The Free Lunch is not over yet – Systematic Exploration of Numerical Thresholds in Phylogenetic Inference

Julia Haag[1*], Lukas Hübner[1,2], Alexey M. Kozlov[1] and Alexandros Stamatakis[1,2]

[1*]Computational Molecular Evolution Group, Heidelberg Institute for Theoretical Studies, Heidelberg, Germany.
[2]Institute for Theoretical Informatics, Karlsruhe Institute of Technology, Karlsruhe, Germany.

*Corresponding author. E-mail: julia.haag@h-its.org;

Abstract

Motivation: Maximum Likelihood (ML) is a widely used model for inferring phylogenies. The respective ML implementations heavily rely on numerical optimization routines that use internal numerical thresholds to determine convergence. We systematically analyze the impact of these threshold settings on the log-likelihood (LnL scores) and runtimes for ML tree inferences with RAxML-NG, IQ-TREE, and FastTree on empirical datasets.
Results: We provide empirical evidence that we can substantially accelerate tree inferences with RAxML-NG and IQ-TREE by changing the default values of two such numerical thresholds. At the same time, changing these settings does not significantly influence the quality of the inferred trees according to statistical significance tests. For RAxML-NG, increasing the likelihood thresholds ϵ_{LnL} and ϵ_{brlen} to 10 and 10^3 respectively results in an average speedup of 1.9 ± 0.6 on *Data collection 1* and 1.8 ± 1.3 on *Data collection 2*. Increasing the likelihood threshold ϵ_{LnL} to 10 in IQ-TREE results in an average speedup of 1.3 ± 0.4 on *Data collection 1* and 1.3 ± 0.9 on *Data collection 2*.
Availability and Implementation: All MSAs and results our analyses are based on are available for download at https://cme.h-its.org/exelixis/material/freeLunch˙data.tar.gz. Our data generation scripts are available at https://github.com/tschuelia/ml-numerical-analysis.
Contact: julia.haag@h-its.org
Supplementary information: Supplementary data are available online.

1 Introduction

Phylogenetic trees have many important applications in biology and medicine, for example, in drug development [7], forensics [13], or the analysis of SARS-CoV-2 genomes [15]. A widely used approach for reconstructing phylogenetic trees from a multiple sequence alignment (MSA) is the maximum likelihood (ML) method [24]. Popular ML-based tools are RAxML-NG [10], IQ-TREE [14], and FastTree [16]. Finding the most likely tree is $\mathcal{NP}$-hard [3] due to the super-exponential number of possible tree topologies. ML tree inference tools therefore implement tree search heuristics that attempt to iteratively optimize the log-likelihood (LnL score) by improving the tree topology, branch lengths, and substitution model parameters. These heuristics heavily rely on a plethora of numerical optimization routines (e.g., the Brent method [1] and the Broyden-Fletcher-Goldfarb-Shanno (BFGS) method [6])

that use specific internal numerical convergence thresholds. To the best of our knowledge, the impact of these threshold settings on inference times and LnL scores has never been systematically assessed, while anecdotal observations do exist. For instance, when analyzing SARS-CoV-2 data, Morel *et al.* [15] observed that one of these numerical thresholds, the minimum allowed branch length (*minBranchLen*), impacts the LnL scores of trees inferred with RAxML-NG and IQ-TREE. Here, we systematically investigate if we can reproduce this effect on other MSAs as well as for distinct numerical thresholds. In addition to RAxML-NG and IQ-TREE, we also investigate the behavior of FastTree. We explore the influence of up to seven distinct numerical thresholds on LnL scores and runtimes for these three ML inference tools. RAxML-NG and IQ-TREE offer two basic execution modes: tree evaluation and tree inference. In tree evaluation mode, the given (user-defined) tree topology remains fixed while the branch lengths and substitution model parameters are being optimized. In tree inference mode, the tools attempt to optimize the tree topology, the branch lengths, and the substitution model parameters. In our analyses, we investigate the influence of the numerical thresholds on both: tree inference and tree evaluation. Our analyses comprise two main studies:

Study 1: We analyzed the influence of up to 7 numerical thresholds on the LnL scores and runtimes of RAxML-NG, IQ-TREE, and FastTree. For RAxML-NG, IQ-TREE, and FastTree we analyzed the influence of the numerical thresholds when varied for tree inferences. For RAxML-NG and IQ-TREE, we also analyzed their influence when varied for tree evaluations. In study 1, we exclusively analyzed unpartitioned DNA MSAs (*Data collection 1*). To verify our findings for the likelihood epsilons ϵ_{LnL} and ϵ_{brlen} in RAxML-NG, and ϵ_{LnL} in IQ-TREE, we subsequently analyzed a more comprehensive as well as representative MSA collection including DNA, amino-acid (AA), and partitioned MSAs (*Data collection 2*).

Study 2: In our second study, we conducted a more detailed analysis of the likelihood epsilons in RAxML-NG as it is being actively developed in our lab. Since RAxML-NG uses the same threshold ϵ_{LnL} for four distinct tree search operations, we separated this threshold into four distinct fine-grained likelihood epsilons. The goal was to assess if appropriate fine-grained threshold settings could further improve the runtime.

The remainder of this paper is organized as follows: In Section 2, we outline the numerical thresholds we analyze and their usage in ML inference tools, our experimental setup, and the metrics we used to assess the influence of the numerical thresholds on tree quality and runtime. In Section 3 we present our key findings and in Section 4 we discuss the results of our analyses with a focus on the ϵ_{LnL} and ϵ_{brlen} thresholds in RAxML-NG and IQ-TREE. We conclude in Section 5.

All MSAs we used for our analyses, as well as all results, are available for download at https://cme.h-its.org/exelixis/material/freeLunch˙data.tar.gz. Our data generation scripts are available at https://github.com/tschuelia/ml-numerical-analysis.

2 Methods

2.1 Numerical Thresholds

Due to the extremely large tree space, an exhaustive search to identify the most likely tree is simply not feasible. ML-based tree inference tools therefore typically implement iterative tree improvement techniques, which they apply to an initial tree. Such an initial topology is obtained via heuristic tree inference methods, such as randomized stepwise addition order [2] or maximum parsimony [4, 5]. In our analyses, we focus on the three widely used ML inference tools RAxML-NG, IQ-TREE, and FastTree. Each tool iteratively optimizes the tree topology, the branch lengths, and the substitution model parameters starting from an initial tree. For example, RAxML-NG iteratively applies Subtree Pruning and Regrafting (SPR) moves followed by branch length and substitution model parameter optimizations. A more detailed description of the tree search heuristics is provided in Section 1 of the supplementary information. We analyze the influence of the following seven numerical thresholds:

- Likelihood epsilon ϵ_{LnL}: Threshold for LnL score improvement after one complete iteration (tree topology, branch lengths, and model parameters). The optimization only continues if the likelihood improvement is higher than this threshold.

- Branch length likelihood epsilon ϵ_{brlen}: RAxML-NG specific threshold for LnL score improvement. This epsilon is used during a so-called fast branch length optimization to rapidly approximate the LnL score of potential SPR moves.
- Minimum branch length (*minBranchLen*): Lower limit for branch length values.
- Maximum branch length (*maxBranchLen*): Upper limit for branch length values.
- Model likelihood epsilon ϵ_{model}: Threshold for substitution model parameter improvement. The substitution model parameters are only further optimized if the LnL score improvement exceeds this threshold.
- *num_iters*: Threshold to control the maximum number of iterations during Newton-Raphson based branch length optimization in RAxML-NG.
- *bfgs_factor*: This RAxML-NG specific threshold controls the convergence of the L-BFGS-B method used for optimizing substitution rates and stationary frequencies. The L-BFGS-B is a variant of the standard BFGS method, optimized for limited memory, and is extended to incorporate bound constraints in variables [25].

For RAxML-NG we analyze the influence of all seven thresholds, for IQ-TREE we analyze the influence of ϵ_{LnL}, *minBranchLen*, *maxBranchLen*, and ϵ_{model}. For FastTree we analyze the influence of ϵ_{LnL} and *minBranchLen*. As stated above, we mainly focus on the thresholds ϵ_{LnL} and ϵ_{brlen}. In Study 1 we find that decreasing the default setting does not substantially improve the LnL scores. To economize on computational resources and runtime, we thus only compare the current default setting to settings larger than the current default. For IQ-TREE, the current default setting for ϵ_{LnL} is 10^{-3}. We analyze the potential more liberal/superficial settings $\{10^{-3}, 10^{-2}, \ldots, 10^{3}\}$. For RAxML-NG the current default setting for both, ϵ_{LnL} and ϵ_{brlen}, is 10^{-1}. We analyze potential new and more superficial settings of $\{10^{-1}, 1, \ldots, 10^{3}\}$

2.2 Data Collections

In Study 1 we analyze 22 empirical unpartitioned DNA MSAs (*Data collection 1*). To verify these results, we analyze an additional collection of 19 empirical MSAs, including AA and partitioned MSAs (*Data collection 2*). For one additional AA dataset with excessive memory and runtime requirements, we only compare the results of the default threshold settings to the suggested new default settings. We exclusively analyze empirical datasets, because it was shown that reconstructing the best tree is more difficult on empirical datasets than it is on simulated datasets [8]. Section 2 in the supplementary information provides a detailed overview of all MSAs used.

2.3 Experimental Setup

We analyze each threshold and each ML inference tool separately. For each threshold and for each possible threshold setting, we infer 50 trees using the standard/default tree inference mode of the respective tool. Subsequently, we re-evaluate each inferred tree using the tree evaluation mode. For tree evaluation, we set the numerical thresholds to their corresponding default values. In the set comprising *all* inferred trees under *all* analyzed threshold settings, we determine the tree with the best LnL score (further referred to as *best-known tree*) and compare it to all other trees using several distinct phylogenetic statistical significance tests. For reasons, we detail further below, we do not compare all trees at once, but always conduct a pairwise comparison of each tree with the best-known tree. We collect trees that pass *all* significance tests in a so-called plausible tree set (see [15] for the introduction of the term). All trees in such a plausible tree set are not significantly worse than the best-known tree under all statistical significance tests. For the unpartitioned DNA MSAs we use the general time reversible (GTR) model [22] of nucleotide substitution as it is the most flexible and general model of nucleotide substitution. To account for among site rate heterogeneity, we additionally use four discrete Γ rate categories. The AA equivalent of the GTR model is the GTR20 (or PROTGTR) model. However, this model for AA data is very parameter rich. In particular, on datasets with weak phylogenetic signal (see below) the corresponding parameter estimates might thus be unstable. Instead, we use the LG substitution model [11] with four discrete Γ rate categories for unpartitioned AA MSAs.

2.4 Evaluation Metrics

In the following, we compare LnL scores in percent rather than via absolute LnL unit difference,

since the datasets yield a broad range of absolute likelihood values (LnL scores range between approximately −90 (D4) and −13 000 000 (D37)). Thus, as LnL scores are reported on a log scale, the observed effects are greater than the percentages might suggest. Therefore, we use two additional quality metrics: statistical significance tests and Robinson-Foulds distances (RF-Distances) [17] which we describe in the following. For evaluating the runtimes of the tree inferences, we compare the runtime of each tree inference in relation to the average runtime under the respective default setting. We report all speedups as mean ± standard deviation. Note that in our analyses we do not compare inferred trees, LnL scores, runtimes, or evaluation metrics across ML inference tools. All described analyses and evaluations metrics are applied separately and independently for each tool.

2.4.1 Significance Tests

In order to compare the trees inferred under different threshold settings, we use the statistical significance tests implemented in IQ-TREE. IQ-TREE implements the following significance tests: the Kishino-Hasegawa (KH) test [9] and the Shimodaira-Hasegawa (SH) test [18], both in their weighted and unweighted variants, the Approximately Unbiased (AU) test [19], as well as the Expected Likelihood Weight (ELW) test [21]. We use the default IQ-TREE settings for the number of resampling of estimated log-likelihoods (RELL) replicates (10 000) and significance level ($\alpha = 0.05$). Since the significance tests can be biased by the number of trees in the candidate set [21], we remove identical tree topologies from the set of inferred trees prior to applying the tests. Despite this tree set cleaning, we observed some unexpected behavior by the significance tests. First, the ELW test computes a c-ELW score (posterior weight) for each tree, sorts the trees according to this score and accepts trees as being not significantly different until the sum of c-ELW scores exceeds a predefined threshold. In our case, numerous trees in the inferred tree set have highly similar LnL scores despite their topologies being different. Yet, the c-ELW score for such trees is identical. Therefore, for trees that have a c-ELW that is close to exceeding the predefined significance threshold, only some trees with the exact same c-ELW score are accepted while the remaining ones are rejected. This leads to trees being rejected despite having the same LnL score as some accepted trees. Further, re-running the significance tests with the same trees but in a different order leads to a different subset of trees being accepted. Instead of re-estimating the substitution model parameters of each candidate tree, IQ-TREE uses a given best tree to optimize these parameters. As stated above, numerous trees have identical LnL scores, and therefore choosing the best tree according to the LnL score is ambiguous. We observe that the results of the significance tests vary largely depending on what tree is passed as the best tree despite identical LnL scores. We provide an example for both scenarios in the supplementary information. For the above reasons, instead of comparing all trees in the inferred tree set to each other at once, we only compare each inferred tree separately via all significance tests in a pairwise manner to the best-known tree. However, the c-ELW test is not intended for pairwise comparisons and only rejects one of the trees if the LnL scores deviate largely. Therefore, we additionally use the RF-Distance metric, which we describe in the following section.

2.4.2 RF-Distances

For the tree inference experiments, we fix the random seed to ensure that tree inferences always initiate their search on the same starting tree, despite using different numerical threshold settings. Therefore, we can directly compare trees inferred under different numerical threshold settings that started on the same starting tree. We compare these trees in a pairwise manner via the relative RF-Distance. If the RF-Distance between two trees, for example, one inferred under $\epsilon_{\text{LnL}} = 10^{-1}$ and one inferred under $\epsilon_{\text{LnL}} = 10^{3}$ is 0.0, then the tree inference converged to the same topology despite the different ϵ_{LnL} setting. However, an RF-Distance > 0 does not necessarily indicate that the tree is worse. For example, in general, the plausible tree set comprises multiple distinct tree topologies. Yet, they are not distinguishable via statistical significance tests. Therefore, when using this metric, we further compare these RF-Distances to the average pairwise RF-Distance between all plausible trees inferred under the default numerical threshold setting per tool

(*default plausible trees*). We further refer to this RF-Distance as *default RF-Distance*. This *default RF-Distance* provides a notion of how topologically scattered the plausible trees are under the default numerical threshold settings. The higher the *default RF-Distance* is, the more rugged the tree space will be. If the *default RF-Distance* is greater or equal to the RF-Distance between trees inferred under different numerical threshold settings, we assume that these differences are due to the ruggedness of the tree space rather than the trees being worse.

2.4.3 Phylogenetic Signal

The properties of the MSA influence the phylogenetic inference [20]. The stronger the so-called *phylogenetic signal* in the data is, the easier the phylogenetic analysis will be. This phylogenetic signal provides a notion of how informative the data is about the underlying evolutionary process [12]. In our study, we use the sites-per-taxa ratio as a proxy for the phylogenetic signal. In general, the higher the sites-per-taxa ratio of the MSA, the better is the phylogenetic signal of the data. In the following analyses, we will refer to MSAs with a sites-per-taxa ratio ≥ 80 as *good phylogenetic signal*. We refer to MSAs with a lower sites-per-taxa ratio as MSA with an *intermediate* or *weak phylogenetic signal*.

3 Results

In study 1, we observe a substantial runtime impact on tree inferences for two likelihood epsilons in RAxML-NG (ϵ_{LnL} and ϵ_{brlen}). We find that we can increase the settings of both thresholds without compromising the quality of the inferred trees, while obtaining a speedup of 1.9 ± 0.6. We make a similar observation for one numerical threshold (ϵ_{LnL}) in IQ-TREE that yields a speedup of 1.3 ± 0.4. All other thresholds we analyzed for RAxML-NG and IQ-TREE, as well as all thresholds analyzed for FastTree show no substantial influence neither on runtime nor on LnL scores as long as the settings remain within a reasonable range. For all analyzed ML inference tools, their current default settings fall within this reasonable range. We further observe that the runtime of the evaluation phase, as expected, is small compared to the corresponding tree inference time. Despite the impact of some numerical thresholds on tree evaluation runtimes, we therefore recommend using a conservative numerical threshold setting for tree evaluation. Our analyses on the more comprehensive collection of MSAs confirm our observations regarding tree inferences: the speedup in *Data collection 2* for RAxML-NG is 1.8 ± 1.1 and 1.3 ± 0.9 for IQ-TREE. Analogous to our results on *Data collection 1*, we do not observe a significant impact on the quality of the inferred trees according to our evaluation metrics. Study 2 shows that separating the ϵ_{LnL} into four distinctx thresholds does not further improve the runtime. Our analyses show a similar behavior for all four thresholds. We hence conclude that such a fine-grained distinction of threshold settings is neither necessary nor beneficial.

In the following discussion, we focus on the analysis of ϵ_{LnL} and ϵ_{brlen} in RAxML-NG and IQ-TREE on *Data collection 2*. In the supplementary information, we discuss the less interesting analysis results of all numerical thresholds on *Data collection 1*, the tree evaluation phase, the FastTree results, and the results of Study 2.

4 Discussion

The threshold with the highest impact on the runtime is the likelihood epsilon ϵ_{LnL}. We observe an impact for all three analyzed ML inference tools. Further, the branch length likelihood epsilon ϵ_{brlen} influences the runtime of RAxML-NG. For both thresholds, higher settings improve the runtime. We observe that increasing these likelihood epsilon settings for RAxML-NG and IQ-TREE leads to equally good results while requiring lower runtimes. All figures in the following section show the results summarized over all MSAs of *Data collection 2*. For better visualization of the speedup, we removed outliers using Tukey's fences [23] with $k = 3$ for all figures depicting a speedup. For the sake of completeness, we provide all speedup figures including all outliers in Section 4.3 in the supplementary information. In all box plots, the dashed vertical line indicates the mean, and the solid vertical line the median value.

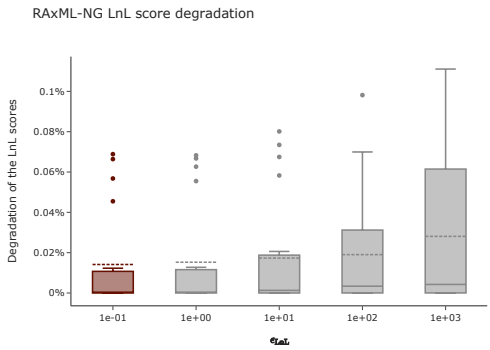

Fig. 1 Influence of the ϵ_{LnL} setting on the LnL scores of RAxML-NG. The highlighted box indicates the default setting. The y-axis shows the LnL score degradation per inferred tree in percent relative to the LnL score of the best-known tree. Higher percentages indicate worse LnL scores.

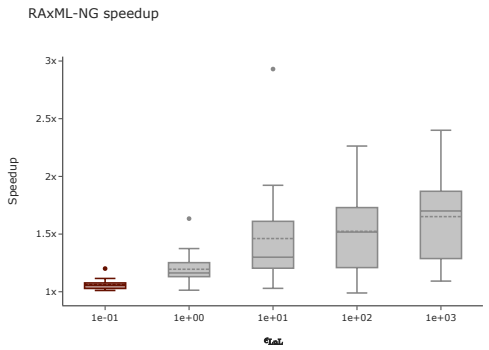

Fig. 2 Influence of the ϵ_{LnL} setting on the runtime of RAxML-NG tree inferences. The highlighted box indicates the default setting. The y-axis shows the speedup relative to the average runtime under the default setting.

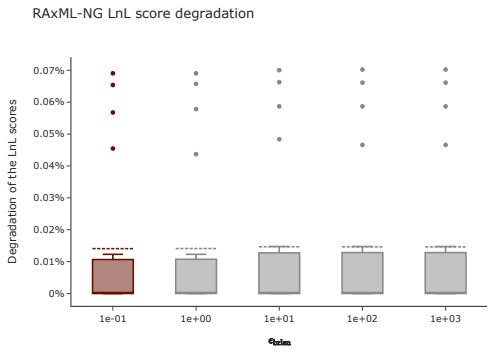

Fig. 3 Influence of the ϵ_{brlen} setting on the LnL scores of RAxML-NG. The highlighted box indicates the default setting. The y-axis shows the LnL score degradation per inferred tree in percent relative to the LnL score of the best-known tree. Higher percentages indicate worse LnL scores.

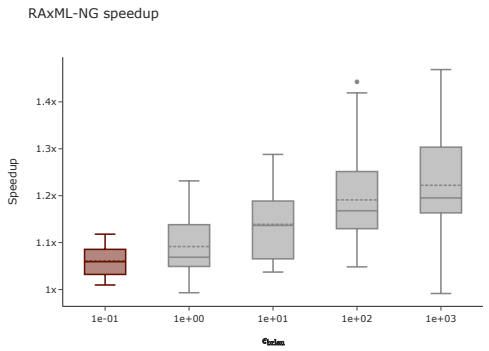

Fig. 4 Influence of the ϵ_{brlen} setting on the runtime of the RAxML-NG tree inference. The highlighted box indicates the default setting. The y-axis shows the speedup relative to the average runtime under the default setting.

4.1 RAxML-NG

With increasing ϵ_{LnL} threshold in RAxML-NG, we observe an expected decrease in LnL scores for higher settings. Especially for ϵ_{LnL} settings $\geq 10^2$ the LnL scores deteriorate noticeably (Figure 1). This is reflected by the proportion of tree inferences yielding a tree that is included in the plausible tree set (henceforth called a plausible tree) as well. For the RAxML-NG default setting $\epsilon_{\text{LnL}} = 10^{-1}$ on average 85 % of tree inferences yield a plausible tree, for 10^3 on average only 83 % yield a plausible tree. For all datasets (except D15) the RF-Distances between trees inferred under $\epsilon_{\text{LnL}} \leq 10$ compared to the default setting $\epsilon_{\text{LnL}} = 10^{-1}$ are smaller or equal to the *default RF-Distance*. However, for settings of 10^2 and 10^3 this is not the case. The average RF-Distances between trees inferred under these settings compared to the default setting are higher than the *default RF-Distance*. The topological differences among trees inferred under settings of 10^2 and 10^3 to trees inferred under the current default setting 10^{-1} can therefore not only be explained by the rugged tree space alone. This observation holds true even for datasets with a good phylogenetic signal. We conclude that for ϵ_{LnL} settings $\geq 10^2$ RAxML-NG

infers worse trees than for settings below 10^2. The runtimes of RAxML-NG tree inferences decrease with higher ϵ_{LnL} settings (Figure 2). On average, tree inferences under $\epsilon_{\text{LnL}} = 10^3$ run approximately twice as fast as tree inferences under $\epsilon_{\text{LnL}} = 10^{-1}$.

Given these observations, we conclude that the ϵ_{LnL} setting can be increased to 10. The quality of the trees is not affected by this more superficial optimization, but the tree inferences run on average 1.4 ± 0.6 times faster.

With RAxML-NG, we also analyze the influence of the ϵ_{brlen} threshold. Similar to the ϵ_{LnL} threshold, the runtimes for ϵ_{brlen} improve with increasing settings (Figure 4). According to our analyses, the LnL score is unaffected by the ϵ_{brlen} setting (variations between settings $\leq 0.007\,\%$; Figure 3). Across all MSAs the number of tree inferences yielding a plausible tree is identical for all ϵ_{brlen} settings we analyze. For MSAs with a good phylogenetic signal, we observe that the ϵ_{brlen} setting does not affect the final tree topology: for all tested settings the inferred tree topologies are identical (RF-Distance = 0.0). For all other MSAs, the average RF-Distance between trees inferred under different settings is below the *default RF-Distance*. We conclude that the ϵ_{brlen} threshold does not substantially influence the tree inference in RAxML-NG and the ϵ_{brlen} setting can be increased to 10^3. In our analyses this observation holds true for all analyzed MSAs independently of the magnitude of the LnL scores. RAxML-NG uses the ϵ_{brlen} to optimize the three branch lengths that are adjacent to the node at which a subtree is regrafted via an SPR move. We suspect that since all branch lengths are optimized at a later step during the tree inference, conducting a thorough optimization of these three branch lengths does not substantially improve the LnL score and can thus be terminated early.

Since we suggest changing two likelihood epsilons in RAxML-NG, we further analyze the influence of simultaneously changing both settings on the quality and the runtimes of tree inferences. To limit the computational effort, we only compare the default combination $(\epsilon_{\text{LnL}}, \epsilon_{\text{brlen}}) = (10^{-1}, 10^{-1})$ with the suggested new combination $(\epsilon_{\text{LnL}}, \epsilon_{\text{brlen}}) = (10, 10^3)$. As expected, the LnL scores are worse under the new setting compared to the old setting (Figure 5), but the tree inferences are faster (Figure 6). Averaged across all MSAs, the LnL scores between the current default and the suggested new combination vary by less than $0.004\,\%$. The percentage of tree inferences yielding a plausible tree is identical under both setting combinations ($87\,\%$). For all MSAs the RF-Distances between trees inferred under the current default combination versus the new combination are smaller or equal to the *default RF-Distance*. We conclude that increasing both threshold settings does not substantially decrease the LnL scores of the inferred trees and does therefore not affect the quality of the inferred trees. With the MSAs of *Dataset collection 1* we observe a speedup of 1.9 ± 0.6, on *Data collection 2* we observe a speedup of 1.8 ± 1.1.

4.2 IQ-TREE

Analogous to RAxML-NG, the runtime of the tree inference improves with higher ϵ_{LnL} settings for IQ-Tree as well. Tree searches under the default setting of $\epsilon_{\text{LnL}} = 10^{-3}$ run on average approximately twice as long as tree searches with $\epsilon_{\text{LnL}} = 10^3$ (Figure 8). However, IQ-TREE appears to be more sensitive to the ϵ_{LnL} setting than RAxML-NG in terms of LnL scores. Under higher ϵ_{LnL} settings, the LnL score degradation is an order of magnitude worse than for RAxML-NG (on average $\leq 0.2\,\%$ for IQ-TREE vs. $\leq 0.03\,\%$ for RAxML-NG; Figure 7). For ϵ_{LnL} values ≤ 10 the LnL scores are on average approximately equal. Based on the plausible tree set size under various settings, we observe that IQ-TREE is more sensitive to the ϵ_{LnL} setting. We observe that for $\epsilon_{\text{LnL}} = 10^3$ averaged over all MSAs, noticeably fewer tree inferences yield a plausible tree than for any other setting ($58\,\%$ vs. $76\,\%$ for $\epsilon_{\text{LnL}} = 10^{-3}$). This effect is less pronounced for MSAs with a good phylogenetic signal. For MSAs with a sites-per-taxa ratio ≥ 80 we observe that the ϵ_{LnL} setting does not affect the final tree topology: under all tested settings the inferred tree topologies are identical (RF-Distance = 0.0). For MSAs with a worse phylogenetic signal, the RF-Distance between trees inferred under the default setting 10^{-3} and settings of 10^2 and 10^3 exceed the average RF-Distance in the plausible tree set. We conclude that for MSAs with an intermediate or weak phylogenetic signal, the trees inferred under ϵ_{LnL} settings $\geq 10^2$ are worse than under

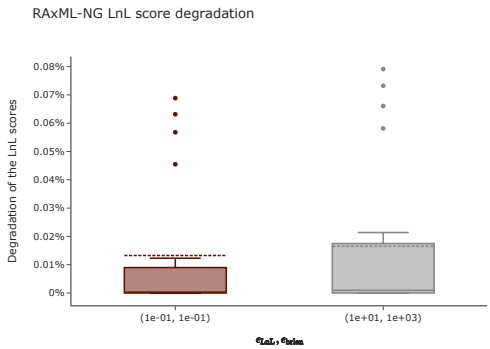

Fig. 5 Influence of simultaneously changing both likelihood epsilon settings on the LnL scores of RAxML-NG. The highlighted box indicates the default combination. The y-axis shows the LnL score degradation per inferred tree in percent relative to the LnL score of the best-known tree. Higher percentages indicate worse LnL scores.

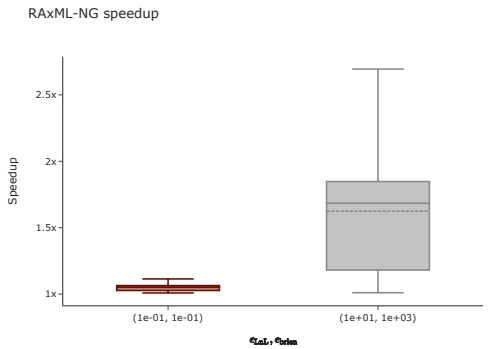

Fig. 6 Influence of simultaneously changing both likelihood epsilon settings on the runtime of the RAxML-NG tree inference. The highlighted box indicates the default combination. The y-axis shows the speedup relative to the average runtime under the default combination.

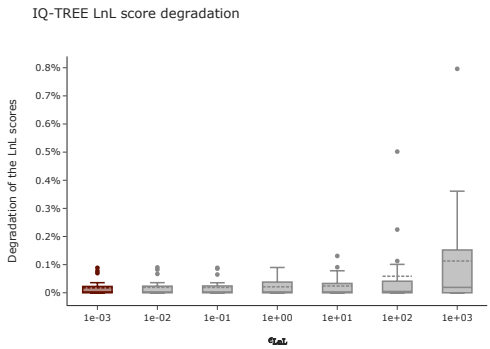

Fig. 7 Influence of the ϵ_{LnL} setting on the LnL scores of IQ-TREE. The highlighted box indicates the default setting. The y-axis shows the LnL score degradation per inferred tree in percent relative to the LnL score of the best-known tree. Higher percentages indicate worse LnL scores.

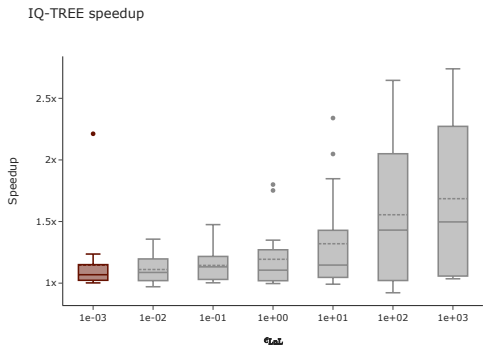

Fig. 8 Influence of the ϵ_{LnL} setting on the runtime of the IQ-TREE tree inference. The highlighted box indicates the default setting. The y-axis shows the speedup relative to the average runtime under the default setting.

lower settings. According to our evaluation metrics across all analyzed MSAs the ϵ_{LnL} setting can be set to 10 without compromising the quality of the inferred trees. In our analyses, this results in an average speedup of 1.3 ± 0.9.

As mentioned before, we observe a higher sensitivity to the ϵ_{LnL} setting in IQ-TREE than in RAxML-NG. We suspect that this is caused by the random Nearest Neighbor Interchange (NNI) topology optimization moves in IQ-TREE's search algorithm. IQ-TREE implements these random NNI moves to escape local NNI maxima (see the supplementary information for a more detailed description of the IQ-TREE inference heuristic). To explore this hypothesis, we modify IQ-TREE and disable this randomness in the search algorithm. As a consequence, IQ-TREE then only optimizes the tree topology using standard NNI moves. We refer to the standard IQ-TREE as *random IQ-TREE* and to the IQ-TREE algorithm without random NNI moves as *de-randomized IQ-TREE*. We re-analyze four MSAs using the de-randomized IQ-TREE version. Without the random NNI moves, the IQ-TREE search heuristic can explore the tree space less, thus, we expect the LnL scores for de-randomized IQ-TREE to

be worse than for random IQ-TREE, which we indeed observe in our analyses. To compare the influence of the ϵ_{LnL} threshold, we again compute the proportion of tree inferences yielding a plausible tree. We observed that when using de-randomized IQ-TREE, noticeably more tree inferences yield a plausible tree under $\epsilon_{\text{LnL}} \geq 10^2$ than when using the random IQ-TREE variant. We conclude that large ϵ_{LnL} settings ($\geq 10^2$) distort the random NNI moves in IQ-TREE, causing a premature termination of the tree inference. This also explains the vast runtime improvement under these settings.

5 Conclusion

Increasing the RAxML-NG settings for the likelihood epsilons ϵ_{LnL} and ϵ_{brlen} to 10 and 10^3 respectively does not significantly influence the quality of the inferred trees according to statistical significance tests. By changing both settings, we observe a speedup of 1.9 ± 0.6 on *Data collection 1* and 1.8 ± 1.1 on *Data collection 2*. With IQ-TREE, increasing the ϵ_{LnL} to 10 has no significant impact on the LnL scores, and we observe a speedup of 1.3 ± 0.4 on *Data collection 1* and 1.3 ± 0.9 on *Data collection 2*. Our observations are independent of the magnitude of the LnL scores of the analyzed MSAs. For MSAs with a good phylogenetic signal, the inferred tree topologies under the current default settings and the suggested new settings are identical for both, RAxML-NG and IQ-TREE (RF-Distance = 0.0). For MSAs with an intermediate or weak phylogenetic signal, the topological differences between threshold settings can be explained by the rugged tree space, and the RF-Distances between inferred trees under different settings are less than or equal to the *default RF-Distance*. It is important to note that the tree evaluation after tree inference should not be omitted and performed under conservative likelihood epsilon settings, for example the default settings in RAxML-NG and IQ-TREE.

Funding

This work was financially supported by the Klaus Tschira Foundation. This work was supported by a grant from the Ministry of Science, Research and the Arts of Baden-Württemberg (Az: 33-7533.-9-10/20/2) to Peter Sanders and Alexandros Stamatakis.

References

[1] R. P. Brent. An algorithm with guaranteed convergence for finding a zero of a function. *The Computer Journal*, 14(4):422–425, 1971. doi: 10.1093/comjnl/14.4.422.

[2] L. L. Cavalli-Sforza and A. W. F. Edwards. Phylogenetic analysis. Models and estimation procedures. *Evolution*, 21(3):550–570, 1967. doi: 10.1111/j.1558-5646.1967.tb03411.x.

[3] Benny Chor and Tamir Tuller. Maximum likelihood of evolutionary trees: hardness and approximation. *Bioinformatics*, 21:i97–i106, 2005. doi: 10.1093/bioinformatics/bti1027.

[4] James S. Farris. Methods for Computing Wagner Trees. *Systematic Biology*, 19(1):83–92, 1970. doi: 10.1093/sysbio/19.1.83.

[5] Walter M. Fitch. Toward Defining the Course of Evolution: Minimum Change for a Specific Tree Topology. *Systematic Zoology*, 20(4):406–416, 1971. doi: 10.2307/2412116.

[6] R. Fletcher. *Practical Methods of Optimization*. John Wiley & Sons, Ltd, 2000. doi: 10.1002/9781118723203.ch3.

[7] IvanV Gregoretti, Yun-Mi Lee, and Holly V Goodson. Molecular Evolution of the Histone Deacetylase Family: Functional Implications of Phylogenetic Analysis. *Journal of Molecular Biology*, 338(1):17–31, 2004. doi: 10.1016/j.jmb.2004.02.006.

[8] John P. Huelsenbeck. Performance of Phylogenetic Methods in Simulation. *Systematic Biology*, 44(1):17–48, 1995.

[9] H Kishino and M Hasegawa. Evaluation of the maximum likelihood estimate of the evolutionary tree topologies from DNA sequence data, and the branching order in hominoidea. *Journal of molecular evolution*, 29(2):170–179, 1989. doi: 10.1007/bf02100115.

[10] Alexey M Kozlov, Diego Darriba, Tomáš Flouri, Benoit Morel, and Alexandros Stamatakis. RAxML-NG: a fast, scalable and user-friendly tool for maximum likelihood phylogenetic inference. *Bioinformatics*, 35(21):4453–4455, 2019. doi: 10.1093/bioinformatics/btz305.

[11] Si Quang Le and Olivier Gascuel. An Improved General Amino Acid Replacement Matrix. *Molecular Biology and Evolution*, 25(7):1307–1320, 2008. doi: 10.1093/molbev/msn067.

[12] Philippe Lemey, Marco Salemi, and Anne-Mieke Vandamme. *The Phylogenetic Handbook: A Practical Approach to Phylogenetic Analysis and Hypothesis Testing*. Cambridge University Press, 2 edition, 2009. doi: 10.1017/CBO9780511819049.

[13] Michael L. Metzker, David P. Mindell, Xiao-Mei Liu, Roger G. Ptak, Richard A. Gibbs, and David M. Hillis. Molecular evidence of HIV-1 transmission in a criminal case. *Proceedings of the National Academy of Sciences*,

99(22):14292–14297, 2002. doi: 10.1073/pnas.222522599.

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

[15] Benoit Morel, Pierre Barbera, Lucas Czech, Ben Bettisworth, Lukas Hübner, Sarah Lutteropp, Dora Serdari, Evangelia-Georgia Kostaki, Ioannis Mamais, Alexey M Kozlov, Pavlos Pavlidis, Dimitrios Paraskevis, and Alexandros Stamatakis. Phylogenetic Analysis of SARS-CoV-2 Data Is Difficult. *Molecular Biology and Evolution*, 38(5):1777–1791, 2020. doi: 10.1093/molbev/msaa314.

[16] Morgan N. Price, Paramvir S. Dehal, and Adam P. Arkin. FastTree 2 – Approximately Maximum-Likelihood Trees for Large Alignments. *PLOS ONE*, 5 (3):1–10, 2010. doi: 10.1371/journal.pone.0009490.

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

[18] H Shimodaira and M Hasegawa. Multiple Comparisons of Log-Likelihoods with Applications to Phylogenetic Inference. *Molecular Biology and Evolution*, 16(8), 1999. doi: 10.1093/oxfordjournals.molbev.a026201.

[19] Hidetoshi Shimodaira. An Approximately Unbiased Test of Phylogenetic Tree Selection. *Systematic biology*, 51: 492–508, 2002. doi: 10.1080/10635150290069913.

[20] Alexandros Stamatakis. *Phylogenetic Search Algorithms for Maximum Likelihood*, chapter 25, pages 547–577. John Wiley & Sons, Ltd, 2011. doi: 10.1002/9780470892107.ch25.

[21] Korbinian Strimmer and Andrew Rambaut. Inferring confidence sets of possibly misspecified gene trees. In *Proceedings. Biological sciences*, pages 137–142, 2002. doi: 10.1098/rspb.2001.1862.

[22] S. Tavaré. Some probabilistic and statistical problems on the analysis of DNA sequences. *Lectures on Mathematics in the Life Sciences*, 17:57–86, 1986.

[23] J. Tukey. *Exploratory Data Analysis*. Addison-Wesley, 1977.

[24] Ziheng Yang, Nick Goldman, and Adrian Friday. Maximum Likelihood Trees from DNA Sequences: A Peculiar Statistical Estimation Problem. *Systematic Biology*, 44: 384–399, 1995. doi: 10.2307/2413599.

[25] Ciyou Zhu, Richard H. Byrd, Peihuang Lu, and Jorge Nocedal. Algorithm 778: L-BFGS-B: Fortran Subroutines for Large-Scale Bound-Constrained Optimization. *ACM Trans. Math. Softw.*, 23(4):550–560, 1997. doi: 10.1145/279232.279236.