancestral sequence. We start simulating sequences at the root of missing subtrees and proceed down to the tips of the subtrees via a pre-order traversal.

Note that the parameters (state frequencies, α shape parameter for the Γ distribution, and substitution rates) required for conducting the simulations are given. They have already been optimized using the existing data in the partition under consideration. Also, *all* branch lengths are already available, since the undetermined ones have been stolen from other partitions in the previous step. Thus, computing the probability transition matrix, P, for each discrete Γ rate and each stolen branch in our prediction algorithm is straight-forward.

Once this is done, we can transform each P matrix into a cumulative matrix C to simplify the stochastic state selection process. The matrix C is also a squared matrix. Each entry $C(i,j)$ contains the cumulative probability for a mutation from state i to state j. In other words, $C(i,j) = \sum_{k=0}^{j}(P[i,k])$. Thus, the entry $C(i,j)$ is simply the probability for moving from a state i to a state $s \mid s \leq j$. Given the current state i and by drawing a uniform random number from $[0,1]$, we can thus easily select a new state using C.

In the following we outline the overall prediction algorithm.

For each partition, we initially determine the set of taxa with missing data, T. For each taxon $t \in T$, we then determine the rooting branch.

Subsequently, for each rooting branch in each partition, we compute the MAPV for the node at the root of the determined subtree. Then, we steal the branch lengths (see Section 2). Once this is done, we have all the data at hand that is required to predict missing sequences.

As already mentioned, we designed two alternative approaches for predicting missing sequences. The first one uses a sequence simulation process. Here, we compute the ancestral sequence of the undetermined subtree by simply determining the most probable marginal ancestral state at each site, given the MAPV at the root of the determined subtree. Subsequently, we evolve this sequence down the subtree toward the tips where data is missing. Thereby, at each inner node we generate a simulated ancestral sequence. This method is summarized in Algorithm 1 in the supplementary information.

The second strategy consists in progressively and explicitly calculating MAPVs from the subtree root toward the tips (excluding the tips) via a pre-order traversal of the undetermined subtree. Note that the calculation of MAPVs is an entirely deterministic process based on the MAPV at the undetermined subtree root node and on the given model parameters as well as branch lengths. Unlike in the ancestral sequence simulation strategy, the stochastic/randomized selection of the final states at the tips is carried out only along terminal branches leading from a MAPV to a tip.

The MAPV (M) is a vector with n elements with s entries each, where s is the number of states (4 for nucleotides and 20 for amino acids) and n is the number of alignment sites. The entries of each

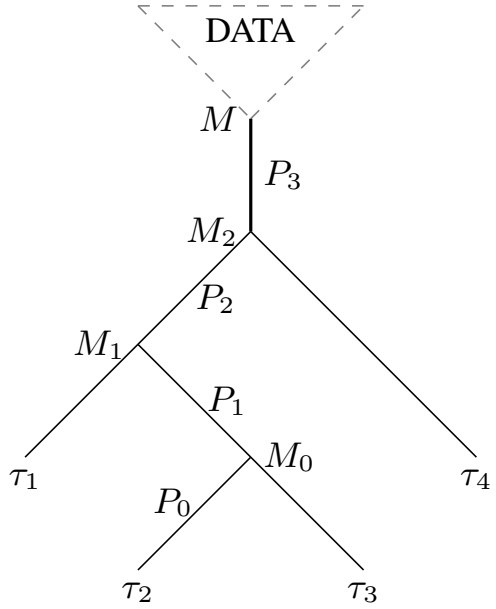

Figure 2: MAPVs (M) and P matrices for predicting the sequence at τ_2. h, the height of the root node of the undetermined subtree, is 3. $P_h = P_3$ is the P matrix for the stolen branch length at the rooting branch.

element in M sum to 1.0, since their values represent the probability of being in a specific state at the corresponding ancestral node. The M_h vector at the root is propagated down the subtree and multiplied with all transition probability matrices for the inner branches on its path to a tip. Figure 2 depicts an example of this process. In this example, the MAPV M_0 of the immediate ancestor of the tip we wish to predict, is computed as $M_0 = M_1P_1 = M_2(P_2P_1) = M(P_hP_{r-1}...P_2P_1)$.

The final tip sequence is predicted by choosing an ancestral state according to the probabilities in M_0 and the probabilities on the transition matrix P_0 leading to the tip. We can first calculate the most probable ancestral sequence based on M_0 and then stochastically select a tip state using P_0. The MAPV-based approach is summarized in Algorithm 2 in the supplementary information.

3.1 Example

Assume we have a tree with N taxa and K partitions, as shown in Figure 1. In this tree, three taxa have missing data in partition $\mathcal{P}_2$. Further assume, that the model parameters and branch lengths have been optimized independently for each partition using the input alignment *with* missing data. Initially, branch lengths for the $\tau_{3,m}, \tau_{4,m}, \tau_{5,m}$ subtree are obtained using the branch stealing process described above.

In our example, we carry out the following steps:

1. Determine the rooting branch b, and the root of the subtree containing data (A_0).
2. Calculate the ancestral sequence at A_0, S_{A_0}, by selecting the most probable states from the MAPV.
3. Carry out a pre-order tree traversal on the undetermined subtree and evolve sequences for child nodes. $S_{A_1} = S_{A_0}$ will be mutated into S_{A_2} and S_{τ_5}. The sequence S_{A_2} is ancestral to τ_3 and τ_4.
   a. Compute a P matrix for the stolen branch length between the current node and the ancestor (i.e., parental branch length) for each discrete Γ rate category. The P matrix determines the probability of observing a substitution at a site, given a parental state.
   b. Transform P into the cumulative matrix C
   c. Randomly select a new state for each site using C. With respect to handling rate heterogeneity, there are three options: (i) assign a single discrete Γ rate category randomly to each site of the undetermined subtree, (ii) assign the most likely discrete rate category to each site using information from the determined subtree, and (iii) calculate the average probability over all discrete Γ rate categories.

3.2 Implementation – ForeSeqs

We developed an open-source sequence prediction tool, called ForeSeqs, that implements the branch stealing and sequence prediction methods described in the two preceding sections. ForeSeqs uses the Phylogenetic Likelihood Library (PLL) (Flouri *et al.*, 2014) that provides functions for optimizing substitution model parameters, branch lengths, and topologies, as well as functions for computing MAPVs and ancestral sequences.

The main purpose of ForeSeqs is to predict missing data for a given MSA via the simulation process outlined above.

The input of ForeSeqs is a MSA with missing data, a phylogenetic reference tree (e.g., best-known ML tree), and a file with the partitioning scheme. One also needs to specify parameters to select among the different algorithms for branch length stealing, to choose the prediction mode (ancestral sequences or MAPVs), and to set the number r of prediction replicates. The output is a set of r MSAs without missing data.

4 EVALUATION

Our evaluation strategy was designed as follows:

1. We initially selected/generated a set of partitioned MSAs with no missing data, that is, each partition has some data for all taxa/sequences.
2. For each MSA, we created a set of evaluation samples (as described in Section 4.1) by removing one or more sequences and replacing them by missing data for a specific, randomly chosen partition of the reference alignment. We denote these samples as '*missing*' samples. Randomly removed sequences can either span an entire subtree of the reference tree (systematic removal) or not (random removal). Subsequently, we infer a ML tree on each missing sample (missing tree), which we then use as input for the prediction process with ForeSeqs.
3. For each missing sample, we predict the missing branch lengths via the branch length stealing algorithms, and the missing sequences using the respective sequence prediction algorithms.
4. We infer a ML tree on the predicted alignment (predicted tree).

5. Finally, we evaluate the differences between the reference, missing, and predicted tree, as well as between the reference and predicted branch lengths and also the alignments.

We refer to these three MSA versions and their corresponding trees as '*reference*', '*missing*', and '*predicted*', respectively. We initially compare the reference and predicted alignments by computing the Hamming distance between all reference sequences and the corresponding predicted sequences. Note that the Hamming distance does not represent a good metric for the prediction quality, because our prediction is based on a stochastic process and it exhibits a high variance with respect to the discrete character states they generate. At the same time, predicted sequences should generate consistent results in downstream analyses (e.g., tree inference). Therefore, we only included Hamming distances for the sake of completeness. We also compare the reference and predicted sequences with respect to their *guanine-cytosine content* (GC content). The GC content can be used to assess DNA stability (Petersheim and Turner, 1983). We exclude sites containing gaps in the reference alignment from Hamming distance and GC content calculations because our simulation process does not generate *indels*.

We also compare trees inferred from reference, missing, and predicted alignments using (i) the relative Robinson-Foulds (RF) distance (Robinson and Foulds, 1981) and (ii) the Kuhner-Felsenstein branch score difference (BS) (Kuhner and Felsenstein, 1994).

To compute the BS difference between two trees, $d_{BS}(T_1, T_2)$, we initially create a set of all splits or bipartitions present in at least one of the trees. Then, for each tree, every bipartition in the set is scored with either 0 if it is not present in the tree, or scored by the the branch length if the bipartition *is* present in the tree. The BS difference is calculated as sum over squared scores assigned to the bipartitions by either tree. Finally, the BS difference is normalized by the number of branches in the tree $d^*_{BS}(T_1, T_2) = d_{BS}(T_1, T_2)/(2N - 3)$.

The BS measure is more appropriate in our context, since it calculates a tree distance that also takes branch length differences into account.

We also disentangle to which extent the observed BS differences between predicted and missing trees are due to stand-alone branch length stealing and branch length stealing in conjunction with subsequent sequence prediction. For each branch stealing method, we simply replace the undetermined branches by stolen branches in the missing tree. Thus, the topology with stolen branches (but without sequence prediction) is identical to the missing tree. We denote this tree as *'unpredicted'* tree. We assess the impact of stand-alone branch stealing by comparing the respective *unpredicted* and *predicted* trees to the reference trees.

4.1 Test Datasets & Experimental Setup

We used two sets of reference MSAs for testing. The first set includes three empirical MSAs and partitioning schemes, that are described in Table 1.

For each of the three empirical MSAs, we created two groups of samples by (i) removing sequences from partitions at random, and by (ii) systematically removing the sequences of taxa located in subtrees. The number of samples generated by each of the two removal strategies is depicted in Table 1. The number of samples was determined as a function of alignment size (#taxa and # partitions).

Table 1. Summary of the empirical datasets evaluated.

	Wiegmann	**Wiens**	**Baker**
Clade	holometabolous insects	squamate reptiles	arecoid palms
Num.Taxa	12	16	173
Seq.Len	5 736	15 794	3 223
Num.Loci	6	22	2
% Gaps	19.56	4.01	57.99
Tree Len	8.026	1.086	4.285
Avg BL	0.365	0.0362	0.0124
#s rand	72	352	346
#s syst	48	264	100
Reference	Wiegmann *et al.* (2009)	Wiens *et al.* (2010)	Baker *et al.* (2011)

The second test set comprises 100 synthetic datasets, containing between 10 to 40 taxa and 4 to 10 partitions each, with a per-partition length ranging from 500 to 1200 sites.

For each synthetic dataset, we first simulated a non-ultrametric phylogenetic tree with a tree length drawn from a uniform distribution between 1.0 and 12.0.

For each partition, we then scaled the branch lengths of the underlying per-partition tree by using two different multipliers: (i) a global multiplier in $U(0.5, 2.0)$ that equally affects all branches, and (ii) a local multiplier in $U(0.8, 1.2)$ that is drawn for each branch in the partition separately. The local multiplier generates more difficult test cases because it increases branch length heterogeneity.

To then generate the sequences for each partition we chose a GTR+Γ model, with rates and frequencies drawn from Dirichlet distributions $D(1, 1, 1, 1, 1, 1)$ and $D(1, 1, 1, 1)$ respectively, and a $Gamma$ shape parameter drawn from an exponential distribution $E(2)$ truncated between 0.5 and 5.0. These truncated values for the α parameter cover a wide and representative range of low (5.0) and high (0.5) among-site rate heterogeneity.

From each simulated reference MSA, we created two *missing* MSAs by removing sequences of (i) a random number of taxa and (ii) a random number of taxa located in a subtree containing between 5 and 50% of the taxa in the tree.

Finally, we evaluated the impact of missing data and hence inaccurate branch lengths on phylogenetic post-analysis by example of species delimitation using PTP (Zhang *et al.*, 2013). See Supplementary Material for details.

4.2 Results and Discussion

4.2.1 Synthetic alignments The simulation process described in Section 4.1 assumes the same underlying topology for each partition and uses two types of branch length scalers to modify per-partition branch lengths. While the branch lengths among partitions differ, the data in each partition supports the same underlying topology. Keep in mind that the sequence prediction is conducted on the missing tree. Thus, we do not expect to observe large RF distances between the missing and predicted data trees. Overall, we obtained low RF distances to the reference trees.

The average results over all datasets are shown in Table 2. BS differences to the reference tree improve by one to two orders of magnitude for *randomly* and *systematically* removed sequences when using ForeSeqs compared to the missing trees. Note that systematically removing sequences requires stealing additional branches

that connect inner nodes. Therefore, we initially expected to obtain higher BS differences than for the random removal experiments due to cumulative branch stealing errors. Contrary to this prior expectation, we did not observe any significant differences.

When comparing the distances between the unpredicted/predicted trees to the reference trees, we see that in all cases the branch length stealing process contributes the most to branch length correction. Sequence prediction induced a difference of 10 to 25% on the remaining error after stealing branch lengths. Thus, stand-alone branch length stealing is sufficient to correct branch lengths if predicted sequences are not required. Overall, the mean RF distances to the reference trees are below 1%.

In Figure 3 we present scatter plots based on linear regression for the BS difference as a function of the percentage of missing data. We observed that branch length stealing with averaging is generally more accurate, in particular for low fractions of missing data. However, the local regression (LOWESS) shows a deviation in the tendency towards a nearly zero slope in the branch length stealing approach with scaling. In other words, the differences between the two branch stealing approaches are negligible when the fraction of missing data increases. Based on these results, the accuracy of the branch stealing approach with scaling is less sensitive to the fraction of missing data.

We observed that for predicting subtrees (*systematic* removal tests), the ancestral sequence prediction approach yields slightly more accurate branch lengths (see Table 2) than the MAPV approach. The similarity in GC content was also higher for the prediction based on ancestral sequences ($\sim$0.1% for the random removal experiments and $\sim$2% for the systematic removal experiments. In general, predicting entire subtrees increases the branch length error. Boxplots are provided in Supplementary material (Figure S2).

The BS difference between corresponding reference and missing trees was 0.101 on average, with a standard deviation of 0.204. The BS difference between corresponding reference and predicted trees has a mean of only 0.002 with a standard deviation of 0.002.

4.2.2 Empirical alignments Table 3 summarizes the results of experiments with empirical alignments. Sequence prediction does not appear to have a substantial impact on the tree topologies compared to trees inferred from MSAs with missing data. In the *random* removal experiments, differences in RF distances between missing and predicted trees to the true trees lie below 4% for the Wiegmann and Wiens alignments, and below 10% for the Baker dataset. In the *systematic* removal experiments, RF distances are higher than for *random* removals, as one might expect. Boxplots are provided in Supplementary material (Figure S3).

We do observe a substantial improvement in the BS differences to the reference tree for the predicted tree. As for the *synthetic* experiments, the predicted trees show a BS difference to the reference tree that is one to two orders of magnitude smaller than BS difference between the reference tree and the missing tree.

In the Baker alignment, scaling stolen branch lengths produced a BS difference that is noticeably higher than for the averaged stealing procedure. Also, the Hamming distance between the predicted and the reference alignment is very high ($> 50\%$), and the GC content presents worst estimates than in other cases (similarity below 95%). The following three observations explain this behavior. Firstly, the alignment has a large number of taxa compared to Wiegmann and Wiens. Therefore, a small number of removed taxa corresponds to a

Table 2. Robinson Foulds (RF) and Branch Score (BS) distances between reference trees and inferred trees for synthetic MSA; and Hamming distance (H) and similarity in GC content (GC) between the reference and predicted sequences. From the datasets, either random taxa (Random) or complete subtrees (Systematic) were removed. 'Removed' is the tree inferred from datasets with missing data. 'Unpred' is the tree inferred from datasets with missing data, but with stolen branch lengths. The branch length stealing strategies used average per-partition (Avg) and tree-wide branch length scalers (Scaler). The sequence prediction strategies used ancestral sequences (Seq) and marginal ancestral probabilities vectors (MAPVs). Here, μ is the average score, and Δ the average difference with respect to the 'Removed' tree.

			RF		BS		H(%)	GC
			μ	Δ	μ	Δ		
Random	**Removed**		0.0092		0.1025			
	Avg	**Unpred**	0.0092		0.0011	-0.1014		
		Seq	0.0087	-0.0005	0.0013	-0.1012	23.23	0.9607
		MAPV	0.0062	-0.0030	0.0014	-0.1011	22.66	0.9615
	Scale	**Unpred**	0.0092		0.0014	-0.1011		
		Seq	0.0076	-0.0014	0.0016	-0.1009	24.81	0.9630
		MAPV	0.0074	-0.0016	0.0016	-0.1009	24.20	0.9623
Systematic	**Removed**		0.0044		0.1089			
	Avg	**Unpred**	0.0044		0.0013	-0.1076		
		Seq	0.0046	0.0002	0.0014	-0.1075	32.11	0.9539
		MAPV	0.0043	-0.0001	0.0017	-0.1072	26.68	0.9364
	Scale	**Unpred**	0.0044		0.0059	-0.1030		
		Seq	0.0053	0.0009	0.0044	-0.1045	33.03	0.9574
		MAPV	0.0069	0.0024	0.0049	-0.1040	28.05	0.9369

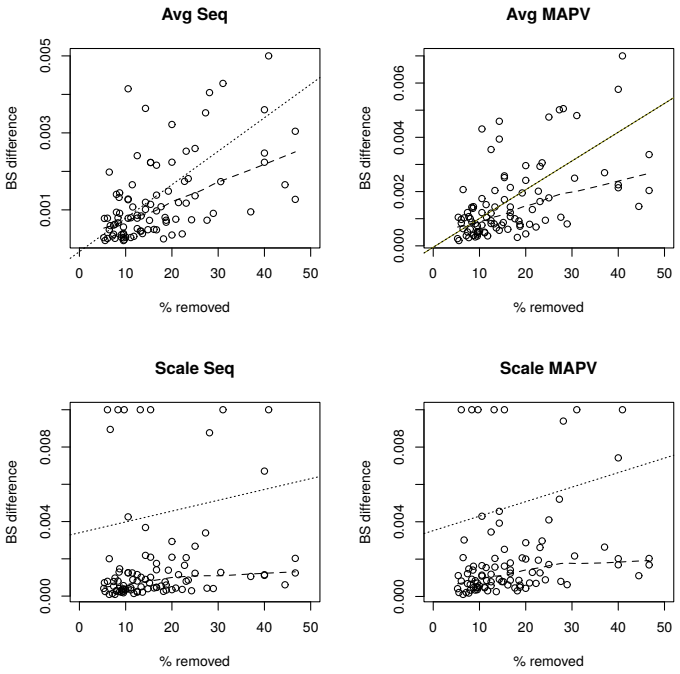

Figure 3: Scatterplot of the branch score differences and percentage of removed taxa for synthetic alignments with systematically removed data. We also depict the linear (dotted) and local (dashed) regressions.

low overall fraction of missing sequences. According to our findings for simulated data, this *has* an effect on the BS difference. Secondly, the ratio of the branch lengths between the two partitions in the alignment has a standard deviation of $\sigma = 3614.40$. Such a high variance among per-branch scalers means that using a scaler can introduce a significant bias in stolen branch length values, irrespective of the fraction of sequences removed. Finally, the number of 'true' alignment gaps (not missing data) is high and close to 80% in some per-partition sequences which leads to long branches as well. Thus, the sequence prediction will either use too short or too long branches caused by the long stretches of gaps in sequences that have some data. We can observe that the RF and BS distances increase proportionally to the amount of 'true' MSA gaps in the empirical alignments (19.56%, 4.01%, and 57.99% for the Wiegmann, Wiens, and Baker alignments, respectively). Also the high differences in the average Hamming distance to the reference alignment between the alignments predicted using the *scale* and *average* branch stealing methods suggests that there is a substantial difference in the expected number of substitutions which is directly related to the stolen branch length values. Unlike in the simulated alignments, in a general way any branch length stealing or prediction strategy performed better than others estimating the GC content.

4.2.3 Species delimitation We find that the impact of missing per-gene data on branch lengths can substantially bias the number of species delimited by PTP. Note that, PTP only relies on the branch lengths of a given phylogeny to delimit species. For trees inferred on MSAs with missing data, the average species number difference is 5.99 (18.13%, standard deviation: 7.95) with respect to PTP delimitations with trees inferred on the original MSA. With the average per-partition branch length scaler, this difference decreases to 1.18 (3.58%, standard deviation: 1.05) and 1.25 (3.79%, standard deviation: 1.36) for trees inferred on predicted sequences. In contrast to this, tree-wide branch length scalers increase the difference to 10.72 (32.48%, standard deviation: 10.31) for stolen branch lengths and 9.71 (29.42%, standard deviation: 11.96). The results are summarized in Figure S4 of the Supplementary Material.

5 CONCLUSION

We presented a method and a tool for predicting missing data in partitioned datasets. We described two procedures for approximating (stealing) the branch lengths of bipartitions with missing data.

Using empirical *and* synthetic datasets we designed realistic test scenarios to evaluate our methods. The stealing and prediction methods yield significant improvements in branch length accuracy of ML trees compared to trees inferred from MSAs with missing data. The BS differences calculated between the trees inferred from the predicted alignments and the reference trees were one to two orders of magnitude smaller than for the missing data trees. We find that branch stealing contributes by far the most to improving branch length estimates. By example of the PTP species delimitation tool, we demonstrate that branch stealing can substantially improve the accuracy of a phylogenetic post-analysis. Sequence prediction can be omitted when one only needs to correct branch lengths for missing data. As we show, sequence prediction *can* be useful for estimating statistical properties of the missing sequences, such as, for instance, their GC content.

Overall, for small fractions of missing data predictions using MAPVs yielded slightly better results than predictions based on discrete ancestral sequences. Nevertheless, predictions based on discrete ancestral sequences outperformed the MAPV-based strategy in most of our tests.

Finally, we observed that predicting sequences in general is difficult for alignments that contain a high amount of 'true' alignment gaps that are treated as missing data in all standard likelihood implementations. While we observed an improvement in BS differences when using prediction compared to alignments with missing data, the average and the variance of the BS increased proportionally to the amount of 'true' alignment gaps in the reference alignments. Also note that prediction can not correct incorrect topologies that have been inferred from alignments with missing data. However, despite the fact that previous studies have suggested that missing data can strongly bias phylogenetic inferences (Lemmon *et al.*, 2009), there is an ongoing debate regarding the topological impact of missing data. Wiens and Morrill (2011) concluded, for instance, that missing data might not be an issue for correctly placing taxa with incomplete data into a given reference tree.

ACKNOWLEDGEMENT

Funding: D.D. and A.S. are funded via institutional funding from the Heidelberg Institute for Theoretical Studies.

REFERENCES

Baker, W. J., Norup, M. V., Clarkson, J. J., Couvreur, T. L., Dowe, J. L., Lewis, C. E., Pintaud, J.-C., Savolainen, V., Wilmot, T., and Chase, M. W. (2011). Phylogenetic relationships among arecoid palms (arecaceae: Arecoideae). *Annals of botany*, **108**(8), 1417–1432.

Flouri, T., Izquierdo-Carrasco, F., Darriba, D., Aberer, A., Nguyen, L.-T., Minh, B., Von Haeseler, A., and Stamatakis, A. (2014). The phylogenetic likelihood library. *Systematic biology*, page syu084.

Jukes, T. and Cantor, C. (1965). *Evolution of protein molecules*. New York: Academic Press.

Kuhner, M. K. and Felsenstein, J. (1994). A simulation comparison of phylogeny algorithms under equal and unequal evolutionary rates. *Molecular Biology and Evolution*, **11**(3), 459–468.

Lemmon, A. R., Brown, J. M., Stanger-Hall, K., and Lemmon, E. M. (2009). The effect of ambiguous data on phylogenetic estimates obtained by maximum likelihood and bayesian inference. *Systematic Biology*, **58**(1), 130–145.

Petersheim, M. and Turner, D. H. (1983). Base-stacking and base-pairing contributions to helix stability: thermodynamics of double-helix formation with ccgg, ccggp, ccggap, accggp, ccggup, and accggup. *Biochemistry*, **22**(2), 256–263.

Robinson, D. and Foulds, L. R. (1981). Comparison of phylogenetic trees. *Mathematical Biosciences*, **53**(1), 131–147.

Sanderson, M. J., McMahon, M. M., and Steel, M. (2011). Terraces in phylogenetic tree space. *Science*, **333**(6041), 448–450.

Stamatakis, A. (2014). The raxml v8.1.x manual.

Wiegmann, B. M., Trautwein, M. D., Kim, J.-W., Cassel, B. K., Bertone, M. A., Winterton, S. L., and Yeates, D. K. (2009). Single-copy nuclear genes resolve the phylogeny of the holometabolous insects. *BMC biology*, **7**(1), 34.

Wiens, J. J. and Morrill, M. C. (2011). Missing data in phylogenetic analysis: reconciling results from simulations and empirical data. *Systematic Biology*, page syr025.

Wiens, J. J., Kuczynski, C. A., Townsend, T., Reeder, T. W., Mulcahy, D. G., and Sites, J. W. (2010). Combining phylogenomics and fossils in higher-level squamate reptile phylogeny: molecular data change the placement of fossil taxa. *Systematic Biology*, **59**(6), 674–688.

Table 3. Robinson Foulds (RF) and Branch Score (BS) distances from real data alignments. For further explanations please refer to Table 2

			Random						Systematic					
			RF		BS		H(%)	GC	RF		BS		H(%)	GC
			μ	Δ	μ	Δ			μ	Δ	μ	Δ		
Wiegmann	**Removed**		0.0354		0.1845				0.1253		0.4095			
	Avg	**Unpred**	0.0354		0.0062	-0.1783			0.1253		0.0101	-0.3994		
		Seq	0.0354	0.0000	0.0063	-0.1782	31.80	0.9657	0.1229	-0.0024	0.0128	-0.3967	34.72	0.9592
		MAPV	0.0354	0.0000	0.0063	-0.1782	31.80	0.9658	0.1253	0.0000	0.0119	-0.3976	33.50	0.9594
	Scale	**Unpred**	0.0354		0.0061	-0.1784			0.1253		0.0100	-0.3995		
		Seq	0.0354	0.0000	0.0061	-0.1784	31.64	0.9624	0.1348	0.0095	0.0141	-0.3954	34.40	0.9621
		MAPV	0.0354	0.0000	0.0061	-0.1784	31.60	0.9627	0.1300	0.0047	0.0118	-0.3977	33.18	0.9603
Wiens	**Removed**		0.0192		0.0125				0.0313		0.0236			
	Avg	**Unpred**	0.0192		0.0001	-0.0124			0.0313		0.0001	-0.0235		
		Seq	0.0195	0.0003	0.0001	-0.0124	9.86	0.9877	0.0313	0.0000	0.0001	-0.0235	12.39	0.9858
		MAPV	0.0198	0.0006	0.0001	-0.0124	9.86	0.9876	0.0313	0.0000	0.0001	-0.0235	11.07	0.9869
	Scale	**Unpred**	0.0192		0.0001	-0.0124			0.0313		0.0001	-0.0235		
		Seq	0.0204	0.0012	0.0001	-0.0124	9.92	0.9879	0.0322	0.0009	0.0001	-0.0235	12.52	0.9856
		MAPV	0.0204	0.0012	0.0001	-0.0124	9.92	0.9878	0.0313	0.0000	0.0001	-0.0235	11.16	0.9872
Baker	**Removed**		0.0846		0.0197				0.1311		0.0545			
	Avg	**Unpred**	0.0846		0.0001	-0.0196			0.1311		0.0001	-0.0544		
		Seq	0.0973	0.0127	0.0001	-0.0196	2.91	0.9914	0.0793	-0.0518	0.0010	-0.0535	5.86	0.9780
		MAPV	0.0969	0.0123	0.0001	-0.0196	2.91	0.9912	0.0771	-0.0540	0.0011	-0.0535	4.87	0.9803
	Scale	**Unpred**	0.0846		0.0136	-0.0061			0.1311		0.0372	-0.0173		
		Seq	0.0936	0.0091	0.0110	-0.0087	53.82	0.9680	0.0827	-0.0484	0.0280	-0.0265	67.01	0.9433
		MAPV	0.0939	0.0093	0.0110	-0.0087	53.82	0.9668	0.0818	-0.0493	0.0243	-0.0302	61.01	0.8773

Yang, Z. (2006). *Computational molecular evolution*, volume 21. Oxford University Press Oxford.

Yang, Z. and Rannala, B. (2014). Unguided species delimitation using dna sequence data from multiple loci. *Molecular biology and evolution*, page msu279.

Zhang, J., Kapli, P., Pavlidis, P., and Stamatakis, A. (2013). A general species delimitation method with applications to phylogenetic placements. *Bioinformatics*, **29**(22), 2869–2876.