

IQ-TREE 2: New Models and Efficient Methods for Phylogenetic Inference in the Genomic Era

Bui Quang Minh ^{*,1,2} Heiko A. Schmidt,³ Olga Chernomor,³ Dominik Schrempf,^{3,4} Michael D. Woodhams,⁵ Arndt von Haeseler,^{†,3,6} and Robert Lanfear^{†,2}

¹Research School of Computer Science, Australian National University, Canberra, ACT, Australia

²Department of Ecology and Evolution, Research School of Biology, Australian National University, Canberra, ACT, Australia

³Center for Integrative Bioinformatics Vienna, Max Perutz Labs, University of Vienna and Medical University of Vienna, Vienna, Austria

⁴Department of Biological Physics, Eötvös Lóránd University, Budapest, Hungary

⁵Discipline of Mathematics, University of Tasmania, Hobart, TAS, Australia

⁶Bioinformatics and Computational Biology, Faculty of Computer Science, University of Vienna, Vienna, Austria

[†]These authors contributed equally to this work.

*Corresponding author: E-mail: m.bui@anu.edu.au.

Associate editor: Emma Teeling

Abstract

IQ-TREE (<http://www.iqtree.org>, last accessed February 6, 2020) is a user-friendly and widely used software package for phylogenetic inference using maximum likelihood. Since the release of version 1 in 2014, we have continuously expanded IQ-TREE to integrate a plethora of new models of sequence evolution and efficient computational approaches of phylogenetic inference to deal with genomic data. Here, we describe notable features of IQ-TREE version 2 and highlight the key advantages over other software.

Key words: phylogenetics, phylogenomics, maximum likelihood, models of sequence evolution.

IQ-TREE is a widely used and open-source software package for phylogenetic inference using the maximum likelihood (ML) criterion. The high performance of IQ-TREE results from the efficient integration of novel phylogenetic methods that improve the three key steps in phylogenetic analysis: fast model selection via ModelFinder (Kalyaanamoorthy et al. 2017), an effective tree search algorithm (Nguyen et al. 2015), and a novel ultrafast bootstrap approximation (Minh et al. 2013; Hoang et al. 2018). Zhou et al. (2018) independently showed that the tree search algorithm in IQ-TREE exhibits good performance in terms of both computing times and likelihood maximization when compared with other popular ML phylogenetics software such as RAxML (Stamatakis 2014) and PhyML (Guindon et al. 2010). IQ-TREE also plays a vital role in the software ecosystem for biomedical research. For instance, it is an integral component of many popular open-source applications such as Galaxy (Afgan et al. 2018), Nextstrain (Hadfield et al. 2018), OrthoFinder (Emms and Kelly 2015), and QIIME 2 (Bolyen et al. 2019).

Since the release of IQ-TREE version 1.0 in 2014, we have continuously developed IQ-TREE to integrate a plethora of new evolutionary models and efficient methods for analyzing large phylogenomic data sets. Here, we present IQ-TREE version 2 and highlight the key new features and improvements. To demonstrate its performance and compare it with other software, we used a 2-GHz CPU server to analyze 2 large sequence alignments: a DNA alignment of 110 vertebrate

species and 25,919 sites (Fong et al. 2012) which we call the DNA-data set, and an amino acid alignment of 76 metazoan species and 49,388 sites (Whelan et al. 2017) which we call the AA-data set.

Time-Reversible Models of Sequence Evolution

IQ-TREE 2 supports more than 200 time-reversible evolutionary models, including all standard substitution models for DNA, protein, codon, binary, and multistate-morphological data (Felsenstein 2004; Lemey et al. 2009). Rate heterogeneity across sites can be accommodated either by the discrete Γ distribution (Yang 1994b) with invariant sites (Gu et al. 1995) or by a distribution-free rate model (Kalyaanamoorthy et al. 2017). Site-specific rates can be estimated by the empirical Bayesian method via the `-rate` option or by ML (Mayrose et al. 2004) via the `-mlrate` option. These estimated site-specific rates can be useful for downstream analysis such as quantification of phylogenetic informativeness, signal, and noise (Dornburg et al. 2016). For single nucleotide polymorphism or morphological data, the absence of invariant sites can be accounted for by an ascertainment bias correction (Lewis 2001).

Moreover, IQ-TREE 2 offers a number of advanced models for phylogenomic data including partitioned models (Lanfear et al. 2012; Chernomor et al. 2016), mixture models (Le et al. 2008, 2012; Le and Gascuel 2010), posterior-mean

site frequency models (Wang et al. 2018), and heterotachy models (Crotty et al. 2019). For allele-frequency data, IQ-TREE 2 implements polymorphism-aware models (Schrempf et al. 2016, 2019). With partitioned models, one can specify either a partition file (as in IQ-TREE 1) or a directory of single-locus alignments (a new feature in IQ-TREE 2). In the latter case, IQ-TREE 2 will load and concatenate all alignments within the directory, eliminating the need for users to manually perform this step. In addition to implementing existing mixture models, IQ-TREE 2 goes beyond the mixture models employed in PhyML-mixtures (Le et al. 2008) and RAXML-NG software (Kozlov et al. 2019), by allowing for user-defined mixture models using the “MIX{model₁,... ,model_k}” syntax. IQ-TREE 2 is also substantially faster than PhyML-mixtures. For example, optimization of the model parameters for the LG4X model on the AA-data set took 1.9 min in IQ-TREE 2, 3 min in RAXML-NG, and 17.6 min in PhyML-mixtures.

Because the aforementioned substitution models assume time reversibility, IQ-TREE 1 only enabled inference of *unrooted* trees (Felsenstein 1981). In IQ-TREE 2, we included non-time-reversible models (e.g., Norris 1997), meaning that IQ-TREE 2 enables inference of *rooted* trees.

Nonreversible Substitution Models

IQ-TREE 2 allows users to reconstruct rooted trees using non-reversible models, a feature not available in most ML packages due to numerical and computational expense. We substantially revised the IQ-TREE code to overcome these obstacles. First, due to nonreversibility of the rate matrix Q , the naïve computation of the transition probability matrix $P(t) = e^{Qt}$, where t is the branch length, is unstable due to complex eigenvalues of Q (Moler and Loan 1978). Eigen-decomposition and scaling-squaring techniques, for example, as provided in the Eigen3 library (Guennebaud et al. 2010) remove the numerical problems. IQ-TREE 2 employs eigen-decomposition to diagonalize Q into its (complex) eigenvalues, eigenvectors and inverse eigenvectors, which are used to compute $P(t)$. If Q is not diagonalizable, then IQ-TREE 2 switches to the scaling-squaring technique to compute $P(t)$. The eigen-decomposition is fast but sometimes unstable, whereas the scaling-squaring is slow but stable. Second, IQ-TREE 2 uses a rooted tree data structure and an adjusted pruning algorithm for computing likelihoods of nonreversible models on rooted trees (Boussau and Gouy 2006). Third, we adapted the hill climbing nearest neighbor interchange, part of the tree search heuristic in IQ-TREE to account for rooted trees. Fourth, we introduced a root search operation that moves the root to the neighboring branches (by default up to 2 branches away from the current root branch) and retains the rooting position with the highest likelihood. Users can increase this parameter (–root-dist option) to test for the position of the root across more or all branches.

Based on these improvements, we could efficiently implement 99 nonreversible DNA models known as Lie Markov models (Woodhams et al. 2015), the unrestricted model (UNREST) for DNA (Yang 1994a) and—for the first time—

the general nonreversible model for amino acid sequences that we call NONREV. For the DNA-data set (Fong et al. 2012), tree search under the general time-reversible model took 2.8 min using 4 CPU cores, whereas the UNREST model took 10.7 min. For the AA-data set (Whelan et al. 2017) tree search under the protein general time-reversible model took 2.52 h, whereas the NONREV model took 5.8 h. The implementation of nonreversible models in IQ-TREE 2 opens new avenues of evolutionary research.

Fast Likelihood Mapping Analysis

IQ-TREE 2 provides a fast and parallel implementation of the quartet likelihood mapping (Strimmer and von Haeseler 1997) to visualize phylogenetic information in alignments or to study the relationships of taxon-groups in large data sets. To this end, IQ-TREE 2 evaluates the exact ML value of all relevant quartets. Application of the original implementation of quartet likelihood mapping in TREE-PUZZLE (Schmidt et al. 2002) (i.e., with 10,000 random quartets and exact ML quartet evaluation) to the DNA-data set took 282 min, whereas the improved implementation in IQ-TREE 2 required only 1 min using one CPU core and 21 s using four cores. Similarly, analysis of the AA-data set took 99.5 h in TREE-PUZZLE, 17 min in IQ-TREE 2 with one CPU core, and 4.5 min using four cores. Together with the extended repertoire of sequence evolution models, likelihood mapping facilitates a thorough investigation of much larger sequence alignments.

New Options for Tree Search

IQ-TREE 2 allows users to perform a constrained tree search (–g option), such that the resulting ML tree will respect a set of user-defined splits, which may also contain polytomies. This option is helpful to enforce the monophyly of certain groups. A tree test (see below) can be performed to ensure that the constrained tree is not significantly worse than the unconstrained tree.

IQ-TREE 2 also provides a fast tree search (–fast option) using an algorithm resembling that implemented in FastTree2 (Price et al. 2010). Here, IQ-TREE 2 computes two starting trees using Maximum Parsimony and Neighbor-Joining (Gascuel 1997), which are then optimized by the hill climbing nearest neighbor interchange moves. For our example DNA-data set, the default tree search took 27 min, whereas the fast IQ-TREE search and FastTree2 needed 82 and 85 s, respectively. For the AA-data set, the default tree search took 5.7 h, whereas the fast IQ-TREE search and the FastTree2 search took 13.9 and 4.3 min, respectively. The speed of FastTree2 was accomplished at the cost of producing substantially worse trees than RAXML, PhyML, and IQ-TREE (Zhou et al. 2018).

IQ-TREE 2 also provides a –runs option, which conducts multiple independent tree searches, summarizes the resulting trees, and reports the tree(s) with the highest log-likelihood. This option is recommended for difficult data sets, for

example, with many taxa and/or limited phylogenetic signal, to increase the probability that the true ML tree is found.

Systematically Accounting for Missing Data

Missing data—when some loci or sites are absent for some species—are almost unavoidable in phylogenomic data sets. In the presence of missing data, each species tree can be associated with a corresponding set of induced single-locus trees. For each locus, an induced tree is obtained from the species tree by removing species with unavailable sequences for that locus. Missing data can create phylogenetic terraces (Sanderson et al. 2011), where for two or more species trees the associated sets of induced single-locus trees are identical, leading to identical likelihoods under an edge-unlinked partitioned model.

IQ-TREE 2 employs the terraphast library (Biczok et al. 2018) to automatically report the inferred ML trees that reside on a terrace. If so, users are advised to gather more data or filter out gappy taxa/loci. Moreover, IQ-TREE 2 generalizes the terrace concept to partial terraces (Chernomor et al. 2015), where a subset of the induced single-locus trees is identical. IQ-TREE 2 exploits partial terraces to improve tree search under partitioned models and achieves up to 4.5- and 8-fold speedups compared with IQ-TREE 1 and RAXML, respectively (Chernomor et al. 2016).

Single-Locus Tree Inference

IQ-TREE 2 enables users to infer individual locus trees (-S option), which can be used for subsequent coalescent (e.g., Mirarab et al. 2014) or concordance analyses (e.g., Minh et al. 2018). Users need to specify either a partition file that delineates the borders between loci or a directory containing the individual locus alignments in individual files. In both cases, IQ-TREE 2 will perform separate model selection and tree searches for each locus, which are automatically scheduled on the k CPU cores. The loci are ranked according to their expected computational costs, where the costs are estimated as a product of the number of sequences, distinct site patterns, and character states (4 for DNA and 20 for protein). The k loci with highest costs will be assigned to one of the k cores. When a core has finished computation the next locus in the ranked list will be assigned to that core. This process continues until all computations are complete.

We compared our scheduling approach with the program ParGenes version 1.0.1 (Morel et al. 2019), which uses RAXML-NG for tree search and a more sophisticated scheduling algorithm. For the DNA-data set, IQ-TREE 2 inferred 168 locus trees in 18.6 and 9.2 min using 2 and 4 CPU cores, respectively; whereas the same analysis in ParGenes took 9.17 and 5.98 min. For the AA-data set with 127 loci, IQ-TREE 2 took 2.05 and 1.05 h using 2 and 4 cores, respectively; whereas ParGenes needed 1.53 and 0.81 h. Therefore, IQ-TREE 2 shows a higher parallel efficiency (DNA: 101% and AA: 97.6%) (Grama et al. 2003) than ParGenes (DNA: 76.7% and AA: 94.4%), but ParGenes needs less computational time. The upshot of this is that both programs are likely to perform similarly on very large data sets.

Fast Branch Tests

IQ-TREE 2 provides fast and parallel implementations for several existing branch tests including the approximate likelihood ratio test (aLRT) (Anisimova and Gascuel 2006), the Shimodaira–Hasegawa-like aLRT (SH-aLRT) (Guindon et al. 2010), and the aBayes test (Anisimova et al. 2011). The SH-aLRT is parallelized over the bootstrap samples to maximize load balance and efficiency. These tests can be performed on the reconstructed ML tree or a user-defined tree. For the DNA-data set, PhyML version 3.3.20190321 (Guindon et al. 2010) took 33.9 h to perform the SH-aLRT, whereas IQ-TREE 2 needed only 1.5 min (1,300-fold speedup). On the AA-data set, PhyML took 173.5 h to perform the SH-aLRT tests, whereas IQ-TREE 2 needed just 1.6 min (a greater than 2,000-fold speedup).

Fast Topology Tests

IQ-TREE 2 also provides fast and parallel implementations of existing tree topology tests including the Shimodaira–Hasegawa test (Shimodaira and Hasegawa 1999), the approximately unbiased test (Shimodaira 2002), and the expected likelihood weight (Strimmer and Rambaut 2002). Moreover, IQ-TREE 2 is well suited for partitioned models because it provides the site, gene, and gene-site bootstrap resampling schemes (Hoang et al. 2018).

For the DNA-data set CONSEL (Shimodaira and Hasegawa 2001), the original and unparallelized implementation of the approximately unbiased test, took 5 min to test 100 tree topologies, whereas IQ-TREE 2 took 1.8 min with one CPU core and 1.1 min with four CPU cores. For the AA-data set, CONSEL took 9.4 min and IQ-TREE 2 took 8.7 min with one CPU core and 2.4 min with four CPU cores. IQ-TREE provides added convenience by calculating site log-likelihoods itself, rather than relying on site log-likelihood output provided by other software.

Scalability with Large Data Sets

IQ-TREE 2 implements several features to facilitate the analysis of large data sets. It uses multithreading to speed up computations in a range of areas and can automatically determine the best number of threads for the computer at hand. IQ-TREE 2 also parallelizes the computation across cluster nodes using Message Passing Interface (Snir et al. 1998). IQ-TREE 2 periodically writes a compressed checkpoint file that enables resumption of an interrupted analysis. IQ-TREE 2 also provides a memory-saving mode (Izquierdo-Carrasco et al. 2012), that is automatically invoked when the memory requirement exceeds the RAM size, and a safe mode (-safe option) to avoid numerical underflow for taxon-rich alignments (automatically invoked for data sets with >2,000 sequences).

We benchmarked the memory-saving mode, where the RAM consumption is reduced by half (-mem 0.5). For the DNA-data set, IQ-TREE 2 took 27 and 32.8 m (21% increase) under the full- and half-memory mode, respectively, whereas the same analysis needed 5.7 and 6.2 h (9% increase) for the AA-data set. This increase in computing times is insignificant

compared with 50% saving in memory, which could otherwise be a bottleneck for very large data sets and mixture models requiring hundreds of GB of RAM.

Documentation, User Support, and Workshop Materials

An extensive user manual, quick start guide, tutorials, and command reference are available (<http://www.iqtree.org/doc>, last accessed February 6, 2020). We actively maintain a forum (<https://groups.google.com/d/forum/iqtree>, last accessed February 6, 2020) for user support, bug reports, and feature requests. We regularly teach IQ-TREE at the Workshop on Molecular Evolution (<https://molevolworkshop.github.io>, last accessed February 6, 2020), the Workshop on Virus Evolution and Molecular Epidemiology (<https://rega.kuleuven.be/cev/veme-workshop/>, last accessed February 6, 2020), and the Workshop on Phylogenomics (<http://evomics.org>, last accessed February 6, 2020). Most workshop materials are freely available at <http://www.iqtree.org/workshop/> (last accessed February 6, 2020).

Supplementary Material

Supplementary data are available at *Molecular Biology and Evolution* online.

Acknowledgments

The authors thank Suha Naser, Cassius Manuel, and Lam-Tung Nguyen for fruitful discussions, Lukasz Reszczyński for integrating the terraphast library, Ian Brennan for the logo design, Dimitri Hoehler for checking the IQ-TREE code quality (<https://github.com/adrianzap/softwape/wiki/Code-Quality-Benchmark>, last accessed February 6, 2020), Alexandros Stamatakis and an anonymous reviewer for constructive comments on the manuscript, and more than 100 users for helpful feedback and bug reports. Their names are listed in the full release notes (<http://www.iqtree.org/release>, last accessed February 6, 2020). This work was supported by the Austrian Science Fund (Grant No. I-2805-B29) to A.v.H., by the Australian National University Futures Scheme grant to R.L., and by the European Research Council under the European Union's Horizon 2020 research and innovation programme (Grant No. 714774) to D.S.

References

- Afgan E, Baker D, Batut B, van den Beek M, Bouvier D, Cech M, Chilton J, Clements D, Coraor N, Gruning BA, et al. 2018. The Galaxy platform for accessible, reproducible and collaborative biomedical analyses: 2018 update. *Nucleic Acids Res.* 46(W1):W537–W544.
- Anisimova M, Gascuel O. 2006. Approximate likelihood-ratio test for branches: a fast, accurate, and powerful alternative. *Syst Biol.* 55(4):539–552.
- Anisimova M, Gil M, Dufayard JF, Dessimoz C, Gascuel O. 2011. Survey of branch support methods demonstrates accuracy, power, and robustness of fast likelihood-based approximation schemes. *Syst Biol.* 60(5):685–699.
- Biczok R, Bozsok P, Eisenmann P, Ernst J, Ribizel T, Scholz F, Trefzer A, Weber F, Hamann M, Stamatakis A. 2018. Two C plus plus libraries for counting trees on a phylogenetic terrace. *Bioinformatics* 34:3399–3401.
- Bolyen E, Rideout JR, Dillon MR, Bokulich NA, Abnet CC, Al-Ghalith GA, Alexander H, Alm EJ, Arumugam M, Asnicar F, et al. 2019. Reproducible, interactive, scalable and extensible microbiome data science using QIIME 2. *Nat Biotechnol.* 37(8):852–857.
- Boussau B, Gouy M. 2006. Efficient likelihood computations with non-reversible models of evolution. *Syst Biol.* 55(5):756–768.
- Chernomor O, Minh BQ, von Haeseler A. 2015. Consequences of common topological rearrangements for partition trees in phylogenomic inference. *J Comput Biol.* 22(12):1129–1142.
- Chernomor O, von Haeseler A, Minh BQ. 2016. Terrace aware data structure for phylogenomic inference from supermatrices. *Syst Biol.* 65(6):997–1008.
- Crotty SM, Minh BQ, Bean NG, Holland BR, Tuke J, Jermin LS, von Haeseler A. Forthcoming 2019. GHOST: recovering historical signal from heterotachously-evolved sequence alignments. *Syst Biol.*
- Dornburg A, Fisk JN, Tamagnan J, Townsend JP. 2016. PhyloInformR: phylogenetic experimental design and phylogenomic data exploration in R. *BMC Evol Biol.* 16(1):262.
- Emms DM, Kelly S. 2015. OrthoFinder: solving fundamental biases in whole genome comparisons dramatically improves orthogroup inference accuracy. *Genome Biol.* 16(1): 157.
- Felsenstein J. 1981. Evolutionary trees from DNA sequences—a maximum likelihood approach. *J Mol Evol.* 17(6):368–376.
- Felsenstein J. 2004. Inferring phylogenies. Sunderland (MA): Sinauer Associates.
- Fong JJ, Brown JM, Fujita MK, Boussau B. 2012. A phylogenomic approach to vertebrate phylogeny supports a turtle-archosaur affinity and a possible paraphyletic lissamphibia. *PLoS One* 7(11):e48990.
- 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.
- Grama A, Karypis G, Kumar V, Gupta A. 2003. Introduction to parallel computing. Harlow (UK): Pearson Education Limited.
- Gu X, Fu YX, Li WH. 1995. Maximum-likelihood-estimation of the heterogeneity of substitution rate among nucleotide sites. *Mol Biol Evol.* 12(4):546–557.
- Guennebaud G, Jacob B, et al. 2010. Eigen v3. Version 3. Available from: <http://eigen.tuxfamily.org>. Accessed February 6, 2020.
- Guindon S, Dufayard JF, 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.
- Hadfield J, Megill C, Bell SM, Huddleston J, Potter B, Callender C, Sagulenko P, Bedford T, Neher RA. 2018. Nextstrain: real-time tracking of pathogen evolution. *Bioinformatics* 34(23):4121–4123.
- Hoang DT, Chernomor O, von Haeseler A, Minh BQ, Le SV. 2018. UFBoot2: improving the ultrafast bootstrap approximation. *Mol Biol Evol.* 35(2):518–522.
- Izquierdo-Carrasco F, Gagneur J, Stamatakis A. 2012. Trading memory for running time in phylogenetic likelihood computations. *Bioinformatics Conference*. Vilamoura, Portugal.
- Kalyaanamoorthy S, Minh BQ, Wong TKF, von Haeseler A, Jermin LS. 2017. ModelFinder: fast model selection for accurate phylogenetic estimates. *Nat Methods.* 14(6):587–589.
- Kozlov AM, 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.
- Lanfear R, Calcott B, Ho SY, Guindon S. 2012. PartitionFinder: combined selection of partitioning schemes and substitution models for phylogenetic analyses. *Mol Biol Evol.* 29(6):1695–1701.
- Le SQ, Dang CC, Gascuel O. 2012. Modeling protein evolution with several amino acid replacement matrices depending on site rates. *Mol Biol Evol.* 29(10):2921–2936.
- Le SQ, Gascuel O. 2010. Accounting for solvent accessibility and secondary structure in protein phylogenetics is clearly beneficial. *Syst Biol.* 59(3):277–287.
- Le SQ, Lartillot N, Gascuel O. 2008. Phylogenetic mixture models for proteins. *Philos Trans R Soc B* 363:3965–3976.

- Lemey P, Salemi M, Vandamme A-M. 2009. The phylogenetic handbook: a practical approach to phylogenetic analysis and hypothesis testing. New York: Cambridge University Press.
- Lewis PO. 2001. A likelihood approach to estimating phylogeny from discrete morphological character data. *Syst Biol*. 50(6):913–925.
- Mayrose I, Graur D, Ben-Tal N, Pupko T. 2004. Comparison of site-specific rate-inference methods for protein sequences: empirical Bayesian methods are superior. *Mol Biol Evol*. 21(9):1781–1791.
- Minh BQ, Hahn MW, Lanfear R. 2018. New methods to calculate concordance factors for phylogenomic datasets. *bioRxiv* 487801, doi: 10.1101/487801.
- Minh BQ, Nguyen MAT, von Haeseler A. 2013. Ultrafast approximation for phylogenetic bootstrap. *Mol Biol Evol*. 30(5):1188–1195.
- Mirarab S, Reaz R, Bayzid MS, Zimmermann T, Swenson MS, Warnow T. 2014. ASTRAL: genome-scale coalescent-based species tree estimation. *Bioinformatics* 30(17):i541–548.
- Moler C, Loan CV. 1978. Nineteen dubious ways to compute the exponential of a matrix. *SIAM Rev*. 20(4):801–836.
- Morel B, Kozlov AM, Stamatakis A. 2019. ParGenes: a tool for massively parallel model selection and phylogenetic tree inference on thousands of genes. *Bioinformatics* 35(10):1771–1773.
- Nguyen LT, Schmidt HA, von Haeseler A, Minh BQ. 2015. IQ-TREE: a fast and effective stochastic algorithm for estimating maximum-likelihood phylogenies. *Mol Biol Evol*. 32(1):268–274.
- Norris JR. 1997. Markov chains. Cambridge: Cambridge University Press.
- Price MN, Dehal PS, Arkin AP. 2010. FastTree 2—approximately maximum-likelihood trees for large alignments. *PLoS One* 5(3):e9490.
- Sanderson MJ, McMahon MM, Steel M. 2011. Terraces in phylogenetic tree space. *Science* 333(6041):448–450.
- Schmidt HA, Strimmer K, Vingron M, von Haeseler A. 2002. TREE-PUZZLE: maximum likelihood phylogenetic analysis using quartets and parallel computing. *Bioinformatics* 18(3):502–504.
- Schrempf D, Minh BQ, De Maio N, von Haeseler A, Kosiol C. 2016. Reversible polymorphism-aware phylogenetic models and their application to tree inference. *J Theor Biol*. 407:362–370.
- Schrempf D, Minh BQ, von Haeseler A, Kosiol C. 2019. Polymorphism-aware species trees with advanced mutation models, bootstrap, and rate heterogeneity. *Mol Biol Evol*. 36(6):1294–1301.
- Shimodaira H. 2002. An approximately unbiased test of phylogenetic tree selection. *Syst Biol*. 51(3):492–508.
- Shimodaira H, Hasegawa M. 1999. Multiple comparisons of log-likelihoods with applications to phylogenetic inference. *Mol Biol Evol*. 16(8):1114–1116.
- Shimodaira H, Hasegawa M. 2001. CONSEL: for assessing the confidence of phylogenetic tree selection. *Bioinformatics* 17(12):1246–1247.
- Snir M, Otto SW, Huss-Lederman S, Walker DW, Dongarra J. 1998. MPI: the complete reference—the MPI core. Cambridge (MA): The MIT Press.
- Stamatakis A. 2014. RAxML version 8: a tool for phylogenetic analysis and post-analysis of large phylogenies. *Bioinformatics* 30(9):1312–1313.
- Strimmer K, Rambaut A. 2002. Inferring confidence sets of possibly misspecified gene trees. *Proc R Soc Lond B* 269(1487):137–142.
- Strimmer K, von Haeseler A. 1997. Likelihood-mapping: a simple method to visualize phylogenetic content of a sequence alignment. *Proc Natl Acad Sci U S A*. 94(13):6815–6819.
- Wang HC, Minh BQ, Susko E, Roger AJ. 2018. Modeling site heterogeneity with posterior mean site frequency profiles accelerates accurate phylogenomic estimation. *Syst Biol*. 67(2):216–235.
- Whelan NV, Kocot KM, Moroz TP, Mukherjee K, Williams P, Paulay G, Moroz LL, Halanych KM. 2017. Ctenophore relationships and their placement as the sister group to all other animals. *Nat Ecol Evol*. 1(11):1737–1746.
- Woodhams MD, Fernandez-Sanchez J, Sumner JG. 2015. A new hierarchy of phylogenetic models consistent with heterogeneous substitution rates. *Syst Biol*. 64(4):638–650.
- Yang Z. 1994a. Estimating the pattern of nucleotide substitution. *J Mol Evol*. 39(1):105–111.
- Yang Z. 1994b. Maximum likelihood phylogenetic estimation from DNA sequences with variable rates over sites: approximate methods. *J Mol Evol*. 39(3):306–314.
- Zhou XF, Shen XX, Hittinger CT, Rokas A. 2018. Evaluating fast maximum likelihood-based phylogenetic programs using empirical phylogenomic data sets. *Mol Biol Evol*. 35(2):486–503.