Software for Systematics and Evolution

Syst. Biol. 00(0):1–9, 2025

https://doi.org/10.1093/sysbio/syaf043
Advance Access publication May 30, 2025

Accelerating Maximum Likelihood Phylogenetic Inference via Early Stopping to Evade (Over-)optimization

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

[1]*Computational Molecular Evolution Group, Heidelberg Institute for Theoretical Studies, Schloß-Wolfsbrunnenweg 35, Heidelberg 69118, Germany*
[2]*Institute of Theoretical Informatics, Karlsruhe Institute of Technology, Kaiserstraße 12, Karlsruhe 76131, Germany*
[3]*Institut de Systématique, Evolution, Biodiversité (ISYEB, UMR7205-CNRS, Muséum National d'Histoire Naturelle, SU, EPHE, UA), 75005 Paris, France*
[4]*Biodiversity Computing Group, Institute of Computer Science, Foundation for Research and Technology, N. Plastira 100, Heraklion 70013, Greece*
[*]*Correspondence to be sent to: Computational Molecular Evolution Group, Heidelberg Institute for Theoretical Studies, Schloß-Wolfsbrunnenweg 35, Heidelberg 69118, Germany; E-mail: anastasis.togkousidis@h-its.org*

Received 16 January 2025; reviews returned 26 March 2025; accepted 29 May 2025
Associate Editor: Robert Thomson

Abstract.—Maximum likelihood–based phylogenetic inference constitutes a challenging optimization problem. Given a set of aligned input sequences, phylogenetic inference tools strive to determine the tree topology, the branch lengths, and the evolutionary model parameters that maximize the phylogenetic likelihood function. However, there exist compelling reasons to not push optimization to its limits, by means of early, yet adequate stopping criteria. Because input sequences are typically subject to stochastic and systematic noise, caution is warranted to prevent overoptimization and the risk of overfitting the model to noisy data. To address this, we integrate the Kishino–Hasegawa (KH) test into RAxML-NG as a reliable and fast-to-compute Early Stopping criterion to effectively limit excessive and compute-intensive overoptimization. Initially, we introduce a simplified heuristic tree search strategy in RAxML-NG (sRAxML-NG) as an underlying method for Early Stopping. Subsequently, we use the KH test in combination with sRAxML-NG to statistically assess the significance of differences between intermediate trees prior to and after major optimization steps. The tree search terminates early when improvements are statistically insignificant. We also propose an extension to the standard KH test that allows to correct for multiple testing, which maintains accuracy while achieving even higher speedups. For benchmarking, we use 300 large representative empirical data sets from TreeBASE. For 98% of the DNA data sets, all Early Stopping methods we introduce infer trees that are statistically equivalent to those inferred from RAxML-NG v1.2. For AA data sets, the fraction of data sets where sRAxML-NG, KH, and the KH-multiple testing versions infer statistically equivalent trees is 96%, 95%, and 92%, respectively. In conjunction with sRAxML-NG, the average speedup achieved by the KH-multiple testing version is 5× for DNA and 3.9× for protein data sets compared with RAxML-NG v1.2. We implemented our stopping criteria in RAxML-NG, which is available under GNU GPL at https://github.com/togkousa/raxml-ng/tree/stopping-criteria. [Early Stopping; maximum likelihood; stopping criteria.]

Phylogenetic inference addresses the problem of finding a binary tree that optimally represents the evolutionary relationships among biological sequences, given in the form of a multiple sequence alignment (MSA). There exist a plethora of optimality criteria to infer trees, such as the maximum parsimony (Fitch 1971) or the maximum likelihood (ML) (Felsenstein 1981) method. The NP-hard ML inference (Roch 2006) is a computationally intensive optimization problem and requires finding the optimal tree topology, branch lengths, and evolutionary model parameters that maximize the likelihood score (Yang 2014).

Tools such as RAxML (Stamatakis 2014), RAxML-NG (Kozlov et al. 2019), IQ-TREE (Minh et al. 2020), and PhyML (Guindon et al. 2010) deploy hill-climbing heuristics (St. John 2016) to find a "good" solution, which is typically a local optimum due to the inherent computational complexity of ML. RAxML, RAxML-NG, and IQ-TREE implement strategies that explore multiple tree search trajectories, although their specific approaches vary. By default, RAxML and RAxML-NG conduct 20 independent ML tree inferences that are initiated on 10 parsimony and 10 random starting trees. As an output, they provide the best ML tree inferred by these independent inferences. In contrast, IQ-TREE performs a single tree search while maintaining a pool of 100 candidate trees, initially populated with 99 parsimony and 1 BioNJ (Gascuel 1997) trees. A recent benchmark study (Liu et al. 2024) recommends performing at least 10 independent ML tree inferences when using RAxML-NG or IQ-TREE, which, in the case of IQ-TREE, translates into executing independent runs. PhyML, by comparison, follows a single search trajectory, starting from a BioNJ tree and employs a form of simulated annealing to optimize the topology. This raises the important question of whether phylogenetic inference tools should prioritize more exhaustive heuristics with fewer starting trees, or favor a broader exploration of tree space by conducting multiple, yet potentially less thorough, independent searches. In this study, we advo-

cate for the latter approach. Specifically, we propose that when using RAxML-NG with multiple (e.g., 10 parsimony) starting trees, while employing a strategically refined heuristic, one can achieve comparable accuracy while substantially reducing runtime and associated CO_2 emissions. In the following, we outline the rationale for this approach.

Sequences are subject to noise (Townsend et al. 2012) stemming from both stochastic and systematic sources. Evolution, which is stochastic in nature, induces randomness in the sequences that reflects deviations between the observed data distributions and the theoretical expectations (Hillis and Huelsenbeck 1992; Münkemüller et al. 2012). Upon that, sampling noise is superimposed, as phylogenetic analyses typically rely on sequences that merely represent a small fraction of the corresponding genomes. This raises concerns about their adequacy in approximating the underlying distribution. Systematic noise, which is more prevalent in empirical MSAs, is introduced, for instance, by sequencing errors (Moutsopoulos et al. 2021), alignment errors (Dress et al. 2008), homoplasy (Rokas and Carroll 2006), and gene tree–species tree discordance in supermatrix MSAs (Maddison 1997).

This intrinsic noise in sequence data increases the risk of overfitting because exhaustive optimization may capture both the phylogenetic signal and the noise. A straightforward solution is to employ more superficial, yet substantially faster, tree search heuristics, such as those implemented in FastTree (Price et al. 2010). Benchmarking studies, however, have demonstrated that tools using more thorough heuristics (IQ-TREE, RAxML-NG) yield trees with higher statistical plausibility and bootstrap support (Lemoine et al. 2018; Höhler et al. 2022). Therefore, our main objective in this work is to devise likelihood-based inference stopping criteria that circumvent unnecessary optimization steps, *without* compromising the quality of the inferred trees.

RAxML-NG, PhyML, and FastTree use fixed log-likelihood improvement thresholds (ϵ) to determine the convergence of optimization steps. Specifically, the default threshold for RAxML-NG v1.2 is $\epsilon = 10$ (Haag et al. 2023). These arbitrary ϵ-thresholds may not adequately reflect the phylogenetic signal-to-noise ratio of a specific, given MSA. They might also not capture the convergence dynamics of individual tree inferences.

The predecessor of IQ-TREE, IQPNNI (Vinh and Von Haeseler 2004), employed a sophisticated approach to terminate tree searches by estimating the probability of having found the optimal tree by monitoring log-likelihood improvements over search algorithm iterations and halting the search once this probability attained 95% (using a Weibull distribution of recorded values). This method was later replaced by a more straightforward rule in IQ-TREE. IQ-TREE terminates the search if no better tree is found after a fixed number of iterations (default $N := 100$).

Here, we initially introduce a simplified RAxML-NG heuristic called sRAxML-NG, which serves as an underlying Early Stopping method upon which we develop our stopping criteria. The rationale for implementing this simplified version is detailed in the "Materials and Methods" section. Next, we integrate the Kishino–Hasegawa (KH) test (Kishino and Hasegawa 1989) into sRAxML-NG as a reliable, fast-to-compute, method for Early Stopping. It dynamically adjusts the log-likelihood improvement threshold (ϵ) value based on the convergence dynamics of each individual ML tree inference. We further extend the standard KH test by employing multiple testing correction in order to improve the p-value approximation. We implemented these stopping criteria in RAxML-NG v1.2. Our criteria can also be seamlessly integrated into other phylogenetic inference tools that use numerical convergence thresholds such as PhyML and FastTree. Our experimental results on 300 large representative empirical MSAs (222 DNA and 78 AA) sampled from TreeBASE (Piel et al. 2009) show that, when tree inferences are initiated with parsimony starting trees, for 98% of the DNA MSAs, Early Stopping versions yield trees that are statistically equivalent to those inferred without Early Stopping. For amino acid (AA) MSAs, the corresponding statistical equivalence rates for sRAxML-NG, KH, and KH-multiple testing versions are 96%, 93%, and 92%, respectively. Combined with sRAxML-NG, the KH-multiple testing version yields an average speedup of 5× for DNA and 3.9× for AA data sets, compared with RAxML-NG v1.2. All MSAs we used for our experiments are available for download at https://doi.org/10.5061/dryad.8gtht76zz

Materials and Methods

In this section, we first introduce sRAxML-NG and explain the rationale for this simplification, which is necessary for integrating KH-based stopping criteria. We then describe the application of the KH test within the RAxML-NG framework. Specifically, the KH test is used to statistically assess significant log-likelihood differences between tree topologies *prior to* and *after* each subtree prune and regraft (SPR) round, which constitutes the core topological optimization block in RAxML-NG. The optimization process terminates early when improvements between such subsequent tree topologies are statistically insignificant. Further, we extend the KH test by a multiple testing correction to more accurately approximate p-values.

Note that we also experimented with stopping criteria that attempt to quantify the MSA sampling noise. Although the performance of these criteria is substantially inferior to the methods we present here, we nonetheless describe the approach and document the negative results in our GitHub repository.

Simplified RAxML-NG

When the improvement in log-likelihood drops below a predefined, fixed ϵ-threshold during the standard RAxML-NG tree search algorithm, the search does not terminate immediately, to avoid getting stuck in a local optimum. Instead, RAxML-NG continues the tree search by adjusting SPR-specific parameters for the subsequent SPR rounds. Specifically, it increases the maximum SPR radius or switches from superficial (FAST) to more thorough (SLOW) SPR rounds, after conducting a round of model parameter optimizations (MPOs; see Supplementary Fig. S1). The main difference between FAST and SLOW SPR rounds is that, during FAST SPR rounds, RAxML-NG evaluates each tree topology generated by an SPR move using the existing branch lengths, whereas in SLOW SPR rounds, the lengths of the three branches adjacent to the subtree insertion node are being reoptimized. Further, full branch length optimization (BLO) rounds are conducted after each FAST and SLOW SPR round and prior to MPO. Hence, the standard RAxML-NG tree search only terminates after several unsuccessful SLOW SPR rounds with distinct SPR parametrizations (Kozlov 2018).

Within this complex optimization framework, the impact of convergence thresholds is challenging to assess, as the search does not terminate immediately when the score improvement drops below the predefined, fixed ϵ value. As a consequence, it is difficult to objectively evaluate the impact of distinct stopping criteria that directly modify the ϵ value. Therefore, we initially simplified and accelerated the complex standard search strategy in RAxML-NG, which already yielded an initial Early Stopping implementation. We call this search heuristic *Simplified RAxML-NG (sRAxML-NG)*. sRAxML-NG conducts a series of FAST SPR rounds followed by a series of SLOW SPR rounds. Each series terminates when the log-likelihood improvement drops below a constant default threshold of $\epsilon := 10$ log-likelihood units, as in RAxML-NG v1.2. Unlike the standard search strategy, we simply execute the FAST and SLOW SPR rounds with a fixed maximum subtree reinsertion radius parameter of 10. We conduct full BLO rounds after each FAST and each SLOW SPR round. We invoke MPO rounds on the initial tree topology (after BLO), in-between the series of FAST and SLOW SPR rounds, and on the final tree (before termination). This simplified heuristic substantially accelerates inferences. It is a noteworthy by-product of our study as sRAxML-NG was predominantly developed to seamlessly assess the accuracy of dynamic adaptation of ϵ-convergence thresholds. The KH-based methods described later build upon the sRAxML-NG heuristic by dynamically adjusting the ϵ-thresholds after each SPR round. We provide further details and workflow diagrams for the standard RAxML-NG and the sRAxML-NG heuristics in the Supplementary Material.

KH Test in RAxML-NG

We deploy the KH test to compare subsequent best trees prior to and after each SPR round. Based on the per-site log-likelihood differences of the two trees, an ϵ value can be derived, which depends on the observed variations of the per-site log-likelihood distributions that correspond to the SPR round under consideration. Consequently, this dynamic ϵ-threshold varies over SPR rounds and automatically adapts to the convergence dynamics of each independent ML tree search.

The KH test is a paired test that compares the log-likelihood of each site in an MSA for two different phylogenetic trees. Typically, the KH test assesses whether the likelihood difference between two trees is significant for a given MSA. To avoid optimization bias, it is generally advised that neither tree being compared is inferred on the MSA under study (Markowski and Susko 2024). This is because, if one tree represents a commonly accepted hypothesis about the studied species, whereas the other tree has been inferred on the given MSA, the KH test would inherently favor the latter. In our application, *both* trees are inferred using the studied MSA, thus eliminating bias in favor of one tree over the other. However, because we test many trees during an SPR round, we are faced with multiple testing.

In the following, we first describe how we apply the standard KH test. Then, we outline a fast heuristic solution for multiple testing correction that has negligible time and memory overhead. Several versions of the KH test are described in Goldman et al. (2000). We adopt the fastest version, as implemented in PAUP* (Swofford 2003) and PHYLIP (Felsenstein 2005), which yields results that are comparable to the more computationally intensive resampling estimated log-likelihood (RELL; Kishino et al. 1990) method. Moreover, the faster approach substantially simplifies the multiple-test implementation.

For an MSA with s sites, suppose that the per-site log-likelihood vectors before ($\mathbb{L}$) and after ($\mathbb{L}'$) any given SPR round are

$$\mathbb{L} = (\mathcal{L}_1, \mathcal{L}_2, ..., \mathcal{L}_s),$$
$$\mathbb{L}' = (\mathcal{L}'_1, \mathcal{L}'_2, ..., \mathcal{L}'_s),$$

where

$$L := \sum_{i=1}^{s} \mathcal{L}_i,$$
$$L' := \sum_{i=1}^{s} \mathcal{L}'_i.$$

The KH test can be applied as follows:

(1) Compute the distribution of the (paired) differences between the vector coordinates $(\mathcal{L}'_i - \mathcal{L}_i)$, $i = 1, 2, ..., s$. These differences can be either positive or negative, even if $L' - L > 0$.

(2) Compute the standard deviation σ_{KH} of the distribution of the s differences. Assuming that the per-site differences are i.i.d., the standard deviation of the difference $(L' - L)$ is $\sigma_{KH} \cdot \sqrt{s}$.
(3) In the fast version of the KH test (Goldman et al. 2000), we assume that the difference $(L' - L)$ is normally distributed. This is a consistent approximation due to the law of large numbers, and because we are summing over log-likelihood differences and not the log-likelihood values. In this respect, under the null hypothesis ($L' \equiv L$), the term $\frac{L'-L}{\sigma_{KH}\sqrt{s}}$ follows a normal distribution $N(0, 1)$ with a cumulative distribution function $\Phi(\cdot)$. We continue the tree search (i.e., we reject the null hypothesis that $L' \equiv L$) with 95% confidence if

$$1 - \Phi\left(\frac{L' - L}{\sigma_{KH}\sqrt{s}}\right) \leq 0.05 \Leftrightarrow$$
$$\epsilon = 1.645 \cdot \sigma_{KH}\sqrt{s} \quad (1)$$

KH Test with Multiple Correction

Because the standard KH test does not incorporate adjustments for multiple testing, it may unnecessarily prolong the tree search. An SPR round comprises multiple iterations, where each iteration involves pruning and regrafting each subtree of the current tree. During each SPR iteration, numerous potential regrafting topologies are evaluated (i.e., topologies generated by SPR moves performed on a given subtree within a specified radius), and only the best move is ultimately selected. This is a situation that requires accounting for the multiplicity of the tests. However, the majority of the candidate topologies we generate during an SPR round exhibits a decreased fit (lower likelihood) to the input MSA, especially toward the end of the tree search when we are close to the (local) optimum. To account for this, we propose a solution that does not require additional calculations and data storage for each SPR move that is performed during an SPR round, as required by more sophisticated approaches (e.g., Holm–Bonferroni, approximately unbiased [AU]). Instead, during an SPR round, we track the number of SPR topologies, denoted as N_t, which improve the likelihood relative to the best tree identified prior to the current iteration. This count is accumulated across all iterations within the SPR round. After the end of the SPR round, we apply a Bonferroni correction to the p-value of the best tree generated by the SPR round. The adjusted ϵ-threshold is then derived as follows:

$$1 - \Phi\left(\frac{L' - L}{\sigma_{KH}\sqrt{s}}\right) \leq \frac{0.05}{N_t} \Leftrightarrow$$
$$\epsilon = \sigma_{KH}\sqrt{s} \cdot \Phi^{-1}\left(1 - \frac{0.05}{N_t}\right). \quad (2)$$

For both the KH-simple and KH-multiple testing methods, the convergence ϵ-threshold is always defined as the maximum of the KH threshold and the fixed value of 10 (default in RAxML-NG v1.2) for the following reasons. First, it ensures that KH-based versions terminate earlier than standard RAxML-NG without unnecessarily prolonging the search. Second, it avoids numerical issues that may arise when the log-likelihood difference between the two trees being compared is very small. In such cases, the KH test, or any statistical test for that matter, should not be used for comparison. Typically, the ϵ-threshold that terminates the KH-based searches exceeds 10, yielding this edge case rare. We provide further details about this in the Supplementary Material.

Results

In this section, we present the results from our experiments on 222 DNA and 78 AA large empirical as well as representative MSAs sampled from the TreeBASE database. Here, "large" refers to data sets with either a high number of taxa, ranging from tens to thousands, or a large number of sites, ranging from hundreds up to nearly a million (see Supplementary Fig. S3). For these data sets, sequential execution times of RAxML-NG v1.2 range from 10 min to 23 h (see Supplementary Fig. S5). The selected empirical MSAs cover the full difficulty score spectrum as predicted by the Pythia tool (Haag et al. 2022) and are hence representative. Further, in the Supplementary Material we provide additional results on 1076 simulated MSAs, which generally converge faster, yet allow to assess the impact of our stopping criteria on the topological accuracy with respect to the true tree.

As mentioned in the "Materials and Methods" section, KH-based stopping criteria have been integrated into sRAxML-NG. Thus, our experiments assess three methods: (a) sRAxML-NG alone (with $\epsilon = 10$), (b) KH-simple, and (c) KH-multiple testing versions, which combine sRAxML-NG with stopping criteria. We refer to the sRAxML-NG version and KH-based versions as the *Early Stopping* versions/criteria. In our experimental setup, we compare all Early Stopping versions with RAxML-NG v1.2, as well as the KH-based versions against sRAxML-NG. For each MSA and program version, we conduct two searches, one using 10 distinct parsimony starting trees, and a second one using 10 distinct random starting trees. Therefore, each program version conducts 10 independent tree inferences per execution. All RAxML-NG versions use the exact same set of random and parsimony starting trees, thereby ensuring identical initial conditions across experiments and providing a fair basis for comparison. We refer to the 10 output trees from each execution as the inferred *ML trees*.

Our benchmarking results include a plausibility assessment of ML trees inferred by all RAxML-NG versions, as well as execution time comparisons. For sim-

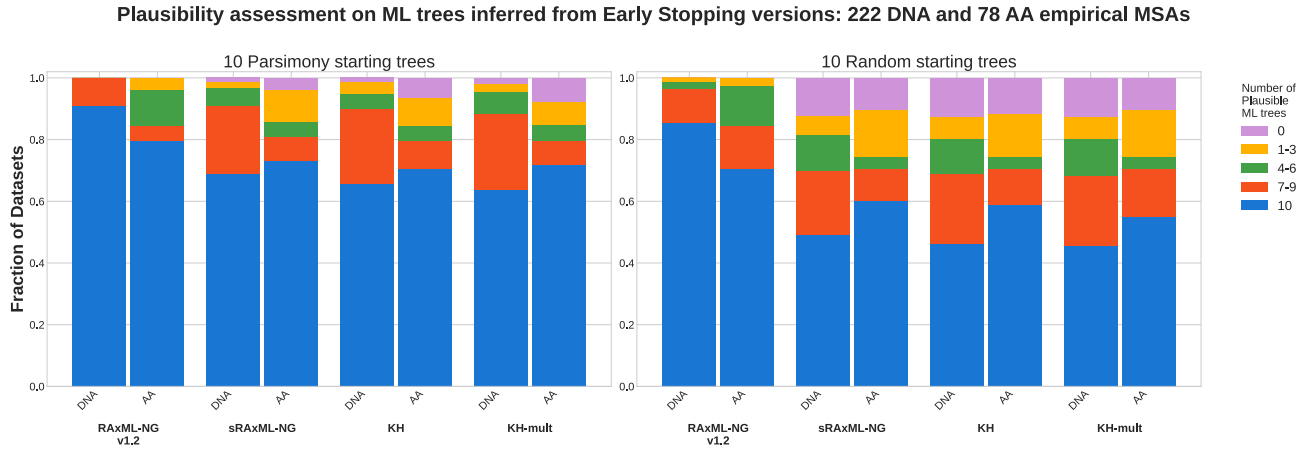

FIGURE 1. Plausibility test results for ML trees inferred by RAxML-NG v1.2 and Early Stopping versions across 222 DNA and 78 AA empirical MSAs. For each data set, all RAxML-NG versions conduct 10 independent tree inferences using the same set of parsimony (left subfigure) or random (right subfigure) starting trees. We evaluated the inferred ML trees using the AU test (over 40 trees, for each type of starting trees) implemented in the CONSEL tool, considering ML trees with a p-value ≥ 0.05 as being plausible.

ulated MSAs, we further conduct topological accuracy comparisons between the best ML trees inferred by each version against the true reference tree (see Supplementary Fig. S11). To assess plausibility, we collect *all* ML trees inferred by all RAxML-NG versions, for the same starting tree type (i.e., one set of ML trees for parsimony starting tree and one for random starting trees), and conduct the AU test (Shimodaira 2002) using the CONSEL tool (Shimodaira and Hasegawa 2001). We consider all ML trees whose p-value is greater than 0.05 as being *plausible*.

To obtain accurate runtime measurements, we executed all program versions sequentially. We have nonetheless already implemented, tested, and released an efficient parallelization of our stopping criteria.

Plausibility Assessment

Figure 1 illustrates the fraction of data sets for which each version under comparison inferred n (out of 10) plausible trees, where n represents specific counts or intervals. The left subfigure corresponds to executions using 10 parsimony starting trees, and the right one to 10 random starting trees. A substantial part of the following analysis focuses on the fraction of MSAs for which a given version infers at least one ($n = 1$) plausible ML tree. In fact, this is a minimal, yet sufficient criterion to ascertain statistical equivalence between versions. When at least one plausible tree is found, the output (i.e., the best ML tree inferred by the given version) is statistically equivalent to the best tree found by the four versions under comparison. We compared the inferred ML trees on random and parsimony starting trees independently, as described in the caption. In the Supplementary Material (Fig. S6), we also jointly compared the ML trees resulting from both starting tree types. The latter approach resembles the default execution of RAxML-NG, where 10 random and 10 parsimony starting trees are used for each version under comparison. The results from the two plausibility assessment frameworks exhibit negligible differences. This indicates that random starting trees do not provide any substantial advantage on our test data sets, although they do induce a substantial runtime overhead (see below for a discussion of the results).

The results in Figure 1 indicate that Early Stopping versions perform substantially better in terms of output ML tree quality when parsimony starting trees are used. This outcome is expected, as parsimony trees provide a reasonable, non-random starting point for searches. Starting from random trees is suboptimal (in fact, the worst possible choice), and it increases the chances of terminating the search prematurely when applying Early Stopping. Specifically, the fraction of DNA MSAs for which the Early Stopping versions infer at least one (out of 10) plausible ML trees, when searches are initiated on parsimony starting trees, is 98.2% across all versions. For AA MSAs, these fractions with parsimony starting trees are 96.2% for sRAxML-NG, 93.6% for KH, and 92.4% for KH-multiple testing version. In contrast, when random starting trees are being used, the corresponding fractions are ∼87% for DNA and ∼90% for AA MSAs across all versions, respectively. RAxML-NG v1.2 infers at least one plausible tree across all data sets (100%) when using either parsimony or random starting trees.

Additionally, in executions with parsimony starting trees, the fractions of DNA MSAs for which all output ML trees (10 out of 10) are plausible are 91% for RAxML-NG v1.2, 69% for sRAxML-NG, and ∼65% for KH-based versions; for AA MSAs, the corresponding fractions are 80% for RAxML-NG v1.2 and ∼72% for the Early Stopping versions. In contrast, when using random starting trees, the fractions for DNA MSAs are 85%

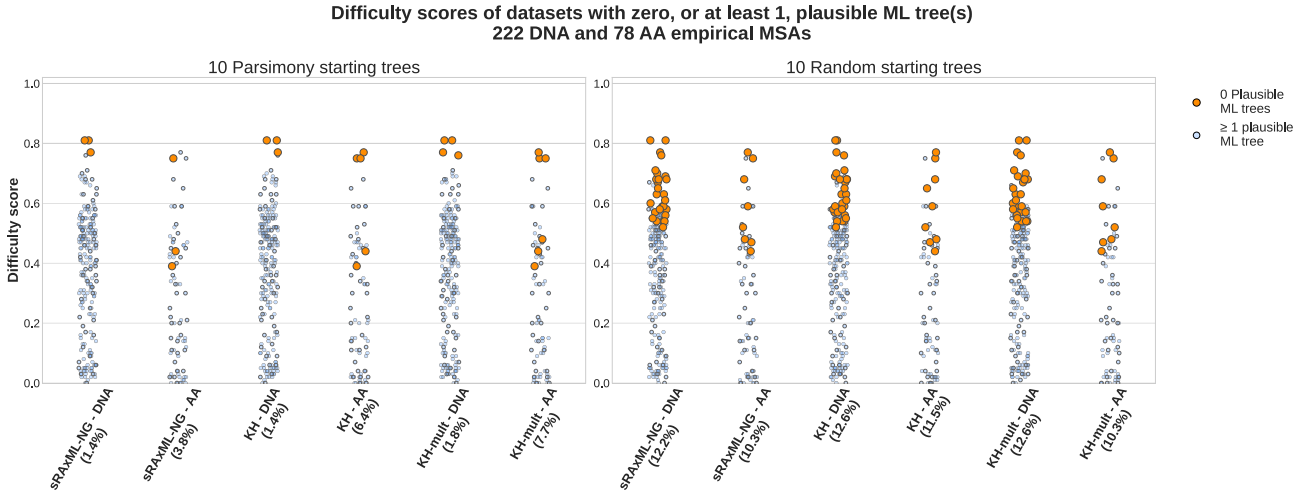

FIGURE 2. Stripplot illustrating correlation between the increased MSA difficulty scores and the inability of Early Stopping versions to infer at least one plausible ML tree, on 222 DNA and 78 AA large empirical MSAs. Each point represents a single data set, with difficulty scores shown on the *y*-axis. The columns on the *x*-axis represent specific combinations of Early Stopping versions (sRAxML-NG, KH, and KH-multiple testing) and data set types (DNA, AA). The percentage next to each x-label indicates the fraction of MSAs for which not a single plausible ML tree could be inferred for the corresponding combination. The left and right subfigures display results for parsimony and random starting trees, respectively. Data sets (points) where no plausible ML trees were inferred are highlighted more emphatically, while those where at least one (out of 10) plausible ML tree was inferred are indicated by lighter marks.

for RAxML-NG v1.2 and below 50% for Early Stopping versions, whereas for AA MSAs, the fractions are ∼70% for RAxML-NG v1.2 and ∼58% for Early Stopping versions. These results indicate that Early Stopping should primarily be used in combination with parsimony starting trees.

There appears to be a correlation between higher Pythia scores in DNA MSAs and the inability of Early Stopping methods to infer at least one plausible ML tree (out of 10), irrespective of starting trees (random vs. parsimony). The stripplot in Figure 2 indicates that all DNA MSAs, for which zero plausible trees were inferred by the Early Stopping versions, have difficulty scores above 0.5. For AA MSAs, however, this trend is less pronounced. Interestingly, the sets of MSAs for which Early Stopping methods failed to yield a single plausible tree are almost identical across all three versions (see Supplementary Tables S1 and S2). This suggests that sRAxML-NG may have been oversimplified for such challenging MSAs, which Pythia predicts to be difficult, particularly so for DNA data. KH-based approaches are not, or only marginally challenged in this case. Improving sRAxML-NG by incorporating strategies such as those proposed in Togkousidis et al. (2023) could potentially improve the performance of all three versions on difficult data sets.

To explore this hypothesis, we present additional analyses in Supplementary Figure S7, where we progressively increase the SPR regrafting radius of sRAxML-NG. This adjustment yields a more thorough search heuristic and therefore allows us to quantify how increasing thoroughness impacts the number of plausible ML trees inferred by Early Stopping. The results (Fig. S7) show that larger SPR regrafting radii improve the performance of Early Stopping versions (sRAxML-NG and KH-multiple testing) on DNA data, especially for searches on random starting trees. However, no improvement is observed for AA MSAs. A detailed assessment of this distinct behavior of the search strategy on AA data is the subject of future work.

To jointly compare the ML trees obtained with parsimony and random starting trees, we performed a plausibility analysis analogous to that in Figure 1, but we now pool together all 80 ML trees obtained from the two starting tree types. This comparison approach is effectively equivalent to having executed each RAxML-NG version using 10 random and 10 parsimony starting trees once, followed by a comparison of the 80 ML trees inferred across all versions (20 per version). The results (Supplementary Fig. S6) show negligible differences to those in Figure 1. Specifically, for those data sets where parsimony starting tree inferences failed to infer at least one (out of 10) plausible ML tree in Early Stopping versions, random starting tree inferences yielded only a single (out of 10) plausible ML tree on the same single protein MSA for all three Early Stopping versions. We provide further details in the Supplementary Material. However, this analysis does show that (i) as expected, optimization is more difficult (resulting in a lower number of plausible trees) when starting from random trees, for all methods; (ii) the difference is more pronounced for the three Early Stopping versions than for standard RAxML-NG; and (iii) for Early Stopping,

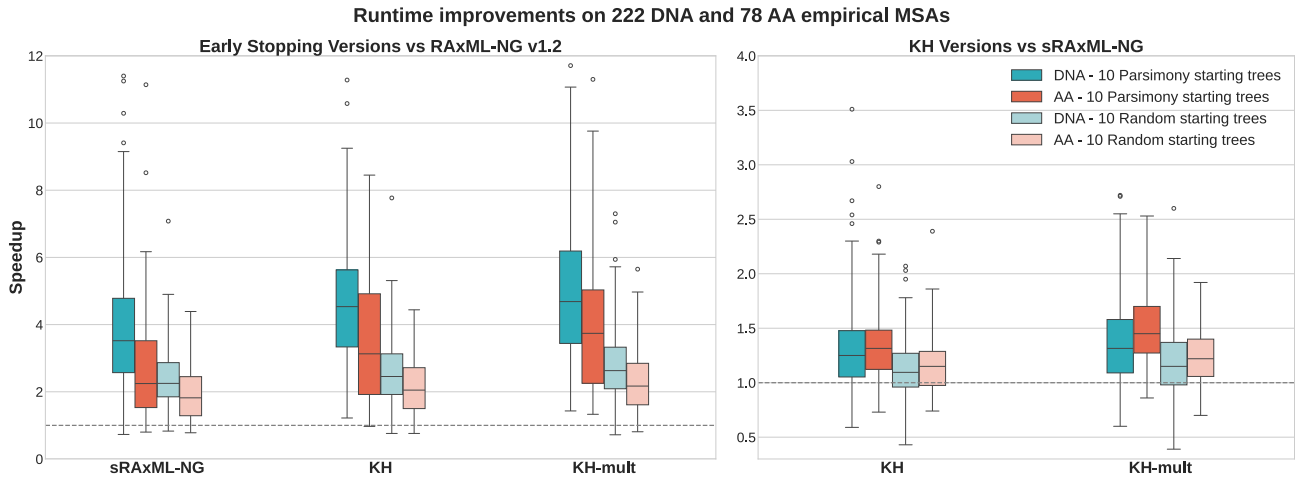

FIGURE 3. Speedup distributions of Early Stopping versions relative to RAxML v1.2 (left subfigure), and of the KH-based versions relative to sRAxML-NG (right subfigure), on 222 DNA and 78 AA large empirical MSAs. The speedups refer to strictly sequential runtimes for either 10 parsimony or 10 random starting trees. The dashed line at the bottom of each subfigure corresponds to a speedup of 1×.

the difference is more pronounced for DNA than for AAs, as expected due to the limitations of parsimony for protein MSAs, which is rarely used for such data, unlike DNA (Simmons et al. 2002).

We also analyzed 1076 simulated DNA MSAs with a wide range of Pythia difficulty scores. The results are generally similar to those observed for empirical data sets (see Supplementary Figs. S9–S11). As expected, there is a stronger phylogenetic signal in the simulated MSAs, which results in a higher number of plausible trees, especially for parsimony starting trees. Importantly, all methods (including standard RAxML-NG) were equally successful in recovering the true tree topology. Although simulated MSAs are easier (due to higher phylogenetic signal) than the empirical data sets, these experiments confirm that the stopping criteria work well from a topological accuracy standpoint.

Speedups

Figure 3 illustrates the speedup distributions across all the 300 large empirical DNA and AA MSAs for Early Stopping versions compared with RAxML-NG v1.2, as well as the runtime improvements of the KH-based versions relative to sRAxML-NG. The figure presents speedups for both parsimony and random starting trees. The highest speedups are observed with parsimony starting trees, reinforcing the preference for parsimony starting trees when using Early Stopping. Specifically, in conjunction with sRAxML-NG, the average speedups for KH and KH-multiple testing versions on parsimony starting trees over RAxML-NG v1.2 are 4.7× and 5× for DNA MSAs and 3.6× and 3.9× for AA MSAs, respectively. With random starting trees, however, speedups are significantly lower at 2.6× and 2.8× for DNA MSAs and 2.2× and 2.3× for AA MSAs, respectively. In the latter case, however, the overall execution times are generally longer (see Supplementary Fig. S5).

A substantial portion of the overall speedup can be attributed to the sRAxML-NG version itself, with KH-based versions providing additional runtime improvements, ranging from 10% to 50%. Specifically, for parsimony starting trees, the KH version is 29% and 35% faster than sRAxML-NG on DNA and AA MSAs, respectively, whereas the KH-multiple testing version is 36% and 49% faster, respectively.

DISCUSSION

Phylogenetic inference tools, such as RAxML-NG, thoroughly optimize the tree topology, the branch lengths, and the model parameters to increase the log-likelihood score of the inferred ML tree. Yet we know empirically that the final stages of this optimization are obsolete with respect to inference accuracy, although they do introduce a risk of overfitting the inferred tree to noisy input data. In deep learning, Early Stopping prevents overfitting by halting training when validation loss increases, even if training loss keeps decreasing. However, in ML-based phylogenetic inference, implementing an analogous approach is challenging, as holding out data for validation inherently alters the inference process. Because traditional validation approaches are not feasible under this setting, we implemented an Early Stopping strategy that is based upon the convergence dynamics of an individual tree search and halts optimization when further improvements become statistically insignificant.

We introduced Early Stopping criteria for ML-based phylogenetic inference and implemented them in RAxML-NG. Our goal was to reduce computing time while maintaining inference accuracy. By integrating the

KH test, and extending it via multiple testing correction, we can dynamically adapt the log-likelihood improvement threshold in a dataset-specific manner. The KH-based criteria rely on a simplified tree search heuristic, sRAxML-NG, which already performs Early Stopping. Our experiments suggest that the standard RAxML-NG search strategy may be unnecessarily complex, especially on easy data sets.

In our experiments on 300 large, representative empirical MSAs, Early Stopping tree searches, when initiated with 10 parsimony starting trees, yielded plausible ML output trees for 98% of the DNA data sets and up to 96% of the AA data sets, compared with RAxML-NG v1.2. These results demonstrate the robustness of our stopping criteria. Further, when all heuristic tree search versions are initiated with 10 parsimony starting trees, Early Stopping methods achieve an up to a 5× speedup for DNA data sets and a 3.9× speedup for AA data sets relative to RAxML-NG v1.2. In contrast, using random starting trees only provide a negligible advantage when combined with Early Stopping. Among data sets where Early Stopping methods failed to infer a plausible ML tree when using parsimony starting trees only, the addition of 10 random starting trees resulted in obtaining a plausible ML tree for only a single AA MSA, but no essential change in DNA MSA results. Despite this minimal impact on inference accuracy, random starting trees substantially increased computational time. Specifically, when comparing Early Stopping versions using 10 parsimony starting trees against the default execution of RAxML-NG v1.2 (which uses 10 random and 10 parsimony starting trees), the average speedups are even more pronounced: 7.8× for sRAxML-NG, 9.6× for KH, and 10.2× for KH-multiple testing versions. Given that the sequential runtime of RAxML-NG on these large data sets can be up to a day (see Supplementary Fig. S5), these speedups translate into hours of saved computing time. Interestingly, RAxML-NG v1.2 successfully inferred a plausible ML tree in all instances when only using parsimony starting trees, raising questions about the general utility of random starting trees in standard heuristics. Further benchmarking, particularly on AA data sets, might be useful to assess potential advantages of random starting trees.

Based on our results, as well as those from Liu et al. (2024), we recommend using Early Stopping methods with multiple (e.g., 10) parsimony starting trees. The Pythia score (Haag et al. 2022) is a good indicator for selecting the appropriate number of starting trees, as discussed in our previous study (Togkousidis et al. 2023). On easier data sets, fewer parsimony starting trees suffice, whereas more challenging data sets require a higher number. Although random starting trees might be useful for difficult MSAs when the primary objective is to maximize the log-likelihood score, we do not recommend this approach. In such cases, the phylogenetic signal is typically weak, and overoptimizing the log-likelihood score is unlikely to yield meaningful biological insights.

Overall, KH-based stopping criteria address a key challenge in large-scale phylogenetic analyses by minimizing inference times while still yielding accurate (plausible) results. Our methods can be seamlessly integrated into other ML-based tree inference tools.

Supplementary Material

Data available from the Dryad Digital Repository: http://dx.doi.org/10.5061/dryad.8gtht76zz

Funding

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

Acknowledgments

The authors thank the reviewers for their constructive feedback, which helped us improve the clarity and overall quality of the manuscript.

Data Availability

Stopping criteria are implemented in RAxML-NG and are available under GNU GPL on GitHub at https://github.com/togkousa/raxml-ng/tree/stopping-criteria. All codes and script files have been uploaded to Zenodo: https://doi.org/10.5281/zenodo.14653326. All data sets used are available for download from the Dryad Digital Repository: https://doi.org/10.5061/dryad.8gtht76zz

References

Dress A.W., Flamm C., Fritzsch G., Grünewald S., Kruspe M., Prohaska S.J., Stadler P.F. 2008. Noisy: identification of problematic columns in multiple sequence alignments. Algorithms Mol. Biol. 3:1–10. doi:10.1186/1748-7188-3-7.

Felsenstein J. 1981. Evolutionary trees from DNA sequences: a maximum likelihood approach. J. Mol. Evol. 17:368. doi:10.1007/BF01734359.

Felsenstein J. 2005. PHYLIP (phylogeny inference package) version 3.6. Distributed by the author. Seattle (WA): Department of Genome Sciences, University of Washington.

Fitch W.M. 1971. Toward defining the course of evolution: minimum change for a specific tree topology. Syst. Biol. 20(4):406–416. doi:10.1093/sysbio/20.4.406.

Gascuel O. 1997. BIONJ: an improved version of the NJ algorithm based on a simple model of sequence data. Mol. Biol. Evol. 14(7):685–695. doi:10.1093/oxfordjournals.molbev.a025808.

Goldman N., Anderson J.P., Rodrigo A.G. 2000. Likelihood-based tests of topologies in phylogenetics. Syst. Biol. 49(4):652–670. doi:10.1080/106351500750049752.

Guindon S., Dufayard J.F., Lefort V., Anisimova M., Hordijk W., Gascuel O. 2010. New algorithms and methods to estimate maximum-likelihood phylogenies: assessing the performance of PhyML 3.0. Syst. Biol. 59(3):307–321. doi:10.1093/sysbio/syq010.

Haag J., Höhler D., Bettisworth B., Stamatakis A. 2022. From easy to hopeless—predicting the difficulty of phylogenetic analyses. Mol. Biol. Evol. 39(12):msac254. doi:10.1093/molbev/msac254.
Haag J., Hübner L., Kozlov A.M., Stamatakis A. 2023. The free lunch is not over yet–systematic exploration of numerical thresholds in maximum likelihood phylogenetic inference. Bioinform. Adv. 3(1):vbad124. doi:10.1093/bioadv/vbad124.
Hillis D.M., Huelsenbeck J.P. 1992. Signal, noise, and reliability in molecular phylogenetic analyses. J. Hered. 83(3):189–195. doi:10.1093/oxfordjournals.jhered.a111190.
Höhler D., Haag J., Kozlov A.M., Stamatakis A. 2022. A representative performance assessment of maximum likelihood based phylogenetic inference tools. bioRxiv. doi:10.1101/2022.10.31.514545.
Kishino H., Miyata T., Hasegawa M. 1990. Maximum likelihood inference of protein phylogeny and the origin of chloroplasts. J. Mol. Evol. 31:151–160. doi:10.1007/BF02109483.
Kishino H., Hasegawa M. 1989. Evaluation of the maximum likelihood estimate of the evolutionary tree topologies from DNA sequence data, and the branching order in hominoidea. J. Mol. Evol. 29: 170–179. doi:10.1007/BF02100115.
Kozlov A. 2018. Models, optimizations, and tools for large-scale phylogenetic inference, handling sequence uncertainty, and taxonomic validation[Ph.D. thesis]. Karlsruhe Institute of Technology.
Kozlov A.M., Darriba D., Flouri T., Morel B., Stamatakis A. 2019. RAxML-NG: a fast, scalable and user-friendly tool for maximum likelihood phylogenetic inference. Bioinformatics 35(21): 4453–4455. doi:10.1093/bioinformatics/btz305.
Lemoine F., Domelevo Entfellner J.B., Wilkinson E., Correia D., Dávila Felipe M., De Oliveira T., Gascuel O. 2018. Renewing Felsenstein's phylogenetic bootstrap in the era of big data. Nature 556(7702):452–456. doi:10.1038/s41586-018-0043-0.
Liu C., Zhou X., Li Y., Hittinger C.T., Pan R., Huang J., Chen X.X., Rokas A., Chen Y., Shen X.X. 2024. The influence of the number of tree searches on maximum likelihood inference in phylogenomics. Syst. Biol. 73(5):807–822. doi:10.1093/sysbio/syae031.
Maddison W.P. 1997. Gene trees in species trees. Syst. Biol. 46(3):523–536. doi:10.1093/sysbio/46.3.523.
Markowski E., Susko E. 2024. Performance of topology tests under extreme selection bias. Mol. Biol. Evol. 41(1):msad280. doi:10.1093/molbev/msad280.
Minh B.Q., Schmidt H.A., Chernomor O., Schrempf D., Woodhams M.D., von Haeseler A., Lanfear R. 2020. IQ-TREE 2: new models and efficient methods for phylogenetic inference in the genomic era. Mol. Biol. Evol. 37(5):1530–1534. doi:10.1093/molbev/msaa015.
Moutsopoulos I., Maischak L., Lauzikaite E., Vasquez Urbina S.A., Williams E.C., Drost H.G., Mohorianu I.I. 2021. noisyR: enhancing biological signal in sequencing datasets by characterizing random technical noise. Nucleic Acids Res. 49(14):e83–e83. doi:10.1093/nar/gkab433.
Münkemüller T., Lavergne S., Bzeznik B., Dray S., Jombart T., Schiffers K., Thuiller W. 2012. How to measure and test phylogenetic signal. Methods Ecol. Evol. 3(4):743–756. doi:10.1111/j.2041-210X.2012.00196.x.
Piel W.H., Chan L., Dominus M.J., Ruan J., Vos R.A., Tannen V. 2009. TreeBASE v.2: a database of phylogenetic knowledge. e-BioSphere.
Price M.N., Dehal P.S., Arkin A.P. 2010. Fasttree 2–approximately maximum-likelihood trees for large alignments. PLoS One 5(3):1–10. doi:10.1371/journal.pone.0009490.
Roch S. 2006. A short proof that phylogenetic tree reconstruction by maximum likelihood is hard. IEEE/ACM Transactions on Computational Biology and Bioinformatics, 3(1):92–94. doi:10.1109/TCBB.2006.4.
Rokas A., Carroll S.B. 2006. Bushes in the tree of life. PLoS Biol. 4(11):e352. doi:10.1371/journal.pbio.0040352.
Shimodaira H. 2002. An approximately unbiased test of phylogenetic tree selection. Syst. Biol. 51(3):492–508. doi:10.1080/10635150290069913.
Shimodaira H., Hasegawa M. 2001. CONSEL: for assessing the confidence of phylogenetic tree selection. Bioinformatics 17(12):1246–1247. doi:10.1093/bioinformatics/17.12.1246.
Simmons M.P., Ochoterena H., Freudenstein J.V. 2002. Amino acid vs. nucleotide characters: challenging preconceived notions. Mol. Phylogenet. Evol. 24(1):78–90. doi:10.1016/S1055-7903(02)00202-6.
St. John K. 2016. Review paper: the shape of phylogenetic treespace. Syst. Biol. 66(1):e83–e94. doi:10.1093/sysbio/syw025.
Stamatakis A. 2014. RAxML version 8: a tool for phylogenetic analysis and post-analysis of large phylogenies. Bioinformatics 30(9):1312–1313. doi:10.1093/bioinformatics/btu033.
Swofford D. 2003. PAUP*. Phylogenetic analysis using parsimony. Version 4. Sunderland: Sinauer Associates.
Togkousidis A., Kozlov O.M., Haag J., Höhler D., Stamatakis A. 2023. Adaptive RAxML-NG: accelerating phylogenetic inference under maximum likelihood using dataset difficulty. Mol. Biol. Evol. 40(10):msad227. doi:10.1093/molbev/msad227.
Townsend J.P., Su Z., Tekle Y.I. 2012. Phylogenetic signal and noise: predicting the power of a data set to resolve phylogeny. Syst. Biol. 61(5):835–835. doi:10.1093/sysbio/sys036.
Vinh L.S., Von Haeseler A. 2004. IQPNNI: moving fast through tree space and stopping in time. Mol. Biol. Evol. 21(8):1565–1571. doi:10.1093/molbev/msh176.
Yang Z. 2014. Molecular evolution: a statistical approach. Oxford: Oxford University Press.

Syst. Biol. 00(0):1–9, 2025

https://doi.org/10.1093/sysbio/syaf043
Advance Access publication May 30, 2025