Syst. Biol. 61(1):1–11, 2012

DOI:10.1093/sysbio/syr074
Advance Access publication on August 9, 2011

Guided Tree Topology Proposals for Bayesian Phylogenetic Inference

SEBASTIAN HÖHNA[1,*] AND ALEXEI J. DRUMMOND[2,3]

[1]*Department of Mathematics, Stockholm University, SE-106 91 Stockholm, Sweden;* [2]*Department of Computer Science, University of Auckland, Private Bag 92019, Auckland 1142, New Zealand; and* [3]*Allan Wilson Centre for Molecular Ecology and Evolution, University of Auckland, Private Bag 92019, Auckland 1142, New Zealand;*
*[*]Correspondence to be sent to: Department of Mathematics, Stockholm University, 106 91 Stockholm, Sweden;
E-mail: hoehna@math.su.se*

*Received 11 June 2010; reviews returned 15 September 2010; accepted 2 March 2011
Associate Editor: Mark Holder*

Abstract.—Increasingly, large data sets pose a challenge for computationally intensive phylogenetic methods such as Bayesian Markov chain Monte Carlo (MCMC). Here, we investigate the performance of common MCMC proposal distributions in terms of median and variance of run time to convergence on 11 data sets. We introduce two new Metropolized Gibbs Samplers for moving through "tree space." MCMC simulation using these new proposals shows faster average run time and dramatically improved predictability in performance, with a 20-fold reduction in the variance of the time to estimate the posterior distribution to a given accuracy. We also introduce conditional clade probabilities and demonstrate that they provide a superior means of approximating tree topology posterior probabilities from samples recorded during MCMC. [Bayesian inference; Gibbs sampling; Markov chain Monte Carlo; phylogenetics; posterior probability distribution; tree topology proposals.]

The Markov chain Monte Carlo (MCMC) algorithm has been known in statistics for many decades (Metropolis et al. 1953; Hastings 1970); however, the full potential of the algorithm was slow to be appreciated. The MCMC algorithm samples a parameter state Ψ (e.g., a phylogenetic tree) from a Markov chain, where new states Ψ' (e.g., new phylogenetic trees) are proposed by proposal distribution $p(\Psi'|\Psi)$ and accepted with probability $\alpha_H = \frac{\pi(\Psi')\times p(\Psi|\Psi')}{\pi(\Psi)\times p(\Psi'|\Psi)}$. Here, $\pi(\Psi)$ denotes the posterior probability of Ψ. The transition matrix defines the probabilities of a jump from state Ψ to Ψ' and being accepted for any discrete parameter. For continuous parameters, the transition kernel, which is the proposal distribution combined with the acceptance probability, describes moves in the parameter space. In this paper, we do not distinguish between transition of discrete parameters and continuous parameters and simply call the mechanism to propose a new state an *operator*.

MCMC has now been applied to many problems, especially those that require approximations of difficult high-dimensional distributions, such as the posterior distribution of phylogenetic trees. Many other applications for the MCMC algorithm are known, such as statistical physics, molecular simulation, dynamic system analysis, and computer vision. MCMC transition kernels need to be adapted for the specific problem domain to achieve a high efficiency. In 1996, Markov chains were first introduced for sampling among phylogenetic trees (Rannala and Yang 1996; Mau and Newton 1997; Li et al. 2000). Since that time, tree proposal distributions have not changed much, although the literature is full of more complex concepts of transition kernels for other applications (e.g., Gilks et al. 1996; Brooks 1998; Liu 2001).

Bayesian phylogenetic inference is inherently difficult because the state space increases superexponentially with the number of taxa under study. In addition, current MCMC approaches propose trees by random perturbations of the current tree, both leading to unnecessarily small effective sample sizes and large variance between runs. An ideal MCMC run would converge fast with a high degree of reproducibility. Although there are many other parameters in a typical MCMC run, the tree topology has a crucial role in phylogenetic analysis. Designing a good proposal kernel for tree topologies is the most challenging aspect of implementing Bayesian MCMC algorithms. Furthermore, samples from the full posterior distribution of other parameters can often only be accurately estimated if the posterior distribution on tree topologies is sampled accurately. Hence, a transition kernel that moves effectively around the space of tree topologies is very desirable.

The current tree proposal distributions implemented in software packages such as BEAST (Drummond and Rambaut 2007) and MrBayes (Ronquist and Huelsenbeck 2003) are constructed relatively simplistically. New trees are proposed at random from some well-defined neighborhood around the current tree. Not much is known about the performance of different transition kernels for phylogenetic inference. Only recently, the first attempt to evaluate such proposal distributions was performed by Lakner et al. (2008) for unrooted trees and a second attempt by Höhna et al. (2008) for time trees (rooted and clock constrained). The results in Lakner et al. (2008) and Höhna et al. (2008) show that the design of the currently used transition kernels leads to a low rate of accepted transitions, unless a small neighborhood is used. Consequently, the MCMC algorithm needs long computation times to give reliable estimates for the parameters under study. New proposal distributions need to be explored to find more efficient MCMC algorithms for sampling the posterior distribution in tree space. In this paper, we will consider new MCMC transition kernels for Bayesian phylogenetic inference, partially inspired by kernel designs for other domains of application.

Materials and Methods

The two most common operations on a tree are the prune-and-regraft algorithm and the subtree-swap (SS) algorithm. The prune-and-regraft algorithm, that is, applied in the Subtree-Prune-and-Regraft (Swofford et al. 1996) operator and the Fixed-Nodeheight-Prune-and-Regraft (FNPR, Höhna et al. 2008) operator, selects a random subtree and reattaches the subtree at a new random branch (e.g., Fig. 1). The SS algorithm (e.g., Drummond et al. 2002) exchanges two random subtended subtrees (Fig. 2). In their current application, both algorithms are stochastic: They propose a new tree by random perturbation of the current tree. This results in many proposals being rejected because they result in a tree with low likelihood. Ideally, the proposal mechanism would propose trees proportional to their posterior probability, obviating the need for MCMC entirely. However, an efficient means of doing this for phylogenetic problems has yet to be demonstrated. An intermediate and achievable goal would be to apply weights to neighboring trees that focus proposals on promising alternatives. These algorithms, such as the Gibbs sampler, are computationally more expensive or require some knowledge of the conditional distribution but generally have a higher efficiency due to a higher acceptance rate.

Gibbs Sampler

The Gibbs sampler is a special case of the Metropolis–Hastings algorithm (Geman and Geman 1984, Gilks1996). The Gibbs sampler samples from the conditional posterior distribution, instead of the full posterior distribution $P(X_1, X_2, \ldots, X_k|Y)$ with k variables. Therefore, the Gibbs sampler fixes all but one variable and samples directly from this conditional distribution $P(X_i|X_1, \ldots, X_{i-1}, X_{i+1}, \ldots, X_k, Y)$. Then, a new parameter X' is selected. The choice of the parameter to be updated can be performed either iteratively or by drawing randomly. Next, a new value for the parameter is conditionally sampled. The Gibbs sampler has an acceptance probability equal to one and therefore a high transition probability. Here, the acceptance probability denotes the probability of accepting the new proposed state (including proposals of the same state as the current state of the chain), and the transition probability denotes the probability of accepting new states (i.e., leaving the current state). However, the Gibbs sampler can be applied only if the conditional distribution of the parameters is known or if each probability can be computed (e.g., when the values are drawn from a discrete state space).

Metropolized Gibbs Sampler

The Gibbs sampler has the highest possible acceptance probability (one) but not the highest transition probability. The transition probability is the probability of leaving the current state, and a high transition probability leads to a faster mixing Markov chain. In this paper, we consider a modification to the Gibbs sampler that increases the transition probability. The transition probability is strongly correlated with the asymptotic variance of the estimates (Peskun 1973; Mira 2001) and therefore strongly influences the performance of the MCMC algorithm

.

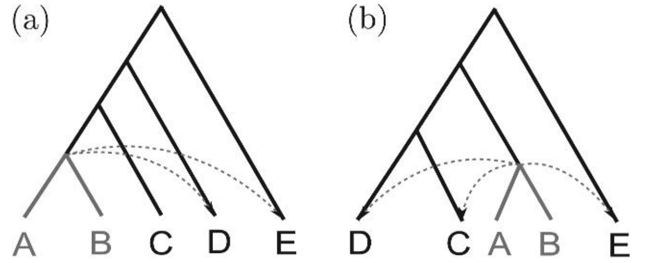

FIGURE 1. A sample SS with five taxa. a) shows the starting tree with clade $\mathcal{C}_{AB}$ chosen first. b) presents the tree after swapping clade $\mathcal{C}_{AB}$ with node D and gives the corresponding backward proposals. Here, the number of possible proposals is not symmetric (the forward and backward proposal probabilities are not equal) and the actual proposal possibilities are indicated. For simplicity of this example, we assume each weight of the tree topology to be equal to one. In a) the probability of choosing clade $\mathcal{C}_{AB}$ first and then proposing to swap with node D is hence $\frac{1}{2}$, as indicated by the arcs. The backward proposal probabilities (exemplified in b) has the probability $\frac{1}{3}$.

A Gibbs sampler can be modified by prohibiting the current state as a new proposal. Therefore, the operator is forced to propose different states more often. Liu (1996) demonstrated that the Metropolized Gibbs Sampler has a higher transition rate and, hence, higher performance. However, the difference decreases with an increasing number of states in the proposal distribution. Therefore, only a discrete parameter can be

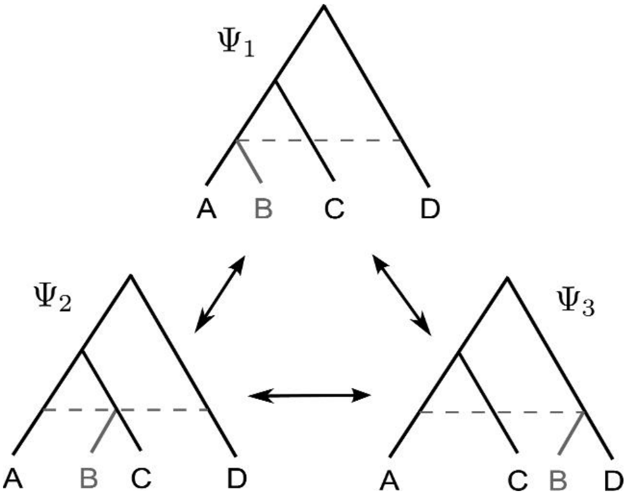

FIGURE 2. A sample proposal distribution for the FNPR/MGFNPR operator on a four-taxon tree. The subtree containing taxon B is pruned from tree Ψ_1. Possible new trees are Ψ_1, Ψ_2, and Ψ_3, whereas each tree can have a weight assigned to it. The Gibbs-like sampler proposes the three trees proportional to their posterior probability and the Metropolized Gibbs sampler excludes tree Ψ_1.

"Metropolized" to achieve a higher transition rate. Nevertheless, in most situations, the chain is in a state that has a high probability compared with most other states in the proposal distribution. This leads to observable improvements when the Metropolized Gibbs Sampler is used for proposing new tree topologies (Liu 1996).

Topology proposals are usually operators in a continuous parameter space because branch lengths and node heights are continuous parameters. Unfortunately, we cannot easily sample from the conditional probability distribution of a tree with arbitrary branch lengths (or node heights). To benefit from using Metropolized versions of the Gibbs sampler, we have developed versions of tree proposals, which propose a state from a discrete set of possible proposals.

First, as a representative for the prune-and-regraft operators, the FNPR operator is extended. The FNPR operator does not change the node heights and hence has a discrete proposal distribution, which is necessary to gain the desired advantage of metropolizing the proposal. Select a node i that is not the root or a direct child of the root. Then, the parent node of i is pruned and all reattachment points j are determined. For each potential reattachment point, a weight is calculated for the corresponding tree topology. Finally, a new tree is proposed by drawing a tree randomly accordingly to the weights (Fig. 2). Although the ordinary Gibbs sampler has an acceptance rate of one, the Hastings ratio needs to be calculated when we use weights other than the posterior probabilities. To keep the operators general, we supply the Hastings ratio. Let us denote the weight for the current tree by w_{ji}, the weight for the proposed tree by w_{ij}, and the set of weights for the possible proposals by $\mathbb{W}_i$ and $\mathbb{W}_j$ at tree Ψ and Ψ', respectively. The Hastings ratio for any weight is computed by

$$\alpha_H = \frac{P(\Psi|\Psi')}{P(\Psi'|\Psi)} = \frac{\dfrac{w_{ji}}{\sum\limits_{\omega \in \mathbb{W}_j} \omega}}{\dfrac{w_{ij}}{\sum\limits_{\omega \in \mathbb{W}_i} \omega}}. \tag{1}$$

Many different weighting functions are feasible, for example, the parsimony score, the Conditional Clade Probability (CCP) score (see below), or the posterior probability. When the posterior probability is used, the operators are Gibbs like. We name the posterior-probability-guided-prune-and-regraft operator the Metropolized-Gibbs Fixed-Nodeheight-Prune-and-Regraft (MGFNPR) operator.

Second, the SS is extended identically. A subtree i is arbitrarily chosen with i not being the root or the older child of the root. Then, every alternative node j is chosen with which i can be swapped. For all resulting trees, a weight is calculated and finally, a tree is proposed randomly, proportional to the weights (see Fig. 1).

The SS operator is an example of an asymmetric proposal distribution. Figure 1 shows the difference in the proposal distribution depending on the the current tree. Consequently, we define the Hastings ratio as follows. Let w_{ij} denote the weight for swapping node i with j at tree Ψ and w'_{ij} at tree Ψ', respectively. Further, $\mathbb{W}_i$ denotes the set of weights for all possible new trees when node i is chosen first and the current tree is Ψ. Then, the Hastings ratio α_H is derived from the following equations:

$$P(\Psi'|\Psi) = \frac{w_{ij}}{\sum\limits_{\omega \in \mathbb{W}_i} \omega},$$

$$P(\Psi|\Psi') = \frac{w'_{ij}}{\sum\limits_{\omega \in \mathbb{W}'_i} \omega},$$

$$\alpha_H = \frac{P(\Psi|\Psi')}{P(\Psi'|\Psi)} = \frac{w'_{ij} \sum\limits_{\omega \in \mathbb{W}_i} \omega}{w_{ij} \sum\limits_{\omega \in \mathbb{W}'_i} \omega}.$$

We name the posterior probability-guided SS operator as the Metropolized-Gibbs Subtree-Swap (MGSS) operator.

Pruned Gibbs Sampling

In the previous section, we developed a Metropolized Gibbs Sampler for clock-constrained rooted phylogenetic trees. However, the Metropolized Gibbs Sampler requires intensive computation for each iteration. Some computations might not be necessary because many proposals are rejected. Hence, the Metropolized Gibbs Sampler is improved by narrowing the proposal distribution to the more likely proposals.

After some initial iterations—the burn-in phase—the current tree of the MCMC run is expected to have a high posterior probability. The posterior probability of a tree can be very sensitive to large changes to the topology or branch lengths. Therefore, local changes on the tree topology are assumed to preserve the posterior probability better than global changes. On the other hand, proposals from a local operator are more correlated to the current tree than proposals from global operators. Furthermore, an operator with a global effect on the tree is more likely to propose new trees from different islands of high posterior probability.

We extend the MGFNPR operator and MGSS operator to propose only trees separated by less than a maximal distance, where the distance is defined by the number of nodes along the path between the original position of the pruned (or swapped) subtree to the newly proposed position. We achieve this restriction by computing a pruning or swapping distance and ensuring that this distance does not exceed a certain threshold. All trees with a larger distance are discarded to save computation time. We suggest a pruning distance of $\frac{2n-1}{10}$ where n is the number of taxa. This value has been chosen arbitrarily but has been proven by simulations to give a good performance in our study. We name these operators as the pruned Metropolized-Gibbs-Fixed-Nodeheight Prune-and-Regraft (pMGFNPR) operator and the pruned MGSS () operator. The idea is similar to the idea of Huelsenbeck et al. (2008) and Lakner et al. (2008), which they called an extended Tree-Bisection-and-Reconnection operator. However, they applied this technique on stochastic operators for unrooted trees.

Performance Analysis for Tree Proposal Operators

The performance of the tree proposal operators can be measured by their ability to converge to the target distribution and the number of samples or time needed for the MCMC to produce sufficient samples from the target distribution, so-called convergence diagnostics. Diagnosing convergence of an MCMC chain is a nontrivial problem and is the subject of extensive theoretical research (see Cowles and Carlin 1996 for a review). Here, we rely on a similarity measurement between the sampled distribution and the target distribution that was introduced by Lakner et al. (2008) and Höhna et al. (2008). We call $\mathbb{C}$ the set of possible clades. For each clade $\mathcal{C}$, it is possible to compute the absolute difference between the clade frequency $s_{\mathcal{C}}$ in the sampled distribution and the clade frequency $t_{\mathcal{C}}$ in the target distribution. We call $\delta = \max_{\mathcal{C}\in\mathbb{C}}(|s_{\mathcal{C}} - t_{\mathcal{C}}|)$ the maximum deviation of the clade frequencies. Once the target distribution of clades is accurately estimated, it is then possible to monitor the convergence of any given MCMC run by monitoring δ as samples are produced. We propose to use the computation time elapsed to reach $\delta < \epsilon$ for the first time as a metric to evaluate the efficiency of operators.

The true target distribution of clades is unkown. To estimate it, we perform a set of long MCMC runs—the so-called "golden runs." Each golden run is simply an extremely long MCMC chain, and therefore, it produces samples whose distribution reflects as accurately as possible the true target distribution. For each data set, 10 golden runs were performed, with the BEAST v1.4.8 defaults. Each run had 1 billion iterations with samples every 1000 iterations, which is far longer than the usual analysis performed on data sets of the size we studied. The estimated error of a data set is taken from the maximal estimated error of all clade frequencies in the data set. Let $\mathrm{SE}_{\bar{x}}$ denote the estimated error, $\mathcal{C}$ any clade in $\mathbb{C}$, $s_{f(\mathcal{C})}$ the standard deviation of the posterior probability for clade $\mathcal{C}$, and $n = 10$ (the number of runs).

$$\mathrm{SE}_{\bar{x}} = \max_{\mathcal{C}\in\mathbb{C}}\left(\frac{s_{f(\mathcal{C})}}{\sqrt{n}}\right).$$

We observed that the maximal estimated error was below 0.04% (Table 1). Even though one might expect the estimated error to increase with the size of the data set, this is not the case.

Conditional Clade Probability

In the section above, we described how the true posterior probability distribution on trees is commonly estimated by the clade frequencies and how this is used as a convergence diagnostic. The approximation error of the posterior probability for infrequently sampled tree topologies is very high. In the following, we introduce a new algorithm to approximate the whole posterior probability distribution on tree topologies, even when only a small fraction of the total tree space has been sampled by a converged MCMC run.

Commonly, the frequency of samples of a particular tree topology is used to approximate its posterior probability. Instead, one can use the posterior probabilities of the clades contained in the tree to approximate the tree topology's posterior probability. The additive binary (AB) coding scheme (Farris et al. 1970; Brooks 1981) is such a method. The AB coding scheme assumes all clades are independent of one another, but, this is not true in general. The constitution of clades depends on the parent clades. For instance, if the clade $\mathcal{C}_1$ consisting of taxa A, B, and C is observed, then the chance of observing the clade $\mathcal{C}_2$ consisting of A and B is $P(\mathcal{C}_2|\mathcal{C}_1)$ (see Fig. 3). Therefore, we introduce the concept of *CCP* and extend the concept of the AB coding scheme.

The weighting for the clades is straightforward. The weight or probability $w_{\mathcal{C}_1|\mathcal{C}_2}$ for clade $\mathcal{C}_1$ given the parent clade $\mathcal{C}_2$ equals the joint frequency of observations for this clade and its parent clade $f(\mathcal{C}_1, \mathcal{C}_2)$ over the frequency of the parent clade being present $f(\mathcal{C}_2)$

$$w_{\mathcal{C}_1|\mathcal{C}_2} = \frac{f(\mathcal{C}_1, \mathcal{C}_2)}{f(\mathcal{C}_2)}.$$

TABLE 1. Details of the 11 real data sets used in this study

Data set	Number of species	Number of nucleotides	Type of data	TreeBASE	Estimated error (in %)
DS 1	27	1949	rRNA, 18s	M336	0.0277
DS 2	29	2520	rDNA, 18s	M501	0.0021
DS 3	36	1812	mtDNA, COII (1–678), cytb (679–1812)	M1510	0.0021
DS 4	41	1137	rDNA, 18s	M1366	0.0048
DS 5	50	378	Nuclear protein coding, wingless	M3475	0.0393
DS 6	50	1133	rDNA, 18s	M1044	0.0052
DS 7	59	1824	mtDNA, COII, and cytb	M1809	0.0003
DS 8	64	1008	rDNA, 28s	M755	0.0007
DS 9	67	955	Plastid ribosomal protein, s16 (rps16)	M1748	0.0038
DS 10	67	1098	rDNA, 18s	M520	0.0338
DS 11	71	1082	rDNA, internal transcribed spacer	M767	0.0010

Notes: rDNA = ribosomal DNA; rRNA = ribosomal RNA; mtDNA = mitochondial DNA; COII = cytochrome oxidase subunit II.

Approximating the posterior probability of a phylogenetic tree using a binary representation.—The algorithm to approximate the posterior probability of a phylogenetic tree is based on the posterior probabilities (or weights) associated with its constituent clades. Let us define a tree Ψ as a set of clades $\mathbb{C}_\Psi$. The posterior probability of the tree is then approximated using

$$\mathcal{P}(\Psi) = \prod_{\mathcal{C}\in\mathbb{C}_\Psi} (w_\mathcal{C} + \epsilon),$$

where $w_\mathcal{C}$ is the weight for the clade given by the binary representation. It is necessary to add a small value ϵ to each weight to avoid zero probabilities. An estimate for ϵ is $1 - \sqrt[m]{\frac{1}{2}}$ where m is the sample size. The reason for choosing this ϵ value is that if we have not observed a tree containing this clade in m samples, then the probability of observing it at least once was likely to be smaller than $\frac{1}{2}$ in m independent trials. Finally, the probability of the tree Ψ is

$$P(\Psi) = \frac{\mathcal{P}(\Psi)}{\sum\limits_{\Psi_i\in\mathbb{T}} \mathcal{P}(\Psi_i)},$$

with $\mathbb{T}$ being the set of all possible trees.

In order to establish the accuracy of the CCP method in approximating the posterior probabilities of tree topologies, we estimated the probability of all 105 trees in a simulated five-taxon data set by the MCMC sample frequency, the CCP model, and a simplistic scoring algorithm referred to as the weighted multiplicative binary (WMB) score. WMB uses the AB coding schemes and unconditional clade probabilities as its weights. We find that the WMB score provides a good approximation for the tree probability distribution only for frequently sampled trees, whereas the CCP model provides a good approximation of the tree probability distribution for all trees (Fig. 4). Furthermore, this can be used as a prior distribution for further Bayesian phylogenetic analysis (Ronquist et al. 2004) or weights for the proposals of the transition kernel. The latter was used during this research. In addition to the posterior probability of a tree, we used the CCP score for a tree as weights in the guided-tree-proposal operators. Compared with the posterior probability, the CCP score is a quick but less accurate approximation.

However, the CCP score needs a presampling step if it used to guide the proposals. A long presampling step tends to give very accurate approximations of the posterior probability but needs a very long time to compute. An insufficient presampling step tends to give inaccurate approximations of the posterior probability and, therefore, can misguide the proposal. We ran MCMC analyses with differing lengths of the presampling step (data no shown) to find a good trade-off between fast but accurate approximations. As a rule of thumb, we used a preburnin run of 10% of the actual chain length to obtain the samples for the CCP scores.

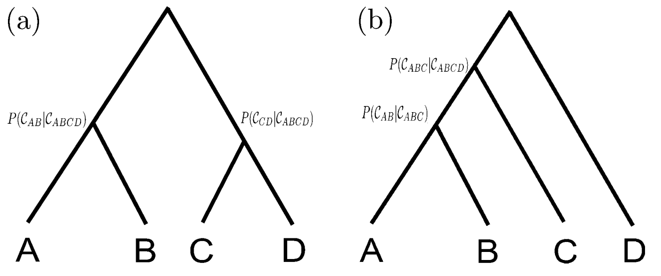
(a)
$P(\mathcal{C}_{AB}|\mathcal{C}_{ABCD})$
$P(\mathcal{C}_{CD}|\mathcal{C}_{ABCD})$
A
B
C
D
(b)
$P(\mathcal{C}_{ABC}|\mathcal{C}_{ABCD})$
$P(\mathcal{C}_{AB}|\mathcal{C}_{ABC})$
A
B
C
D

FIGURE 3. The constitution of conditional clade probabilities shown on two 4-taxon trees. Tree a) consists of the clades $\mathcal{C}_{ABCD}$, $\mathcal{C}_{AB}$, and $\mathcal{C}_{CD}$, where both clades $\mathcal{C}_{AB}$ and $\mathcal{C}_{CD}$ are subclades of clade $\mathcal{C}_{ABCD}$. The CCP score is then composed by $P(\mathcal{C}_{AB}|\mathcal{C}_{ABCD}) \times P(\mathcal{C}_{CD}|\mathcal{C}_{ABCD})$. On the other hand, tree b) consists of the clades $\mathcal{C}_{ABCD}$, $\mathcal{C}_{ABC}$, and $\mathcal{C}_{AB}$ with $\mathcal{C}_{ABC}$ being a subclade of $\mathcal{C}_{ABCD}$ and $\mathcal{C}_{AB}$ a sublade of $\mathcal{C}_{ABC}$. Hence, the CCP score is composed by $P(\mathcal{C}_{ABC}|\mathcal{C}_{ABCD}) \times P(\mathcal{C}_{AB}|\mathcal{C}_{ABC})$.

RESULTS

The data sets used in this study are 11 empirical data sets (see Lakner et al. 2008 for more details). The data sets range from 27 to 71 species and from 378 to 2520 nucleotides (Table 1). We chose these data sets so that we could benchmark our results against previous studies in a feasible amount of time (Tables 4 and 5).

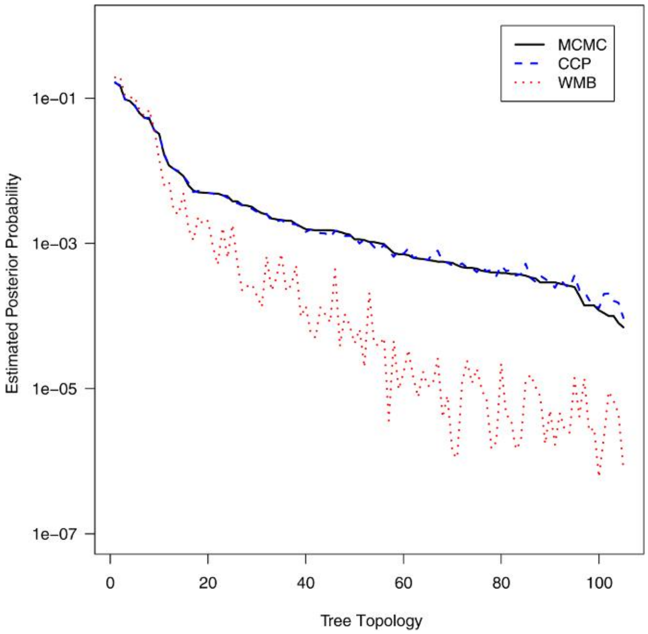

FIGURE 4. The estimated posterior probabilities based on the WMB score, CCP score, and MCMC sampling frequencies on a five-taxa data set containing all 105 distinct rooted tree topologies. The trees are sorted in decreasing sampling frequency. The WMB score and CCP score are established using the same sample of trees as used to plot the MCMC frequencies. The accuracy is high for both clade-based score on the frequently sampled trees but diverges extremely for the WMB score on the fewer sampled trees. This figure is available in black and white in print and in color at *Systematic Biology* online.

Set-up of the Empirical Analysis

In our study, we used a set of commonly used stochastic operators, the newly developed guided operators, and two mixtures of operators (Table 2; see also Höhna et al. 2008 for a review of the stochastic operators). The two mixtures of operators are the previously default setting of BEAST denoted as *BEAST default* and our newly proposed mixture *Mixture of Tree Proposal Operators (MTPO)*. BEAST default is made up of the the operators with weights given in parenthesis: Narrow-Exchange (15), Subtree-Slide (15), SS (3), and Wilson–Balding (3) (for descriptions of the operators, see, e.g., Wilson and Balding 1998; Drummond et al. 2002; Höhna et al. 2008). MTPO is made up of the operators: Narrow-Exchange (3), Nearest-Neighbor-Interchange (NNI) (3), and MGFNPR (4).

We performed 100 test runs for each operator on each of the 11 real data sets. For each data set and operator, the runs were stopped when either a maximal deviation of clade frequencies of 5% or 100,000,000 iterations were reached. The iterations and elapsed time till convergence were reported. Samples were taken every 100 steps. The performance evaluation was performed on a cluster of 94 iMacs with Intel's T7700 2.4 GHz Core 2 Duo Central Processing Units (CPUs). For simplicity, only one core on each machine was used.

Evaluation

Running times varied greatly (Fig. 6). The median CPU time till convergence is used as a proxy for the performance of an operator (Table 3). The variance in running times is also of interest because even an operator with a low median running time can have a substantial fraction of runs that take an extremely long time before they converge, causing estimates to vary significantly when the chain is stopped at an arbitrary time. A numerical comparison of the median running time is achieved by normalizing the performance per data set and averaging the results. Let t_{ij} denote the median running time for operator j on data set i, $\min(t_i)$ denote the minimal median running time on data set i of all operators, and k denote the number of data sets

$$s_j = \frac{\sum_{i=0}^{k} \frac{t_{ij}}{\min(t_i)}}{k}.$$

Table 3 shows the ranking of the operators according to the score defined in the formula above. The median running time represents the factor of how much longer the median running time is, averaged on all data sets. The variance represents the factor of how much worse the variance of running times for an operator is on average, which is determined in the same way as averaged median running times. The MTPO clearly produced the best results. A speed up of 52% over the previous mixture and an order of magnitude (20-fold) reduction in the variance was achieved.

TABLE 2. Characteristics of different tree proposal operators

Operator	Changes on topology	Effect on tree	Branch length	Proposal technique	Guidance function
FNPR	Direct	Global	Preserved	Prune-and-regraft	None
MGFNPR	Direct	Global	Preserved	Prune-and-regraft	Posterior probability
MGSS	Direct	Global	Preserved	SS	Posterior probability
CPP-pune-and-regraft	Direct	Global	Preserved	Prune-and-regraft	CCP
CCP-SS	Direct	Global	Preserved	SS	CCP
Narrow-Exchange	Direct	Local	Preserved	SS[a]	None
NNI	Direct	Local	Changed	SS[a]	None
pMGFNPR	Direct	Global	Preserved	Prune-and-regraft	Posterior probability
pMGSS	Direct	Global	Preserved	SS	Posterior probability
Wilson–Balding	Direct	Global	Changed	Prune and regraft	None
SS	Direct	Global	Preserved	SS	None
Subtree-Slide	Side effect	Mostly local	Changed	Branch change	None

Note: pMGSS = pruned Metropolized-Gibbs-Subtree-Swap.
[a] In this case equal to a prune-and-regraft.

TABLE 3. The median running time and the SD of the running times per operator over all 11data sets

Operator	Medianrunning time	SD of running times
MTPO	1.27	1.50
pMGFNPR	1.87	1.99
BEAST default	1.93	7.37
MGFNPR	2.86	3.50
pMGSS	2.95	39.67
Narrow-Exchange	3.63	28.02
NNI	5.3	42.20
FNPR	10.03	12.77
CCP-prune-and-regraft	14.01	79.53
MGSS	14.59	145.05
Wilson–Balding	25.21	24.02
SS	33.51	49.43
CCP-SS	50.08	195.05
Subtree-Slide	91.36	43.37

Note: The table is sorted according to the median running time.

Analyses on the different data sets show that, on average, the new guided operators converge fastest, followed by the unguided local operators (Narrow and NNI) and then the unguided global operators (FNPR, Wilson–Balding, and SS). Furthermore, it can be established that combining several operators creates a synergy that leads to better performance than the single operators. This synergy is illustrated, for instance, by the better performance of BEAST's default mixture of operators, compared with the individual component operators.

The efficiency of the guided operators depends on the type of guidance function. The posterior-probability-guided operators perform well on all data sets but need intensive computations. Therefore, the efficiency is low on data sets that are easy for other (i.e., stochastic) operators (e.g., data set DS11). However, the efficiency is better in each case compared with the unguided global operators. Furthermore, the efficiency is improved on every data set if the operator uses a restricted neighborhood. On average, 35% less computation time is needed for the pruned versus the unpruned MGFNPR operator. The CCP-guided operators converge faster for some data sets, such as DS3 and DS7. However, these operators rely heavily on the accuracy of the posterior estimation of the CCP. Runs with a poor approximation therefore perform extremely poorly.

Mossel and Vigoda (2005) created an artificial data set, which is difficult for prune-and-regraft operators but easier for SS operators. Ronquist et al. (2006) argued that those situations are unlikely to occur in real data. In the 11 real data sets studied in this research, we can confirm the assumption of Ronquist et al. (2006). No data set was particularly difficult for the prune-and-regraft operators. In contrast, the FNPR operator was always superior to the SS operator and data set DS5 seems particularly difficult for the SS operators (Tables 4 and 5).

The Subtree-Slide operator had a very poor performance on many data sets and produced many runs that failed to converge within the maximum allowed chain length. Nonetheless, the Subtree-Slide operator is at least as good as the global stochastic operators in 5 of the 11 real data sets (see Table 4 and 5). This result contradicts the conclusion of Lakner et al. (2008) who reported the worst efficiency for the Branch–Change operators. Nevertheless, the overall performance was the worst according to the median run time.

Tree Space Visualization

Shortcomings of one operator can be identified by tracing the path of the Markov chain through the tree space. However, tree space is multidimensional and visualizing it is challenging (Billera et al., 2001). Hillis et al. (2005) developed a tool using multidimensional scaling (MDS) to visualize a tree space. In their method, the distance between two trees is defined by the weighted or unweighted Robinson–Foulds (RF) distance. However, the RF distance does not reflect the number of steps an operator has to take on the shortest path between two trees. Matsen (2006) followed the same idea using MDS to turn the tree space into a two-dimensional space. Instead of the RF distance, Matsen used the NNI distance but without showing visual results.

We performed tree space visualization on the posterior distribution of trees for data set DS1. DS1 is the smallest of the real data sets but was comparatively difficult for most of the operators and therefore provided a good insight into the source of difficulties in sampling phylogenetic posterior distributions. Other data sets show similar properties, as they all contain at least two tree islands (data not shown). First, the trees contained in the 95% credible set were extracted. The set of trees is given in Table 6. Next, we calculated the NNI distances between every pair of trees. The resulting tree space was transformed with the MDS algorithm into a two-dimensional space. Figure 5 shows the tree space with all 15 trees and edges between trees with a NNI distance of one. The tree space is separated into 5 tree islands. Each island is built by a group of trees with at least one connection of only one NNI transformation to another tree of that island. Consequently, the distance between two tree islands is at least two NNI transformations.

The optimal scenario is when the operators connect one tree island directly to any other tree island. The likelihood of visiting the more distant islands decreases with the number of operations needed to reach it. Furthermore, if the tree island is not likely to be visited but has a high posterior probability, then the dwell time in that island must be higher to achieve the correct proportion of samples for the tree island. This leads to slow mixing between the distant parts in the tree space and high variance in convergence time.

The distances between islands have a strong influence on the transition probabilities between those islands. If the tree islands are more than three NNI transformations distant, then direct transitions between the tree islands become extremely unlikely. For instance, the only tree island with a distance of three or less from tree

TABLE 4. The median running times (in minutes) and SD in running times for each of the guided tree proposal operators and the two mixtures on the empirical data sets

Operator	MTPO		pMGFNPR		BEAST default		MGFNPR		pMGSS		CCP-prune-and-regraft		MGSS		CCP-SS	
	Mean	SD	Mean	SD	Mean	SD	Mean	SD	Mean	SD	Mean	SD	Mean	SD	Mean	SD
DS1	11.90	100.97	**10.14**	**82.91**	16.80	90.81	23.2	173.79	56.1	561.34	160.71	497.11	175.12	2159.49	559.14	910.58
DS2	1.75	5.52	2.100	9.15	**1.48**	**4.4**	7.97	51.15	6.22	467.36	1.69	247.62	20.51	2611.12	3.45	19.89
DS3	9.32	18.59	6.83	13.43	6.29	10.71	16.30	21.89	**4.2**	**5.4**	6.38	326.74	15.21	26.50	4.20	559.79
DS4	10.96	30.95	14.82	45.70	**5.53**	**8.98**	23.14	55.92	18.16	939.54	9.62	541.32	87.72	209.84	51.16	1884.64
DS5	14.23	22.14	**13.45**	**19.7**	53.37	156.90	16.33	25.35	122.5	533.10	169.6	450.81	580.73	3154.98	4393.54	1236.76
DS6	7.35	**6.93**	11.30	11.62	**6.44**	24.67	14.40	16.92	13.34	72.95	8.80	233.52	84.16	273.57	112.81	1491.8
DS7	5.0	**3.21**	5.67	4.53	5.6	121.48	7.90	4.54	6.6	265.87	3.85	792.65	32.37	2053.33	14.4	2825.78
DS8	3.43	6.36	5.35	9.88	**2.34**	**5.45**	4.86	17.80	3.12	438.37	3.30	691.17	29.7	52.88	15.77	1997.32
DS9	**7.88**	**4.48**	9.46	5.94	9.62	10.48	11.1	5.85	8.48	7.25	1029.36	911.6	46.45	25.71	979.46	1030.92
DS10	**15.43**	18.44	20.78	**14.82**	123.7	338.63	24.43	16.16	24.97	42.21	38.16	506.94	162.58	256.22	38.33	547.26
DS11	15.89	11.2	25.63	20.29	**7.34**	5.51	29.24	23.8	13.63	12.40	30.46	235.52	119.55	96.65	31.64	286.90

Note: The best performances of all operators, guided and stochastic, are marked in bold.

TABLE 5. The median running times (in minutes) and SD in running times for each stochastic tree proposal operator on the empirical data sets

Operator	Narrow-exchange		NNI		FNPR		Wilson–Balding		SS		Subtree-Slide	
	Mean	SD	Mean	SD	Mean	SD	Mean	SD	Mean	SD	Mean	SD
DS1	19.95	155.96	19.56	179.3	70.42	282.76	67.66	255.75	189.5	307.37	66.16	155.12
DS2	1.79	183.81	3.42	158.23	10.99	176.39	25.68	43.26	21.2	342.57	115.17	291.79
DS3	9.47	19.60	18.37	39.5	31.76	49.16	181.47	300.34	37.88	137.9	434.45	448.58
DS4	42.53	439.64	80.48	538.20	54.10	124.68	156.72	326.23	116.94	307.46	109.39	327.71
DS5	106.2	181.52	150.53	237.86	69.9	104.97	115.43	153.44	533.35	107.42	377.97	196.51
DS6	20.64	214.73	15.17	181.97	52.52	59.55	57.56	31.22	150.39	279.29	115.37	378.95
DS7	**3.31**	301.49	16.76	699.66	40.59	49.8	273.8	391.57	412.10	898.48	2232.40	582.46
DS8	2.77	242.97	4.40	355.49	42.9	74.57	43.56	31.21	106.14	143.44	32.97	153.99
DS9	10.90	14.50	12.6	17.66	57.9	30.55	73.96	32.76	107.58	62.42	45.21	46.3
DS10	210.27	393.99	222.24	465.70	115.40	70.47	285.44	241.47	366.93	324.17	504.66	470.40
DS11	7.96	**5.13**	8.24	8.24	123.2	102.33	205.4	104.96	204.94	98.75	63.56	26.51

Note: The best performances of all operators, guided and stochastic, are marked in bold.

TABLE 6. The 95% credible set of data set DS1

Tree	Tree island	Posterior probability (%)
1	1	60.44
2	1	19.27
3	2	7.61
4	1	1.73
5	2	1.01
6	3	0.89
7	1	0.73
8	2	0.68
9	1	0.54
10	4	0.48
11	1	0.46
12	5	0.38
13	1	0.35
14	4	0.35
15	1	0.34

island 3 is tree island 1 (Fig. 5). Hence, almost all transitions from tree island 3 are to tree island 1, although tree island 4 has a distance of only 4. Furthermore, direct transitions between tree islands become unlikely if one shortest path leads via another tree islands. This routing has a strong impact on the visiting times of a tree island of the Markov chain and hence the sampling frequency of trees.

DISCUSSION

Mixing in Tree Space

Two properties of the posterior distribution can contribute to slow convergence in phylogenetic MCMC. First, if the tree space is very flat (has many trees in the 95% credible set and little variations in the posterior probabilities of the trees), then more trees must be sampled in order to represent the posterior probability distribution correctly (e.g., data set DS9). Second, if the tree space is very spiky (has discretely separate tree islands each with high posterior probability, e.g., data set DS1), then the difficulty is in jumping from one spike to another.

Local operators need more transformations to pass valleys of less likely trees, whereas global operators need fewer transformations. But the proposal distribution of the global operators is much larger and fewer proposals are accepted because of the spiky nature of the tree space with only a few high probability trees. The impact of the low acceptance probability is greater than the gain of fewer steps through the valley of the less likely trees. Therefore, empirically it appears that the local operators outperform the global operators.

Visualizing tree space.—The detailed study of data set DS1 shows that the local operators are more efficient in mixing in the tree space than the stochastic global operators. The stochastic global operators perform poorly because of low acceptance probabilities. The transition probabilities between the tree islands are a good proxy for the mixing ability of an operator. Furthermore, the transition probabilities show the shortcomings of an operator. The data sets can be further analyzed by finding the shortest path between any two trees in the 95% credible set, which has the lowest transition probability and then designing an operator, which can propose jumps directly between the two trees. Also, future studies might evaluate the efficiency of mixing between tree islands (e.g., by measuring the mean access time, mean commute time, and mean cover time; Seary and Richards 1997).

Guided MCMC Operators

Global operators can be guided to prefer trees with a high posterior probability so that more tree proposals are accepted and the transition probability is higher. As a result, once the chain is in a valley, guided operators will traverse valleys between tree islands more quickly. The guidance functions can be quick and inaccurate or slow and accurate. The CCP model is used to estimate the probability of a proposal quickly, but it will have low accuracy unless a long period of training is used. This can be achieved by a good presampling of the tree space. The importance distributions used in this research required a substantial presampling phase, especially when the posterior distribution was very flat (e.g., data set DS9). Further improvement would require importance distributions that represent the posterior distribution more accurately with fewer samples. One idea for a better importance distribution would involve taking the marginal clade divergence times into account when evaluating a proposed tree, as a substantial fraction of trees with a high marginal probability may be rejected because the current divergence time assignments are unsuitable for the proposed tree topology.

Guiding the proposals by the posterior probability is slow but accurate, and the likelihood calculations of the proposed trees consumes the majority of the computational time. Nevertheless, a number of optimizations were investigated to make the likelihood calculation faster, such as recalculating only partial likelihoods when only parts of the tree are changed. Because the current implementation of the tree likelihood calculation in BEAST is efficient at reusing partial likelihoods, we believe that guiding the proposals by the likelihood is currently the most promising approach to Metropolized Gibbs Sampler for phylogenetic analysis.

CONCLUSION

This research has shown that single tree proposal operators can be improved when they are guided. The guidance function and the proposal distribution have a large impact on the performance. For the given proposal distributions, the MGFNPR and MGSS have the highest possible transition probabilities. They can only be improved by optimizing the implementation or finding a trade-off between speed and accuracy.

We introduced two new Metropolized Gibbs Sampler operators, and we verified their advantage over the ordinary Gibbs samplers for proposing new tree topologies. The pMGFNPR greatly increases the performance

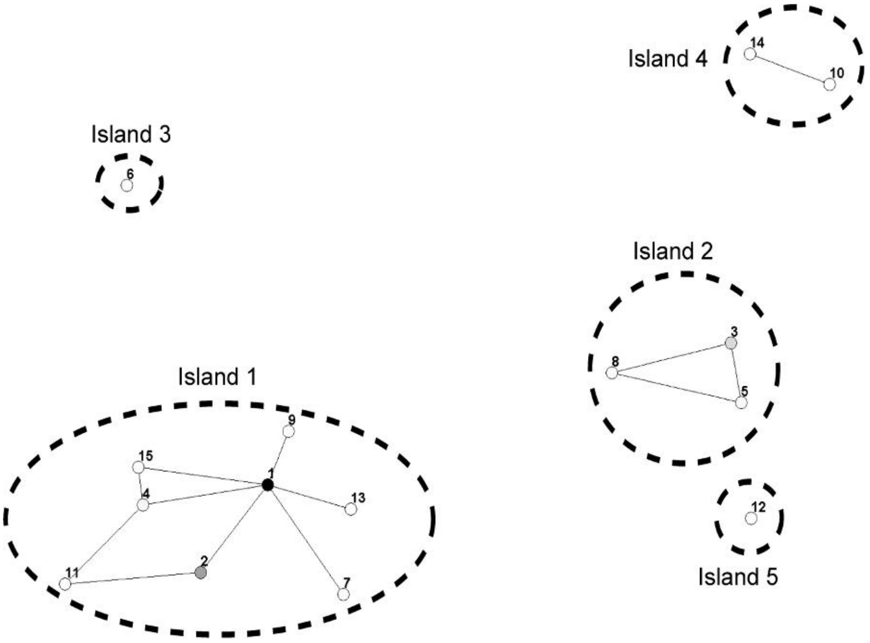

FIGURE 5. A visualization of the tree space of data set DS1 containing the 95% credible set, emphazising the distinct tree islands. The set of trees with their posterior probabilities is given in Table 6. The trees are plotted using MDS and the NNI distance between each pair of trees and filled with a grayscale matching their posterior probability. All trees in one island are connected to at least one other tree in the island by a single NNI transformation.

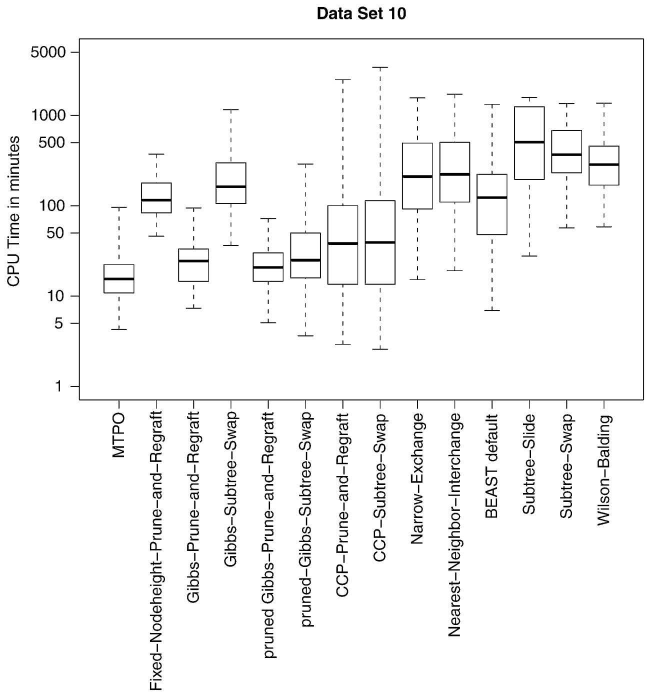

FIGURE 6. The detailed performance analysis of data set DS10. The box plots show the running times for each operator over 100 replicates. The boxes representing the 50% median running times, and the whiskers show the full range including the outliers. Runs were terminated when either they converged closely to the target distribution or they reached the maximum number of iterations.

of the MCMC algorithm. A new mixture, MTPO, which includes the new Metropolized Gibbs Sampler operators, decreases the median running time till convergence by 52% and the variance among the runs 20-fold over the previous settings. We recommend our mixture for fast and more reliable results in Bayesian phylogenetic inference.

The newly introduced CCP score is a good proxy of the posterior distribution on trees when MCMC samples are available. Here, we used the CCP score to guide the operators for proposing trees with higher posterior probability more often. However, the CCP score can be used in many more applications. First, it can be used as a summarizing statistic after a MCMC run is performed. It is a fast algorithm to compare several different tree topologies and estimate their posterior probabilities. Second, the CCP score can be used as a analyses can be combined together. This research only scratches the surface of ideas for improving Bayesian phylogenetics by MCMC, and we anticipate much more work in this area if Bayesian phylogenetics is to mature as a statistical computing discipline.

ACKNOWLEDGMENTS

The authors would like to thank M. Defoin Platel and M. Suchard for helpful discussions. For comments of the manuscript and helpful suggestions, we would like to thank R. DeBry, M. Holder, and three anonymous reviewers.

References

Billera L.J., Holmes S.P., Vogtmann K. 2001. Geometry of the space of phylogenetic trees. Adv. Appl. Math. 27:733–767.

Brooks D.R. 1981. Hennig's parasitological method: a proposed solution. Syst. Zool. 30:229–249.

Brooks S. 1998. Markov chain monte carlo method and its application. Statistician. 47:69–100.

Cowles M.K., Carlin B.P. 1996. Markov chain monte carlo convergence diagnostics: a comparative review. J. Am. Stat. Assoc. 91:883–904.

Drummond A., Rambaut A. 2007. BEAST: Bayesian evolutionary analysis sampling trees. BMC Evol. Biol. 7:214.

Drummond A.J., Nicholls G.K., Rodrigo A.G., Solomon W. 2002. Estimating mutation parameters, population history and genealogy simultaneously from temporally spaced sequence data. Genetics. 161:1307–1320.

Farris J.S., Kluge A.G., Eckardt M.J. 1970. A numerical approach to phylogenetic systematics. Syst. Zool. 19:172–189.

Geman S., Geman D. 1984. Stochastic relaxation, Gibbs distributions and the Bayesian restoration of images. IEEE Trans. Pattern Anal. Mach. Intell. 6:721–741.

Gilks W., Richardson S., Spiegelhalter D. 1996. Markov chain Monte Carlo in practice. London: Chapman & Hall.

Hastings W.K. 1970. Monte Carlo sampling methods using Markov chains and their applications. Biometrika. 57:97–109.

Hillis D., Heath T., John K. 2005. Analysis and visualization of tree space. Syst. Biol. 54:471–482.

Höhna S., Defoin-Platel S., Drummond A. 2008. Clock-constrained tree proposal operators in Bayesian phylogenetic inference. 8th IEEE International Conference on BioInformatics and BioEngineering; 2008. Athens (Greece): BIBE. p. 7.

Huelsenbeck J.P., Ane C., Larget B., Ronquist F. 2008. A Bayesian perspective on a non-parsimonious parsimony model. Syst. Biol. 57:406–419.

Lakner C., van der Mark P., Huelsenbeck J.P., Larget B., Ronquist F. 2008. Efficiency of Markov chain Monte Carlo tree proposals in Bayesian phylogenetics. Syst. Biol. 57:86–103.

Li S., Pearl D.K., Doss H. 2000. Phylogenetic tree construction using Markov chain Monte Carlo. J. Am. Stat. Assoc. 95:493–508.

Liu J. 2001. Monte Carlo strategies in scientific computing. New York: Springer.

Liu J.S. 1996. Peskun's theorem and a modified discrete-state Gibbs sampler. Biometrika. 83:681–682.

Matsen F.A. 2006. A geometric approach to tree shape statistics. Syst. Biol. 55:652–661.

Mau B., Newton M.A. 1997. Phylogenetic inference for binary data on dendograms using Markov chain Monte Carlo. J. Comput. Graph. Stat. 6:122–131.

Metropolis N., Rosenbluth A., Rosenbluth M., Teller A., Teller E. 1953. Equation of state calculations by fast computing machines. J. Chem. Phys. 21:1087–1092.

Mira A. 2001. Ordering and improving the performance of Monte Carlo Markov chains. Stat. Sci. 16:340–350.

Mossel E., Vigoda E. 2005. Phylogenetic MCMC algorithms are misleading on mixtures of trees. Science. 309:2207–2209.

Peskun P.H. 1973. Optimum Monte–Carlo sampling using Markov chains. Biometrika. 60:607–612.

Rannala B., Yang Z. 1996. Probability distribution of molecular evolutionary trees: a new method of phylogenetic inference. J. Mol. Evol. 43:304–311.

Ronquist F., Huelsenbeck J. 2003. MrBayes 3: Bayesian phylogenetic inference under mixed models. Bioinformatics. 19: 1572–1574.

Ronquist F., Huelsenbeck J., Britton T. 2004. Bayesian supertrees. In: Bininda-Emonds, O.R.P., editor. Phylogenetic supertrees: combining information to reveal the tree of life. Dordrecht, The Netherlands: Kluwer Academic. p. 193–224.

Ronquist F., Larget B., Huelsenbeck J.P., Kadane J.B., Simon D., van der Mark P. 2006. Comment on "phylogenetic MCMC algorithms are misleading on mixtures of trees". Science. 312:367.

Seary A., Richards W. 1997. The Physics of Networks. INSNA Sunbelt XVII; February; San Diego. p. 13–17.

Swofford D., Olsen G., Waddell P., Hillis D. 1996. Phylogenetic inference. In: Molecular systematics. Volume 2. p. 407–514.

Wilson I.J., Balding D.J. 1998. Genealogical inference from microsatellite data. Genetics. 150:499–510.