Syst. Biol. 57(1):86–103, 2008
Copyright © Society of Systematic Biologists
ISSN: 1063-5157 print / 1076-836X online
DOI: 10.1080/10635150801886156

Efficiency of Markov Chain Monte Carlo Tree Proposals in Bayesian Phylogenetics

CLEMENS LAKNER,[1,2] PAUL VAN DER MARK,[2] JOHN P. HUELSENBECK,[3] BRET LARGET,[4] AND FREDRIK RONQUIST[1,2]

[1]Department of Biological Science, Section Ecology and Evolution, and
[2]School of Computational Science, Florida State University, Tallahassee, Florida 32306-4120, USA;
E-mail: lakner@scs.fsu.edu (C. L.)
[3]Department of Integrative Biology, 3060 VLSB 3140, University of California, Berkeley, Berkeley, CA 94720-3140, USA
[4]Departments of Botany and of Statistics, University of Wisconsin at Madison, Wisconsin 53706, USA

Abstract.— The main limiting factor in Bayesian MCMC analysis of phylogeny is typically the efficiency with which topology proposals sample tree space. Here we evaluate the performance of seven different proposal mechanisms, including most of those used in current Bayesian phylogenetics software. We sampled 12 empirical nucleotide data sets—ranging in size from 27 to 71 taxa and from 378 to 2,520 sites—under difficult conditions: short runs, no Metropolis-coupling, and an oversimplified substitution model producing difficult tree spaces (Jukes Cantor with equal site rates). Convergence was assessed by comparison to reference samples obtained from multiple Metropolis-coupled runs. We find that proposals producing topology changes as a side effect of branch length changes (LOCAL and Continuous Change) consistently perform worse than those involving stochastic branch rearrangements (nearest neighbor interchange, subtree pruning and regrafting, tree bisection and reconnection, or subtree swapping). Among the latter, moves that use an extension mechanism to mix local with more distant rearrangements show better overall performance than those involving only local or only random rearrangements. Moves with only local rearrangements tend to mix well but have long burn-in periods, whereas moves with random rearrangements often show the reverse pattern. Combinations of moves tend to perform better than single moves. The time to convergence can be shortened considerably by starting with a good tree, but this comes at the cost of compromising convergence diagnostics based on overdispersed starting points. Our results have important implications for developers of Bayesian MCMC implementations and for the large group of users of Bayesian phylogenetics software. [Bayesian inference, Hastings ratio, Markov chain Monte Carlo, topology proposals.]

Bayesian inference was introduced to phylogenetics in the last years of the 20th century (Li, 1996; Mau, 1996; Rannala and Yang, 1996; Mau and Newton, 1997; Yang and Rannala, 1997; Larget and Simon, 1999; Newton et al., 1999; Huelsenbeck et al., 2000; Li et al., 2000) and has become widely adopted since then (for reviews see Huelsenbeck et al., 2001, 2002; Lewis 2001; Holder and Lewis 2003. For a general discussion of Bayesian data analysis, see, for instance, Gelman et al., 2003). Typically, Markov chain Monte Carlo (MCMC) approaches based on the Metropolis-Hastings algorithm (Metropolis et al., 1953; Hastings, 1970) are used to approximate the posterior distribution. A number of different MCMC proposal strategies have been applied but the efficiency of the different mechanisms has never been tested rigorously. Felsenstein (2004) appositely stated: "At the moment the choice of a good proposal distribution involves the burning of incense, casting of chicken bones, magical incantations and invoking the opinions of more prestigious colleagues." In this article, we suggest a framework for testing the efficiency of tree proposals and apply it to seven mechanisms, including commonly used ones as well as some that have not been considered in the literature before.

In Bayesian phylogenetics, all inference is based on the joint posterior distribution of evolutionary trees and substitution model parameters. Let $\psi = \{\tau, \mathbf{v}\}$ denote a phylogenetic tree with topology τ and an associated set of branch lengths $\mathbf{v}$ and let Ψ denote the set of all possible phylogenies on N taxa. Furthermore, let $\Omega = \{\Psi, \Theta\}$ denote the parameter space containing all phylogenetic trees and all possible attributions of values to the substitution model parameters (Θ). We focus here on the Jukes-Cantor model (Jukes and Cantor, 1969) without rate variation among sites (JC), for which only the joint posterior probability distribution of τ and $\mathbf{v}$ needs to be approximated since Θ is empty. The joint posterior distribution for JC can formally be written as

$$f(\tau, \mathbf{v}|X) = \frac{f(X|\tau, \mathbf{v}) f(\tau, \mathbf{v})}{\sum_{\tau} \int_{\mathbf{v}} f(X|\tau, \mathbf{v}) f(\tau, \mathbf{v})\, \mathrm{d}\mathbf{v}}$$

where X denotes observed data.

Calculating this distribution analytically would involve summation over all possible trees and for each tree integrating over all possible combinations of branch lengths (and usually also model parameters). This cannot be accomplished except for very small trees. Instead, the posterior is typically estimated using a Markov chain, which generates dependent samples from the distribution.

In practice, efficient proposal mechanisms are vital for the chain to produce an adequate sample of the posterior probability distribution within the time constraints faced by most users. This is particularly true for complex multimodal distributions, such as the distributions on tree space that result from most phylogenetic problems. In the latter, we can expect multiple posterior probability peaks for each topology, corresponding to different combinations of branch lengths, and—even worse—there is no obvious way of jumping between branch length peaks of different topologies. Several isolated areas of topology space with high probability mass, known as *tree islands*, may also have to be visited before an adequate sample of the posterior is obtained. In phylogenetics, Markov chains that mix well move quickly among all the good trees, whereas chains that mix poorly tend to get stuck in tree space. Markov chains that mix well may also be expected to have shorter burn-in periods. However, bold proposals that tend to be rejected in later stages of an MCMC run may be advantageous during the burn-in phase because they may help the chain quickly explore large portions of the parameter space. More modest proposals generally have higher acceptance rates but are slower in providing adequate coverage and in moving between modes, resulting in longer burn-in periods and poor mixing. Generally, proposals with intermediate acceptance rates are considered optimal.

We focus here mostly on proposals that change topology and branch lengths simultaneously. With JC, such proposals are sufficient for a working Markov chain, allowing us to test their performance individually. The proposals we examine can be divided into two different classes: the *branch-change proposals* and the *branch-rearrangement proposals*. The branch-change proposals modify branch lengths or branch attachment points continuously in a way that produces topology changes in some cases. Of this type, we consider the Continuous Change proposal (CC; Jow et al., 2002) and the LOCAL proposal (Larget and Simon, 1999), used in the software packages PHASE (http://www.bioinf.manchester.ac.uk/resources/phase/) and BAMBE (Simon and Larget, 1998), respectively. The programs MrBayes (Huelsenbeck and Ronquist, 2001; Ronquist and Huelsenbeck, 2003) and PhyloBayes (Lartillot et al., 2007) also implement the LOCAL move.

The branch-rearrangement proposals—used in BEAST (Drummond and Rambaut, 2003), BADGER (Simon and Larget, 2004), MrBayes, PHASE, PhyloBayes, and also in Li et al. (2000) and Suchard et al. (2001)—can be divided into two subtypes: the pruning-regrafting moves and the swapping moves. In a pruning-regrafting move, a subtree is pruned from the rest of the tree and regrafted somewhere else. We test two different strategies for selecting the regrafting point: (1) the point is chosen randomly; and (2) the attachment point is moved one branch at a time according to an extension probability, which results in local rearrangements being favored. The moves we consider are Random Subtree Pruning and Regrafting (rSPR), Extending Subtree Pruning and Regrafting (eSPR), and Extending Tree Bisection and Reconnection (eTBR).

In the second type of branch-rearrangement moves, which we refer to as the swapping moves, two subtrees simply trade places. The moves we consider are Stochastic Nearest Neighbor Interchange (stNNI) and Extending Subtree Swapping (eSTS). We excluded random subtree swapping from our comparison because preliminary data indicated it was too bold to compete successfully with the other moves. There is no meaningful way in which an extension mechanism can be applied to stNNI because its topology changes are always local. The stNNI move links the two subtypes of branch-rearrangements because it can be understood both as a subtree swap and as a special case of subtree pruning and regrafting. All the tested proposals are available in the most recent source code release of MrBayes (Ronquist and Huelsenbeck, 2003, http://sourceforge.net/projects/mrbayes).

We assess the efficiency of the proposals using 12 empirical DNA data sets selected from TreeBASE (http://www.treebase.org) and ranging in size from 27 to 71 taxa and from 378 to 2520 sites (Table 1). The tree proposals are used to repeatedly sample the posteriors of these data sets under difficult conditions, which allows us to challenge the proposals with relatively small data sets. In order to separately assess convergence and mixing behavior of the different proposals, we compare runs that are started from two sets of starting points: overdispersed trees and from trees that were sampled from the target distribution. We also try to eliminate other factors so that we can focus on the performance of the tree proposals. Specifically, we use a fixed, rather small number of generations and we do not employ Metropolis-coupling, a general technique for improving mixing behavior in Bayesian MCMC phylogenetics (Geyer, 1991; Huelsenbeck et al. 2001). Finally, we use the JC model to eliminate substitution model parameters and to make the posterior more difficult to sample from (Nylander, 2004). Convergence is assessed by comparison to reference samples obtained using at least six independent Metropolis-coupled MCMC runs (Table 2).

METHODS

All analyses were conducted with a developer's version of MrBayes 3.1.2, available from C.L. upon request. The parallel version of the program (Altekar et al., 2004) was used for the reference runs (see below). MCMC runs were deployed on a cluster of 57 dual AMD Athlon MP

TABLE 1. Data sets used for the experiments.

Data set	No. of taxa	No. of sites	Type of data	TreeBASE matrix accession no.
1	27	1949	rRNA, 18s	M336
2	29	2520	rDNA, 18s	M501
3	36	1812	mtDNA, COII (1–678), cytb (679–1812)	M1510
4	41	1137	rDNA, 18s	M1366
5	43	1660	rDNA, 18s	M932
6	50	378	Nuclear protein coding, *wingless*	M3475
7	50	1133	rDNA, 18s	M1044
8	59	1824	mtDNA, COII and cytb	M1809
9	64	1008	rDNA, 28S	M755
10	67	955	Plastid ribosomal protein, s16 (*rps16*)	M1748
11	67	1098	rDNA, 18s	M520
12	71	1082	rDNA, internal transcribed spacer (ITS)	M767

TABLE 2. Summary of the reference runs and number of generations for the test runs. The samples from the reference runs, which used Metropolis-coupling and were run until the standard deviation of split frequencies was 0.005, were assumed to be accurate representations of the posterior probability distribution.

			No. of trees in credible set			
Data set	Proposal ratio	No. of generations until convergence	50%	90%	95%	No. of generations for the test runs
1	10 eTBR : 1 LOCAL : 1 stNNI	6,975,000	3	24	42	8×10^6
2	1 LOCAL	1,131,000	2	4	5	1×10^6
	1 eTBR	323,000				
3	1 LOCAL	2,914,000	2	6	15	2×10^6
	1 eTBR	1,683,000				
4	5 eTBR : 1 LOCAL	6,183,000	4	78	189	7×10^6
5	5 eTBR : 1 LOCAL	610,000	2	15	28	1×10^6
6	5 eTBR : 1 LOCAL	4,305,000	7929	62,938	72,624	6×10^6
7	8 eTBR : 1 LOCAL : 1 stNNI	6,771,000	7944	72,739	87,974	8×10^6
8	5 eTBR : 1 LOCAL	3,940,000	26	417	778	5×10^6
9	5 eTBR : 1 LOCAL	5,513,000	37	1273	3,105	6×10^6
10	5 eTBR : 1 LOCAL	3,898,000	87,535	157,700	166,471	1×10^6
11	5 eTBR : 1 LOCAL	8,547,000	17,081	122,217	141,448	9×10^6
12	5 eTBR : 1 LOCAL	3,450,000	77,608	139,709	147,472	5×10^6

2400+ processors running LINUX and a cluster of 64 dual G5 2.0 Ghz processors running Macintosh OS X at the School of Computational Science at Florida State University. Jobs were distributed over the network cluster using Condor version 6.7.1 (Litzkow et al., 1988).

The performance of the topology proposals was tested on 12 empirical DNA data sets (Table 1). Data set 6 was taken from the literature (Brower, 2000) and submitted to TreeBASE as part of this study; the remaining data sets were obtained from previous submissions to TreeBASE.

All analyses were performed under the Jukes Cantor model with no rate variation across sites [MrBayes commands: lset nst=1 rates=equal; prset statefreqpr=fixed(equal)]. A uniform prior (all labeled topologies equally likely) was used on topologies and an unconstrained, exponential prior with mean 0.1 was used on branch lengths [prset topologypr=uniform brlenspr=unconstrained:exponential(10)].

Convergence Diagnostic

To diagnose convergence among tree samples, we used the average standard deviation of split frequencies (the estimated posterior probabilities of splits or taxon bipartitions). Only splits that occurred in more than 10% of the samples in at least one of the runs were included in the calculations because the frequency of the rare splits is more difficult to estimate accurately and the rare splits are less important in characterizing the tree sample.

Qualitatively similar results were obtained when other cutoff levels and convergence-diagnostic statistics were used (maximum standard deviation and maximum absolute difference of split frequencies; see online Supplementary Materials, available at http://www.systematicbiology.org).

Reference Runs

To obtain a reference sample of the posterior probability distribution for each data set, we ran six parallel runs, each using four Metropolis-coupled chains under the default MrBayes heating schema (incremental heating with the temperature of chain i being $1/(1 + \lambda i)$, with $i \in \{0, 1, 2, \ldots\}$ and the tuning parameter $\lambda = 0.2$). In every generation, a single swap was attempted between a randomly drawn pair of chains. The runs were started from different random topologies and were sampled every 100 generations until the average standard deviation of split frequencies fell below 0.005 for the last 75% of the tree samples. In general, a mixture of topology proposals was used (Table 2). For data sets 2 and 3, however, we used only the LOCAL update for three runs and only the eTBR update for three runs. In both cases, the LOCAL and eTBR runs converged to the same stationary distribution. The tuning parameter settings for all reference runs were as follows: eTBR extension probability 0.8, branch length multiplier $2\ln(1.6)$, LOCAL $\lambda = 2\ln(1.1)$, and stNNI branch length multiplier $2\ln(1.6)$.

Test Runs

For each data set, we generated 100 good starting trees and 100 overdispersed random starting trees. The good starting trees were randomly drawn from the reference sample, whereas the overdispersed trees were generated as follows: First, 10,000 random topologies were constructed for each data set and for each pair of topologies, the Robinson-Foulds distance (Robinson and Foulds, 1981) was calculated. Using a minimax and maximin distance design algorithm (Johnson et al., 1990) a subset of 100 topologies was determined to optimally fill the tree space. Finally, the branch lengths were all arbitrarily set to 0.1.

We tested three different tuning parameter settings for each proposal (Table 3). Setting I generated the most modest proposals, setting II intermediate proposals, and setting III the boldest proposals. For the eTBR, eSPR, and eSTS proposals, we changed the extension probability rather than the branch length multiplier tuning parameter, because the former has more effect on the behavior of the proposal than the latter. For the rSPR move, we changed the probability of proposing a topology change, which is likely to be of overriding importance when

TABLE 3. The tuning parameter settings tested for each tree proposal.

		Tuning parameter settings for test runs					
Proposal	Tuning parameter	Data set[a]	I	Data set[a]	II	Data set[a]	III
rSPR	Probability of topology change (P_r)		0.8		0.9		0.95
	Multiplier (λ)		2 ln(1.6)		2 ln(1.6)		2 ln(1.6)
eTBR	Extension probability P_e		0.5		0.8		0.9
	λ		2 ln(1.6)		2 ln(1.6)		2 ln(1.6)
eSPR	P_e		0.5		0.8		0.9
	λ		2 ln(1.6)		2 ln(1.6)		2 ln(1.6)
eSTS	P_e		0.5		0.8		0.9
	λ		2 ln(1.6)		2 ln(1.6)		2 ln(1.6)
stNNI	λ		2 ln(1.2)		2 ln(1.6)		2 ln(2.0)
LOCAL	λ		2 ln(1.05)		2 ln(1.1)		2 ln(1.3)
CC	Standard deviation (σ)	1, 10	0.015	1, 10	0.02	1, 10	0.03
		2, 3, 4	0.04	2, 4	0.06	2, 3, 4	0.08
		5, 8, 9	0.03	3, 5, 8, 9	0.05	5, 8, 9	0.06
		6	0.08	6	0.11	6	0.15
		7	0.02	7, 11, 12	0.03	7, 11, 12	0.05
		11, 12	0.021				

[a] If not applied to all data sets.

rSPR is used as the only proposal mechanism. For the stNNI and LOCAL moves, we varied the tuning parameters that determine the boldness of the branch length modifications. For the continuous change proposal, we followed the procedure used by Jow et al. (Vivek Gowri-Shankar, personal communication): the tuning parameter (the standard deviation of the normal distribution used by the proposal) was adjusted beforehand for each data set to obtain an acceptance probability of approximately 20%.

Each tuning parameter setting was run for a fixed number of generations, using only a single chain (no Metropolis-coupling), on each of the 100 overdispersed and 100 good starting trees. The number of generations for these test runs was determined separately for each data set using the time to convergence for the reference runs as a guide. Because the reference runs used Metropolis-coupling and the test runs did not, we expected a significant fraction of the test runs not to reach convergence within the chosen number of generations. A test run was deemed to have converged if the average standard deviation of split frequencies, when the topology sample was compared to that of the reference run, was below 0.01 at the end of the run. The time to convergence was determined as the number of generations the chain was run before the the standard deviation of split frequencies first fell below 0.01. The computational complexity of the tested proposals is similar, so that the wall-clock time used per generation is essentially constant across the experiments.

In a separate experiment, we tested some of the proposals used above, which all change topology and branch lengths simultaneously, against comparable mixtures of separate topology and branch length proposals. The chosen proposals were rSPR (tuning parameter II, Table 3), eTBR (tuning parameter II), eSPR (tuning parameter II), and stNNI (tuning parameter I). To make the mixtures as similar as possible to the corresponding joint proposals, we used the same algorithm for the mixtures and only modified it such that when a topology change was proposed, branch lengths were left unmodified, with old branch lengths mapping into new ones as suggested in the description of the proposals below. The mixtures were only tested on the overdispersed starting trees. For both the mixtures and the joint proposals, we collected information about the number of proposed and accepted topology changes.

Because most implementations use a combination of several topology and branch length moves, we also tested a mixture of all seven proposals and a mixture of the five branch-rearrangement moves. In both cases, the proposals were chosen with equal probability at every step of the Markov chain. The tuning parameters for these runs were adjusted to the best performing settings from the individual runs.

Dependence of convergence success on the starting tree was assessed in several different ways. First, we used a χ^2 test of independence to examine whether significantly more runs converged for certain starting trees than for others ($P < 0.05$). Second, to test whether runs started from random trees close to the good trees were more likely to converge than others, the random trees were ordered according to their likelihoods and according to their average Robinson-Foulds distances from a sample of 1000 trees from the posterior (the post-burn-in sample of the reference runs). The number of runs that had reached convergence for each tree was summed over all proposals.

Finally, we graphically compared the variance in time to convergence for runs started from the 100 overdispersed trees with that for 100 runs started from the same tree (a randomly selected tree from the 100).

Topology Proposals

The tested tree proposals can be roughly sorted from bold to modest by first focusing on the expected NNI distance between the candidate tree and the current tree (the minimum number of nearest neighbor interchanges required to go from one tree to the other). Among the branch-rearrangement moves, the expected NNI distance gives the rough sequence (from bold to modest) rSPR > eSTS > eTBR > eSPR > stNNI. For small trees, eSTS may actually be bolder than rSPR because the maximum distance of the rSPR move is limited by the number of branches in the tree, and subtree pruning and regrafting is, in itself, a more modest topology change than a subtree swap. Given the same extension probability, eTBR is typically more bold than eSPR since two attachment points are moved in eTBR and only one in eSPR. However, the distribution of topology changes is also different. While the proposal distribution of eSPR has a long tail of rather distant rearrangements, the eTBR distribution is more focused on less dramatic changes. Specifically, if d is the NNI distance and p_e is the extension probability, the distribution is $f(d) = (d+1)p_e^d(1-p_e)^2$ for TBR and $f(d) = p_e^d(1-p_e)$ for SPR, in both cases assuming that the moves are not limited by the size of the tree. The stNNI move is the least bold because it only involves one NNI rearrangement. The two branch-change moves (CC and LOCAL) may arguably be considered more modest than all branch-rearrangement moves because they result in topological changes more rarely and, when they do, the distance between the candidate tree and the current tree is always one NNI rearrangement. The CC move can be considered more modest in that it makes less dramatic branch length changes than LOCAL.

All seven tested tree proposals are briefly described below. A more detailed description including the derivation of the Hastings ratios can be found in the Appendix. The moves fall into two classes: (1) branch-change proposals (LOCAL and CC), which modify branch lengths or branch attachment points in a way that sometimes leads to topology changes; and (2) branch-rearrangement proposals, which either prune and regraft subtrees (eSPR, eTBR, rSPR) or swap subtrees (stNNI, eSTS). The branch-rearrangement proposals either choose pruning and regrafting points or subtrees to swap using an extension mechanism (eSPR, eTBR, eSTS), which favors local rearrangements, or at random (rSPR). The exception is stNNI, which always involves minimal topology changes.

LOCAL.—This update mechanism was introduced by Larget and Simon (1999) and first used in their software package Bayesian Analysis in Molecular Biology and Evolution (BAMBE). It consists of two independent steps that could form separate update mechanisms. The first step changes a subtree attachment point, the second changes three branch lengths. For detailed descriptions of the move see Larget and Simon (1999), Larget (2005), and Holder et al. (2005).

Continuous change (CC).—Introduced by Jow et al. (2002), CC is primarily a branch length proposal. It first picks a branch b at random, whose length is v. A value u is drawn from $N(0, \sigma)$, where σ is a tuning parameter, and we propose the new branch length $v^* = |v + u|$. If $v + u < 0$ and b is internal, then, with equal probability, the topology is changed to one of the two alternative NNI rearrangements centered on b and v^* becomes the length of the new branch. In all other cases, the topology remains unchanged. The proposal ratio of the CC move is 1 (Appendix).

Stochastic nearest neighbor interchange (stNNI).—The stochastic nearest neighbor interchange we implemented first picks a random interior branch b_x with the length v_x. This branch has four subtrees attached to its ends. Randomly label the two subtrees at one end A and B and the two subtrees at the other end C and D. Subtree A sits on a branch with the length v_a, etc., so that the five branches affected by this move can be collected in the local branch length vector $\mathbf{v} = (v_a, v_b, v_c, v_d, v_x)$. Now, with probability 1/3 swap A and C; with probability 1/3 swap B and C; and with probability 1/3 leave the topology unchanged. Finally, propose the new branch lengths $\mathbf{v}^* = \{m_a v_a, m_b v_b, m_c v_c, m_d v_d, m_x v_x\}$, where each m_i represents an independent application of the multiplier move described in the Appendix. The proposal ratio for the topology part of this move is 1; for the factor of the multiplier part see the Appendix.

Extending subtree pruning and regrafting (eSPR).—The eSPR proposal first picks a random interior branch b_a of length v_a (Figure 6a). Randomly label the subtrees attached to that branch A and B. Consider B as rooted at b_a and arbitrarily label each pair of descendant branches in B as left and right descendants. Now prune A from B and choose a regrafting point for A on B using the following mechanism (see Fig. 6b). With probability 1/2, try to move the regrafting point left in B, and with probability 1/2 try to move it right. With probability p_e, which is a tuning parameter called the *extension probability*, move the regrafting point across one node to either the left or right descendant branch with equal probability. With probability $1 - p_e$, choose the current branch as the regrafting point. If the current branch is a terminal branch, the extension mechanism stops and the current branch is called a *constrained* regrafting point. If the extension mechanism could have proceeded at least one step farther, the regrafting point is *unconstrained*.

Label the two branches at the pruning point b_x and b_p, where b_x is the branch in the chosen movement direction and b_p is the other branch (the *pruning* branch). Label the branch chosen for regrafting b_r (the *regrafting* branch) and the branches traversed during the extension phase ($b_1, b_2, \ldots, b_n$; see Fig. 6). In the new topology, branches b_p, b_r, b_a can be mapped together with their lengths (v_p, v_r, v_a) into branches that define the same splits, whereas the branches in the vector $(b_1, b_2, \ldots b_n)$ and their lengths can be mapped into equivalent branches that differ only in the placement of the leaves in A. This leaves us with b_x and v_x, which are mapped into the regrafting point in the direction of the reverse move (see Fig. 6c).

After completing these branch length transfers, we apply the multiplier move independently to the branch lengths $\mathbf{v} = (v_a, v_x)$ to give the new branch lengths $\mathbf{v}^* = (m_a v_a, m_x v_x)$. See the Appendix for the derivation of the Hastings ratio.

In clock trees, the subtree A cannot be picked randomly; it will always have to be from the younger (nonroot) end of the chosen interior branch. In standard unrooted trees, however, it is probably suboptimal to enforce a consistent rooting such that SPR moves always change one of the two attachment points of a given interior branch. This is because the rooted version of the SPR produces a topology space that is less well connected and presumably more difficult to traverse than that of the unrooted version we used here.

Extending tree bisection and reconnection (eTBR).—This is the same move as eSPR except that the extension mechanism is applied to both ends of the picked interior branch b_a. Also, when the pruning and regrafting points are the same, we always modify the lengths v_a and v_x, never v_p.

Extending subtree swapper (eSTS).—This move is similar to the stNNI move in that it involves a swap of two subtrees. However, the swapped subtrees need not be nearest neighbors; they are chosen by an extension mechanism and can be arbitrarily distant (Appendix).

Random subtree pruning and regrafting (rSPR).—This move is similar to eSPR except that the regrafting branch is picked randomly from the branches in subtree B. Because an extension mechanism is not used, the proposal ratio for the topological change is always 1 and the ratio for the branch length change is identical to the one for eSPR. To increase the frequency of branch length changes without topology modifications, which is important when rSPR is used as the only tree proposal, we introduced a tuning parameter p_r, the *rearrangement probability*. With probability p_r, we selected a regrafting point among the branches in B other than the pruning point. With probability $1 - p_r$, we would force the regrafting point to be the same as the pruning point.

Visualizing Tree Space

The Mesquite (Maddison and Maddison, 2005) Tree Set Visualization Module (TSV; Hillis et al., 2005, http://comet.lehman.cuny.edu/treeviz/) was used to illustrate tree space via multidimensional scaling on a topology sample using unweighted Robinson-Foulds distances. The topology sample consisted of 5000 samples drawn from the posterior (i.e., from the reference runs) and 500 samples from each of the chain paths that we wanted to plot in the space. The x- and y-coordinates of each topology were extracted from the postscript file generated by TSV. These coordinates were used to plot the posterior probability of each topology and the chain paths in the resulting space.

Results

For each proposal, we tested three different sets of tuning parameters (Table 3), where the first set (I) gave more modest proposals, the second set (II) intermediate proposals, and the third set (III) bolder proposals. The time to convergence and the convergence success rate were measured under two different scenarios. First, we started from 100 overdispersed random starting trees (Fig. 1a). In this situation (*random starting trees*), we expect the time to convergence to reflect both the burn-in time and the mixing efficiency. Second, we started from 100 trees drawn from the posterior distribution (i.e., the reference sample; Fig. 1b). In this case (*good starting trees*), the time to convergence should be determined solely by the mixing behavior since the burn-in period is negligible.

Our most striking result is that the branch length moves (CC and LOCAL) do very poorly compared to the branch-rearrangement moves (Fig. 1). This is apparently due primarily to long burn-in times for CC and LOCAL, but mixing is also slow, in particular for CC (Fig. 1b). Among the branch-rearrangement moves, eTBR and eSPR do particularly well. Their superior performance is the result of both short burn-in times and rapid mixing. The more modest stNNI move mixes slightly better for some data sets than both eTBR and eSPR (Fig. 1 b) but its burn-in time is considerably longer (Fig. 1a). The boldest proposals, rSPR and eSTS, occasionally do well but their performance is unpredictable. The rSPR move might be expected to perform poorly on large trees, because the rejection rate presumably increases with tree size, but we could not detect such a trend in our data.

Focusing on the success rate (the percentage of runs that converged within the predefined number of generations) instead of the time to convergence, the picture changes slightly (Fig. 2). We show these data only for the runs starting from overdispersed trees, because the success rate was uniformly high when starting from good trees. The boldest proposal, rSPR, does much better when judged by this criterion than by the time to convergence. On average, the rSPR move achieves convergence much slower than eTBR and eSPR (Fig. 1), but it succeeds with a similar number of runs within a fixed number of generations (Fig. 2). For some data sets (5 and 9), it actually succeeds with more runs than eTBR and eSPR. Thus, rSPR can be described as a safe but slow proposal mechanism. The same thing cannot be said for eSTS; its success rate is considerably lower than that of eTBR and eSPR even though it is a bolder proposal.

To illustrate the effect of changing the tuning parameters, we also give the success rates for all three tuning parameter settings (Table 4). The tuning parameters tend to have a distinct but not dramatic effect, with the intermediate settings (II) often doing better than the two extreme settings. An obvious exception is LOCAL, where the success rate was very similar for all tuning parameter settings.

The overall acceptance rate was not a good indicator of convergence success (see online Supplementary Materials). Convergence success correlated better with the number of accepted topology changes and the average NNI distance of the accepted proposals, but the correlation was far from perfect. For data set 12, for instance, rSPR(II) always converged, whereas eTBR(III) was doing considerably worse at 87%. Yet the overall acceptance rate was similar, and the number and topological distance of accepted topological changes were roughly the same as well. A possible explanation is that a few successful distant topology changes helped shorten the burn-in or improve the mixing significantly for the rSPR move on this data set.

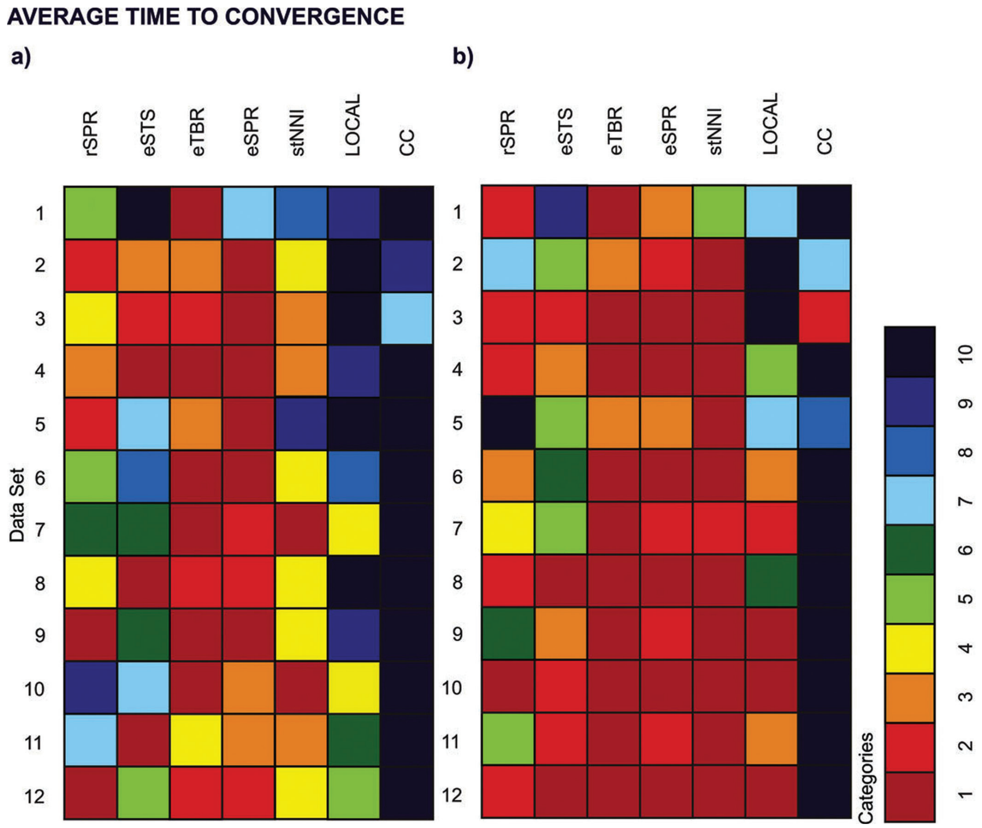

FIGURE 1. The average time to convergence (10 equal-sized intervals from best to worst for each data set) for the random starting trees (a) and for the good starting trees (b) for seven tree proposals. In the first case, convergence time reflects both the burn-in and the mixing behavior; in the second case, it reflects only mixing. The proposals are represented by their best tuning parameter settings for each data set and are sorted from the boldest (rSPR) to the most modest (CC). Branch-rearrangement algorithms (rSPR, eSTS, eTBR, eSPR, and stNNI) tend to converge much faster than branch-change proposals (LOCAL and CC), primarily due to shorter burn-in periods.

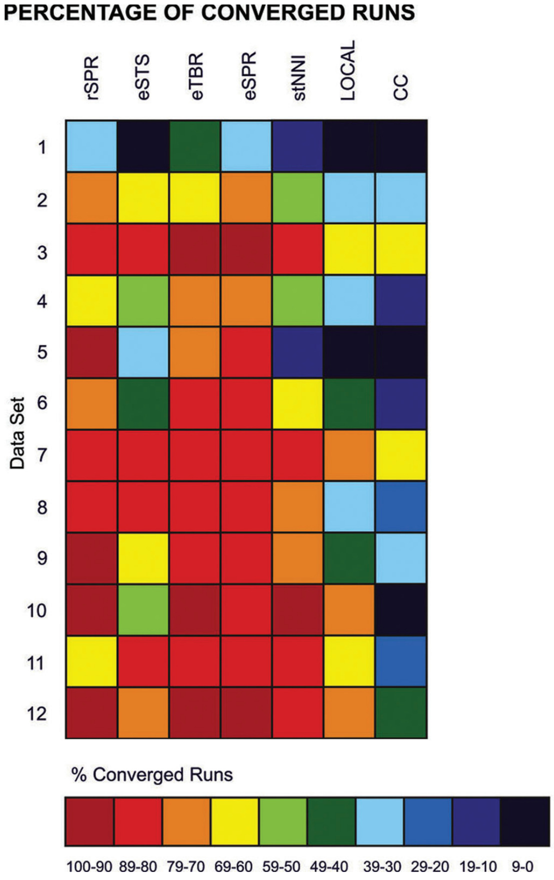

FIGURE 2. The success rate (percentage of runs starting from random trees that converged within a fixed number of generations) for seven tree proposals. The proposals are represented by their best tuning parameter settings for each data set, and are sorted from the boldest (rSPR) to the most modest (CC). In general, the intermediate proposals (eTBR and eSPR) do best.

TABLE 4. Convergence success (percentage of runs that reached convergence) for runs started from overdispersed random trees (bold: best value for the data set; shaded cells: significantly worse than best value).

	Percentage of converged runs																				
	rSPR			eTBR			eSPR			eSTS			stNNI			LOCAL			CC		
Data set	I	II	III	I	II	III	I	II	III	I	II	III	I	II	III	I	II	III	I	II	III
1	32	30	27	32	**46**	43	14	30	16	6	4	4	12	10	9	8	8	7	4	6	2
2	**73**	66	69	60	60	55	47	70	63	59	67	51	57	49	37	38	29	32	29	34	37
3	78	71	83	85	89	**93**	86	83	92	72	81	84	81	74	72	65	57	57	55	47	60
4	59	65	62	**72**	67	59	63	68	70	51	55	51	52	47	28	30	20	21	16	7	8
5	81	90	85	57	75	72	46	**89**	87	25	39	35	16	11	16	5	3	3	3	6	6
6	68	67	73	**86**	80	69	70	80	83	43	40	40	66	61	43	42	42	37	13	6	5
7	80	70	78	**88**	79	69	85	80	80	80	82	75	80	83	78	69	73	62	57	64	41
8	74	81	73	81	87	86	82	87	**88**	71	82	85	70	65	35	39	29	30	29	22	18
9	96	**97**	95	76	89	80	70	83	89	49	58	61	70	61	51	43	45	41	37	26	17
10	82	79	**100**	95	84	51	78	87	61	52	50	31	95	97	90	68	76	75	0	0	0
11	60	49	43	78	80	55	84	85	84	**86**	82	**86**	81	82	76	65	62	58	20	8	1
12	**100**	**100**	**100**	95	95	87	88	95	97	73	73	70	83	85	83	74	73	75	46	37	15

Across all data sets, the stNNI(I) move stood out in terms of its success in modifying the topology. For data set 2, for instance, stNNI(I) had more than 11,000 accepted topology changes; the closest competitors had little more than 3000. Whereas the best LOCAL move, LOCAL(I), succeeded with only 0.4% of its attempts to change the topology for this data set, stNNI(I) succeeded more than four times as often (1.7%). A high number of accepted topology changes correlated to some degree with rapid mixing (Fig. 1b) but not at all with burn-in times (Fig. 1b). Short burn-in times were more often associated with a large distance of accepted topology changes. The obvious exception was eSTS, which often had long burn-in times despite making radical topology modifications.

For the proposals using the extension mechanism, increasing the extension probability resulted in bolder topology rearrangements and fewer accepted proposals. The number of accepted topology changes, however, was less affected than the overall acceptance rate. In some cases, the number of accepted topology changes actually increased when going from the lowest (0.5) to the intermediate (0.8) extension probability. The boldness of the accepted topology changes also often peaked at an extension probability of 0.8. It is possible that this contributed to the intermediate tuning parameter settings (II) of these moves (eSTS, eTBR, eSPR) converging so well (Table 4).

When comparing different moves, it is clear that bolder moves had lower acceptance probabilities than more modest moves (filled circles, Fig. 3). For instance, stNNI topology changes were, on average, more than three times as likely as rSPR changes to be accepted. The acceptance rates also correlated strongly with the topological variance of the posterior distribution. The highest acceptance rates were observed for data sets with large numbers of trees in their 95% credibility sets, such as data sets 6, 7, 10, and 12 (Table 2). The three data sets (2, 3, and 5) with the smallest number of trees in their credible sets also had the lowest acceptance rates for topology changes. In other words, the more informative the data are about the topology, the more difficult it is for topology proposals to get accepted.

The branch-rearrangement proposals we studied all modify branch lengths in the neighborhood of an attempted topology change by using the multiplier move (see Methods). An alternative possibility is to map old branch lengths into the new tree without changing their values at all. We did not use such proposal mechanisms in the main experiment because their performance cannot be evaluated separately; they must be combined with branch length proposals to produce a working Markov chain, and this would have complicated the comparison with the branch-change moves, CC and LOCAL. However, we did evaluate mixtures of separate topology and branch length proposals against our combined approach in a separate experiment. In this experiment, the normal stNNI, eSPR, eTBR, and rSPR moves were contrasted with moves that were programmed identically except that the branch lengths were modified only when the proposed topology was the same as the current one.

In this experiment, we found that the probability of accepting a proposed topology change was higher when topology and branch lengths were updated separately (open circles, Fig. 3) than when they were updated simultaneously (filled circles). The difference was small for many data sets but rather pronounced for the ones with many trees in their credible sets (data sets 10 and 12 in particular). Despite the success of the separate proposals in modifying topology, the convergence success and the times to convergence were roughly comparable to that of the corresponding combined proposals (Table 5). Thus, the higher rate of accepted topology changes for separate proposals (Fig. 3) was not reflected in significant improvements in the convergence behavior.

For most data sets, combinations of the moves performed better when branch-change proposals were excluded (Table 6). With respect to time to convergence, the combined approach often outperformed the single proposal approach, especially in the runs that were started from overdispersed trees.

By visualizing tree space using multidimensional scaling and by following the chain paths through this space, it is possible to illustrate the mixing behavior of the

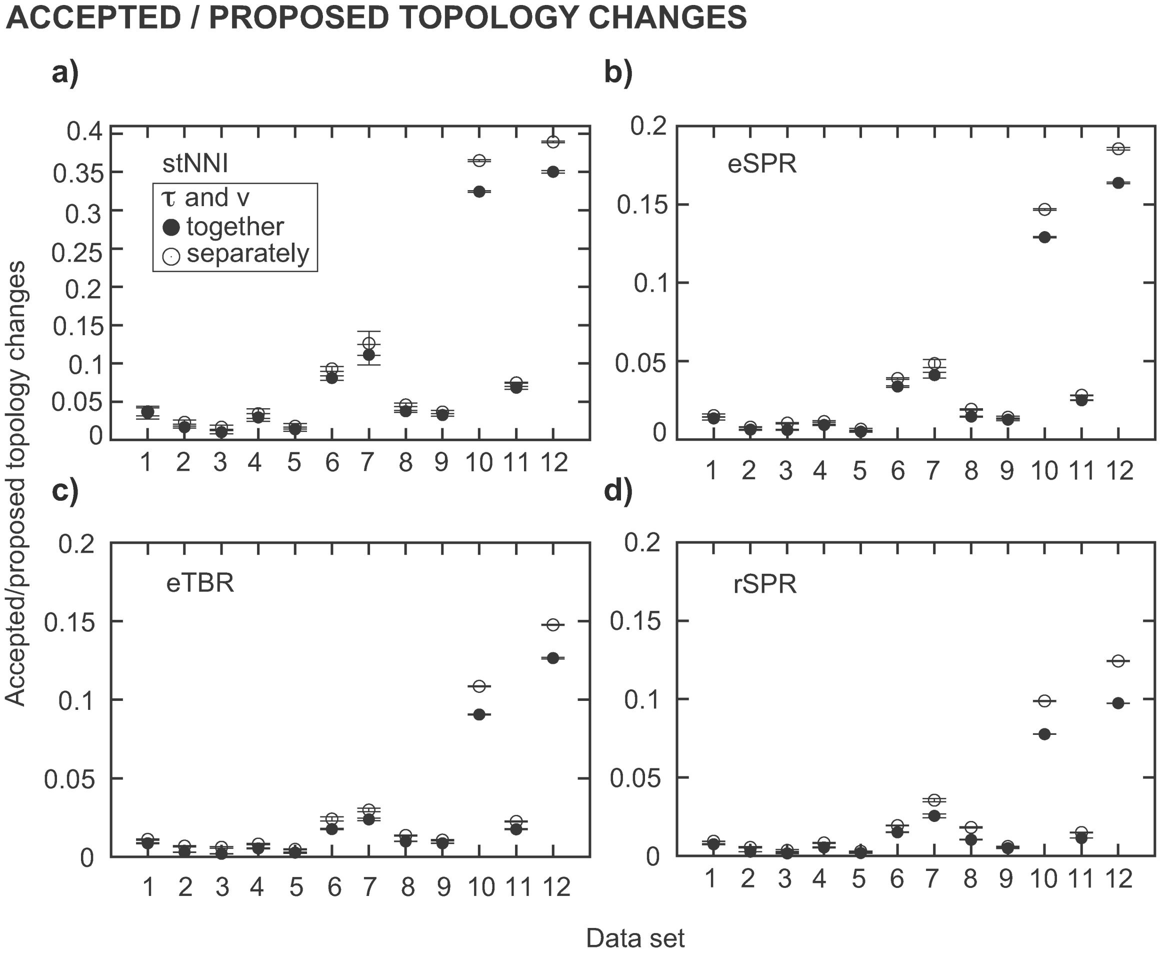

FIGURE 3. The number of proposed and accepted topology changes for four different tree proposals. Each data point is the average over 100 runs started from random trees. Error bars indicate standard deviations. Filled circles represent runs in which topology and branch lengths were updated simultaneously (the standard version of the tree proposals); empty circles represent runs where branch lengths and topology were changed separately. The overall frequency of accepted topology changes increases in the sequence stNNI > eSPR > eTBR > rSPR. The data sets with diffuse topology posteriors (data sets 10 and 12) also have a higher frequency of accepted topology changes. Proposing topology and branch length changes separately increases the frequency of successful topology modifications.

TABLE 5. Convergence success and time to convergence for independent updates of topology and branch lengths (bold: better than or same as best value in Tables 7 and 4; shaded cells: significantly worse than best value in Table 4; Conv.: *converged*).

	rSPR (II)		eTBR (I)		eSPR (II)		stNNI (I)	
Data set	% Conv.	Time	% Conv.	Time	% Conv.	Time	% Conv.	Time
1	25	7,700,000	45	7,300,000	24	7,700,000	15	7,850,000
2	69	**350,000**	54	500,000	69	**350,000**	54	550,000
3	76	550,000	82	400,000	86	350,000	79	500,000
4	56	5,300,000	**76**	**3,900,000**	**73**	**4,050,000**	47	5,100,000
5	**93**	**350,000**	85	**350,000**	78	450,000	22	850,000
6	**88**	3,500,000	79	3,200,000	**86**	**800,000**	70	4,000,000
7	82	3,900,000	75	3,900,000	87	3,000,000	77	3,400,000
8	85	1,700,000	84	1,550,000	**92**	**1,100,000**	66	2,550,000
9	95	**1,800,000**	84	2,500,000	82	2,500,000	64	3,600,000
10	89	700,000	**100**	700,000	**100**	750,000	90	650,000
11	63	7,250,000	79	6,450,000	**92**	4,750,000	**90**	5,300,000
12	**100**	**500,000**	99	850,000	96	1,100,000	87	1,400,000

TABLE 6. Convergence success and time to convergence for combinations of proposals (bold: better than or same as best value in Tables 7 and 4; shaded cells: significantly worse than best value in Table 4; Conv.: *converged*).

	All moves			Branch-rearrangement moves		
Data set	% Conv.[a]	Time 1[a]	Time 2[b]	% Conv.[a]	Time 1[a]	Time 2[b]
1	24	7,750,000	2,308,555	36	7,550,000	1,943,480
2	68	**350,000**	6,395	**73**	**300,000**	5,425
3	79	500,000	14,585	85	400,000	13,220
4	66	4,350,000	**538,990**	68	**4,150,000**	**520,680**
5	84	400,000	31,765	81	400,000	29,760
6	**88**	**2,750,000**	**965,780**	**88**	**2,750,000**	**1,019,650**
7	87	**2,600,000**	969,225	84	3,050,000	997,165
8	77	1,900,000	214,615	81	1,550,000	186,490
9	86	2,200,000	**235,690**	94	**2,100,000**	354,690
10	**100**	**550,000**	402,310	**100**	**550,000**	**384,125**
11	**89**	**5,000,000**	1,847,570	**94**	**4,850,000**	1,923,095
12	**100**	**600,000**	306,475	**100**	**650,000**	306,000

[a] Runs started from overdispersed starting trees.
[b] Runs started from trees with high posterior probability.

different proposals in more detail. For this purpose, we chose our smallest and most difficult data set (data set 1), which has only three topologies in its 50% credible set (1, 2, and 3), whereas some topologies in its 90% credible set (exemplified by topology 4) are separated from these by a broad valley (Fig. 4). Clearly, the success of sampling the posterior depends on having the right proportion of trees sampled from the space around topologies 1 to 3 *and* from the space around topology 4, on opposite sides of the valley. The more times the chain can cross that valley, the more likely it should be to get the proportions right. On average, eTBR outperforms LOCAL on this data set and it also crosses the valley more often, as evidenced by both short and long runs (Fig. 4). In the subsample of 500 trees used to generate the plots, the eTBR crossed the valley multiple times between adjacent sampling points (Fig. 4a and c), whereas the LOCAL crossed the valley only once (Fig. 4b) or twice (Fig. 4d).

Because we used the same set of starting trees for all proposals, we could examine the effect of the starting point on the convergence behavior across proposals. Among the overdispersed trees, some should offer better chances of rapid convergence than others because they are closer to the good trees. Among the good trees, some trees might be better starting points than others because they are situated in the middle of the posterior distribution rather than at some extreme. However, we could find no such effects (see Methods for details). Among the 100 starting trees, whether overdispersed or drawn from the posterior, it was not true that particular trees were more often associated with successful runs than other trees. Similarly, closeness to the good trees in topology space did not predict convergence success, nor did starting trees with a high initial likelihood result more often in convergence than other trees.

To illustrate the lack of starting-point dependence, we compared 100 runs started from different overdispersed trees with 100 runs started from the same tree (this tree being one of the 100). For this experiment we used the LOCAL and eTBR moves. As expected, the distribution of convergence times was essentially identical regardless of whether the starting points were different or the same (Fig. 5). This was true both for the LOCAL move and for eTBR, even though the mixing behaviors of these two moves are quite different, and hence also their distributions of convergence times.

The starting-point independence among random trees and among good trees contrasts starkly with the difference between them. When the runs were started from good trees, convergence was reached much faster than if the runs were started from random trees. The difference in convergence time was an order of magnitude for many data sets (Tables 7 and 8).

Finally, some general patterns are worth pointing out. Despite expectations to the contrary, we could not detect any correlation between the difficulty of sampling the posterior and the number of taxa, the number of characters, or the number of trees in the 50% credible set (Tables 1 and 2, Fig. 1). Overall, the most difficult data set was data set 1, which had the smallest number of taxa and among the smallest number of trees in its credible set. The two data sets with the smallest number of characters (6 and 10) were more difficult than average and the two data sets with the largest number of characters (1 and 2) were among the most difficult (Fig. 1a). Apparently, the exact shape of the posterior distribution was more important in determining the sampling difficulty of our data sets than factors such as tree size, number of characters, or number of trees in the credible set. It should be pointed out, however, that the variation in tree size and number of characters of the tested data sets is rather small.

DISCUSSION

To evaluate the efficiency of different tree proposals in sampling the true posterior probability distribution, it is necessary to know the latter. Typically, it is difficult to calculate this distribution, even for simulated data, unless the tree is very small. Our approach relies on using Metropolis-coupling and several independent, long

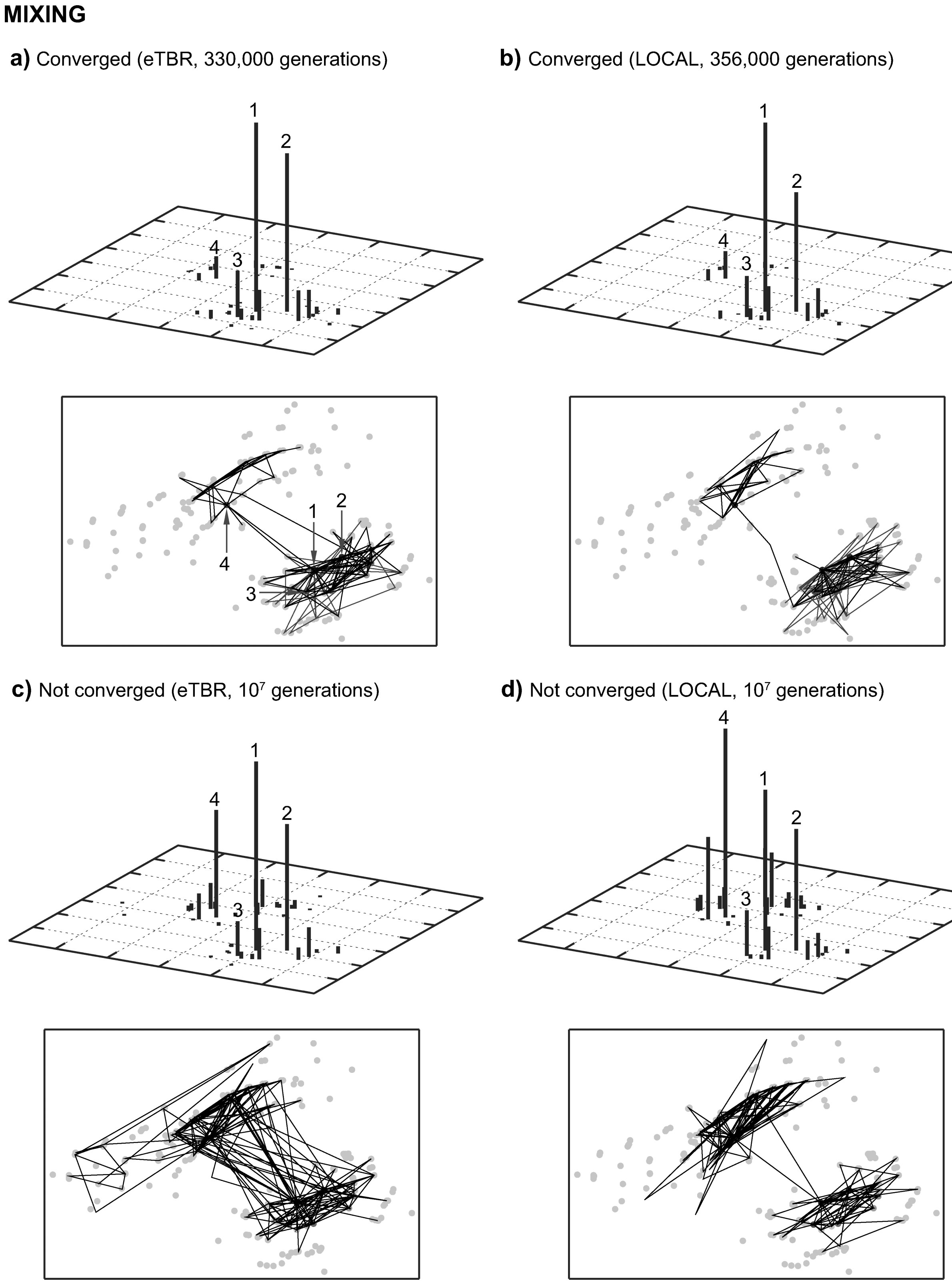

FIGURE 4. The mixing behavior of the eTBR (a and c) and LOCAL (b and d) proposals on data set 1. For each proposal, one run that converged within approximately 350,000 generations is shown (a and b), as well as one run that had not converged within 10 million generations (c and d). All four runs were started from trees sampled from the posterior probability distribution. For data set 1, the three most probable topologies (1, 2, and 3) are similar to each other but some other topologies (exemplified by 4) in the 90% credible set are separated from these by a distinct valley in topology space (a and b). Following the chain path through an evenly spaced subsample of 500 points from each run shows that the eTBR move crosses the valley much more frequently (a and c) than the LOCAL move (b and d). Gray dots indicate trees from the reference run.

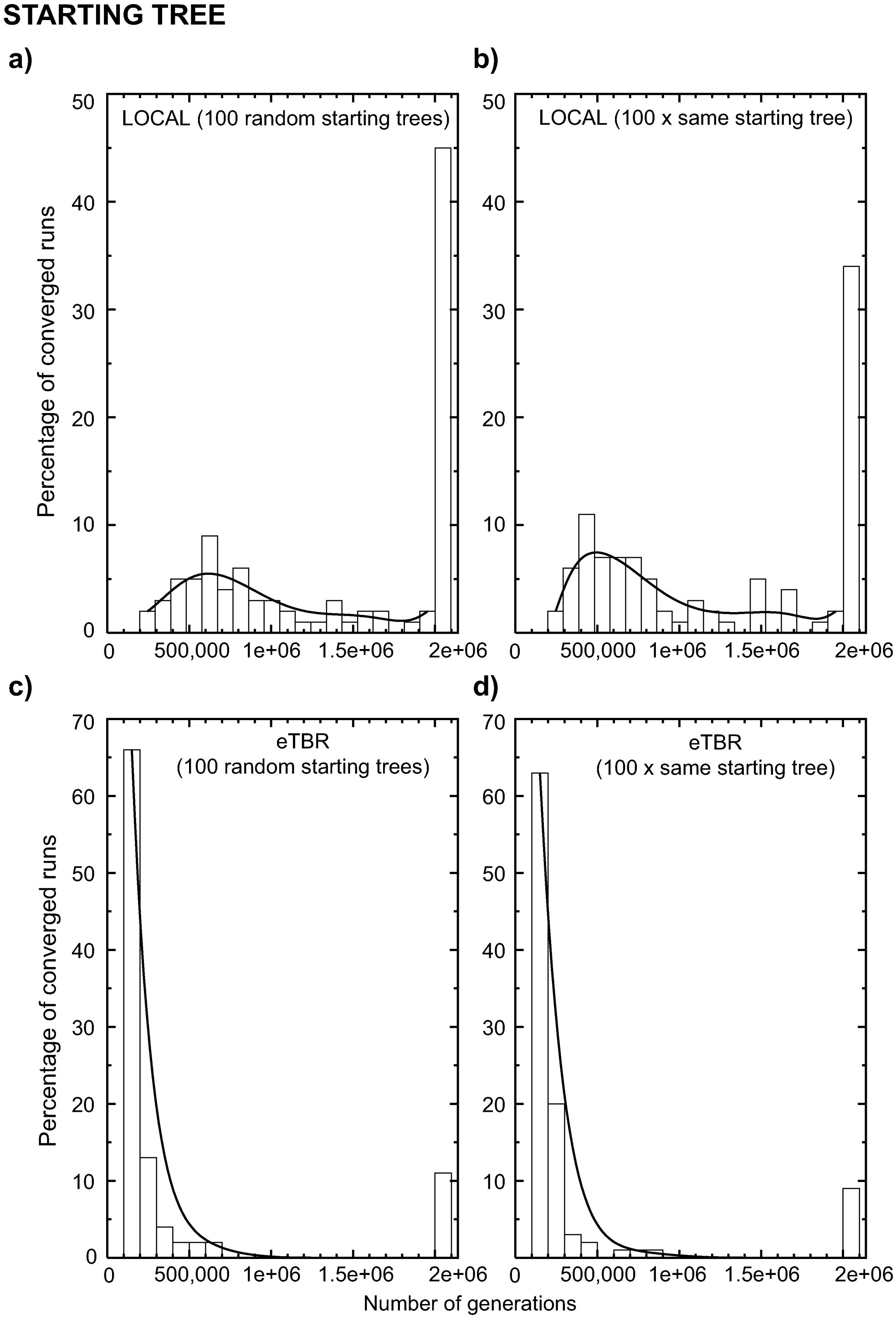

FIGURE 5. Comparison of the distributions of time to convergence for 100 runs started from different random trees (a and c) and 100 runs started from the same random tree (b and d). The plot shows the results for the LOCAL move (a and b) and the eTBR move (c and d) for data set 3. The distributions are very similar, showing that the time to convergence is not determined by the starting point.

runs to obtain a sample of the posterior that is so precise that it can be considered the true posterior. We are confident that this approach was successful in our case based, among other things, on the fact that most test runs, regardless of the proposal mechanism, eventually converged to our estimated "true" posterior.

Perhaps the most important result of our experiments is that the branch-change moves (LOCAL and CC) do not sample topology space very efficiently compared to branch-rearrangement moves. The CC move in particular had long burn-in times and mixed poorly but also the LOCAL move took a long time to burn in and it mixed

TABLE 7. Average time to convergence for runs started from overdispersed random trees (bold: best value for the data set; only values for the best tuning parameter settings are shown).

	Average number of generations until convergence						
Data set	rSPR	eTBR	eSPR	eSTS	stNNI	LOCAL	CC
1	7,600,000	**7,250,000**	7,750,000	8,000,000	7,850,000	7,900,000	8,000,000
2	400,000	450,000	**350,000**	450,000	550,000	750,000	700,000
3	550,000	350,000	**200,000**	450,000	450,000	1,300,000	950,000
4	4,900,000	**4,200,000**	4,350,000	4,850,000	4,800,000	6,400,000	6,700,000
5	450,000	500,000	**350,000**	800,000	900,000	1,000,000	1,000,000
6	4,250,000	3,150,000	**3,050,000**	5,300,000	4,050,000	5,050,000	5,900,000
7	4,300,000	**2,650,000**	3,200,000	4,450,000	2,800,000	3,800,000	5,850,000
8	2,450,000	1,650,000	1,550,000	**1,300,000**	2,400,000	4,450,000	4,550,000
9	**2,100,000**	2,150,000	2,250,000	3,700,000	3,200,000	4,550,000	5,050,000
10	750,000	600,000	700,000	900,000	**550,000**	800,000	1,050,000
11	7,350,000	5,800,000	5,450,000	5,450,000	**5,400,000**	6,900,000	8,800,000
12	**650,000**	1,300,000	1,150,000	2,400,000	1,750,000	2,250,000	4,000,000

well only for the largest trees. We do not think that these results are due to suboptimal tuning parameter settings in our experiments. The LOCAL tuning parameter has little influence on performance and the CC tuning parameter was optimized for each data set according to the suggestions by Jow et al. (2002). One could argue that the LOCAL move, for instance, is inefficient solely because it proposes topology changes less often than the most similar branch-rearrangement move, stNNI. However, even if we restrict our attention to the proposed topology changes, stNNI still succeeded better than LOCAL at getting these accepted. This was true for all our data sets, the difference usually being significant. In conclusion, we recommend that CC and LOCAL be used only in combination with branch-rearrangement moves. In general, stNNI should perhaps also be combined with other branch-rearrangement proposals.

The rSPR move is commonly used in phylogenetics MCMC software, but we found its performance to be unpredictable. In some cases where the move was successful (data sets 2, 5, and 9), the reason appears to be that it found a very small set of good trees in a large tree space more effectively than other moves. It is characteristic for these data sets that the good overall performance of rSPR is due entirely to short burn-in times because its mixing is poor (Fig. 1a and b).

In general, one might expect rSPR to do worse as the trees get larger, because the proportion of local topology changes decreases rapidly with increasing tree size for this move. This could possibly contribute to the poor performance of rSPR on data sets 10 and 11. It is more difficult to explain its good performance on data set 12. Possibly, this data set has its good trees more spread out in topology space so that a random branch-rearrangement is more likely to pick up an improved topology.

Mossel and Vigoda (2005) recently gave an example of a type of tree space that is difficult to sample from using pruning and regrafting moves because such moves have to pass through very poor intermediate topologies to walk between good ones. An interesting property of these tree spaces is that they should be easy to sample from using subtree swapping moves, because the tree islands are separated by only a single such move. We have argued elsewhere that the extreme Mossel-Vigoda tree spaces are unlikely to be encountered in real data (Ronquist et al., 2006); but if they do occur, they should be detectable because of the improved performance of subtree swapping over pruning-regrafting moves. In general, we failed to see such an effect. The single subtree swapping move we examined, eSTS, typically was rather inefficient in sampling the posterior, but there were a

TABLE 8. Average time to convergence for runs started from trees with high posterior probability (good trees; bold: best value for the data set; only values for the best tuning parameter settings are shown).

	Average number of generations until convergence						
Data set	rSPR	eTBR	eSPR	eSTS	stNNI	LOCAL	CC
1	1,841,250	**1,224,560**	2,833,390	6,602,330	3,829,935	5,370,930	7,264,890
2	16,270	8,155	6,940	12,675	**3,905**	24,400	17,175
3	24,840	17,450	11,605	22,660	**5,080**	149,390	28,710
4	957,695	660,725	**618,935**	1,132,205	827,015	1,548,300	2,917,415
5	97,020	37,820	35,540	57,020	**18,350**	69,435	78,740
6	2,290,465	1,144,960	**1,109,165**	3,430,320	1,211,845	2,178,805	5,392,490
7	1,433,180	**759,750**	1,107,670	1,655,070	1,071,670	1,184,080	2,943,680
8	630,205	313,860	296,760	**147,710**	207,485	1,805,050	2,929,400
9	740,560	287,395	362,585	462,105	**250,405**	336,775	1,181,925
10	599,050	489,500	592,880	876,940	**393,220**	686,470	4,151,135
11	4,127,870	1,951,790	2,359,795	2,535,100	**1,744,760**	3,262,490	7,507,360
12	493,225	403,780	429,145	378,080	**235,065**	393,755	2,598,430

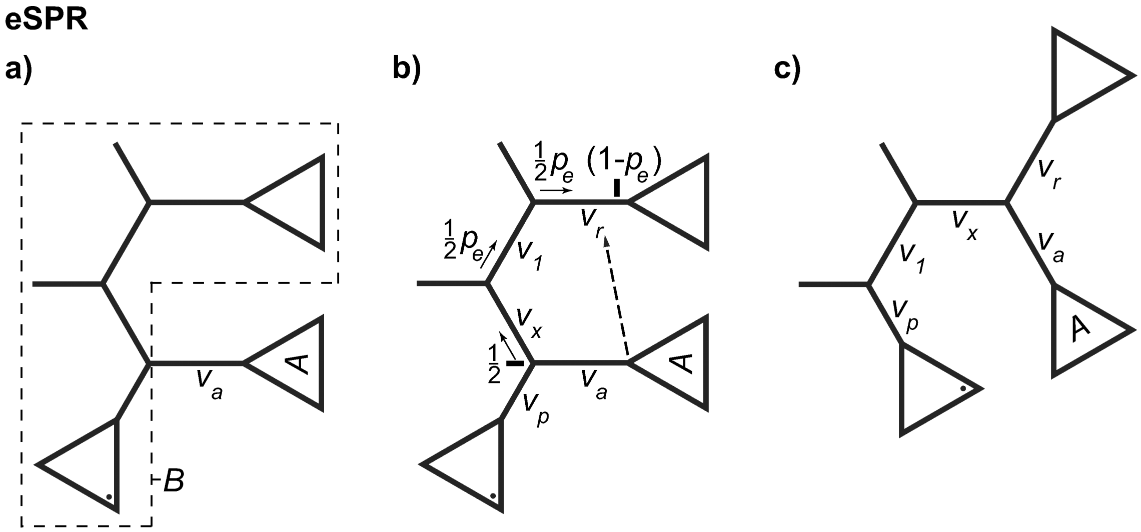

FIGURE 6. The eSPR update mechanism, also illustrating the eTBR, rSPR, and eSTS updates. First, a random interior branch with length v_a is picked at random (a). Randomly label the two subtrees attached to that branch A and B. Prune A from B and choose an initial move direction with probability 1/2. With probability p_e, the extension probability, move the regrafting point across one node, choosing one of the two possible adjacent branches with equal probability. With probability $1 - p_e$, select the current branch as the regrafting point (b). The pruning point is on a branch with length v_p and the regrafting point is on a branch with length v_r. The first branch in the moving direction has length v_x and the following branches the lengths ($v_1, v_2, \ldots, v_n$). Map these lengths into the new tree as shown (c). The rSPR move is similar except that the regrafting branch is chosen randomly in B after A has been pruned away. The eTBR move prunes and regrafts both ends of the chosen branch, whereas eSTS swaps the subtrees A and C, the latter sitting at the distal end of the regrafting branch.

couple of data sets (8 and 11) where eSTS did well. It is possible that these data sets are characterized by some valleys in the pruning-regrafting topology space that disappeared or became less prominent in the subtree-swapping space.

One might expect topology updates without branch length changes to be ineffective because they would tend to be trapped by situations where their deterministic mapping of branch lengths persistently resulted in poor branch lengths on the new tree. Some random modifications of the branch lengths should help the chain to avoid such situations. However, our comparison of separate topology and branch length updates with combined updates (Fig. 3) indicates that the separate approach in general produces a *higher* rate of accepted topology changes. This should lead to faster and more reliable convergence even though we were unable to show this with our data; at least the difference was not dramatic (Table 5). It is possible that improved branch length update strategies may support combined proposals.

The increased acceptance rate for the separate approach could potentially be due to the fact that the branch lengths converged more slowly under this approach (because branch lengths are changed more rarely, and poor branch lengths tend to increase the acceptance rates of topology changes). However, because topology changes are generally accepted infrequently, they contribute little to branch length convergence. The acceptance rates of topology changes also stabilized early in our runs, again suggesting that slower branch length convergence did not significantly affect the acceptance rates for the separate updates. Thus, separate updates do appear to be more efficient. This is also supported by the fact that the higher acceptance rates for the separate approach were observed at equilibrium. It appears worthwhile to examine whether this can be associated with a significant performance advantage in more extensive experiments than ours.

As expected, our results indicate that the details of the posterior probability distribution may affect the success of different tree proposals. However, an empiricist is typically faced with the problem of choosing a tree proposal without knowing anything about the true posterior distribution. Thus, there is considerable interest in identifying a single proposal or a mix of proposals that will do well over most types of data sets. In our results, the eTBR stands out as the obvious choice from this perspective. It mixes extremely well over most posteriors (Fig. 1b) and its convergence success rate is high throughout (Fig. 2). When it is not the best proposal, it is never far behind. Perhaps the weakest aspect of the eTBR is that its burn-in phase can be significantly longer than for the best move (Fig. 1a). However, the move with the shortest burn-in is different for each data set, so that it appears impossible to select one that always does better during the burn-in than eTBR. It may even seem doubtful whether the efficiency of eTBR can be improved much by combining it with other moves. However, we did not test proposal mixtures and some mixtures definitely deserve further exploration, such as eTBR/rSPR, eTBR/eSPR, eTBR/eSTS/rSPR, or eTBR/eSTS/eSPR. Another interesting idea that our results point to is to let the Markov chain shift during the run from bolder to more modest proposals. One could, for instance, start with an rSPR/eSTS/eSPR mixture and then gradually shift over to an eTBR/stNNI mixture according to a predetermined schema.

The fact that we could not detect any starting-point dependence among the random trees or among the good trees, but a striking difference in convergence times between them, suggests to us that the posterior distribution on tree space is accurately described by the witch's hat analogy (Geyer, 1992; Polson, 1992). Most of the posterior probability is concentrated in a tiny subspace and the remainder is distributed rather evenly across the rest of the space. The time it takes to find the tiny subspace—the good trees—from a random starting point is largely dependent on chance and walking speed, not on the distance to the subspace. Thus, it appears that the bolder topology proposals have an advantage during this burn-in phase because they tend to cover more ground in a smaller number of generations. However, the brim of the hat apparently also has a lot of local structure, so that it is not only the boldness of the proposal that matters but also the precise relation between the current and proposed topologies. For instance, the eSTS move proposes very bold topology changes but that does not help in cutting down its burn-in time, presumably because the topology neighbors separated from the current tree by a subtree swap typically have significantly lower posterior probabilities.

Because of the size and shape of most tree spaces, it is obvious that a strategy of systematically trying a large number of starting trees spread out over tree space using some space-filling algorithm is not likely to be very productive. The chance of finding a tree that is sufficiently close to the good trees to significantly speed up convergence is simply too small. A better strategy may be to find a good starting tree using a quick and dirty algorithm, such as neighbor-joining or parsimony without branch swapping. This will undoubtedly speed up convergence considerably but it comes at the cost of invalidating commonly used convergence diagnostics that rely on overdispersed starting points, such as the average standard deviation of split frequencies. A compromise solution would be to start independent runs from different trees obtained by randomly perturbing the same neighbor-joining or parsimony tree. Given an appropriate number of NNI perturbations, for instance, one should be able to obtain an overdispersed mixture of starting trees while still preserving some of the convergence speed-up. An alternative approach would be to use a space-filling algorithm to select starting trees among a set of trees produced using a partly stochastic procedure, such as stepwise addition with random addition sequences. However, the starting trees could never be guaranteed to be overdispersed with respect to the posterior distribution with these approaches, so some of the power of the topological convergence diagnostics would inevitably be lost.

We think that our results also apply to larger trees and more realistic substitution models, in particular because of the relatively consistent performance differences we observed between moves across data sets. However, it would of course be valuable to have experimental confirmation of this. In particular, we think it would be interesting to look at data sets with more characters and more taxa than the ones we studied.

A related question is how the heating used in Metropolis-coupling affects the efficiency of different topology moves. In general, one might expect bolder proposals to do better in heated chains than in the cold chain, but it is unclear how significant this effect is and whether it might be productive to run cold and heated chains with different tree proposals. The two reference samples that we obtained repeatedly with different moves (for data sets 2 and 3; Table 2) indicate that the general performance differences we observed between eTBR and LOCAL in the test runs generalize well to Metropolis-coupling with the same tree proposal applied to all chains.

Our study focused entirely on tree proposals for standard nonclock trees. Clock trees impose constraints on node depths that have important consequences for tree proposals. For instance, tree bisection and reconnection is not feasible for clock trees because the attachment point in the crown part of the tree cannot be moved to another branch, at least not without great difficulty. Similarly, both the LOCAL and CC moves need to be modified so extensively to work with clock trees that it is hardly justifiable to use the same name for them in that context (Larget and Simon, 1999). The remaining four moves that we examined (rSPR, eSPR, eSTS, and stNNI) are all relatively easy to adapt to clock trees. However, it is still an open question to what extent the performance data we collected for these moves in the nonclock context extend to clock trees.

Although random branch-rearrangement proposals are commonly used today, we think our results clearly show that it is advantageous to use an extension mechanism instead. The extending proposals can be viewed as random proposals with a bias in the proposed topologies, such that local rearrangements are favored over more distant ones. Local rearrangements are more likely to be accepted because their posterior probability is more often comparable to that of the current tree, and the higher acceptance rate for topology changes tends to speed up convergence. However, the extension mechanism we use is a rather primitive way of generating a proposal bias. We are currently in the process of examining more sophisticated mechanisms, such as favoring new topologies that have good parsimony or likelihood scores. Such proposals range from true likelihood-based Gibbs samplers, which are likely to be slow, to faster parsimony or likelihood-based Metropolis samplers. If necessary, it is quite feasible to test that the latter are correctly balanced by using them, with the bias, to sample from the prior, even though they are likely to mix poorly in this setting. In contrast, they should mix very well when sampling from the posterior. Another possibility is to improve the proposed branch lengths. Many new topologies proposed by the current moves are rejected not because the new topology has a low posterior probability but because the proposed branch lengths fit that topology poorly.

However, improving proposed topologies and branch lengths is difficult. One reason is the unavoidable tradeoff between precision and computational complexity. For instance, an increased acceptance rate due to better precision in proposed branch lengths may be completely offset by an increase in the time needed to compute each proposal. Another difficulty is the unfavorable proposal ratios that result from strongly biased proposals. Generally speaking, a strongly biased proposal is favorable only if it is matched by an equally strong response in posterior probabilities. Nevertheless, topology and branch lengths are currently the most difficult parameters to sample from in most Bayesian MCMC phylogenetics problems, so the potential payoff is significant and further research in this area should be a high priority.

ACKNOWLEDGMENTS

We would like to thank Paul Lewis, Nicolas Lartillot, and Marc Suchard for a number of helpful suggestions and comments. We would also like to thank Mark Holder, Jim Wilgenbusch, Johan Nylander, and Dave Swofford for helpful comments, suggestions, and discussion of several ideas relating to this paper. This research was supported by NIH grant R01 GM-069801 (to JPH, FR, and BL).

REFERENCES

Altekar, G., S. Dwarkadas, J. P. Huelsenbeck, and F. Ronquist. 2004. Parallel Metropolis-coupled Markov chain Monte Carlo for Bayesian phylogenetic inference. Bioinformatics 20:407–415.

Brower, A. V. Z. 2000. Phylogenetic relationships among the Nymphalidae (Lepidoptera) inferred from partial sequences of the wingless gene. Proc. R. Soc. Lond. B. 267:1201–1211.

Drummond, A. J. and A. Rambaut. 2003. BEAST v1.0. Available at http://evolve.zoo.ox.ac.uk/beast/.

Felsenstein, J. 2004. Inferring phylogenies. Sinauer Associates, Sunderland, Massachusetts.

Gelman, A., J. B. Carlin, H. S. Stern, and D. B. Rubin. 2003. Bayesian data analysis, second edition. Chapman and Hall, London.

Geyer, C. J. 1992. Practical Markov chain Monte Carlo. Stat. Sci. 7:473–483.

Green, P. J. 2003. Three-dimensional Markov chain Monte Carlo. Pages 175–194 *in* Highly structured stochastic systems (P. J. Green, N. L. Hjort, and S. Richardson, eds.). Oxford University Press, Oxford, UK.

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

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

Holder, M. T. and P. O. Lewis. 2003. Phylogeny estimation: Traditional and Bayesian approaches. Nat. Rev. Genet. 4:275–284.

Holder, M. T., P. O. Lewis, D. L. Swofford, and B. Larget. 2005. Hastings ratio of the LOCAL proposal used in Bayesian phylogenetics. Syst. Biol. 54:961–965.

Huelsenbeck, J. P., B. Larget, R. E. Miller, and F. Ronquist. 2002. Potential applications and pitfalls of Bayesian inference of phylogeny. Syst. Biol. 51:673–688.

Huelsenbeck, J. P., B. Rannala, and B. Larget. 2000. A Bayesian framework for the analysis of cospeciation. Evolution 54:352–364.

Huelsenbeck, J. P. and F. Ronquist. 2001. MrBayes: Bayesian inference of phylogeny. Bioinformatics 17:754–755.

Huelsenbeck, J. P., F. Ronquist, R. Nielsen, and J. P. Bollback. 2001. Bayesian inference of phylogeny and its impact on evolutionary biology. Science 294:2310–2314.

Johnson, M. E., L. M. Moore, and D. Ylvisaker. 1990. Minimax and maximin distance designs. J. Stat. Plan. Infer. 26:131–148.

Jow, H., C. Hudelot, M. Rattray, and P. G. Higgs. 2002. Bayesian phylogenetics using an RNA substitution model applied to early mammalian evolution. Mol. Biol. Evol. 19:1591–1601.

Jukes, T. H., and C. R. Cantor. 1969. Evolution of protein molecules. Pages 21–132 *in* Mammalian protein metabolism Academic Press, New York.

Larget, B. 2005. Introduction to Markov chain Monte Carlo methods in molecular evolution. Pages 45–62 *in* Statistical methods in molecular evolution (R. Nielsen, ed.). Springer Verlag. New York, New York.

Larget, B., and D. L. Simon. 1999. Markov chain Monte Carlo algorithms for the Bayesian analysis of phylogenetic trees. Mol. Biol. Evol. 16:750–759.

Lartillot, N., H. Brinkmann, and H Philippe. 2007. Suppression of long-branch attraction artefacts in the animal phylogeny using a site-heterogeneous model. BMC Evol. Biol. 7(Suppl 1):S4.

Lewis, P. O. 2001. Phylogenetic systematics turns over a new leaf. Trends Ecol. Evol. 16:30–37.

Li, S. 1996. Phylogenetic tree construction using Markov chain Monte Carlo. Ph.D. thesis, Ohio State University, Columbus.

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

Litzkow, M., M. Livny, and M. Mutka. 1988. Condor—A hunter of idle workstations. Pages 104–111 *in* Proceedings of the 8th International Conference of Distributed Computing Systems, San Jose, California.

Maddison, W. P., and D. R. Maddison. 2005. Mesquite: A modular system for evolutionary analysis. Version 1.06. Available at http://mesquiteproject.org.

Mau, B. 1996. Bayesian phylogenetic inference using Markov chain Monte Carlo methods. Ph.D. thesis, University of Wisconsin, Madison.

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

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

Mossel, E., and E. Vigoda. 2005. Phylogenetic MCMC algorithms are misleading on mixtures of trees. Science 30:2207–2209.

Newton, M. A., B. Mau, and B. Larget. 1999. Markov chain Monte Carlo for the Bayesian analysis of evolutionary trees from aligned molecular sequences. Pages 143–162. *in* Statistics in molecular biology, Volume 33 (F. Seillier-Moseiwitch, T. P. Speed, and M. Waterman, ed.). Institute of Mathematical Statistics, Hayward, CA.

Nylander, J. A. A. 2004. MrModeltest 2.0. Program distributed by the author. Evolutionary Biology Centre, Uppsala University. Norbyvägen 18 D. SE-752 36, Uppsala, Sweden.

Polson, N. G. 1992. Comment on "Practical Markov chain Monte Carlo" by Charles Geyer. Stat. Sci. 7:490–491.

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

Robinson, D. F., and L. R. Foulds. 1981. Comparison of phylogenetic trees. Math. Biosci. 53:131–147.

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

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

Simon, D., and B. Larget. 1998. Bayesian analysis in molecular biology and evolution (BAMBE). Department of Mathematics and Computer Science, Duquesne University, Pittsburgh, Pennsylvania. Available at http://www.mathcs.duq.edu/larget/bambe.html.

Simon, D., and B. Larget. 2004. Bayesian analysis to describe genomic evolution by rearrangement (BADGER), version 1.01 beta. Department of Mathematics and Computer Science, Duquesne University. Available at http://badger.duq.edu/.

Suchard, M. A., R. E. Weiss, and J. S. Sinsheimer. 2001. Bayesian selection of continuous-time Markov chain evolutionary models. Mol. Biol. Evol. 18:1001–1013.

Yang, Z., and B. Rannala. 1997. Bayesian phylogenetic inference using DNA sequences: A Markov chain Monte Carlo method. Mol. Biol. Evol. 14:717–724.

First submitted 28 February 2007; reviews returned 25 June 2007; final acceptance 27 September 2007
Associate Editor: Marc Suchard

APPENDIX

Tree Proposals

We derive the proposal ratios for each of the tree update mechanisms below following the approach suggested by Green (2003), in which the proposal ratio is broken into a jump-probability ratio, a random-variable-density ratio, and a Jacobian. Suppose that we draw a vector of random values $\mathbf{u}$ and use them to deterministically transform the current parameter vector $\mathbf{x}$ to the new parameter vector $\mathbf{x}^*$. The vector $\mathbf{u}$ is drawn from a distribution with the density function $g(\mathbf{u})$. To make the reverse move, we need to draw the vector of values $\mathbf{u}^*$ from the density function $g^*(\mathbf{u}^*)$ (we will assume here that $g^* = g$). We pick the move with probability $j_m(\mathbf{x})$ for the forward proposal and with probability $j_m(\mathbf{x}^*)$ for the reverse proposal. Now, the proposal ratio is

$$\frac{f(\mathbf{x}|\mathbf{x}^*)}{f(\mathbf{x}^*|\mathbf{x})} = \frac{j_m(\mathbf{x}^*)}{j_m(\mathbf{x})}\frac{g^*(\mathbf{u}^*)}{g(\mathbf{u})}\,|J\,|$$

where the last factor is the absolute value of the the Jacobian determinant, J, which is defined as

$$J = \left|\frac{\partial(\mathbf{x}^*, \mathbf{u}^*)}{\partial(\mathbf{x}, \mathbf{u})}\right|$$

We simplify the calculation of the proposal ratio by breaking the moves into independent steps, if possible. Sometimes, computational elements involved in a move can be considered either as part of the deterministic transformation of $\mathbf{x}$ or as part of the density function for $\mathbf{u}$. For example, random variables are typically generated in the computer by some transformation of a uniform random variable generated on the interval [0, 1). In such cases we prefer to reduce the complexity of the transformation of $\mathbf{x}$ by choosing a random variable u from a more complex density function. Finally, it is worth noting that the relevant Jacobian is on the random variables and the model parameters. If a move is described in terms of auxiliary variables, then the Jacobian of the transformation from the model space to the auxiliary space must be considered as well.

It is not obvious how branch lengths on one topology should be mapped into branch lengths on another. In principle, the marginal posteriors of all branch length parameters change when the topology changes, so topologies could be considered different models with completely separate branch length parameters. However, it is commonly assumed that branches that define the same splits (taxon bipartitions) are identical even if they belong to different topologies, and we follow this convention here. Thus, unless otherwise stated, if both the new and the old tree have a branch defining the same split, it is assumed that the old branch length is simply transferred to the new branch defining the same split. This transfer of branch lengths has a proposal ratio of 1 because it is deterministic and does not involve a change in model dimensionality (we only consider bifurcating topologies here).

Most of the tree proposals use the same multiplier mechanism to change branch lengths. Because this can always be described as an independent step and is sometimes used as a separate proposal mechanism, we describe it and derive its proposal ratio first. This is followed by descriptions of the proper tree proposals.

Multiplier.—The multiplier is essentially a sliding window on the log scale. Draw a multiplier value m from the distribution $g(m) = 1/(\lambda m)$ in the interval $(1/e^{\lambda/2}, e^{\lambda/2})$, where λ is a tuning parameter. If the tuning parameter is given in the form $\lambda = 2\ln a$, then m will be in the interval $(1/a, a)$. In a computer program, m would typically be generated by drawing a uniform random value u on [0,1) and then applying the transformation $m = e^{\lambda(u-0.5)}$, but the simulation procedure need not be considered in deriving the proposal ratio. Assume we apply the multiplier to a single branch length v to get the proposed branch length $v^* = mv$. For the reverse move, we need to draw the multiplier value $m^* = 1/m$. This gives us the Jacobian

$$J = \begin{vmatrix} \dfrac{\partial m^*}{\partial m} & \dfrac{\partial m^*}{\partial v} \\ \dfrac{\partial v^*}{\partial m} & \dfrac{\partial v^*}{\partial v} \end{vmatrix} = \begin{vmatrix} \dfrac{-1}{m^2} & 0 \\ v & m \end{vmatrix} = \frac{-m}{m^2}$$

Thus, the Jacobian simplifies to the product of the stretching factor for m, which is $-1/m^2$, and the stretching factor for v, which is m.

The proposal ratio using Green's method is then (ignoring j_m, which is the same for the forward and backward moves)

$$\begin{aligned} \frac{f(v|v^*)}{f(v^*|v)} &= \frac{g(m^*)}{g(m)}|J\,| \\ &= \frac{1/\lambda(1/m)}{1/(\lambda m)}\left|\frac{-m}{m^2}\right| \\ &= m^2\frac{m}{m^2} = m \end{aligned}$$

Note that the part of the Jacobian that is the stretching factor for m cancels the density ratio $g(m^*)/g(m)$, whereas the Jacobian is multiplied by m for any branch length to which the multiplier is applied. Thus, if the same multiplier m is applied to n branch lengths, the overall proposal ratio is m^n. If n different multipliers $\mathbf{m} = \{m_1, m_2, m_3, ..., m_n\}$ are applied to different branch lengths, the proposal ratio is simply their product, $\prod_i m_i$.

LOCAL.—The proposal ratio for the LOCAL was initially reported to be m^2 but was later corrected to m^3 by Holder et al. (2005) and by Larget (2005). Both of the latter papers derive the proposal ratio by considering the entire move (multiplier and topology change) and a modified parameter space. The Hastings ratio can be derived more easily for this move by considering it as two separate proposals, one a branch length multiplier move affecting three branches and one a topology proposal that does not affect branch lengths. It is easy to show that the Hastings ratio is m^3 for the first move and 1 for the latter.

Continuous change (CC).—To reverse the move we need to draw the value $u^* = -u$ from $\mathrm{N}(0, \sigma)$ as well as the right topology if there was a topology change in the forward move. Clearly, $g(-u) = g(u)$, and we also have $j_m(\tau^*, v^*) = j_m(\tau, v)$ and $J = -1$ so the proposal ratio is 1.

Stochastic nearest neighbor interchange (stNNI).—The proposal ratio for the multiplier part of the stNNI is

$$\frac{f(\tau, \mathbf{v}|\tau^*, \mathbf{v}^*)}{f(\tau^*, \mathbf{v}^*|\tau, \mathbf{v})} = m_a m_b m_c m_d m_x = \prod_i m_i.$$

Extending Subtree Pruning and Regrafting (eSPR).—We first derive the proposal ratio of the topology change part of the proposal. Consider each pair of pruning and regrafting points, (b_p, b_r), as a separate move. Then we have that the probability of a particular move, given that $b_p \neq b_r$ (Fig. 6) and that they are separated by $n+1$ branches and that we choose to move the right end of the right internal branch in the right direction, is:

$$j_m(\tau) = \left(\frac{1}{2}p_e\right)^{n+1}(1-p_e)$$

when b_r is unconstrained and

$$j_m(\tau) = \left(\frac{1}{2}p_e\right)^{n+1}$$

when b_r is constrained. This means that the probability ratio of the backward move to the forward move is

$$\begin{aligned} \frac{j_m(\tau^*)}{j_m(\tau)} &= \frac{\left(\frac{1}{2}p_e\right)^{n+1}(1-p_e)}{\left(\frac{1}{2}p_e\right)^{n+1}} \\ &= 1-p_e \end{aligned}$$

when the forward move is constrained and the backward move is not, while it is $1/(1 - p_e)$ in the reverse case. When both the forward and backward moves are constrained or both are unconstrained, the ratio is 1. Without branch length changes, it is trivial to show that the other factors of the proposal ratio are 1. Adding in the branch length changes, the proposal ratio is multiplied by $\prod_i m_i$ as shown above.

When the regrafting point is the same as the pruning point, we just randomly choose to apply the branch length multiplier either to (v_a, v_x) or to (v_a, v_p). The proposal ratio in this case is the same as the proposal ratio of the multiplier.

Extending Subtree Swapping (eSTS).—First pick a random branch b_a, a subtree A, a pruning branch b_p, and a regrafting branch b_r using the same procedure as the eSPR proposal except that the branch b_a is picked from the set of all branches, internal and terminal (otherwise the move would not always be reversible). Label the subtree rooted at the distant end of the regrafting branch, b_r, C (Fig. 6). Now swap the subtrees A and C along with the branches they sit on, b_a and b_r. Note that the subtrees A and C will consist of only a tip node in some cases.

Map old branches and branch lengths into the new tree as follows. The branches (b_a, b_p, b_r) all map to branches defining identical splits in the new tree. The branches $(b_1, b_2, \dots, b_n)$ and b_x map to branches in the new tree that define splits that are identical except that the taxa in subtrees A and C have traded places.

Finally, we apply the multiplier independently to the branch lengths (v_a, v_p, v_r, v_x) and to $(v_1, v_2, \dots, v_n)$ by drawing $n + 4$ different multipliers. If no topology change is made, then we apply the multiplier separately to v_a and v_x.

The proposal ratio for eSTS is simply the product of the length multipliers, $\prod_i m_i$. The proposal ratio is 1 for the topology part of the proposal because each subtree swap can be arrived at in two different ways, either starting at A and going to C or starting at C and going to A. Thus, the probability of choosing a particular subtree swap is necessarily the same for the forward and backward moves.