Supplement to “Much Ado About Nothing: Accelerating Maximum Likelihood Phylogenetic Inference via Early Stopping to evade (Over-)optimization"

Anastasis Togkousidis[1,2], Alexandros Stamatakis[3,1,2], and Olivier Gascuel[4]

[1]Computational Molecular Evolution Group, Heidelberg Institute for Theoretical Studies, Germany; [2]Institute of Theoretical Informatics, Karlsruhe Institute of Technology, Karlsruhe, Germany; [3]Biodiversity Computing Group, Institute of Computer Science, Foundation for Research and Technology - Hellas; [4]Institut de Systématique, Evolution, Biodiversité (ISYEB, UMR7205 - CNRS, Muséum National d'Histoire Naturelle, SU, EPHE, UA), Paris, France

1. Introduction

In this Supplement, we first introduce two additional Early Stopping criteria based on Sampling Noise (SN) quantification for a given MSA. These methods are included here because, while they did not perform as well as the KH-based methods, they may provide inspiration for readers interested in developing new techniques to accelerate ML tree inference. We then provide additional details on the sRAxML-NG and KH-based methods that were omitted from the main text to enhance readability. Next, we include command-line instructions for invoking all Early Stopping versions, and outline the HPC cluster details used in our experiments. Finally, we present results on an additional 725 empirical and 506 simulated MSAs, which are generally shorter than the datasets used in the main experiments.

2. Sampling Noise methods

As mentioned in the main text, MSAs are subject to noise stemming from both, stochastic, and systematic sources. Evolution, which is stochastic in nature, induces a form of *stochastic noise* in the sequences, onto which *Sampling Noise* is superimposed. Stochastic noise reflects the extent to which the observed data distribution deviates from the theoretically expected values. Sampling Noise occurs because the sequences typically used for a phylogenetic analysis merely represent a small fraction of the corresponding genomes. Thus, the appropriateness of such a sample to approximate the original distribution may be put into question. The following two criteria strive to quantify the inherent Sampling Noise (SN) in the input MSA. Specifically, the first method assumes a Normal Distribution to approximate Sampling Noise (SN-Normal), while the second employs a non-parametric RELL-like (Kishino et al., 1990) approach (SN-RELL).

2.1. Sampling Noise Normal Distribution (SN-Normal)

In the case of sampling noise, the "true" log-likelihood value is the expected log-likelihood value when drawing a sample of size s (i.e., the number of sites in the MSA) from a from a substantially larger (full genome) MSA. The observed variations around the true value correspond to sampling noise and allow to quantify the ability of the sample to approximate the complete distribution. We adopt a simple model wherein we posit that the log-likelihood is subject to a type of noise that is invariant *before* and *after* the fundamental optimization block (SPR round) and that it follows the same normal distribution (SN-Normal). In other words:

$$L = \Lambda + n$$

and:

$$NL = N\Lambda + n'$$

Here, n and n' denote independent noise drawn from the same distribution, Λ and $N\Lambda$ denote the "true" log-likelihood values corresponding to the trees before and after an SPR round, respectively, and L and NL denote the noisy log-likelihood values for the same tree topology ($NL \geq L$). By assuming independence between n and n', we can estimate the parameters of this distribution (i.e., the mean and standard deviation, assuming normality) merely using a single, yet reasonable, that is non-random (e.g., parsimony) starting tree. We propose the following straight-forward parametric solution:

1. Consider the set of per-site log-likelihood values for the s sites in the MSA. Their sum is L, and the standard deviation is σ_{SN}.
2. The standard deviation of the sum over the per-site log-likelihoods (equal to L) is $\sigma_{SN} \cdot \sqrt{s}$.
3. Under these parametric assumptions, the quantity $\frac{NL-L}{\sigma_{SN}\sqrt{2s}}$ follows a normal distribution $N(0,1)$. We can derive the 95% confidence interval via the cumulative distribution function $\Phi(\cdot)$ of a standard normal distribution. We continue the search with 95% confidence, if:

$$\Phi\left(\frac{NL-L}{\sigma_{SN}\sqrt{2s}}\right) \geq 0.95 \quad \Rightarrow$$

$$NL - L \geq 1.645 \cdot \sigma_{SN}\sqrt{2s} = \epsilon \qquad \textbf{(S1)}$$

This ϵ value is based on a normal assumption and uses the central limit theorem because we sum over a sufficiently large number of independent variables (i.e., per-site log-likelihoods). The simplicity of this approach allows for rapid computation of the ϵ-threshold, while maintaining reasonable accuracy, yet under certain conditions only, as we show via our experiments.

2.2. Sampling Noise RELL Approximation (SN-RELL)

In analogy to the SN-Normal stopping rule, we also propose a non-parametric version via the RELL bootstrap approach (SN-RELL), as follows:

1. Resample s per-site log-likelihoods *with* replacement from a reasonable (e.g., parsimony) starting tree, and compute the sum L_1^* over these s values.

2. Repeat the experiment in step 1 N times (e.g., $N := 1{,}000$) to obtain N $L_i^*, i = 1, 2, ..., N$ values. The non-parametric, empirical distribution of L_i^* values represents the variability of L and NL. In the next step we estimate the distribution of $NL - L$ assuming independence of NL and L and the Null Hypothesis being $N\Lambda = \Lambda$.

3. Randomly and independently draw M (e.g., $M =: 1{,}000$) pairs of L_i^* values, where NL^* and L^* denote the maximum and minimum of a single pair, and represent NL and L, respectively. Compute $NL^* - L^*$ and the 95% quantile of the distribution of M differences of random pairs. This yields an ϵ-threshold analogous to the one we propose in (eq. S1), but is non-parametric. In other words, when $NL - L$ exceeds the 95% quantile of the distribution of differences, we are confident that the optimization step significantly improved the log-likelihood score. Otherwise, the difference $NL - L$, that is the log-likelihood difference cannot be distinguished from Sampling Noise.

2.3. Limitations

The independence assumptions between NL and L underlying the proposed Sampling Noise methods might be strong and not necessarily valid. The ϵ values generated by the two methods are comparatively large in practice, of the order of thousands of log-likelihood units (see Figure S10). Based on our experience with developing ML heuristics and the results of Haag *et al.* (2023), these ϵ values might result in stopping too early and, hence, caution is warranted when employing such thresholds.

3. sRAxML-NG and KH-based methods

Here, we provide additional information on the implementation and specific considerations of the Early Stopping methods. This includes a detailed outline of the heuristic of the Simplified RAxML-NG version (sRAxML-NG), and specific boundary cases that required dedicated solutions in the KH-based versions.

3.1. RAxML-NG v1.2 vs sRAxML-NG

Figure S1 provides a schematic representation of the RAxML-NG v1.2 tree search heuristic. This representation is simplified and omits unnecessary details, such as the exact implementation of Subtree Prune and Regraft (SPR) rounds as they are identical among the different versions. The purpose of this Figure is to provide an overview of the thoroughness of the original RAxML-NG heuristic and to highlight the limited as well as hard-to-interpret role of the ϵ-thresholds in terminating the search. Additional technical details on the standard RAxML-NG heuristic can be found in Kozlov *et al.* (2018).

Branch-Length Optimization and Model-Parameter Optimization rounds are denoted by BLO and MPO, respectively. After each SPR round a full BLO round is conducted on the X best trees found during this round (see Kozlov (2018) for details). For the sake of the simplicity we do not display this BLO step in Figure S1 but consider it as being an implicit part of the SPR round block in the diagram. The initial value of `best_radius` parameter is 5.

The RAxML-NG v1.2 heuristic can be divided into three successive and different types of several SPR rounds, corresponding to the three closed loops of Figure S1. These SPR rounds are denoted by the following functions in the workflow:

1. `AUTODETECT FAST SPR ROUND(min_radius, max_radius)`: This function takes as input the minimum and maximum SPR radius parameters, which denote the minimum and maximum range (in terms of number of nodes away from the original branch where the subtree was pruned) of regrafting distances of an SPR move on the tree topology. These SPR rounds are essentially `FAST SPR` rounds (see below). The purpose of this initial sequence of SPR rounds is to automatically determine the `best_radius` parameter for the dataset at hand, which is used by `FAST SPR` rounds as the maximum regrafting distance. The algorithm successively increases the minimum and maximum radius parameters by 5 units after each SPR round, and the sequence terminates when the log-likelihood improvement is less than the ϵ-threshold or when the global maximum radius setting of 25 units has been reached.

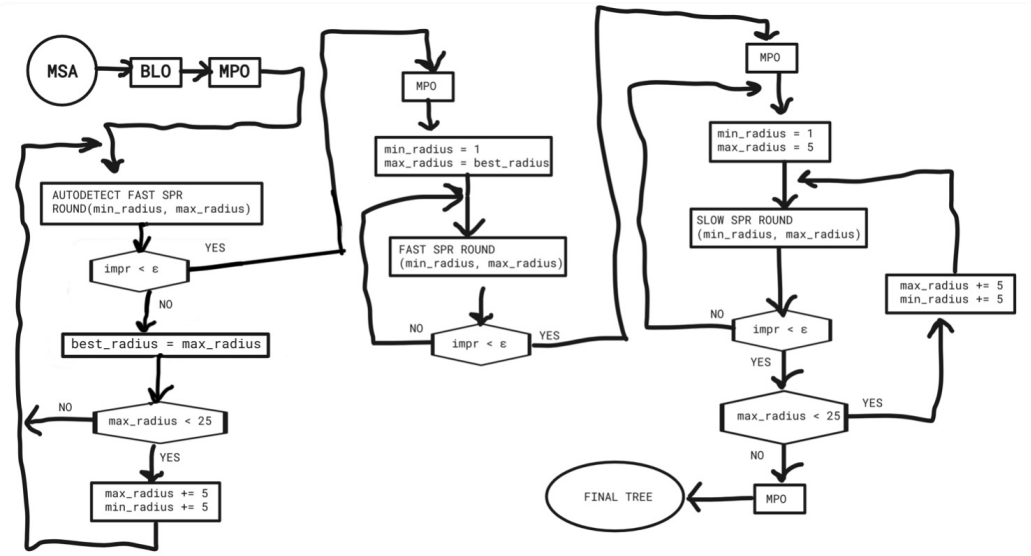


Figure S1. A schematic representation of the RAxML-NG v1.2 search heuristic.

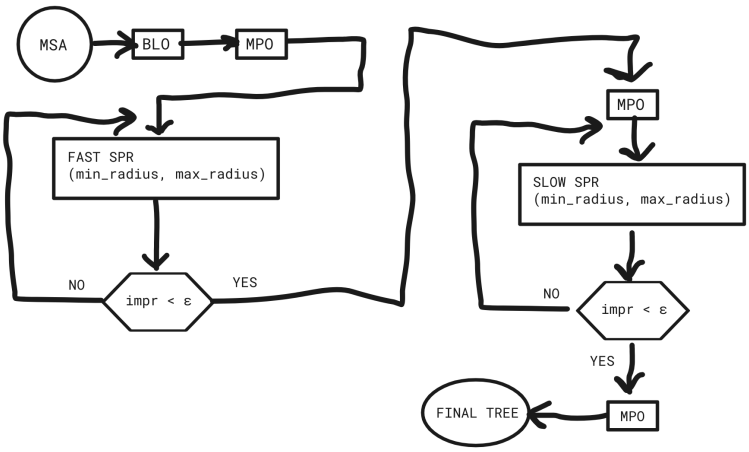


Figure S2. sRAxML-NG heuristic

2. `FAST SPR ROUNDS(min_radius, max_radius)`: This function takes as input the minimum regrafting distance which is always set to 1 and the maximum regrafting distance as given by the `best_radius` parameter and determined via the preceding `AUTODETECT FAST SPR` rounds sequence. In `FAST SPR` rounds, RAxML-NG evaluates each tree topology generated by an SPR move (i.e., each subtree insertion) using the existing branch lengths. The sequence of consecutive `FAST SPR` rounds terminates when the log-likelihood improvement is less than the ϵ-threshold.

3. `SLOW SPR ROUNDS(min_radius, max_radius)`: In analogy to the preceding SPR rounds, this function also takes as input the minimum and maximum regrafting distances, that are initially set to 1 and 5, respectively. The main difference between `FAST` and `SLOW SPR` rounds is that, during slow rounds, RAxML-NG evaluates each tree topology resulting from an SPR move by re-optimizing the lengths of the three adjacent branches around the insertion node. If, after an SPR round, the log-likelihood improvement is below the ϵ-threshold, the minimum and maximum parameters are both increased by 5; otherwise, the next SPR round the minimum and maximum radii remain 1 and 5, respectively. The sequence of `SLOW SPR` rounds is terminated when the log-likelihood improvement is below the ϵ-threshold for five consecutive SLOW SPR round invocations, with increasing radius intervals (1-5, 6-10, 11-15, 16-20, 21-25) .

From the above description, it is evident that RAxML-NG v1.2 can terminate when several distinct conditions are met. One critical factor is the ϵ-threshold parameter. However, termination is also depends on other factors, such as the completion of three distinct sequences of increasingly thorough SPR rounds. Additionally, in the final sequence of SPR rounds, the algorithm terminates only if the log-likelihood improvement does not exceed the ϵ-threshold for five consecutive SPR rounds with increasing minimum and maximum radii values. This thorough heuristic was designed to maximize the likelihood of the inferred topology to the best possible degree. As noted in the main text, Maximum Likelihood (ML) phylogenetic inference is an NP-hard optimization problem, as the number of possible topologies that need to be evaluated grows super-exponentially with the number of taxa in the input Multiple Sequence Alignment (MSA). Consequently, a superficial heuristic has a relatively high probability of becoming stuck in local optima.
In the main text, we argue that there are compelling reasons to not push the optimization to its limits. Therefore, we proposed four distinct stopping criteria that dynamically determine the ϵ parameter based on the noise present in the MSA and the convergence dynamics of the specific search process. The role of the ϵ-threshold in terminating RAxML-NG v1.2 is confounded with other stopping conditions. In addition, multiple unnecessary SPR rounds may be conducted despite the fact that the stopping criteria we propose may already indicate that execution should be halted. This emphasizes the need to simplify the RAxML-NG v1.2 heuristic such that solely the ϵ parameter determines termination.
Figure S2 illustrates the sRAxML-NG heuristic. This heuristic only comprises two sequences of SPR rounds: one conducts `FAST SPR` and the other conducts `SLOW SPR` rounds. In both types of SPR rounds, the `min_radius` parameter is set to 1. The `max_radius` parameter has a fixed default value of 10, but users can modify this value using the `--spr-radius` option in the command line. Both sequences of SPR rounds terminate when the log-likelihood improvement after an SPR round is below the ϵ-threshold. sRAxML-NG uses the same ϵ-threshold as in RAxML-NG v1.2, that is 10.0 log-likelihood units. In the versions with adaptive stopping criteria, the ϵ-threshold is either computed once at the beginning of the tree search (SN-based versions) or after each SPR round (KH-based versions).

3.2. KH-based versions

As mentioned in the main text, certain boundary cases require special handling in the implementation of the KH-based versions. These boundary cases are listed below:

- If the log-likelihood improvement ($NL - L$) after an SPR round is negligible (down to numerical round-off error), it is expected that the standard deviation σ_{KH} will also be negligible. Consequently, the quantity $\frac{NL-L}{\sigma_{KH}\sqrt{s}}$, used as an argument in the cumulative distribution function $\Phi(\cdot)$ of the standard normal distribution (see Sections 2.4 and 2.5 of the main text), becomes undefined, inducing numerical instability. To avoid this, when $NL - L < 10$, the algorithm exits the current sequence of SPR rounds. Essentially, the minimum value that the ϵ-threshold can take in both KH-based methods is 10. This prevents the KH-based methods from assigning a value to the ϵ-threshold that is lower than the one used in RAxML-NG v1.2, which has been empirically determined to suffice for infering high-quality trees.

- If the quantity $\frac{NL-L}{\sigma_{KH}\sqrt{s}}$ exceeds 3.5, we assume that the Null Hypothesis is rejected (in both KH-based versions), as this value is typically the maximum in Z-score tables. In this case, we set $\epsilon := NL - L - 1$, and the algorithm proceeds to the next SPR round.

- Occasionally, $NL - L > 0$ may occur even if the topology remains unchanged before and after the SPR round, due to branch-length optimizations within and after the SPR round, again due to round-off error propagation. In such instances, we set $\epsilon := NL - L + 1$ and the algorithm exits the current SPR round sequence.

4. Commands and bwForCluster Helix details

The stopping criteria are implemented in the `stopping-criteria` branch[1] of the RAxML-NG repository. To disable the adaptive heuristic (Togkousidis et al., 2023) and run the standard version of RAxML-NG v1.2, we use the `--adaptive off` command. sRAxML-NG can be invoked via `--extra simplified-on`. Stopping criteria are selected using the `--stopping-criterion` argument; available options are: `{sn-normal | sn-rell | KH | KH-mult}`, corresponding to the SN-Normal, SN-RELL, simple KH test, and KH test with multiple testing correction criteria. When stopping criteria are specified, the algorithm automatically executes the sRAxML-NG heuristic search, hence `--extra simplified-on` does not need to be specified explicitly. The commands we used to invoke each version and generate our results are the following (e.g., using 10 parsimony starting trees):

Standard RAxML-NG v1.2:

```
./raxml-ng-adaptive --adaptive off --threads 1 --msa {msa} --model {model} --tree pars{10}
--seed 0
```

sRAxML-NG:

```
./raxml-ng-adaptive --threads 1 --msa {msa} --model {model} --tree pars{10} --seed 0
--extra simplified-on
```

SN-Normal version:

```
./raxml-ng-adaptive --threads 1 --msa {msa} --model {model} --tree pars{10} --seed 0
--stopping-criterion sn-normal
```

SN-RELL version:

```
./raxml-ng-adaptive --threads 1 --msa {msa} --model {model} --tree pars{10} --seed 0
--stopping-criterion sn-rell
```

KH version:

```
./raxml-ng-adaptive --threads 1 --msa {msa} --model {model} --tree pars{10} --seed 0
--stopping-criterion KH
```

KH multiple testing correction version:

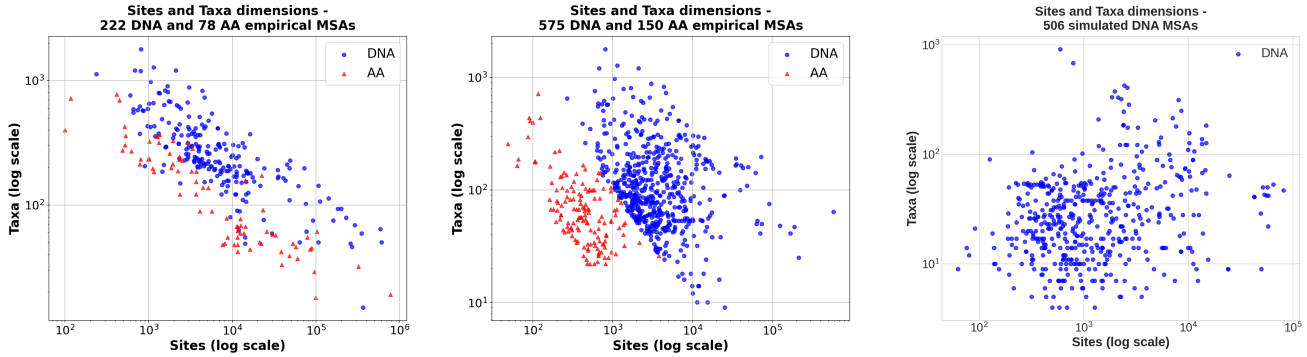
```
./raxml-ng-adaptive --threads 1 --msa {msa} --model {model} --tree pars{10} --seed 0
--stopping-criterion KH-mult
```

In the above commands, since the seed is fixed among all version invocations, and the modifications we introduced only affect the tree optimization stages, we are sure that all versions initiate the exact same trees.
We ran our experiments on the bwForCluster Helix, located at the Heidelberg University Computing Centre. We utilized the `cpu-single` cluster partition, which allocates AMD nodes for submitted jobs. Each AMD node is equipped with two AMD Milan EPYC 7513 Processors, running at 2.60GHz, providing a total of 64 cores per node. The operating system is RedHat Linux.

5. Datasets

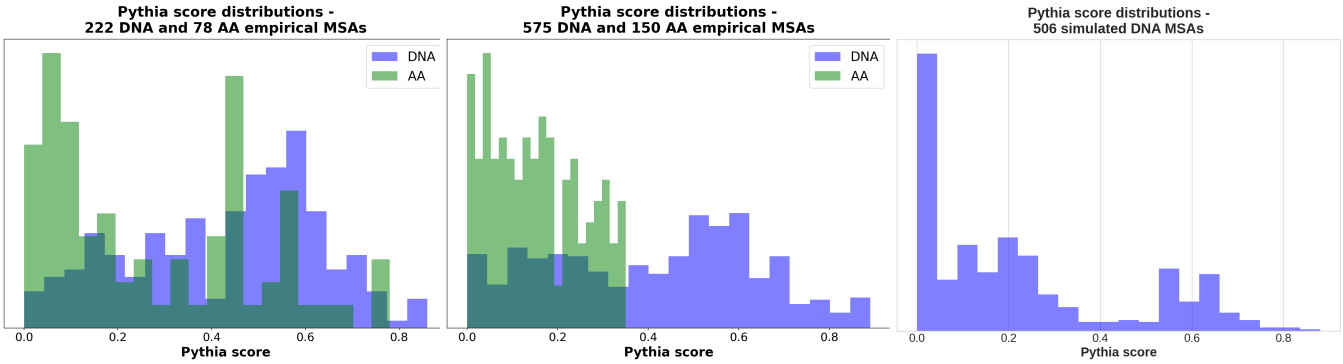
In the main text we analyzed the results from 300 large empirical MSAs, from which 222 are DNA and 78 are AA MSAs. Here, we further present the results for 725 intermediate-sized empirical and 506 simulated MSAs. Among the empirical datasets, 575 are DNA and 150 are AA MSAs. All empirical MSAs were sampled from the TreeBASE databse (Piel et al., 2009) and the simulated MSAs from the datasets used in a recent study on simulated data, conducted by our group and colleagues (Trost et al., 2024). For the analysis of the DNA MSAs we used the GTR+Γ model (Tavaré, 1986), and for AA datasets we used the LG+Γ model (Le and Gascuel, 2008). Information regarding the dimensions of the MSAs are provided in the scatter plots of Figure S3. The Pythia score distributions (Haag et al., 2022) of the sampled datasets are presented in Figure S4. Moreover, in Figures S5 and S6 we illustrate the absolute runtime distributions of RAxML-NG v1.2, on both 300 large, and 725 intermediate-sized empirical MSAs. All datasets used for the sebsequent analyses can be found in the following link: https://cme.h-its.org/exelixis/material/stopping_criteria_data.tar.gz

[1] https://github.com/togkousa/raxml-ng/tree/stopping-criteria



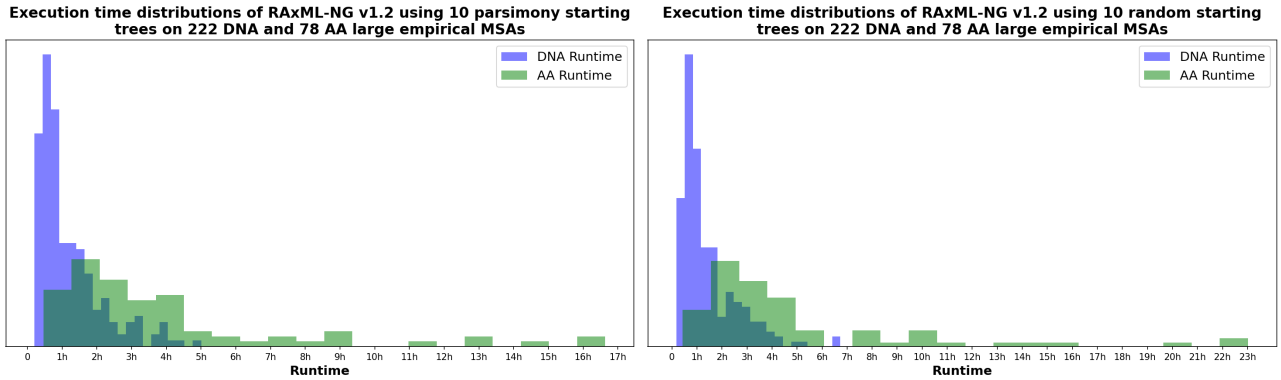
(a) **(b)** **(c)**

Figure S3. Scatter plot showing the number of taxa and sites for each dataset in **(a)** 300 large empirical MSAs **(b)** 725 intermediate-sized empirical MSAs and **(c)** 506 simulated DNA MSAs.



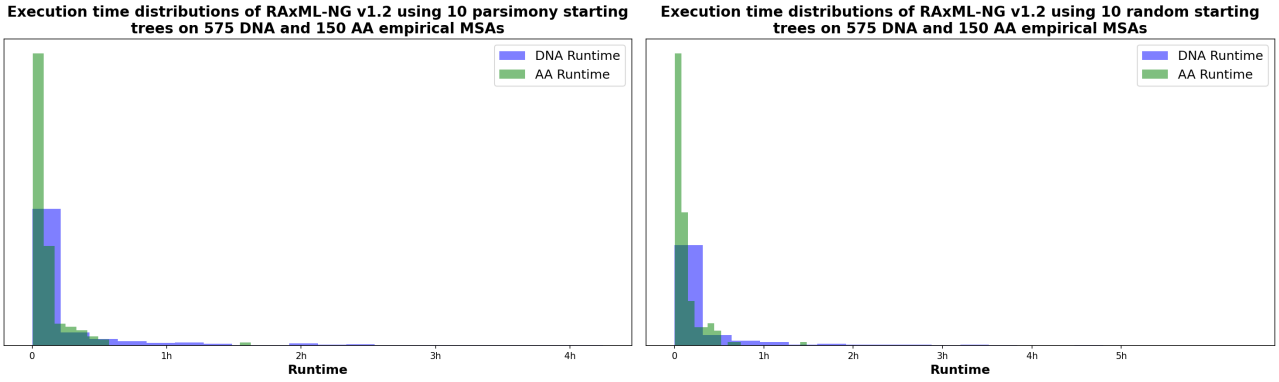
(a) **(b)** **(c)**

Figure S4. Distributions of Pythia scores of **(a)** 300 large empirical MSAs, **(b)** 725 intermediate-sized empirical MSAs, and **(c)** 506 simulated DNA MSAs.



(a) **(b)**

Figure S5. Absolute runtime distributions of RAxML-NG v1.2 on 300 large empirical MSAs, using **(a)** 10 parsimony starting trees, **(b)** 10 random starting trees.



(a) **(b)**

Figure S6. Absolute runtime distributions of RAxML-NG v1.2 on 725 intermediate-sized empirical MSAs, using **(a)** 10 parsimony starting trees, **(b)** 10 random starting trees.

6. Supplementary Results

In this section, we first present the experimental results from 725 intermediate-sized empirical MSAs, followed by results from 506 simulated DNA MSAs. These comparisons also include the SN-based versions.

6.1. Results on 725 intermediate-sized empirical MSAs

Similar to Figure 1 in the main text, Figure S7 illustrates the fraction of datasets, for which each version under comparison inferred x (out of 10) plausible trees, where x represents specific counts or intervals. The left subfigure corresponds to executions using 10 parsimony starting trees, while right one to inferences with 10 random starting trees.

Compared to Figure 1 in the main text, Figure S7 reveals that all Early Stopping versions perform significantly better on intermediate-sized MSAs than on large MSAs when parsimony starting trees are used. In fact, with parsimony starting trees, all Early Stopping versions (including the SN-based versions) infer at least one plausible ML tree in 99.5% of cases. However, the robustness of the SN-based versions is substantially lower than that of sRAxML-NG and KH-based versions. For instance, sRAxML-NG and the KH-based versions infer 10 (out of 10) plausible ML trees in 88%, 86%, and 85% of cases, respectively, with parsimony starting trees. In contrast, the SN-Normal and SN-RELL versions achieve this rate in only 80% of cases. An analogous drop is observed for AA MSAs.

The difference in performance between SN-based and KH-based stopping criteria is more prevalent when random starting trees are used. In this case, sRAxML-NG and KH-based versions infer at least one plausible tree in approximately 94% of DNA MSAs, while the corresponding rates for SN-Normal and SN-RELL are 89% and 87%, respectively. For AA MSAs, the former versions infer at least one plausible tree in 98% of cases, compared to 92% for the SN-based versions. Furthermore, for DNA MSAs, the fraction of datasets where 10 (out of 10) plausible ML trees were inferred is approximately 70% for sRAxML-NG and KH-based versions, while it drops to only 30% for SN-based versions. These results indicate that SN-based versions are less robust and their performance depends heavily on the starting tree type. As already reported, the ϵ-thresholds computed by the SN-based versions on random starting trees are substantially larger than those computed on parsimony starting trees (see Figure S10). This outcome aligns with one of the basic assumptions for quantifying Sampling Noise, that is, that the starting tree should be "reasonable" with respect to the data (see Section 2). Consequently, SN-based methods should not use random starting trees.

There also appears to be a correlation between higher difficulty scores of DNA MSAs and the inability of KH-based methods to infer at least one plausible ML tree (out of 10), regardless of whether random or parsimony starting trees are used. The stripplot in Figure S8 shows that all DNA MSAs (with the exception of three outliers when random starting trees are used) for which KH-based versions inferred zero plausible trees have difficulty scores above 0.5. In contrast, SN-based methods fail to infer plausible trees even in less difficult MSAs when starting from random trees. For AA MSAs, further investigation is required, as all intermediate-sized MSAs sampled have difficulty scores below 0.3.

Figure S9 shows the speedup distributions for all Early Stopping versions relative to RAxML-NG v1.2 (left subfigure) and the speedups of SN- and KH-based versions relative to sRAxML-NG (right subfigure). The SN-based versions generally provide higher average speedups than the KH-based versions. For example, when combined with sRAxML-NG, the average speedup of SN-based versions on DNA MSAs (when parsimony starting trees are used) is $\tilde{3}$.8x, while on AA MSAs is $\tilde{3}$.4x. The corresponding speedups for the KH-standard and KH-multiple testing versions are $\tilde{3}$.4x and $\tilde{3}$.7x for DNA, and $\tilde{2}$.8x and $\tilde{3}$.1x for AA MSAs, respectively. However, as discussed above, this speedup comes at the cost of the inferred tree quality. Aligning with the results in the main text, the average speedup achieved by Early Stopping versions is greater when using parsimony starting trees than when using random starting trees.

Finally, in Figure S10, we present the ϵ-threshold values used by each Early Stopping version for a single tree inference on a randomly selected subset of 50 (out of 575) intermediate-sized, empirical DNA MSAs. Figure S10a shows the ϵ-thresholds when using a parsimony starting tree, while Figure S10b when using a random starting tree. This plot illustrates the range of ϵ values used by each Early Stopping version. For the KH-based methods, where the ϵ value is recomputed after each SPR round and adjusted based on the respective convergence dynamics and noise in the per-site log-likelihood values, only the minimum and maximum values are shown.

SN-based stopping criteria yield larger thresholds, ranging from hundreds to thousands of log-likelihood units, depending on how well the starting tree on which they are computed fits the input data. If we compare the two plots, we observe that the ϵ-thresholds of SN-based methods are substantially larger on a random starting trees. Despite these large convergence thresholds, SN-based methods typically find statistically plausible trees in the majority of cases, especially when parsimony starting trees are used. This is because the vast tree space, which grows super-exponentially with the number of taxa, contains multiple search paths that hill-climbing heuristics can follow to reach local optima. SPR moves, being highly effective topological moves, often result in substantial improvements in the log-likelihood score. The worse the initial log-likelihood score before the SPR round, the greater the subsequent

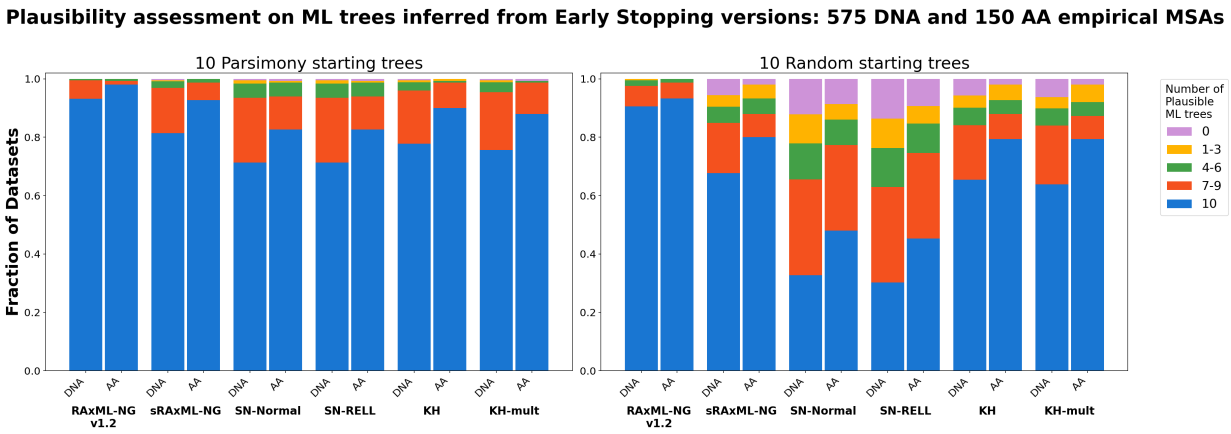


Figure S7. Plausibility test results for ML trees inferred by RAxML-NG v1.2 and early stopping versions across 575 DNA and 150 AA empirical MSAs. For each dataset, all RAxML-NG versions conduct 10 independent tree inferences using the same set of parsimony (left subfigure) or random (right subfigure) starting trees. We evaluated the inferred ML trees using the AU test implemented in the CONSEL tool, considering ML trees with a p-value > 0.05 as plausible.

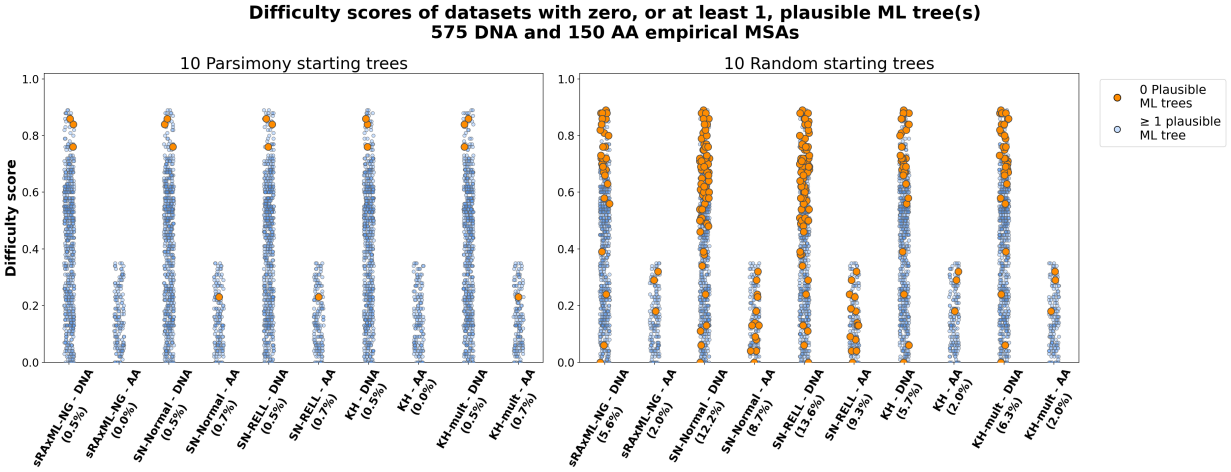


Figure S8. Stripplot illustrating correlation between the increased dataset difficulty scores and the inability of Early stopping versions to infer at least one plausible ML tree, on 575 DNA and 150 AA intermediate-sized empirical MSAs. Each point represents a single dataset, with difficulty scores shown on the y-axis. The columns on the x-axis represent specific combinations of Early Stopping versions (sRAxML-NG, SN-Normal, SN-RELL, KH, and KH-multiple testing) and dataset types (DNA, AA). The left subfigure corresponds to parsimony starting trees, while the right subfigure corresponds to random starting trees. Points where zero plausible ML trees were inferred by a given version for a dataset are drawn more emphatically, while points for datasets where at least one (out of 10) plausible ML tree was inferred are shown with a lighter mark.

improvement after the SPR round. Consequently, the large ϵ-thresholds set by SN-based methods are frequently exceeded by these improvements (especially by the improvement of the first SPR round). When starting from a parsimony tree, one or two SPR rounds are often sufficient for Early Stopping versions to infer trees that are statistically equivalent to the best ML tree inferred by RAxML-NG v1.2.

For the KH-based versions, the KH method with multiple testing correction yields higher ϵ-thresholds than the simple KH method, especially during the initial SPR rounds, where hundreds of SPR topologies (N_t) improve the log-likelihood score of the input tree (see eq. (2) in the main text). This illustrates the advantage of a dynamic approach that adapts the ϵ value as a function of the data and intermediate trees. Moreover, we observe that the minimum ϵ value for both versions is 10. This constraint arises from the necessary numerical restrictions in the implementation of the KH test (see Section 3.2) to ensure that the ϵ value computed by the KH-based methods is not lower than the threshold used in RAxML-NG v1.2, which is 10.

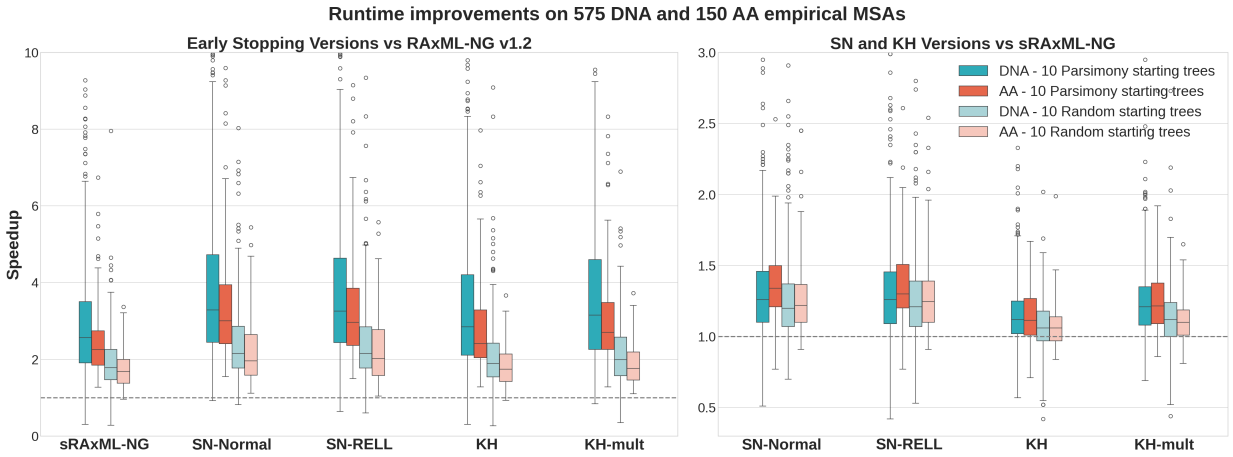
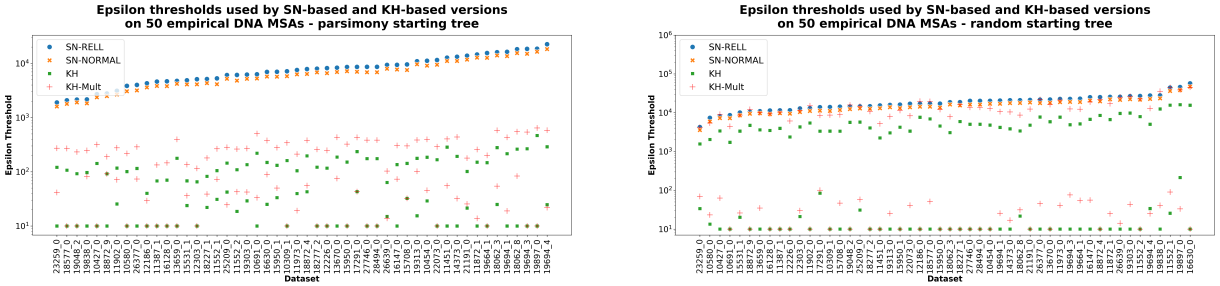


Figure S9. Speedup distributions of Early Stopping versions relative to RAxML v1.2 (left subfigure), and of the SN-, and KH-based versions relative to sRAxML-NG (right subfigure), on 575 DNA and 150 AA intermediate-sized empirical MSAs. The speedups refer to runtimes measured on sequential executions using 10 parsimony or 10 random starting trees. The dashed line at the bottom of each subfigure corresponds to a speedup of 1x.



(a) **(b)**

Figure S10. Convergence ϵ-thresholds used by Early Stopping versions for a single tree inference on 50 intermediate-sized empirical DNA MSAs, using a parsimony starting tree. The datasets are sorted based on the ϵ-thresholds determined in the SN-RELL version. For the KH-based methods, we only plot the maximum and minimum ϵ-thresholds used throughout the tree inference. **(b)** Same description as in (a), however each version uses a single random starting tree instead of a parsimony starting tree. The datasets on the x-axis are the same in Figures (a) and (b).

6.2. Results on 506 simulated DNA MSAs

In this final section, we present benchmarking results for the Early Stopping RAxML-NG versions on 506 simulated MSAs. Simulated MSAs are generally easier to analyze and tend to converge faster due to their clearer and stronger phylogenetic signal. Thus, in this section, we provide an overview of the plausibility assessment and runtime improvement plots without extensive numerical details, as they are of limited relevance. The primary focus of this section is on the topological accuracy plots, presented at the end.

Figure S11 shows the plausibility assessment plot, while Figure S12 shows the speedup distributions plot, as described above for empirical MSAs. In the plausibility assessment, all stopping criteria appear to infer high-quality trees on simulated MSAs when using parsimony starting trees. However, with random starting trees, the performance of SN-based methods decreases, despite the reduced complexity of simulated MSAs. In terms of speedups, parsimony starting tree inferences yield substantially higher average speedups than those initiated with random starting trees. Moreover, SN-based methods demonstrate faster average runtimes compared to KH-based versions.

Regarding topological accuracy, we compare the best ML trees inferred from all versions with the reference tree using the Robinson–Foulds (RF) distance (Robinson and Foulds, 1981). Here, the reference tree is the "true" tree used to generate the simulated sequences. Figure S13 illustrates the distributions of RF distances between the best ML trees inferred from each RAxML-NG version and the reference tree topologies, when parsimony or random starting trees are used. The plot demonstrates that for simulated data, there is no significant difference in the distributions of relative RF distances between RAxML-NG v1.2 and the Early Stopping versions in terms of topological accuracy. In fact, all RF distance distributions exhibit the same average value of 0.128 for parsimony starting trees, and 0.132 for random starting trees, respectively.

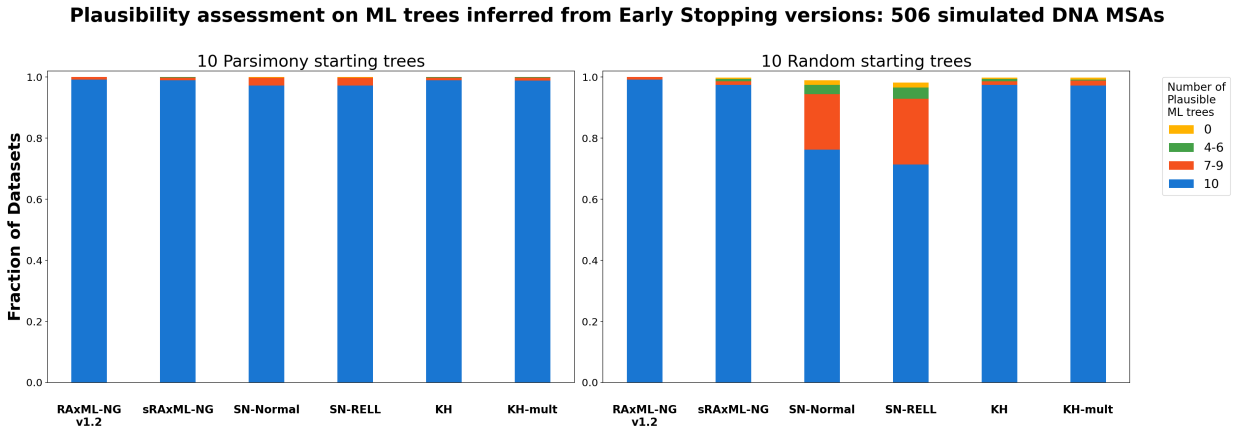


Figure S11. Plausibility test results for ML trees inferred by RAxML-NG v1.2 and early stopping versions across 506 simulated DNA MSAs. For each dataset, all RAxML-NG versions conduct 10 independent tree inferences using the same set of parsimony (left subfigure) or random (right subfigure) starting trees. We evaluated the inferred ML trees using the AU test implemented in the CONSEL tool, considering ML trees with a p-value > 0.05 as plausible.

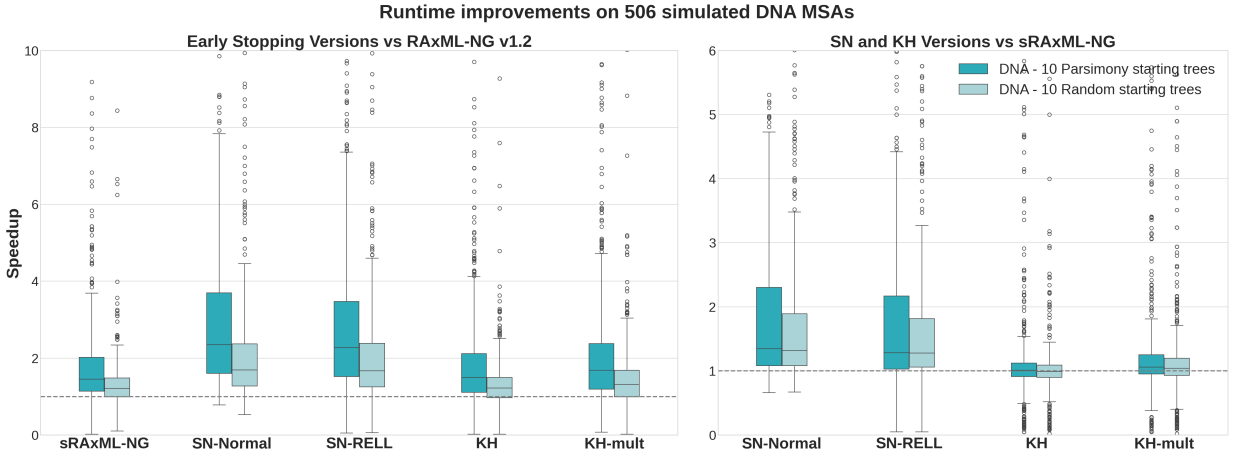


Figure S12. Speedup distributions of Early Stopping versions relative to RAxML v1.2 (left subfigure), and of the SN-, and KH-based versions relative to sRAxML-NG (right subfigure), on 506 simulated DNA MSAs. The speedups refer to runtimes measured on sequential executions using 10 parsimony or 10 random starting trees. The dashed line at the bottom of each subfigure corresponds to a speedup of 1x.

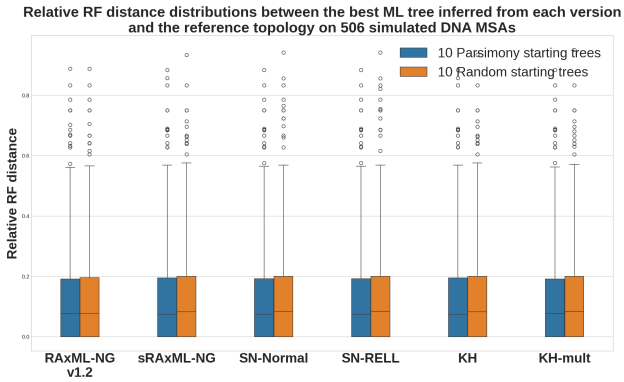


Figure S13. Distributions of relative RF distances between the best ML trees inferred from each RAxML-NG version and the corresponding reference tree topologies, where each version conducted 10 independent tree inferences on 506 simulated DNA MSAs, using parsimony or random starting trees. The reference tree topology is the "true" tree used to generate the sequences.

Acknowledgment

This study was financially supported by the Klaus Tschira Foundation and by the European Union (EU) under Grant Agreement No 101087081 (Comp-Biodiv-GR). O.G. was supported by PRAIRIE (ANR-19-P3IA-0001). The authors acknowledge support by the state of Baden-Württemberg through bwHPC and the German Research Foundation (DFG) through grant INST 35/1597-1 FUGG.

References

Haag, J., Höhler, D., Bettisworth, B., and Stamatakis, A. (2022). From easy to hopeless—predicting the difficulty of phylogenetic analyses. *Molecular Biology and Evolution*, 39(12):msac254.

Haag, J., Hübner, L., Kozlov, A. M., and Stamatakis, A. (2023). The free lunch is not over yet—systematic exploration of numerical thresholds in maximum likelihood phylogenetic inference. *Bioinformatics Advances*, 3(1):vbad124.

Kishino, H., Miyata, T., and Hasegawa, M. (1990). Maximum likelihood inference of protein phylogeny and the origin of chloroplasts. *Journal of Molecular Evolution*, 31:151–160.

Kozlov, A. *Models, Optimizations, and Tools for Large-Scale Phylogenetic Inference, Handling Sequence Uncertainty, and Taxonomic Validation.* PhD thesis, Karlsruhe Institute of Technology, (2018).

Le, S. Q. and Gascuel, O. (2008). An improved general amino acid replacement matrix. *Molecular biology and evolution*, 25(7):1307–1320.

Piel, W. H., Chan, L., Dominus, M. J., Ruan, J., Vos, R. A., and Tannen, V. (2009). TreeBASE v. 2: A Database of Phylogenetic Knowledge. *e-BioSphere 2009*.

Robinson, D. and Foulds, L. (1981). Comparison of phylogenetic trees. *Mathematical Biosciences*, 53(1):131–147. doi: https://doi.org/10.1016/0025-5564(81)90043-2.

Tavaré, S. (1986). Some probabilistic and statistical problems in the analysis of dna sequences. *Lect Math Life Sci (Am Math Soc)*, 17:57–86.

Togkousidis, A., Kozlov, O. M., Haag, J., Höhler, D., and Stamatakis, A. (2023). Adaptive raxml-ng: Accelerating phylogenetic inference under maximum likelihood using dataset difficulty. *Molecular Biology and Evolution*, 40(10):msad227.

Trost, J., Haag, J., Höhler, D., Jacob, L., Stamatakis, A., and Boussau, B. (2024). Simulations of sequence evolution: how (un) realistic they are and why. *Molecular biology and evolution*, 41(1): msad277.