Version dated: February 6, 2018

An Annealed Sequential Monte Carlo Method for Bayesian Phylogenetics

Abstract.—Bayesian phylogenetics, which approximates a posterior distribution of phylogenetic trees, has become more and more popular with the development of Monte Carlo methods. Standard Bayesian estimation of phylogenetic trees can handle rich evolutionary models, but requires expensive Markov chain Monte Carlo (MCMC) simulations, which suffers from two difficulties, the curse of dimensionality and the local-trap problem. Recent research has shown that sequential Monte Carlo (SMC) methods can serve as good alternatives to MCMC in posterior inference over phylogenetic trees. However, the existing SMC methods mainly focus on the clock trees and have limited choices of proposal distributions. In this paper, we propose an SMC Sampler for general unrooted trees that can incorporate the MCMC kernels from the rich literature of Bayesian phylogenetics. We illustrate our method using simulation studies and real data analysis.

(Keywords: sequential Monte Carlo; phylogenetics; Markov chain Monte Carlo; continuous-time Markov chain)

Introduction

In Bayesian phylogenetics (Lemey et al. (2009); Drummond and Suchard (2010); Huelsenbeck and Ronquist (2001); Ronquist and Huelsenbeck (2003); Ronquist et al. (2012); Suchard and Redelings (2006)), the main challenge is to compute a posterior over a phylogenetic tree space. This challenging posterior computation is typically carried out by running Markov chain Monte Carlo (MCMC) algorithms for long periods Rannala and Yang (1996); Yang and Rannala (1997); Mau et al. (1999); Larget and Simon (1999); Li et al. (2000); Holder and Lewis (2003); Rannala and Yang (2003); Lakner et al. (2008); Höhna et al. (2008); Höhna and Drummond (2012). Many user-friendly software packages have been developed for implementing MCMC for phylogenetics, such as MrBayes Ronquist et al. (2012), BEAST Drummond and Rambaut (2007), and BAli-Phy Suchard and Redelings (2006). Due to combinatorial constraints, the distribution over tree space is often a complex multimodal distribution Lakner et al. (2008), and the main difficulty lies in the efficiency with which topology proposals sample the tree space. It is mainly the proposal distribution for tree topology that determines the performance of an MCMC method in Bayesian phylogenetics. The tree topology proposals include the simple moves such as Nearest Neighbor Interchange (NNI) Lakner et al. (2008) and more complicated move such as Subtree Prune and Regraft (SPR) Lakner et al. (2008); Höhna et al. (2008); Höhna and Drummond (2012). There are several drawbacks for MCMC in phylogenetic inference. First, it's very challenge to design MCMC with good mixing due to the complex posterior tree distribution. At every MCMC iteration, only very small MCMC moves are allowed as large moves result in high rejection rate. Second, it's hard to determine if the Markov chain gets converged and decide the burn in stage. Third, in order to do Bayesian model selection, the marginal likelihood is challenge and computationally expensive to achieve.

Recent research Teh et al. (2008); Görür and Teh (2009); Görür et al. (2012); Bouchard-Côté et al. (2012); Wang et al. (2015) has shown that sequential Monte Carlo (SMC)

methods can serve as good alternatives to MCMC algorithms in posterior inference over phylogenetic trees. However, most of the previous work on applying SMC to phylogenetics (Teh et al. (2008); Görür and Teh (2009); Görür et al. (2012); Bouchard-Côté et al. (2012)) are limited in the types of phylogenetic proposals they can use and cannot handle unrooted trees in a natural framework. This is an important limitation, as most current work in phylogenetics relies on unrooted tree models. To overcome the limitation, Wang et al. (2015) proposed an SMC algorithm for unrooted trees based on a graded partially ordered set on an extended combinatorial space and proposed a method to jointly infer the phylogenetic tree and the associated evolutionary parameters based on particle MCMC Andrieu et al. (2010). However, the proposal distribution of their SMC algorithm for tree topology is not flexible enough due to the imposed graded partially ordered set on the extended tree topology space. Dinh et al. (2016) focused on the theoretical framework for the online phylogenetic inference using SMC approaches. Everitt et al. (2016) described an online phylogenetic inference targeting the spaces of varying dimension for coalescent trees. Smith et al. (2017) jointly estimated the phylogenetic tree and disease transmission model via sequential Monte Carlo methods.

In contrast, We have three main contributions in this paper. First, we develop an SMC algorithm for general unrooted trees based on the framework of SMC sampler in Del Moral et al. (2006, 2007). We explore using several commonly used proposal distributions in Bayesian phylogenetics with MCMC in the proposed SMC sampler. We investigate and compare two adaptive schemes of temperature scheduling in the SMC sampler. Secondly, we focus on three estimation methods for the normalizing constant in Bayesian phylogenetics: stepping stone (widely used in MrBayes), SMC, and linked importance sampling. Thirdly, we explore different approaches to design the artificial sequence of intermediate distributions. Our implementation is available at `https://github.com/...`. We illustrate the performance of the proposed method through simulation studies and real data analysis.

Bayesian Phylogenetics

A phylogenetic tree t represents the relationship among observed taxa via a tree topology and a set of branch lengths. We consider the general unrooted trees that can handle non-constant evolutionary rates Thorne et al. (1998); Drummond and Suchard (2010).

Phylogenetic reconstruction is based on observed information located at the leaves of phylogeny. Our objective is to use n observed biological sequences, denoted $\mathcal{Y}$, to estimate the phylogenetic tree. There are some unknown parameters, denoted θ, in the evolutionary model. In the Bayesian framework, we need to specify the prior distribution and likelihood function. Let $p(\theta)$ be the prior density for θ. For a tree $t \in \mathcal{X}$, the prior density given θ is denoted by $p(t|\theta)$. Branch lengths are here considered as being part of t, not part of θ. For example, a common prior over unrooted trees consists of a uniform distribution over topologies and a product of independent exponential distributions over the branch lengths. The probability of the observed data $\mathcal{Y}$ given parameters θ and tree t is $\mathbb{P}(\mathcal{Y}|\theta, t)$.

Bayesian inference relies on the joint posterior density,

$$p(\theta, t|\mathcal{Y}) = p(\theta|\mathcal{Y})p(t|\mathcal{Y}, \theta) = \frac{\mathbb{P}(\mathcal{Y}|\theta, t)p(t|\theta)p(\theta)}{\mathbb{P}(\mathcal{Y})}, \tag{1}$$

where the normalization, $\mathbb{P}(\mathcal{Y}) = \int\int \mathbb{P}(\mathcal{Y}|\theta, t)p(t|\theta)p(\theta)\,\mathrm{d}\theta\,\mathrm{d}t$, is intractable.

In phylogenetic literature, the sites of a biological sequence are often assumed to be independent, and a continuous-time Markov chain (CTMC) is used to model the evolution of each site. Let Q denote the rate matrix of the continuous-time Markov chain. If t is rooted, the full likelihood model, $\mathbb{P}(\mathcal{Y}|\theta, t)$, is described by a directed graphical model. Unrooted trees are approached by restricting the CTMC to be reversible, a common assumption in phylogenetics. In this case, all rootings keep the likelihood invariant, so $\mathbb{P}(\mathcal{Y}|\theta, t)$ can be computed by picking an arbitrary rooting.

In a Bayesian model, the rate matrix Q is obtained from a parametric function depending

on the unknown parameter(s) θ. In this paper, for simplicity, we use the Kimuras two parameter (K2P) model Kimura (1980). The only unknown parameter, the transition/transversion rate, is denoted by κ.

The space under consideration is a joint space of all the possible trees and all the evolutionary parameters, denoted $\Omega = \Theta \times \mathcal{X}$. An MCMC algorithm generates a sequence of dependent samples of phylogenetic trees and evolutionary parameters from the space Ω that are distributed approximately according to the posterior distribution. In the next section, we will propose an SMC sampler as an alternative Monte Carlo method for Bayesian phylogenetics.

Methodology

Sequential Monte Carlo Samplers

The SMC sampler framework proposed by Del Moral et al. (2006, 2007) is a very general method for obtaining a set of samples from a sequence of distributions which can exist on the same or different spaces. This is a generalization of the standard SMC method Doucet et al. (2001) in which the target distribution exists on a space of strictly increasing dimension.

The SMC sampler mainly addressed the case of the sequence of target distributions $\{\pi_r\}$ that are defined on a common continuous space $\mathcal{X}$, e.g. π_r is the posterior distribution of a parameter given the data collected until time r, i.e. $\pi_r(x) = p(x|y_{1:r})$. The corresponding unnormalized distributions are denoted by $\{\gamma_r\}$. This SMC sampler can be obtained by defining a sequence of distributions that admit the distribution of interest, $\pi_r(x_r)$, as the recent iteration marginal

$$\tilde{\pi}_r(\boldsymbol{x}_r) = \pi_r(x_r) \prod_{j=1}^{r-1} L_j(x_{j+1}, x_j),$$

where $L_j(x_{j+1}, x_j)$ is the artificial backward Markov kernels from iteration $j+1$ to j. Then we apply the standard SMC on this sequence of distributions. Sample at iteration r,

$$x_{r,k} \sim K_r(x_{r-1,k}, \cdot),$$

where K_r is a Markov kernel defined on $E_{r-1} \times \mathcal{E}_r$. The resulting sampler has a weight update

$$W_{r,k} \propto \frac{\pi_r(x_{r,k})L_{r-1}(x_{r,k}, x_{r-1,k})}{\pi_{r-1}(x_{r,k})K_r(x_{r-1,k}, x_{r,k})},$$

which is different from the one in a standard SMC.

Algorithm 1 An SMC Sampler

sample $x_{1,k} \sim q_1(\cdot)$
set its unnormalized weight $w_{1,k} = \gamma_1(x_{1,k})/q_1(x_{1,k})$.
normalize weights $W_{1,k} = w_{1,k}/\sum_{k=1}^K w_{1,k}$
resample $\{x_{1,k}, W_{1,k}\}$ to obtain new particles denoted $\{\tilde{x}_{1,k}\}$
for $r \in 2, \ldots, R$ **do**
  sample $x_{r,k} \sim K_r(\tilde{x}_{r-1,k}, \cdot)$
  compute

$$w_{r,k} = w(\tilde{x}_{r-1,k}, x_{r,k}) = \frac{\gamma_n(x_{r,k})}{\gamma_{r-1}(\tilde{x}_{r-1,k})} \cdot \frac{L_{r-1}(x_{r,k}, \tilde{x}_{r-1,k})}{K_r(\tilde{x}_{r-1,k}, x_{r,k})}$$

  normalize weights $W_{r,k} = w_{r,k}/\sum_{k=1}^K w_{r,k}$
  resample $\{x_{r,k}, W_{r,k}\}$ to obtain new particles denoted $\{\tilde{x}_{r,k}\}$
end for

Algorithm 1 summarizes the SMC sampler. A common approach in SMC samplers is to choose $K_r(x_{r-1}, x_r)$ to be π_r-invariant, typically MCMC kernels. A convenient backward Markov kernel that allows an easy evaluation of the importance weight is

$$L_{r-1}(x_r, x_{r-1}) = \frac{\pi_r(x_{r-1})K_r(x_{r-1}, x_r)}{\pi_r(x_r)}.$$

With this backward kernel, the incremental importance weight becomes

$$\begin{aligned} w_r &= w(x_{r-1}, x_r) = \frac{\gamma_r(x_r)}{\gamma_{r-1}(x_{r-1})} \cdot \frac{L_{r-1}(x_r, x_{r-1})}{K_r(x_{r-1}, x_r)} \\ &= \frac{\gamma_r(x_r)}{\gamma_{r-1}(x_{r-1})} \cdot \frac{\pi_r(x_{r-1})K_r(x_{r-1}, x_r)}{\pi_r(x_r)} \cdot \frac{1}{K_r(x_{r-1}, x_r)} \\ &= \frac{\gamma_r(x_{r-1})}{\gamma_{r-1}(x_{r-1})}. \end{aligned}$$

MCMC Kernels for phylogenetics within an SMC sampler

The SMC sampler in Del Moral et al. (2006, 2007) provides a framework of converting an MCMC algorithm for a static distribution π into an SMC algorithm by doing MCMC moves within SMC iterations. In this section, we propose to use the standard MCMC kernels in Bayesian phylogenetics within an SMC sampler. The idea of the proposed SMC sampler is to design a sequence of artificial intermediate distributions that goes from a tractable (easy-to-sample) distribution π_1 to a distribution of interest, π_R. Each SMC iteration uses an MCMC kernel to propose artificially intermediate states, which are full trees with tempering parameters.

In Bayesian phylogenetics, the target distribution of interest is the joint posterior of a phylogenetic tree t and evolutionary parameters θ, i.e. $\pi(t, \theta) \equiv p(t, \theta|\mathcal{Y})$. For simplicity of notation, we denote $x = (t, \theta)$.

We define

$$\pi_r(x) \propto p(\mathcal{Y}|x)^{\phi_r}\pi(x), \tag{2}$$

where $0 \leq \phi_1 < \cdots < \phi_R = 1$, and $\pi_R(x) = \pi(x) = p(x|\mathcal{Y})$.

We will use the SMC sampler (Algorithm 1) with the backward kernel,

$$L_{r-1}(x_r, x_{r-1}) = \pi_r(x_{r-1})K_r(x_{r-1}, x_r)/\pi_r(x_r).$$

With this backward kernel, the incremental importance weight becomes $\gamma_r(x_{r-1})/\gamma_{r-1}(x_{r-1})$. More precisely, using Equation (2), we have

$$\gamma_r(x_{r-1})/\gamma_{r-1}(x_{r-1}) = \{p(\mathcal{Y}|x_{r-1})\}^{\Delta_r},$$

where $\Delta_r = \phi_r - \phi_{r-1}$.

A common choice for the Markov kernels, $K_r(x_{r-1}, \cdot)$, is to use MCMC kernels Del Moral et al. (2006, 2007). A typical MH kernel used in an SMC sampler is composed of the following steps:

1. Let $q(x_{r-1}, \cdot)$ be a proposal distribution. Propose a new tree and new evolutionary parameters, denoted x_r^*, from $q(x_{r-1}, \cdot)$.

2. The MH ratio is computed as

$$\alpha(x_{r-1}, x_r^*) = \min\left\{1, \frac{\pi_r(x_r^*)q(x_r^*, x_{r-1})}{\pi_r(x_{r-1})q(x_{r-1}, x_r^*)}\right\}.$$

3. With probability $\alpha(x_{r-1}, x_r^*)$, the proposal x_r^* is accepted, and with (1-$\alpha(x_{r-1}, x_r^*)$) probability, x_{r-1} remains.

In phylogenetics, there is a rich literature on using MCMC algorithms to sample the posterior phylogenetic trees. In order to take an advantage of these methods, we can combine different MCMC samplers into mixtures and cycles of several individual samplers. This is justified by a very powerful and useful property of MCMC Tierney (1994); Andrieu et al. (2003): if each of the transition kernels $\{K^i\}, i = 1, \cdots, M$, have invariant distribution π, then the *cycle hybrid kernel* $\prod_{i=1}^{M} K^i$ and the *mixture hybrid kernel* $\sum_{i=1}^{M} p_i K^i$, $\sum_{i=1}^{M} p_i = 1$, are also transition kernels with invariant distribution π.

Algorithm 2 summarizes the SMC sampler for phylogenetics where the proposal K_r^i can be any MCMC kernel, including those proposed in Bayesian phylogenetics literature, e.g. Larget

and Simon (1999); Lakner et al. (2008); Li et al. (2000); Holder and Lewis (2003). In this paper, we used the proposals K_r^i defined as follow:

1. K_r^1: the *multiplicative branch proposal*. This proposal picks one edge at random and multiply its current value by a random number distributed uniformly in $[1/a, a]$ for some fixed parameter $a > 1$ (controlling how bold the move is) Lakner et al. (2008).

2. K_r^2: the *global multiplicative branch proposal* that proposes all the branch lengths by applying the above multiplicative branch proposal to each branch.

3. K_r^3: the *stochastic NNI proposal*. We consider the nearest neighbor interchange (NNI) Jow et al. (2002) to propose a new tree topology.

4. K_r^4: the *stochastic NNI proposal with resampling the edge* that uses the above NNI proposal in (3) and the multiplicative branch proposal in (1) for the edge under consideration.

5. K_r^5: the *Subtree Prune and Regraft (SPR) move* that selects and removes a subtree from the main tree and reinserts it elsewhere on the main tree to create a new tree.

Note that here we only describe the MCMC kernels for phylogenetic trees. For estimating evolutionary parameters θ, we just need to use $\{K_r^i\}$ to propose θ.

Algorithm 2 An SMC sampler for phylogenetic trees

$x_{1,k} \leftarrow \perp, \forall k \in \{1, \cdots, K\}$
$w_{1,k} \leftarrow 1/K$
for $r \in 2, \ldots, R$ **do**
  Sample $x_{r,k} \sim \sum_{i=1}^{M} p_i K_r^i(x_{r-1,k}, \cdot)$, $\sum_{i=1}^{M} p_i = 1$
  $w_{r,k} \leftarrow \{p(\mathcal{Y}|x_{r-1,k})\}^{\phi_r - \phi_{r-1}}$
  Normalize weights $W_{r,k} \propto w_{r,k}$, and resample $\{x_{r,k}, W_{r,k}\}$
end for

Normalizing Constant

Estimate from SMC

A byproduct of the SMC algorithm is an estimate of the normalizing constant Z. We can rewrite the first constant normalizing constant as

$$Z_1 = \int \frac{\gamma_1(x_1)}{q_1(x_1)} q_1(x_1) dx_1 = \int w_1(x_1) q_1(x_1) dx_1.$$

Correspondingly, an estimate of Z_1 is

$$Z_{1,K} = \frac{1}{K} \sum_{k=1}^{K} w_{1,k}.$$

Similarly, we can rewrite the ratio of the normalizing constants as

$$\begin{aligned}
\frac{Z_r}{Z_{r-1}} &= \frac{\int \gamma_r(\boldsymbol{x}_r) d\boldsymbol{x}_r}{Z_{r-1}} = \frac{\int \gamma_r(\boldsymbol{x}_r) d\boldsymbol{x}_r}{\gamma_{r-1}(\boldsymbol{x}_{r-1})/\pi_{r-1}(\boldsymbol{x}_{r-1})} \\
&= \int \frac{\gamma_r(\boldsymbol{x}_r)}{\gamma_{r-1}(\boldsymbol{x}_{r-1})} \pi_{r-1}(\boldsymbol{x}_{r-1}) d\boldsymbol{x}_r \\
&= \int \frac{\gamma_r(\boldsymbol{x}_r)}{\gamma_{r-1}(\boldsymbol{x}_{r-1}) q_r(\boldsymbol{x}_{r-1} \to x_r)} \pi_{r-1}(\boldsymbol{x}_{r-1}) q_r(\boldsymbol{x}_{r-1} \to x_r) d\boldsymbol{x}_r \\
&= \int w_r(\boldsymbol{x}_r) \pi_{r-1}(\boldsymbol{x}_{r-1}) q_r(\boldsymbol{x}_{r-1} \to x_r) d\boldsymbol{x}_r.
\end{aligned}$$

Straightforwardly, an estimate of Z_r/Z_{r-1} is provided by

$$\widehat{\frac{Z_r}{Z_{r-1}}} = \frac{1}{K} \sum_{k=1}^{K} w_{r,k}.$$

Since the estimate of the normalizing constant can be rewritten as

$$Z \equiv Z_R = Z_1 \prod_{r=2}^{R} \frac{Z_r}{Z_{r-1}},$$

an estimate of the normalizing constant Z is

$$Z_{R,K} = \prod_{r=1}^{R}\left(\frac{1}{K}\sum_{k=1}^{K} w_{r,k}\right) = \prod_{r=1}^{R}\left(\frac{1}{K}\sum_{k=1}^{K}\{p(\mathcal{Y}|x_{r-1,k})\}^{\phi_r-\phi_{r-1}}\right), \tag{3}$$

which can be obtained from an SMC algorithm readily. Moreover, Equation (3) is an unbiased estimator of the marginal likelihood $p(\mathcal{Y})$ Del Moral (2004).

Stepping Stone

Stepping Stone (SS) (Xie et al. 2010) is an alternative method to provide an marginal likelihood estimator. The basic idea of SS is to introduce a list of annealed posterior distributions to connect the posterior distribution and the prior distribution. Let $d = 1, 2, \cdots, D$ denote the D intermediate distributions we have introduced, the normalizing constant Z can be written as

$$Z \equiv Z_R = Z_1 \prod_{d=2}^{D} \frac{Z_d}{Z_{d-1}}.$$

The ratio of Z_d and Z_{d-1} is approximated using importance sampling with importance distribution $g(x) = \pi_{d-1}(x)$, then

$$\widehat{\frac{Z_d}{Z_{d-1}}} = \frac{1}{N}\sum_{i=1}^{N}\{p(\mathcal{Y}|x_{d-1,k})\}^{\phi_d-\phi_{d-1}},$$

where $x_{d-1,k}$ is obtained by running MCMC algorithms. The number of MCMC chains we run is a trade-off between computing cost and accuracy. A larger number of MCMC chains can provide a better importance sampling approximation, but the computational cost will be higher. To make fair comparison between the marginal likelihood estimators provided by Annealed SMC and SS, we set $K_{SMC}R_{SMC} = N_{SS}D_{SS}$. Another factor that will impact the SS estimator is the strategy we choose the sequence of power $\{\phi_d\}_{d=1,2,\cdots,D}$. Xie et al. (2010) recommended to use a Beta$(a, 1)$ distribution to choose the values of ϕ_d, where a is between 0.2 and 0.4. One specific example is to choose ϕ_d, such that $\phi_d = (d/D)^{1/a}$.

Linked Importance Sampling

Stepping stone uses simple importance sampling to approximate the ratio of normalizing constant for two intermediate distributions. The IS approximation would be poor if the two successive distributions don't have enough overlaps. Linked Importance Sampling (LIS) (Neal 2005) improves the performance of IS by introducing bridge distributions, e.g. 'geometric' bridge: $\gamma_{j-1*j}(x) = \sqrt{\gamma_{j-1}(x)\gamma_j(x)}$. We could obtain the ratio of two normalizing constants by using

$$\frac{Z_d}{Z_{d-1}} = \frac{Z_{d-1*d}}{Z_{d-1}} / \frac{Z_{d-1*d}}{Z_d} = \frac{1}{N_{d-1}} \sum_{k=1}^{N_{d-1}} \frac{\gamma_{d-1*d}(x_{d-1,k})}{\gamma_{d-1}(x_{d-1,k})} / \frac{1}{N_d} \sum_{k=1}^{N_d} \frac{\gamma_{d-1*d}(x_{d,k})}{\gamma_d(x_{d,k})}.$$

LIS provides an unbiased marginal likelihood estimator. We described the LIS procedure as follows:

1. Sample an index v_1 randomly from $\{1, 2, \cdots, N_1\}$, and sample $x_{1,v_1} \sim \pi_1(\cdot)$.

2. For $d = 1, 2, \cdots, D$, sample N_d states from π_d as follows:

   (a) If $d > 1$: Sample an index v_d from $\{1, 2, \cdots, N_d\}$, and set $x_{d,v_d} = x_{d-1*d}$.

   (b) For $k = v_d + 1, \cdots, N_d$, sample $x_{d,k}$ from the forward kernel $x_{d,k} \sim K_d(x_{d,k-1}, \cdot)$.

   (c) For $k = v_d - 1, \cdots, 1$, sample $x_{d,k}$ from the backward kernel $x_{d,k} \sim L_d(x_{d,k+1}, \cdot)$.

   (d) If $d < D$, sample μ_d from $\{1, 2, \cdots, N_d\}$ according to the following probabilities:

$$p(\mu_d | x_d) = \frac{\gamma_{d-1*d}(x_{d,\mu_d})}{\gamma_d(x_{d,\mu_d})} / \sum_{k=1}^{N_d} \frac{\gamma_{d-1*d}(x_{d,k})}{\gamma_d(x_{d,k})},$$

   and set x_{d*d+1} to x_{d,μ_d}.

3. Compute the likelihood estimate

$$\hat{Z}_{LIS} = \prod_{d=2}^{D} \left[\frac{1}{N_{d-1}} \sum_{k=1}^{N_{d-1}} \frac{\gamma_{d-1*d}(x_{d-1,k})}{\gamma_{d-1}(x_{d-1,k})} / \frac{1}{N_d} \sum_{k=1}^{N_d} \frac{\gamma_{d-1*d}(x_{d,k})}{\gamma_d(x_{d,k})} \right].$$

We could also run the above procedure for multiple times, then take an average to obtain the marginal likelihood estimator.

Temperature scheduling

A simple choice for the temperature sequence is to use a deterministic schedule. That is, we choose $\phi_i = i/R$, where R is the total number of SMC iterations. In this case, the difference of successive temperatures is $\Delta_i = 1/R$. An annealed SMC with a larger number of R is computationally more expensive but has a better performance.

Adaptive scheme

The main difficulty for the temperature scheduling relies on choosing the successive temperature difference Δ_i. ESS (Del Moral et al. (2012)) at time r is

$$\mathrm{ESS}_r = \frac{1}{\sum_{k=1}^{K}\left(\frac{W_{r-1,k}w_{r,k}}{\sum_{j=1}^{K} W_{r-1,j}w_{r,j}}\right)^2} = \frac{(\sum_{k=1}^{K} W_{r-1,k}w_{r,k})^2}{\sum_{j=1}^{K} W_{r-1,j}^2 w_{r,j}^2}.$$

ESS takes values between 1 and K. ESS_r represents the number of perfect samples we are approximating π_r. A high ESS value is a necessary condition for good SMC approximation. If we choose Δ_i that is too large, then with high probability most of the particles will have very small or zero weights, which will lead to low ESS and collapse of annealing SMC algorithm. A smaller Δ_i can help improve the performance of algorithm, but the computational cost is higher, the particles may move too slow to the target distribution.

Inspired by the work of Del Moral et al. (2012), we aim to control the ESS over iterations by selecting the differences of successive temperatures Δ_r such that

$$\mathrm{ESS}_r(\Delta_r) = \alpha \mathrm{ESS}_{r-1},$$

where $0 < \alpha < 1$, and it should be close to 1 (for example, 0.99). The advantage of this adaptive

scheme is that we can automatically determine the temperatures to prevent the algorithm from being collapsed. Note that $w_{r,k} = \{p(\mathcal{Y}|x_{r-1,k})\}^{\phi_r - \phi_{r-1}}$, $\text{ESS}_r(\Delta_r)$ doesn't involve the particles at time r. We could use bisection method to find an approximate solution for Δ_r.

Conditional ESS (CESS)

If the resampling step is not conducted at iteration $r-1$, the ESS is not able to reflect the discrepancy between two successive intermediate distributions π_{r-1} and π_r. Zhou et al. (2016) propose to use the conditional ESS to measure the discrepancy. The CESS can be written in the following form

$$\text{CESS}_r = \left[\sum_{k=1}^{K} KW_{r-1,k}\left(\frac{w_{r,k}}{\sum_{k=1}^{K} KW_{r-1,k}w_{r,k}}\right)^2\right]^{-1} = \frac{K(\sum_{k=1}^{K} W_{r-1,k}w_{r,k})^2}{\sum_{k=1}^{K} W_{r-1,k}(w_{r,k})^2}.$$

Note that the CESS will be equal to the ESS when resampling is conducted at every iteration.

Simulation Studies

Data simulation and tree distance

We evaluate the proposed annealed SMC using some simulation studies. In order to simulate the datasets, we first generate a set of random unrooted trees, including topology and branch lengths, as the reference trees. The tree topology is sampled from a uniform distribution Desper and Gascuel (2004). Each branch length is generated from an exponential distribution with rate 10.0.

Then, for each reference tree, we simulate DNA sequences using the K2P model with parameter $\kappa = 2.0$. We choose an arbitrary point on the simulated reference tree as its root (the model is reversible). The data generation starts from the root of a tree by randomly sampling from the stationary distribution of the CTMC. Assuming site independence, we generate the data for

the children of the root using the transition probability computed with Q. This procedure is recursively implemented until reaching the leaves. We discard the data at the internal nodes and take the data on leaves as the simulated observations.

We summarize the sample of phylogenetic trees from the SMC sampler using the *majority-rule consensus tree* which consists of those groups that are present in no less than a half of the trees Felsenstein (1981). Then we measure the distance between a reference tree and an estimated consensus tree using three types of tree distance metrics: Robinson Foulds (RF) metric Robinson and Foulds (1981), the partition metric (PM) Felsenstein (2003), and Kuhner Felsenstein (KF) metric Kuhner and Felsenstein (1994). A small tree distance between an estimated consensus tree and its reference tree indicates good performance of the SMC sampler.

We first discard the edge directions from rooted trees to get unrooted trees. Each branch on an unrooted tree can partition the whole set of leaves into two unordered subsets, called one bipartition. We use $S(t)$ to denote the set of all the bipartitions of t: $S(t) = \{B_i, i = 1, \cdots, n_e\}$, where B_i is the bipartition resulting from the i-th edge. The set of different bipartitions of t and t' is denoted by $D(t, t') = S(t) \triangle S(t')$, where $A_1 \triangle A_2$ denotes the symmetric difference of sets A_1 and A_2. The partition metric of t and t' is defined as the number of their different bipartitions, denoted $|D(t, t')|$. The RF metric of t and t' is defined as $\sum_{B \in D(t,t')} |b(B; t) - b(B; t')|$, where $b(B; t)$ denotes the length of the branch corresponding to the bipartition B on tree t. The KF metric is defined as $\sum_{B \in D(t,t')} (b(B; t) - b(B; t'))^2$.

Comparison of normalizing constant estimates $\hat{Z}$

In this section, we emphasize the marginalized likelihood estimates provided by two schemes of Annealed SMC, LIS and SS. The CESS for adaptive annealing SMC between successive steps is set to be $\text{CESS}_r(\Delta_r) = 0.99999 \cdot \text{CESS}_{r-1}$. We run the deterministic SMC with same temperatures obtained from the adaptive SMC.

We simulate unrooted trees of various numbers of taxa: 10, 15, 20, and 25. For each tree, we generate 1 data set of DNA sequences and each sequence with length 100. The DNA sequences are simulated via continuous time Markov chain using K2P model with $\kappa = 2$. In stepping stone and linked importance sampling, we set the total number of heated chains $D = 50$, and the temperature scheme is assumed to be $\phi_d = (d/D)^3$, where $d = 1, 2, \cdots, D$. In order to make fair comparison, we set $N_{SMC}R_{SMC} = N_{SS}D_{SS} = N_{LIS}D_{LIS}$. In Figure 1, we compare the performance of the four methods in terms of the normalizing constants. SMC sampler can achieve higher marginalized likelihood estimator than SS and LIS.

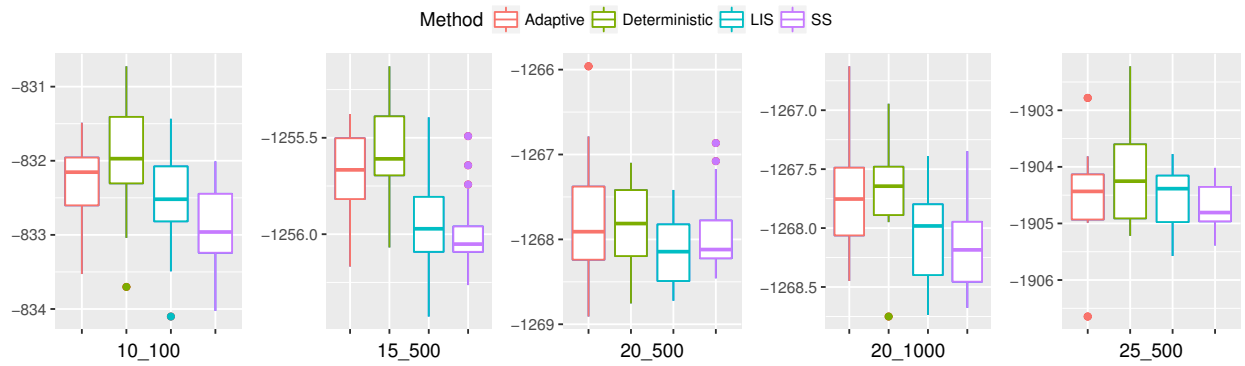

Figure 1: Normalizing constant for different number of taxa and particles, 100 sites for each sequence, $\alpha = 0.99999$ and 20 replicates.

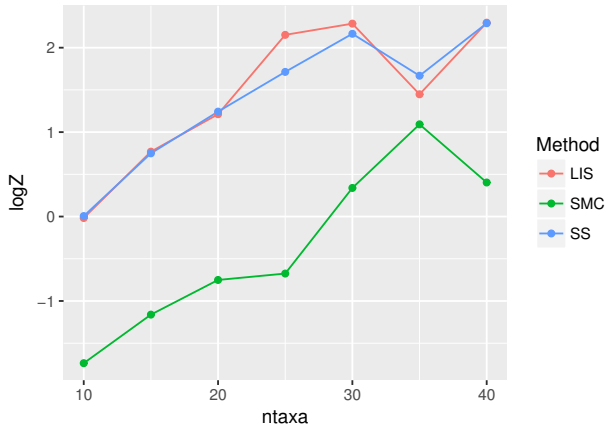

Figure 2: Logarithm of coefficient of variation versus # taxa

We use coefficients of variation to measure the variation of normalizing constant for data

sets with different number of taxa. The coefficients of variation is defined as $CV = sd(\hat{Z})/E(\hat{Z})$. Table **??** displays the logarithm of CV for $\hat{Z}$ $\left(\log(CV) = \log(sd(\hat{Z})) - \log(E(\hat{Z}))\right)$ using SS and LIS with fixed computational cost (50000 MCMC iterations). Two data sets are simulated for each number of taxa, we run 100 replicates for each case. With the increment in number of taxa, the logarithm of CV increases.

Comparison of tree metrics

In this section, we focus on the comparison of tree metrics using Annealed SMC algorithm and MCMC algorithm. To make fair comparison, we set the number of MCMC runs to be equal to $N_{SMC}R_{SMC}$. The number of taxa is set to be 50, there are 2000 sites in each sequence. We simulate one tree and one data set. We repeat two SMC algorithms and MCMC 20 times. (Now only 9 replicates). Figure **??** displays the four tree metrics, Consensus Log-likelihood, Partition Metric, Robinson-Foulds (RF) Metric and Kuhner-Felsenstein (KF) metrics obtained from SMC algorithms and MCMC. With same computational cost, the Consensus log likelihood provided by the two SMC algorithms are much higher than MCMC. The RF and KF metrics provided by SMC are much lower than MCMC. Among all these replicates, the partition metric provided by adaptive SMC are all equal to 0. MCMC has much higher partition metric.

Adaptive SMC provides higher Consensus Log-likelihood than the other two, while MCMC provides the lowest. Deterministic SMC can provide lower RF and KF metric than the other two, the metric provided by MCMC is the worst.

Table 1 displays the tree metric obtained from SMC and MCMC. Even if we run MCMC for a very long time (10 million iterations), the tree metric is still worse compared with SMC with lower computational cost (1.7 million MCMC iterations). If we use the consensus tree obtained from Adaptive SMC as the initial tree for MCMC algorithm, the MCMC is also able to achieve good performance in terms of tree metric (MCMC2 in Table 1).

Method	R	N	Metric	Value
SMC	54876	100	ConsensusLogLL	-72787.99
	54876	100	BestSampledLogLL	-72826.17
	54876	100	PartitionMetric	0
	54876	100	RobinsonFouldsMetric	0.70623
	54876	100	KuhnerFelsenstein	0.00990
MCMC	1.0E+07		ConsensusLogLL	-72833.82
	1.0E+07		PartitionMetric	0
	1.0E+07		RobinsonFouldsMetric	0.92031
	1.0E+07		KuhnerFelsenstein	0.03138
MCMC2	5.49E+06		ConsensusLogLL	-72784.86
	5.49E+06		PartitionMetric	0
	5.49E+06		RobinsonFouldsMetric	0.73644
	5.49E+06		KuhnerFelsenstein	0.01066

Table 1: Tree Distance for 50 taxa, 2000 sites each sequence using SMC and MCMC, α = 0.999999 and 100 for SMC.

Comparison of Adaptive and Deterministic Annealed SMC

In this section, we focus on the comparison of two annealing SMC algorithms in terms of marginalized likelihood estimator and tree metrics with different number of particles and different α. The simulation setup for trees is the same as previous sections (30 taxa, the length of sequence is equal to 1500). We first use an example to show one advantage of using SMC algorithm over MCMC algorithm. We ran adaptive annealing SMC on one simulated data set, using $N_{SMC} = 1000$ and $\alpha_{SMC} = 0.99$, with 100 replicates. Figure 3 displays the computing time versus different number of threads. The results indicate that by increasing the number of cores, the speed of SMC algorithm can be increased notably. In our experiments, the propagation step in SMC algorithm is paralleled.

We compare the performance of SMC algorithms using different number of particles, with $\alpha = 0.99999$. We choose four different number of particles $N_{SMC} = 100, 300, 1000, 3000$. Figure 4 indicates that both the marginal likelihood estimator and tree distance improved when we increase the number of particles. Figure 5 displays the performance of two SMC algorithms using different values of α, with number of particles equals to 1000. We select five different values,

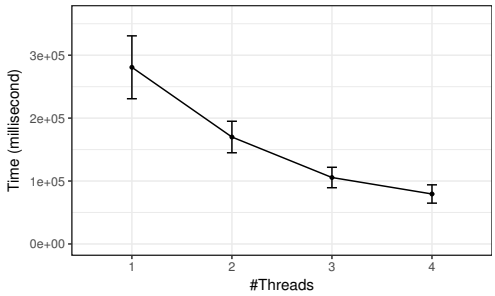

Figure 3: Computing time of Adaptive SMC using multiple threads.

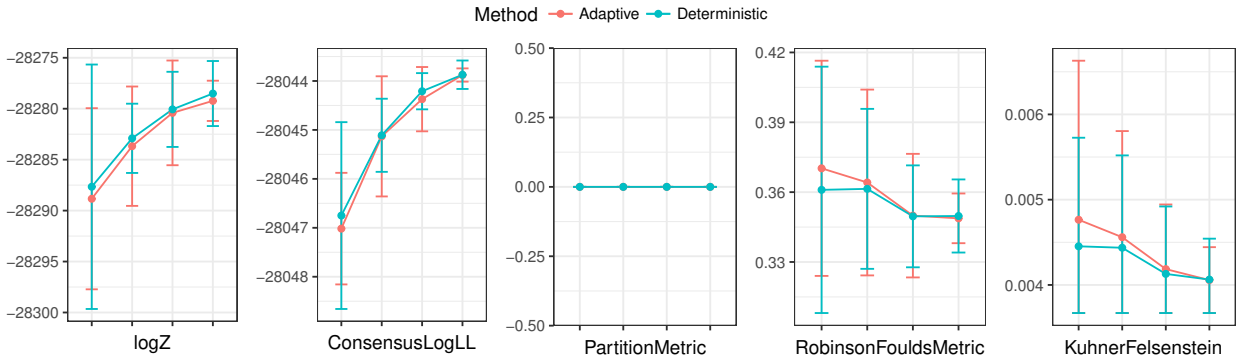

Figure 4: Comparison of two SMC algorithms with different number of particles, from left to right $N = 100, 300, 1000, 3000$.

$\alpha = 0.999, 0.9999, 0.99995, 0.99999, 0.999995$. The marginal likelihood estimates and tree distance can be improved when α increases, they tend to be stable when α reaches 0.99999.

Real datasets

Difficult datasets from TreeBASE

We use two datasets in Table 1 from "Efficiency of Markov Chain Monte Carlo Tree Proposals in Bayesian Phylogenetics" Lakner et al. (2008). The datasets are downloaded from TreeBase. The first real data we use is M336, with number of taxa equal to 27, number of sites equal to 1949. We compare the results provided by SMC algorithm with Mr.Bayes. The log marginal likelihood estimated from Annealing SMC is -65314.1, while the one estimated from Mr.Bayes using stepping stone is -7107.88. **We implement our own SS on this data set with**

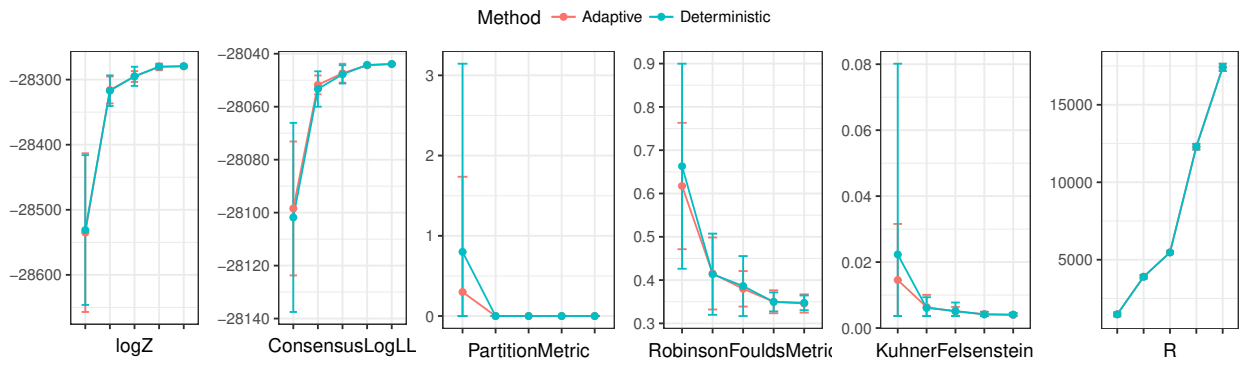

Figure 5: Comparison of two SMC algorithms with different α, from left to right $\alpha = 0.999, 0.9999, 0.99995, 0.99999, 0.999995$.

MCMC iterations 1.1 **million, the marginalized likelihood estimates is** -65316.54**.** The log likelihood estimated from Mr. Bayes is several magnitudes higher than the estimated Consensus log likelihood, which is not reasonable. **We use a small subset of M336** (10 **out of** 27 **taxa), run SMC and MB with same computational cost** ($N = 100$ **and** $\alpha = 0.99999$)**. The consensus log likelihood obtained from both SMC and MB are very close, with** $SMC = -22381.03$ **and** $MB = -22381.06$**. However, the marginalized likelihood estimates are not on the same scale, with** $SMC = -22440.95$ **and** $MB = -4165.39$**. The stepping stone results from MB doesn't seem to be reasonable. Further, we implement our own stepping stone on this data set, the marginalized likelihood is** $SMC = -22440.53$**, which is close to the estimates obtained from SMC.** Table 2 displays the tree metrics provided by SMC and MB. The computational cost is quite similar for these two methods, $R_{SMC} \cdot N_{SMC} \approx R_{MB}$. The Consensus Log likelihood estimated from SMC is higher than that from MB. The Partition Metric are both 0. The RF and KF metrics estimated from MB are slightly lower than SMC.

The second real data we use is M1809, with number of taxa equal to 59, number of sites equal to 1824. The log marginal likelihood estimated from Annealing SMC is -36212.33, the one estimated by Mr.Bayes using stepping stone is -36052.68. Table 2 displays the tree metrics provided by SMC and MB, with $R_{SMC} \cdot N_{SMC} \approx R_{MB}$. The Consensus log likelihood estimated

Method	R	N	Metric	Value
SMC	11029	1000	ConsensusLogLL	-65102.5
	11029	1000	BestSampledLogLL	-65108.9
	11029	1000	PartitionMetric	0
	11029	1000	RobinsonFouldsMetric	0.01412
	11029	1000	KuhnerFelsenstein	6.09E-06
MB	1.16E+07		ConsensusLogLL	-65132.9
	1.16E+07		PartitionMetric	0
	1.16E+07		RobinsonFouldsMetric	0.00512
	1.16E+07		KuhnerFelsenstein	1.31E-06

Table 2: TreeBASE: M336. CESS is set to 0.99999.

from SMC is higher than that from MB.

Method	R	N	Metric	Value
SMC	57029	1000	ConsensusLogLL	-35688.53
	57029	1000	BestSampledLogLL	-35702.58
	57029	1000	PartitionMetric	4.0
	57029	1000	RobinsonFouldsMetric	0.11211
	57029	1000	KuhnerFelsenstein	5.25E-4
MB	5.70E+07		ConsensusLogLL	-35691.58
MB	5.70E+07		PartitionMetric	4.0
MB	5.70E+07		RobinsonFouldsMetric	0.11533
MB	5.70E+07		KuhnerFelsenstein	6.17E-4

Table 3: TreeBASE: M1809. CESS is set to 0.999999.

Conclusion

The SMC sampler with MCMC moves provides another flexible framework to exploit the previous work in Bayesian phylogenetics using MCMC moves in an SMC algorithm. This method looks very similar to parallel tempering MCMC (Swendsen and Wang (1986)) in which subchains of tempered target distributions are implemented in parallel and value-swapping moves among subchains are used to help the chain for the target distribution to converge faster. The difference between the two methods is subtle. SMC samplers bypass the awkward

value-swapping moves. In an SMC sampler, each tempered target distribution is approximated by a set of weighted particles at each SMC iteration. Contrary to running subchains with various temperatures in parallel, an SMC sampler starts from a very flat distribution and then approaches the target distribution gradually by increasing the temperature little by little. In this way, we can alleviate the main problem of using MCMC in phylogenetics, i.e. inefficient exploration in the tree space because only small moves are allowed.

In this paper, we have considered using the SMC sampler for Bayesian phylogenetic inference. The main advantage includes that the MCMC moves designed for standard MCMC algorithms in phylogenetics can be used in the SMC sampler. The challenge mainly lies in the difficulty of determining the temperature scheduling. In order to maker the SMC sampler work well, the general rule is to choose a small temperature difference between successive SMC iterations, which might be computationally expensive due to the large number of SMC iterations. It is essential to design an efficient adaptive schedule for the temperature scheduling. We have investigated and compared two adaptive temperature scheduling schemes.

MCMC imposes relatively strict constraints on the types of proposals that can be used. More precisely, to alleviate the problem of a high rejection rate, only small moves are allowed in proposals, making it challenging to design fast mixing algorithms. In future, it is desirable to design more bold MCMC moves that are more suitable for SMC samplers. For example, we can use the "Adaptive Specification of Proposals of Toward Automatic model selection" to improve the MCMC tree moves in the SMC sampler.

*

References

Andrieu, C., N. de Freitas, A. Doucet, and M. I. Jordan. 2003. An introduction to MCMC for machine learning. Machine Learning 50:5–43.

Andrieu, C., A. Doucet, and R. Holenstein. 2010. Particle Markov chain Monte Carlo methods. Journal of the Royal Statistical Society B 72:269–342.

Bouchard-Côté, A., S. Sankararaman, and M. I. Jordan. 2012. Phylogenetic inference via sequential Monte Carlo. Systematic Biology 61:579–593.

Del Moral, P. 2004. Feynman-Kac Formulae: Genealogical and Interacting Particle Systems with Applications. Springer, New York.

Del Moral, P., A. Doucet, and A. Jasra. 2006. Sequential Monte Carlo samplers. Journal of the Royal Statistical Society B 68:411–436.

Del Moral, P., A. Doucet, and A. Jasra. 2007. Sequential Monte Carlo for Bayesian computation. Bayesian Statistics 8:1–34.

Del Moral, P., A. Doucet, and A. Jasra. 2012. An adaptive sequential monte carlo method for approximate bayesian computation. Statistics and Computing 22:1009–1020.

Desper, R. and O. Gascuel. 2004. Theoretical foundation of the balanced minimum evolution method of phylogenetic inference and its relationship to weighted least-squares tree fitting. Molecular Biology and Evolution 21:587–598.

Dinh, V., A. E. Darling, I. Matsen, and A. Frederick. 2016. Online bayesian phylogenetic inference: theoretical foundations via sequential monte carlo. arXiv preprint arXiv:1610.08148 .

Doucet, A., N. de Freitas, and N. Gordon. 2001. Sequential Monte Carlo methods in practice. Springer-Verlag, New York.

Drummond, A. and A. Rambaut. 2007. Beast: Bayesian evolutionary analysis by sampling trees. BMC Evolutionary Biology 7:214.

Drummond, A. and M. Suchard. 2010. Bayesian random local clocks, or one rate to rule them all. BMC biology 8:114.

Everitt, R. G., R. Culliford, F. Medina-Aguayo, and D. J. Wilson. 2016. Sequential bayesian inference for mixture models and the coalescent using sequential monte carlo samplers with transformations. arXiv preprint arXiv:1612.06468 .

Felsenstein, J. 1981. Evolutionary trees from DNA sequences: a maximum likelihood approach. J. Mol. Evol. 17:368–376.

Felsenstein, J. 2003. Inferring phylogenies. Sinauer Associates.

Görür, D., L. Boyles, and M. Welling. 2012. Scalable inference on Kingman's coalescent using pair similarity. Journal of Machine Learning Research 22:440–448.

Görür, D. and Y. W. Teh. 2009. An efficient sequential Monte Carlo algorithm for coalescent clustering. *in* Advances in Neural Information Processing Systems (NIPS).

Höhna, S., M. Defoin-Platel, and A. Drummond. 2008. Clock-constrained tree proposal operators in Bayesian phylogenetic inference. Pages 1–7 *in* 8th IEEE International Conference on BioInformatics and BioEngineering.

Höhna, S. and A. J. Drummond. 2012. Guided tree topology proposals for Bayesian phylogenetic inference. Syst. Biol. 61:1–11.

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

Huelsenbeck, J. P. and F. Ronquist. 2001. MRBAYES: Bayesian inference of phylogenetic trees. Bioinformatics 17:754–755.

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.

Kimura, M. 1980. A simple method for estimating evolutionary rates of base substitutions through comparative studies of nucleotide sequences. J. Mol. Evol. 16:111–120.

Kuhner, M. K. and J. Felsenstein. 1994. A simulation comparison of phylogeny algorithms under equal and unequal evolutionary rates. Mol. Biol. Evol. 11:459–468.

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

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

Lemey, P., A. Rambaut, A. J. Drummond, and M. A. Suchard. 2009. Bayesian phylogeography finds its roots. PLoS Computational Biology 5:e1000520.

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

Mau, B., M. Newton, and B. Larget. 1999. Bayesian phylogenetic inference via Markov chain Monte Carlo. Biometrics 55:1–12.

Neal, R. M. 2005. Estimating ratios of normalizing constants using linked importance sampling. arXiv preprint math/0511216 .

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

Rannala, B. and Z. Yang. 2003. Bayes estimation of species divergence times and ancestral population sizes using DNA sequences from multiple loci. Genetics 164:1645–1656.

Robinson, D. and L. Foulds. 1981. Comparison of phylogenetic trees. Mathematical Biosciences 53:131–147.

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

Ronquist, F., M. Teslenko, P. van der Mark, D. L. Ayres, A. Darling, S. Hohna, B. Larget, L. Liu, M. A. Suchard, and J. P. Huelsenbeck. 2012. MrBayes 3.2: Efficient Bayesian phylogenetic inference and model choice across a large model space. Syst. Biol. 61:539–542.

Smith, R. A., E. L. Ionides, and A. A. King. 2017. Infectious disease dynamics inferred from genetic data via sequential monte carlo. Molecular Biology and Evolution Page msx124.

Suchard, M. A. and B. D. Redelings. 2006. BAli-Phy: simultaneous Bayesian inference of alignment and phylogeny. Bioinformatics 22:2047–2048.

Swendsen, R. H. and J.-S. Wang. 1986. Replica Monte Carlo simulation of spin-glasses. Phys. Rev. Lett. 57:2607–2609.

Teh, Y. W., H. Daumé III, and D. M. Roy. 2008. Bayesian agglomerative clustering with coalescents. *in* Advances in Neural Information Processing Systems (NIPS).

Thorne, J. L., H. Kishino, and I. S. Painter. 1998. Estimating the rate of evolution of the rate of molecular evolution. Mol. Biol. Evol. 15:1647–1657.

Tierney, L. 1994. Markov chains for exploring posterior distributions. Annals of Statistics 22:1701–1762.

Wang, L., A. Bouchard-Côté, and A. Doucet. 2015. Bayesian phylogenetic inference using a combinatorial sequential monte carlo method. Journal of the American Statistical Association 110:1362–1374.

Xie, W., P. O. Lewis, Y. Fan, L. Kuo, and M.-H. Chen. 2010. Improving marginal likelihood estimation for Bayesian phylogenetic model selection. Systematic biology 60:150–160.

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

Zhou, Y., A. M. Johansen, and J. A. Aston. 2016. Toward automatic model comparison: An adaptive sequential monte carlo approach. Journal of Computational and Graphical Statistics 25:701–726.