PROTEIN FOLDING AND PHYLOGENETIC TREE RECONSTRUCTION USING STOCHASTIC APPROXIMATION MONTE CARLO

A Dissertation

by

SOOYOUNG CHEON

Submitted to the Office of Graduate Studies of
Texas A&M University
in partial fulfillment of the requirements for the degree of

DOCTOR OF PHILOSOPHY

May 2007

Major Subject: Statistics

PROTEIN FOLDING AND PHYLOGENETIC TREE RECONSTRUCTION USING STOCHASTIC APPROXIMATION MONTE CARLO

A Dissertation

by

SOOYOUNG CHEON

Submitted to the Office of Graduate Studies of
Texas A&M University
in partial fulfillment of the requirements for the degree of

DOCTOR OF PHILOSOPHY

Approved by:

Chair of Committee,	Faming Liang
Committee Members,	Jianhua Huang
	Ruzong Fan
	Jerry Tsai
Head of Department,	Simon J. Sheather

May 2007

Major Subject: Statistics

ABSTRACT

Protein Folding and Phylogenetic Tree Reconstruction Using Stochastic
Approximation Monte Carlo. (May 2007)

Sooyoung Cheon, B.A., Korea University, Korea;
M.S., Korea University, Korea
Chair of Advisory Committee: Dr. Faming Liang

Recently, the stochastic approximation Monte Carlo algorithm has been proposed by Liang *et al.* (2005) as a general-purpose stochastic optimization and simulation algorithm. An annealing version of this algorithm was developed for real small protein folding problems. The numerical results indicate that it outperforms simulated annealing and conventional Monte Carlo algorithms as a stochastic optimization algorithm. We also propose one method for the use of secondary structures in protein folding. The predicted protein structures are rather close to the true structures.

Phylogenetic trees have been used in biology for a long time to graphically represent evolutionary relationships among species and genes. An understanding of evolutionary relationships is critical to appropriate interpretation of bioinformatics results. The use of the sequential structure of phylogenetic trees in conjunction with stochastic approximation Monte Carlo was developed for phylogenetic tree reconstruction. The numerical results indicate that it has a capability of escaping from local traps and achieving a much faster convergence to the global likelihood maxima than other phylogenetic tree reconstruction methods, such as BAMBE and MrBayes.

To Hyunjoo and Jehyun

ACKNOWLEDGEMENTS

I wish to express my appreciation to my advisor, Dr. Faming Liang, for his confident and patient direction of my Ph.D. program. His insights have been extremely helpful in the preparation of this dissertation. I shall forever be in debt to him for the influence that he has been in my life. I would also like to thank the other members of my committee, Dr. Jianhua Huang, Dr. Ruzong Fan and Dr. Jerry Tsai for their continuous assistance and advice related to my research. I am also grateful to faculty, staff, and colleagues at the Department of Statistics for their friendship and support during my time at Texas A&M University. I am grateful to the Department of Statistics for funding my trip to the Joint Statistical Meetings held in August 2006 where I presented a paper.

My sincere thanks also go to Dr. John Walker at the Texas Agriculture Research and Extension Center. His support and encouragement are greatly appreciated.

My most appreciate and loving thanks go to my parents. Without their constant love and unlimited support, this dissertation would not have been possible. I would also acknowledge my parents-in-law for their encouragement and prayers. Last but not least, I am truly indebted to my wife, Hyunjoo Cho, for her endless love and support.

TABLE OF CONTENTS

LIST OF FIGURES

FIGURE Page

LIST OF TABLES

CHAPTER I

INTRODUCTION

This dissertation has utilized the stochastic approximation Monte Carlo (SAMC) algorithm for protein folding and phylogenetic tree reconstruction. Quite recently, SAMC has been proposed by Liang *et al.* (2005), which overcomes the weaknesses of the Wang-Landau algorithm (Wang and Landau, 2001) in convergence. Liang *et al.* found that the Wang-Landau algorithm is not trapped by local energy minima, but there does not exist a rigorous theory to support its convergence and thus the estimates produced by the algorithm can only reach a limited statistical accuracy. Because the self-adjusting nature of the SAMC algorithm enables it to overcome any barrier of the energy landscape, it can escape from local traps. Thus it is an excellent tool for Monte Carlo optimization.

The main problem in protein folding is the difficulty of the prediction of native structures from its sequence. The difficulties are that the dimension of the system is usually high because it is in the same order with the number of atoms involved in the system, and a multitude of local minima can be separated by high-energy barriers (i.e., the energy landscape of the system is usually complicated). Phylogenetic tree reconstruction also has high dimensionality difficulty because the number of all possible trees will increase much faster than exponential time as the number of taxa increases and local trap difficulty because the posterior probability distribution of trees can contain multiple energy minima.

This dissertation follows the style of *Journal of the American Statistical Association.*

1.1 Stochastic Approximation Monte Carlo

Before describing annealing stochastic approximation Monte Carlo (ASAMC) and sequential stochastic approximation Monte Carlo (SSAMC) algorithms, we first give a brief description for SAMC (see Liang *et al.,* 2005).

Suppose that we are interested in sampling from the following distribution,

$$p(x) \;=\; \frac{1}{Z_\tau} exp\left\{-\frac{U(x)}{\tau}\right\}, \quad x \in \mathcal{X},$$

where $\mathcal{X}$ is the sample space, Z_τ is the normalizing constant, τ is called the temperature, and $U(x)$ is called the energy function. In Bayesian statistics, $U(x)$ corresponds to the negative log-posterior density/mass function. Without the loss of generality, we assume that $\mathcal{X}$ is continuous and compact. Then the sample space $\mathcal{X}$ can be restricted to the region $\{x : U(x) \leq U_{max}\}$, where U_{max} is sufficiently large such that the region $\{x : U(x) > U_{max}\}$ is not of interest. Suppose that $\mathcal{X}$ has been partitioned according to a chosen parameterization, say the energy function $U(x)$, into m disjoint subregions denoted by $E_1 = \{x : U(x) \leq u_1\}$, $E_2 = \{x : u_1 < U(x) \leq u_2\}, \cdots, E_m = \{x : U(x) > u_{m-1}\}$ where $u_1, u_2, \cdots, u_{m-1}$ are $m-1$ specified real numbers. Let $\psi(x)$ be a non-negative function defined on the sample space $\mathcal{X}$ with $0 < \int_{\mathcal{X}} \psi(x)dx < \infty$, and $g_i = \int_{E_i} \psi(x)dx$. In practice, we often set $\psi(x) = exp\left\{-\frac{U(x)}{\tau}\right\}$, and g_i turns out to be the normalizing constant of the truncated distribution of $p(x)$ on the subregion E_i. Let $\hat{g_i}^{(t)}$ denote the estimate of g_i obtained at iteration t.

Let $\theta_{ti} = log(\hat{g_i}^{(t)})$ and $\theta_t = (\theta_{t1}, \cdots, \theta_{tm})$. The distribution can be written as

$$p_{\theta_t}(x) \;=\; \frac{1}{Z_t} \sum_{i=1}^{m} \frac{\psi(x)}{e^{\theta_{ti}}} I(x \in E_i), \quad x = 1, 2, \cdots, m.$$

Assume that $\theta_t \in \Theta$ for all t, where Θ is a compact set and set $\Theta = [-B_\Theta, B_\Theta]^m$ with $B_\Theta = 10^{20}$ for all examples in this paper, although as a practical matter this is essentially equivalent to setting $\Theta = \Re^m$. Let $T(x, y)$ denote a proposal distribution which is not necessarily symmetric and satisfy the following condition. For every $x \in \mathcal{X}$, there exist $\epsilon_1 > 0$ and $\epsilon_2 > 0$ such that $|x - y| \leq \epsilon_1 \Longrightarrow T(x, y) \geq \epsilon_2$. This is a natural condition in a study of MCMC theory (Mengersen and Tweedie, 1996). In practice, this kind of proposal can be designed easily for both continuum and discrete systems. For a continuum system, $T(x, y)$ can be set to the random walk Gaussian proposal $y \sim N(x, \sigma^2 I)$ with σ^2 being calibrated to have a desired acceptance rate. For a discrete system, $T(x, y)$ can be set to a discrete distribution defined in a neighborhood of x by assuming that the states have been ordered in a certain way. For a continuum system with the sample space consisting of several disconnected regions, we often employ a global proposal (i.e, $T(x, y) > 0$ for all $x, y \in \mathcal{X}$).

Let the desired sampling distribution $\pi = (\pi_1, \ \cdots, \ \pi_m)$ be a m-vector with $0 < \pi_i < 1$ and $\sum_{i=1}^m \pi_i = 1$ which defines the desired sampling frequency for each of the subregions. Let $\{\gamma_t\}$ be a positive, non-increasing sequence satisfying the two conditions,

$$(i) \ \sum_{t=1}^{\infty} \gamma_t = \infty, \quad (ii) \ \sum_{t=1}^{\infty} {\gamma_t}^{\zeta} < \infty$$

for some $\zeta \in (1, 2)$. In this paper we set

$$\gamma_t = \left[\frac{t_0}{max(t_0, t)} \right]^{\tau}, \quad t = 1, 2, \cdots$$

for some specified value of $t_0 > 1$ and $\tau \in \left(\frac{1}{2}, 1\right]$. A large value of t_0 will allow the sampler to reach all subregions very quickly even for a large system. With the above notations, the SAMC simulation proceeds as follows.

Let $x^{(t)}$ and $\hat{g}^{(t)}(E_i)$ denote the sample and the estimate of $g^{(t)}(E_i)$, respectively, at the tth iteration of the simulation. The simulation starts with the initial estimates $\hat{g}^{(0)}(E_1) = \cdots, = \hat{g}^{(0)}(E_m) = 1$, and then iterates as follows:

1. Propose a new configuration x^* in the neighborhood of $x^{(t)}$ according to a prespecified proposal distribution $T(x^{(t)}, \cdot)$

2. Accept x^* with probability

   $$min\left\{\frac{\hat{g}^{(t)}(E_{I_{x^{(t)}}})}{\hat{g}^{(t)}(E_{I_{x^*}})}\frac{\psi(x^*)}{\psi(x^{(t)})}\frac{T(x^* \to x^{(t)})}{T(x^{(t)} \to x^*)}, 1\right\},$$

   where I_z denotes the index of the subregion where z belongs to. If it is accepted, set $x^{(t+1)} = x^*$; otherwise, set $x^{(t+1)} = x^{(t)}$.

3. Set $\theta^* = \theta + \gamma_t(e_{t+1} - \pi)$, where $e_{t+1} = (e_{t+1,1}, e_{t+1,2}, \cdots, e_{t+1,m})$ and $e_{t+1,i} = 1$ if $x^{(t+1)} \in E_i$ and 0 otherwise. If $\theta^* \in \Theta$, set $\theta_{t+1} = \theta^*$; otherwise, set $\theta_{t+1} = \theta^* + c^*$ where $\mathbf{c}^* = (c^*, \cdots, c^*)$ can be an arbitrary vector which satisfies the condition $\theta^* + c^* \in \Theta$.

Under mild conditions, Liang *et al.* (2005) showed that

$$\theta_{ti} = \begin{cases} C + log\left(\int_{E_i} \psi(x)dx\right) - log(\pi_i + d) & \text{if } \mathrm{E_i} \neq \emptyset \\ -\infty & \text{if } \mathrm{E_i} = \emptyset \end{cases}$$

as $t \to \infty$, where $d = \sum_{j \in \{i: E_i = \emptyset\}} \pi_j/(m - m_0)$ and m_0 is the number of empty subregions, and C is an arbitrary constant. To determine the value of C, extra information is needed, e.g., $\sum_{i=1}^{m} e^{\theta_{ti}}$ is equal to a known number. Let π_{ti} denote the realized sampling frequency of the subregion E_i at iteration t. As $t \to \infty$, $\hat{\pi}_{ti}$ converges to $\pi_i + d$ if $E_i \neq \emptyset$ and 0 otherwise. Note that for a non-empty subregion, its sampling frequency is independent of its probability $\int_{E_i} p(x)dx$. This implies that SAMC is capable of exploring the whole sample space, even for the regions with tiny probabilities. Potentially, SAMC can be used to sample rare events from a large sample space. In practice, SAMC tends to lead to a "random walk" in the space of non-empty subregions being proportional to $\pi_i + d$.

The critical difference between SAMC and other stochastic approximation MCMC algorithms (Younes, 1988, 1999; Moyeed and Baddeley, 1991; Gu and Kong, 1998; Gelfand and Banerjee, 1998; Delyon, Lavielle and Moulines, 1999; Gu and Zhu, 2001) is regarding sample space partitioning. Sample space partitioning improves its performance of stochastic approximation in optimization. With sample space partitioning, SAMC can be applied to protein folding and phylogenetic tree reconstruction for optimization. Control of the sampling frequency also effectively prevents the system from getting trapped into local energy minima in simulations.

1.2 Annealing Stochastic Approximation Monte Carlo for Folding Small Proteins

The annealing stochastic approximation Monte Carlo (ASAMC) algorithm (Liang, 2006) has been applied to the study of the BLN model protein. This model uses three residue types: hydrophobic (B), hydrophilic (L), and neutral (N). The self-adjusting nature of SAMC enables it to overcome any barrier of the energy landscape. The ASAMC algorithm is an accelerated version of the SAMC algorithm for optimization problems. We propose one method for the use of secondary structures in protein folding to improve the prediction of protein folding. Our numerical results show that ASAMC is a very promising algorithm for finding the ground configurations of proteins.

1.3 Sequential Stochastic Approximation Monte Carlo for Phylogeny Reconstruction

Sampling from high-dimensional systems often suffers from the curse of dimensionality. In this chapter, we explore the use of sequential structures in sampling from high-dimensional systems with the aim of eliminating the curse of dimensionality and propose an algorithm called sequential stochastic approximate Monte Carlo (SSAMC) for alleviating the problem of local optimum traps in phylogenetic tree reconstruction (PTR). Numerical results suggest that the SSAMC algorithm is a promising tool for sampling from high-dimensional systems, it has a capability of escaping from local traps and achieving a much faster convergence to the global likelihood maxima than other phylogenetic tree reconstruction methods, such as BAMBE (Bayesian Analysis in Molecular Biology and Evolution; Simon and Larget, 2001) and MrBayes (Bayesian phylogenetic inference; Ronquist and Huelsenbeck, 2003).

CHAPTER II

ANNEALING STOCHASTIC APPROXIMATION MONTE CARLO FOR FOLDING SMALL PROTEINS

2.1 Introduction

In recent years there has been a great deal of interest in studying the prediction of protein folding from its amino acid sequence in biophysics. Proteins are not linear molecules such as a “string” of amino acid sequences. Rather, this “string” folds into an intricate three-dimensional structure that is unique to each protein. This three-dimensional structure allows proteins to function. Native-state topology often plays a dominant role in the kinetics of this folding process. This implies that interactions among 20 different amino acids give rise to cooperative formation of native structure through backbone hydrogen bonding and specific side-chain packing of the native-state core.

Molecular modeling attempts to predict the structure of a protein *ab initio* (i.e., by trying to apply laws of physics to describe the protein molecule) rather than by using databases of known structures. It is assumed that the native state of the molecule is given by those parameter values that minimize the energy function. The energy function may have multiple minima; in this case the molecule is assumed to have multiple native states, which indeed occurs in reality. Energy functions take into account all atoms in a protein molecule. Since the number of amino acids in proteins ranges from 25 to 3000 and the number of atoms ranges from around 500 to more than 10000, dealing with even the simplest energy functions may be a difficult computational task.

Some good prediction results can be obtained by using methods such as threading based on information on known structure (Moult *et al.,* 1999; Venclovas *et al.,* 1999). However, for small proteins the information contained in the amino acid sequence determines sufficiently the tertiary structure of protein with the lowest minimum energy value. Lu *et al.* (2003) noticed this point and suggested a new minimization function called "relative entropy". But, it is difficult to find the thermodynamically stable state of the protein; it is a NP-hard problem. The difficulties of the problem are that the dimension of the system is usually high because it is in the same order with the number of atoms involved in the system, and the energy landscape of the system can be characterized by a multitude of local minima separated by high-energy barriers. At low temperatures, traditional Monte Carlo and molecular dynamic simulations tend to get trapped in local minima. Hence, only a small fraction of the phase space is sampled, and the thermodynamic quantities cannot be estimated accurately. Thus it makes the simulations ineffective.

Many studies to alleviate this problem have been developed and successfully applied to various problems such as methods searching for the lowest potential energy conformation [Monte Carlo with minimization (Li and Scheraga, 1987), simulated annealing (Kirkpatrick *et al.,* 1983), and genetic algorithms (Holland, 1975; Goldberg, 1989)] and methods sampling the phase space with more efficient samplers [multicanonical (Berg and Neuhaus 1991; 1992), entropic sampling (Lee, 1993), parallel tempering (Geyer, 1991; Hukushima and Nemoto, 1996), simulated tempering (Marinari and Parisi, 1992), $1/k$-ensemble sampling (Hesselbo and Stinchcomb, 1995), chain growth algorithms (Rosenbluth and Rosenbluth, 1955; Grassberger, 1997; Frauenkron *et al.,* 1998; Bastolla *et al.,* 1998), and Metropolis algorithms with long range moves (Ramakrishnan *et al.,* 1997; Deutch, 1997)].

Honeycutt and Thirumalai (1990, 1992) and Veitshans *et al.* (1997) introduced an off-lattice model protein. This off-lattice model uses an α carbon trace to represent the protein backbone. The so-called BLN proteins are modeled as a sequence of beads of three types: hydrophobic (B), hydrophilic (L), and neutral (N). These models exhibit many similarities with real proteins, and thus are particularly useful in the study of predicting the native structure of a protein. Honeycutt and Thirumalai (1990, 1992) studied a 46-residue BLN protein, and Guo *et al.* (1992), Honeycutt and Thirumalai (1992) and Thirumalai and Guo (1995) demonstrated that the folding kinetics of the BLN 46-mer is very similar to that of real proteins. Brown *et al.* (2003) showed that the sequence mapping from a 20-letter amino acid code to the three-letter reduced code is sufficient for determining the folding to a target topology. Furthermore, there have been efforts in finding the global minimum-energy conformation by studying this off-lattice model protein to avoid the high energy barriers between local minima in recent years (Kim *et al.*, 2003).

In this chapter, we apply the ASAMC algorithm to an off-lattice (BLN) model protein. The SAMC algorithm is a generalization of the Wang-Landau algorithm (Wang and Landau, 2001) and the 1/k-ensemble algorithm (Hesselbo and Stinchcomb, 1995). It can be used for both continuous and discrete systems. The self-adjusting nature of the algorithm enables it to overcome any barrier of the energy landscape. The ASAMC algorithm is an accelerated version of the SAMC algorithm for optimization problems. Our numerical results show that ASAMC is a very promising algorithm for finding the native topology of proteins.

The remaining part of this chapter is organized as follows. In Section 2.2, we describe the BLN model. In Section 2.3, we review the ASAMC algorithm briefly. Section 2.4 presents numerical results for real small proteins. The applications of the use of secondary structures in protein folding are given in Section 2.5.

2.2 Model

The model is a simple BLN protein model proposed by Kim *et al.* (2003). The protein chain is modeled as a sequence of beads of three types: hydrophobic (B), hydrophilic (L), and neutral (L) (Table 1). Hydrophobic beads tend to pull each other to form a strong core. Hydrophilic beads tend to push other beads, balancing the forces and reducing the bias of the correct native fold. The neutral beads also typically give a signal of the turn regions in the sequence.

The total potential energy function of the BLN model with M residues is given by

$$\begin{aligned} V \;=\; & \sum_{i=2}^{M} \frac{K_r}{2}(|r_i - r_{i-1}| - a)^2 \;+ \sum_{i=3\ (angles)}^{M} \frac{K_\theta}{2}(\theta_i - \theta_0)^2 \\ + & \sum_{i=4\ (dihedrals)}^{M} [A_i(1+cos\phi_i) + B_i(1+cos3\phi_i)] \\ + & \; 4\epsilon \sum_{i=1}^{M-3} \sum_{j=i+3}^{M} C_{ij}\left[\left(\frac{\sigma}{r_{ij}}\right)^{12} - D_{ij}\left(\frac{\sigma}{r_{ij}}\right)^{6}\right], \end{aligned}$$

where the force constants are given by $k_r = 400\epsilon/a^2$ and $k_\theta = 20\epsilon/(rad)^2$, ϵ is the energy constant, a is the average bond length, r_i is the position of the ith residue, σ is the Lennard-Jones parameter, and r_{ij} is the distance between two nonbonded residues i and j given by $r_{ij} = |r_i - r_j|$.

Table 1: Sequence mapping between 20-letter (20) amino acid and coarse-grained three-letter (3) code

20	3	20	3	20	3	20	3
Ala	B	Met	B	Gly	N	Asn	L
Cys	B	Val	B	Ser	N	His	L
Leu	B	Trp	B	Thr	L	Gln	L
Ile	B	Tyr	B	Glu	L	Lys	L
Phe	B	Pro	N	Asp	L	Arg	L

Bond lengths are held rigid. The bond angle θ_i is defined by three residuals $i-2$, $i-1$, and i and is maintained by a harmonic potential with force constant k_θ and equilibrium bond angle $\theta_0 = 1.8326$ rad or 105^o. First and second parts in the total energy are called the bond-stretching energy and the bond-angle bending energy, respectively. In the dihedral or torsional angle energy, each dihedral ϕ_i in the chain is defined by four residues $i-3, i-2, i-1$, and i and predefined: $A_i = 0$ and $B_i = 0.2\epsilon$ if two or more of the four residues are neutral; $A_i = B_i = 1.2\epsilon$ for all the other cases. Finally, the non-local interactions in the van der Waals energy are given by $C_{ij} = D_{ij} = 1$ for BB interactions, $C_{ij} = 2/3$ and $D_{ij} = -1$ for LL and LB interactions, and $C_{ij} = 1$ and $D_{ij} = 0$ for all interactions involving N residues. The attractive forces in the model responsible for collapse are due to the interactions between hydrophobic beads (B-B interactions). The interactions among all other combinations of beads are repulsive, although different strengths of repulsion are used depending on the bead types involved (Brown *et al.,* 2003). All simulations are performed in reduced units, with a, ϵ and σ all set equal to unity.

2.3 The Annealing Stochastic Approximation Monte Carlo Algorithm

In order to accurately predict protein folding, we must focus on minimizing $U(x)$. Liang (2004) proposed a space annealing version of a contour Monte Carlo algorithm for structure optimization in an off-lattice protein model and showed that it can be applied successfully to finding ground states. Liang (2006) also suggested the space annealing version of the SAMC (ASAMC) algorithm for neural network training and numerical results indicated that ASAMC is superior to simulated annealing and the gradient-based algorithms in MLP training. Thus, we make use of this annealing concept to accelerate the optimization process of protein folding. The basic concept is that the sample space is limited at each iteration of SAMC to a small region to accelerate the process because the process may be slow due to the broadness of the sample space. We give a brief description for this ASAMC algorithm [see Liang (2004, 2006)].

Suppose that the subregions $E_1, \cdots, E_m$ have been arranged in ascending order by energy. Let $\varpi(u)$ denote the index of the subregion that a sample x with energy $U(x) = u$ belongs to. For example, if $x \in E_i$ then $\varpi(U(x)) = i$. Let $\mathcal{X}^{(t)}$ denote the sample space at iteration t. The simulation process is as follows:

1. Start with $\mathcal{X}^{(1)} = \bigcup_{i=1}^{m} E_i$; i.e., all subregions are used.

2. Set iteratively

$$\mathcal{X}^{(t)} = \bigcup_{i=1}^{\varpi(U_{min}+\triangle)} E_i,$$

   where U_{min} is the minimum energy value obtained so far in the run, and $\triangle > 0$ is a user-specified parameter.

The phase space $\mathcal{X}^{(t)}$ shrinks iteration by iteration. In this sense, this modified algorithm is called annealing SAMC (ASAMC).

We considered several issues for an effective implementation of ASAMC:

- Partition of the sample space. The sample space can be partitioned according to the energy function which allows for minimizing the energy function. The maximum energy difference in each subregion should be bounded by a reasonable number (e.g., 1) which ensures that the local Metropolis Hastings moves within the same subregion have a reasonable acceptance rate.

- Choice of $\triangle$. The performance of ASAMC depends on the value of $\triangle$. If $\triangle$ is too large or small, ASAMC may take a long time to locate the global minimum due to the broadness of the sample space. In this case, the sample space may contain only a few separated regions, and most of proposed transitions will be rejected. It is generally believed that allowing a sampler to jump to intermediate states of high energy will increase the probability of transitions from one local energy minimum to others. The proposal distribution used in ASAMC should be more spread out than that used in SAMC in order to reduce the negative effect of the sample space restriction. In this chapter, we set $\triangle = 50$. This value works well for all cases considered in my research.

- Desired sampling distribution π. Generally, π should be chosen to bias sampling to the low energy subregions in order to increase the chance of finding the global minima because it controls the sampling frequency of each subregion. However, sampling in ASAMC has been restricted to the low energy subregions by choosing an appropriate value for $\triangle$. Thus, considering a good choice of π is not necessary for ASAMC. For protein folding, we set π to be uniform on the subregions $E_1, \cdots, E_{\varpi(U_{min}+\triangle)}$.

- Choice of N, τ, and t_0. Here N is the total number of iterations of a run, and τ and t_0 determine the gain factor $\{\gamma_t\}$. The gain factor controls the ability of ASAMC moving across subregions. We fixed $\tau = 1.0$ and $t_0 = 2500$ in all simulations of this case. The appropriateness of the choice of t_0, τ, and N can be diagnosed by examining the convergence of the run. In ASAMC, the desired sampling distribution has been set to be equal to the realized sampling frequencies of these subregions. As suggested by Wang and Landau (2001), because it is impossible to obtain a perfectly flat histogram, convergence of a run means that the sampling frequency for each subregion is not less than 80%

of the average sampling frequency; that is,

$$\epsilon_f = min\left\{\frac{f_i}{\overline{f}}\ :\ i = 1, \cdots, \varpi(U_{min} + \triangle),\ E_i \neq \emptyset\right\} \geq 80\%$$

where f_i denotes the realized sampling frequency of the subregion E_i, and $\overline{f}$ is the average sampling frequency of the subregions included in $\mathcal{X}^{(\infty)}$.

2.4 Numerical Results

A simple representation was adopted as the BLN model, in which a residue was reduced to a bead and its coordinate was in the position of the C_α atom of the residue. Two small proteins, 1a7f and 9ins, were selected as the tested targets from the RCSB Protein Data Bank (Berman *et al.*, 2000; www.rcsb.org). For both proteins, we partitioned the phase space into $E_1, \cdots, E_{201}$ with an equal energy bandwidth of 1.0; that is, we set $E_1 = \{x \in \mathcal{X} : H(x) \leq -100\}$, $E_2 = \{x \in \mathcal{X} : -100 < H(x) \leq -99\}, \cdots$, $E_{200} = \{x \in \mathcal{X} : 99 < H(x) \leq 100\}$, and $E_{201} = \{x \in \mathcal{X} : H(x) > 100\}$. In simulations, we set $\tau = 1.0,\ t_0 = 2500$ and $n = 10^8$, where n denotes the number of iterations performed in the simulation. We had three types of local moves as follows. These moves happen equally likely at each iteration. Let $x^{(t)} = (\theta_3, \cdots, \theta_N, \phi_4, \cdots, \phi_N)$ denote the current state. In the type-I move, a component of $x^{(t)}$ is selected randomly by modifying with Gaussian random variable $\epsilon \sim N(0, s^2[H(x^{(t)}) - H_0])$, where s is a user tunable parameter and H_0 is a user-guessed lower bound for $H(x^{(t)})$. We set $H_0 = -100$. The variance of ϵ suggests a different step size $s\sqrt{H(x^{(t)}) - H_0}$ for different states. The step size is large for high-energy states, and the step size is small for low-energy states. This allows the sampler to move through the high-energy region fast and explore the low-energy region in detail. The type-II move is the same with the type-I move, except for which two components are picked up randomly. In the type-III move, a spherical proposal distribution is used. A direction is generated uniformly, and then the radius is drawn from $N(0, 2s^2[H(x^{(t)}) - H_0])$. In this case, we set $s = 4.0$ for 1a7f, and $s = 3.0$ for 9ins.

The ASAMC algorithm was run 5 times independently. The computational results are summarized in Table 2. It was found that the root mean square deviation (r.m.s.d.) data are not in the reasonable range (with r.m.s.d. values ranging from 3 to 7.5 for small proteins; Lu *et al.,* 2003). However, the comparison shows that the ASAMC algorithm has made a significant improvement over the simulated annealing (SA) and the conventional Metropolis Monte Carlo (MH) method in locating the ground states for the BLN model protein. For all proteins, the average minimum energy found by the ASAMC algorithm is better than the minimum energy by SA and MH in all runs. The differences of the energy values come from the differences of the folding predicted angles. The values of r.m.s.d. were obtained from distance between the folded predicted protein structures and their native structures. Protein structures tested are as follows: 1a7f (29 residues - BBLLLBBNNL BBLBBLBBBN LLNNBBLNL); 9ins (30 residues - BBLLLBBNNL BBLBBBBBBN LLNBBBLNLB). Figure 1 shows that the folded predicted structure using the ASAMC algorithm is somewhat similar to, but is not exactly the same as, the native structure (9ins). That is, there may be a problem in the turn direction of the predicted structure. Thus, we need to improve our method.

2.5 The Use of Secondary Structure in Protein Folding

Levinthal (1968) and Wetlaufer (1973) argued that the time for a random search of all possible structures would be unrealistically long for even a small protein, and that something like a nucleation event must occur to permit structure formation in biologically feasible time. In this study, they found that distinct structural regions

Table 2: Comparison of ASAMC with SA and the conventional Metropolis

Protein	NE[a]	Metropolis(rmsd)[b]	SA(rmsd)[c]	ASAMC(rmsd)[d]
1A7F (29)	60.7142	24.7843(19.584)	14.5720(16.012)	-3.8187(8.374)
9INS (30)	61.9850	34.3073(15.576)	-3.9989(21.600)	-6.5567(9.552)

a : The energy of the native structure. b : The average minimum energy value and r.m.s.d. data of sample by the Metropolis method. c : The average minimum energy value and r.m.s.d. data of sample by the simulated annealing. d : The average minimum energy value and r.m.s.d. data of samples by the ASAMC algorithm.

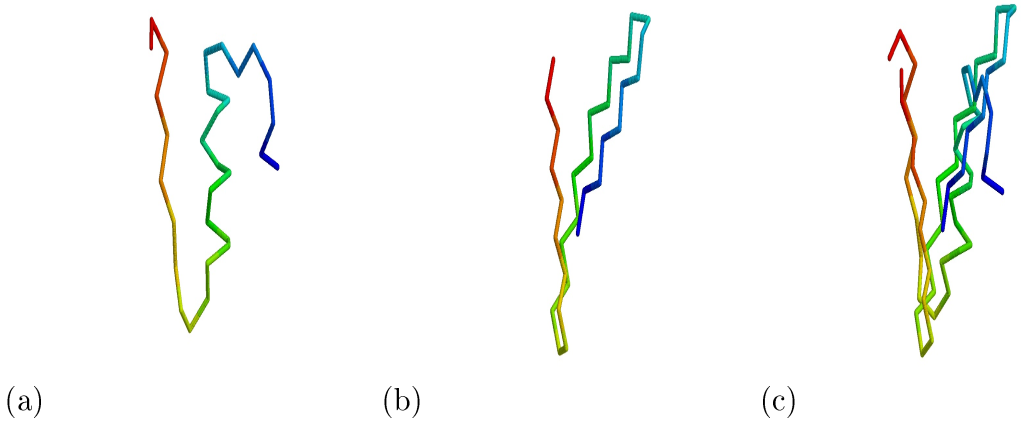


Figure 1: Comparison native structure with best target structure predicted by ASAMC for 9ins. (a) A native structure. (b) A folded best structure generated by ASAMC. (c) The native structure superimposed on the best structure found by ASAMC.

have been found in several globular proteins composed of single polypeptide chains, and proteins fold much too fast to involve an exhaustive search. This is the so called Levinthal paradox: how can a protein find a native state without a globally exhaustive search? Experiments show that a protein folds to its native state according to a relatively small number of pathways [i.e., it folds by a specific sequence of molecular events (Creighton, 1978; Kim and Baldwin, 1990)]. A pathway is defined by the intermediate states and the transition states which occur between the initial and final states (Creighton, 1978). Some pathways are strongly favored [i.e., folding is "cooperative" in that a "nucleating" HH contact acts as a constraint that restricts local conformational freedom on 2D hydrophobic-hydrophilic (HP models) and speeds the "zipping up" of other contacts nearby]. Mainly two types of interaction contribute to protein folding: (i) the helix-coil propensities, among monomers that are connected neighbors in the chain sequence; (ii) the hydrophobic and solvent interactions among monomers that may either be near each other or far apart in the chain sequence (Miller *et al.,* 1992; Dill *et al.,* 1993). Based on this information, Liang and Wong (2001) proposed one method for the use of secondary structures for protein folding in 2D HP model and showed that it is very successful in finding low energy states.

Motivated by the above studies, we add the following steps using the secondary structures to the ASAMC algorithm for speed-up of the simulation:

1. Sample the subsequences at random in the library consisting of the number of residues, bond and dihedral angles, which will possibly fold to secondary structures in the native states of known proteins.

2. Perform sampling on the constrained conformation space where some subsequences are subject to possible secondary structures.

The representation we use for protein subsequences is based on a library of various fragments of protein backbone according to the number of residues, and consists of bond and dihedral angles of α-helix and β-sheet, respectively, in its C_α atoms. To generate the library we considered 122 protein domains based on Rost and Sander database (Rost and Sander, 1993) whose three-dimensional structure was accurately determined. These secondary structures, α-helix and β-sheet, were obtained from the RCSB Protein Data Bank. Each of these fragments (sequences) was grouped using MCLUST (Fraley and Raftery, 2002), a software package for model-based clustering, density estimation, and discriminant analysis interfaced to the S-PLUS commercial software, to cluster angles. The possible secondary structures folded by subsequences of α-helix and β-sheet are illustrated in Figures 2(a), 2(b), 2(c) and 2(d), which correspond to β-sheet (with 2 residues), and α-helix (with 8, 10 and 11 residues) of a real protein, respectively.

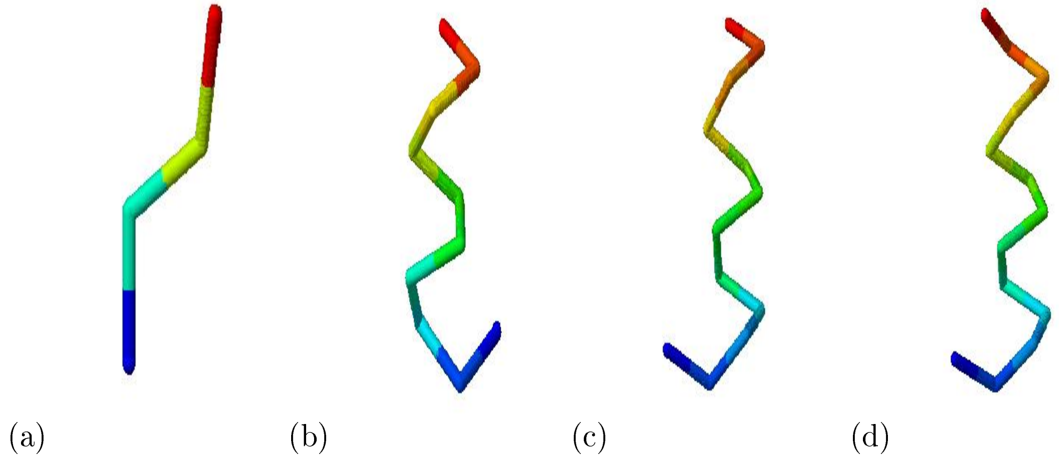


Figure 2: Secondary structures folded by subsequences of α-helix and β-sheet. (a) Extended β-Sheet with 2 residues. (b) α-helix with 8 residues. (c) α-helix with 10 residues. (d) α-helix with 11 residues.

However, the total number of secondary structures that could be folded by the subsequence may be huge as the number of amino acid sequence increases. Thus, an essentially arbitrary distribution should be assigned to these secondary structures with each structure having a nonzero mass value. For example, a secondary structure for β-Sheet with 2 residues like Figure 2(a) consists of bond and dihedral angles. Bond angles were divided into two groups with probabilities 0.34536 and 0.65464, respectively, and dihedral angles were divided into two groups with probabilities 0.54639 and 0.45361, respectively. Each of these groups has a different mean and variance. Once one group with a specified mean and variance is selected according to its probability for each angle, new bond and dihedral angles are generated. These angles make a three-dimensional structure for protein folding. For example, we assigned 2 residues of the structure [shown in Figure 2(a)] with a probability 0.65464 and 0.54639 for bond and dihedral angles, respectively. These groups have different means and variances. This function will then work as a proposal transition function for the move of the block of residues; the resulting simulation will be ergodic.

Under the constrained conformation space, we consider 1a7f (29 residues), 9ins (30 residues), and two new small proteins such as 1ejg (46 residues) and 1crn (46 residues). By sampling on the constrained conformation space, we fold all these proteins rapidly to their lowest energy states. The primary sequences of new proteins are given as follows: 1crn/1ejg (46-LLBBNNBBBL NLBLBBLBNN LNLBBBBLBL NBBBBNNBLB NNLBBL). We set $\tau = 1.0$, $t_0 = 2500$, $n = 10^8$, and $H_0 = -50$. For 1a7f and 9ins, we partitioned the phase space into $E_1, \cdots, E_{201}$ with an equal energy bandwidth of 1.0; that is, we set $E_1 = \{x \in \chi : H(x) \leq -50\}, E_2 = \{x \in \chi : -50 < H(x) \leq -49\}, \cdots, E_{201} = \{x \in \chi : H(x) > 150\}$. For 1crn and 1ejg, we partitioned the phase space into $E_1, \cdots, E_{301}$ with an equal energy bandwidth of 1.0; that is, we set $E_1 = \{x \in \chi : H(x) \leq -50\}, E_2 = \{x \in \chi : -50 < H(x) \leq -49\}, \cdots, E_{301} = \{x \in \chi : H(x) > 250\}$. For a step size, we set $s = 4.0$ for all proteins.

The positions of secondary structure sequences (Table 3) used in this example were assigned with the DSSP program designed by Kabsch and Sander (1983), and these subsequences were chosen randomly with assigned mean and variance according to each type of their secondary structures. The ASAMC algorithm was run 5 times independently for each of the proteins. The reported values are the lowest energy values and the r.m.s.d. data achieved during the most efficient run. The folded predicted structures found by the constrained ASAMC sampler are shown in Figure 3 for 1a7f, 9ins, 1crn, and 1ejg. Figure 3 indicates the folded target structures predicted by our method are very similar to the native structure for all proteins.

The computational results are summarized in Table 3. The average minimum energy found by the ASAMC algorithm is better than the minimum energy by SA and MH algorithms in all runs. The r.m.s.d. data are also in the reasonable range (with r.m.s.d. values ranging from 3 to 7.5 for small proteins) based on information of known structures (Lu *et al.,* 2003).

Table 3: Protein folding simulations with the use of secondary structures

Protein[a]	NE[b]	Metropolis(rmsd)[c]	SA(rmsd)[d]	ASAMC(rmsd)[e]
1A7F (29)	60.7142	34.1537(16.446)	19.7759(13.296)	15.4885(6.934)
9INS (30)	61.9850	37.6464(15.202)	19.1160(16.494)	15.2736(6.762)
1CRN (46)	101.2824	221.4140(23.358)	74.2322(13.674)	32.2536(7.446)
1EJG (46)	103.2615	102.3827(20.146)	90.0267(12.766)	28.3862(7.140)

a: In 1a7f, the subsequence (residues 9-18) is constrained to the secondary structure as α-helix. In 9ins, the subsequence (residues 7-17) is constrained to the secondary structure as α-helix. In 1crn, the subsequence (residues 7-17, 23-30 as α-helix and 33-34 as β-sheet) is constrained to the secondary structure. In 1ejg, the subsequence (residues 7-17, 23-30) is constrained to the secondary structure as α-helix. The positions of all subsequences were assigned with DSSP. b,c,d,e: same with Table 2.

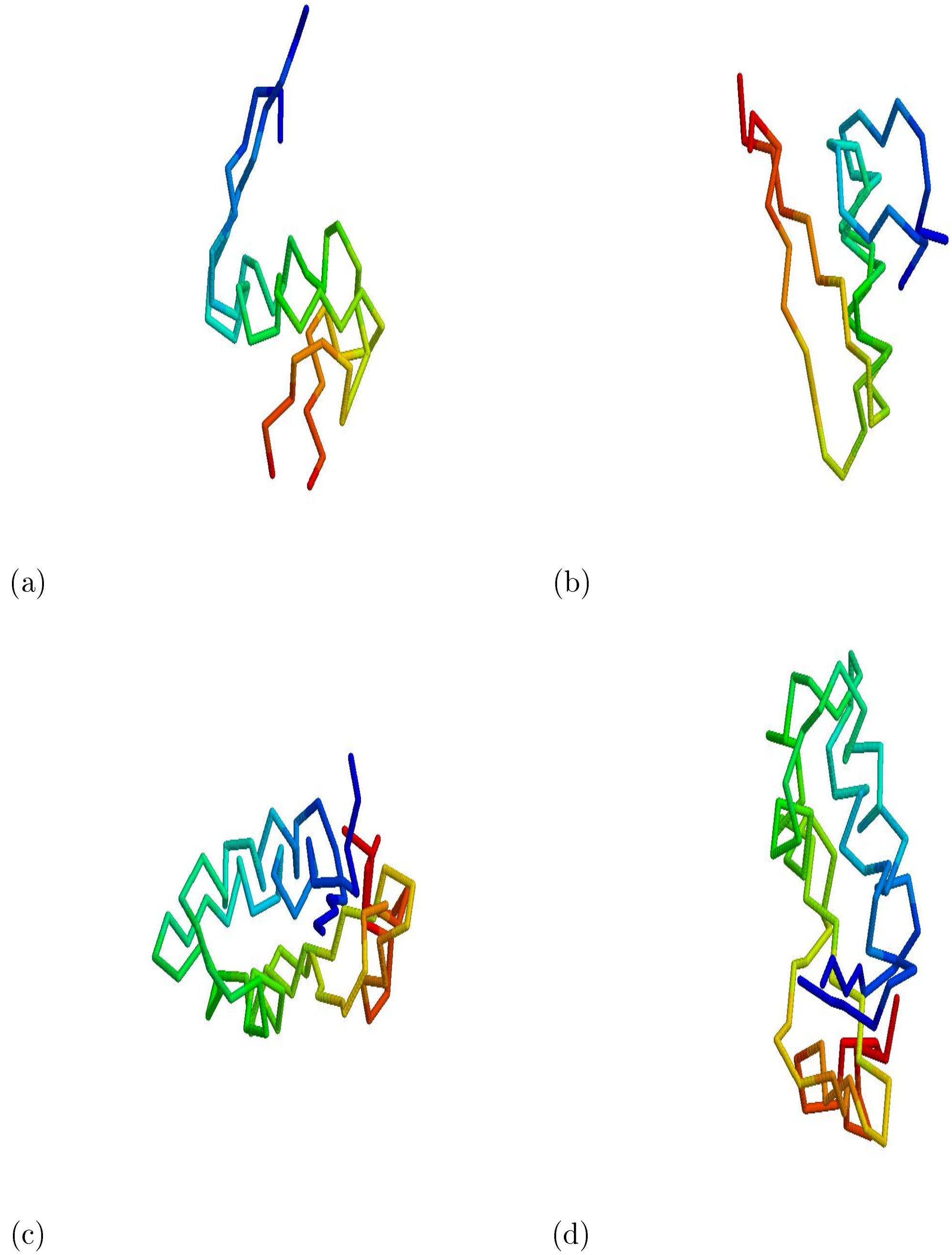


Figure 3: The native structure superimposed on the best structure predicted by ASAMC with the use of secondary structures. (a) 1a7f. (b) 9ins. (c) 1crn. (d) 1ejg.

CHAPTER III

SEQUENTIAL STOCHASTIC APPROXIMATION MONTE CARLO FOR PHYLOGENY RECONSTRUCTION

3.1 Introduction

Phylogenetic trees have been used in biology for a long time to graphically represent evolutionary relationships among species and genes. However, computational complexity makes it difficult to develop statistical approaches for phylogenetic inference. Many studies have been developed which attempts to alleviate this problem (e.g., Bayesian inference and Markov chain Monte Carlo techniques have been introduced). Bayesian inference is a recently designed method for estimating phylogenetic trees (Rannala and Yang, 1996; Yang and Rannala, 1997; Mau and Newton, 1997; Mau *et al.*, 1999; Larget and Simon, 1999; Li *et al.*, 2000). These papers showed that Markov chains based on the conventional Metropolis Monte Carlo algorithm were computationally more efficient than other phylogeny estimation methods, such as the maximum likelihood (Felsenstein, 1981; Kishino *et al.,* 1990), maximum parsimony (Fitch, 1971; Lake 1987), and neighbor joining (Saitou and Nei, 1987) methods. These methods have many weaknesses; i.e., they do not produce valid inferences beyond point estimates, measures of uncertainty rely exclusively on computer-intensive and approximate bootstrap analysis (Felsenstein, 1985; Newton, 1996), and mathematical and computational complexity are limited to extremely small problems (Evans and Speed, 1993; Sinsheimer *et al.,* 1996).

However, a Bayesian inference method that incorporates MCMC has several advantages over other methods of phylogeny inference. First, it takes care of the uncertainty of trees automatically (Rannala and Yang, 1996; Mau and Newton, 1997;

Mau *et al.*, 1999; Larget and Simon, 1999). Second, it can be used to infer for the models of sequence mutation (Rannala and Yang, 1996). Third, it makes analysis of large data sets more tractable (Mau *et al.*, 1999; Li *et al.*, 2000). Samples from the posterior distribution of the trees can be used to construct a consensus tree. This is much faster than bootstrap resampling (Larget and Simon, 1999), an alternative method for a consensus tree construction.

A Bayesian analysis of phylogenetic trees requires the evaluation of high-dimensional summations and integrals. The computation suffers from two difficulties. First, it suffers from the curse of dimensionality. As the number of taxa increases, the number of all possible trees will increase in a speed $((2n-3)!/[2^{n-2}(n-2)!]$ for rooted trees and $((2n-5)!/[2^{n-3}(n-3)!]$ for unrooted trees. This speed is much faster than exponential time. Thus, the search time for the optimal tree will increase drastically as the number of taxa increases. Second, it suffers from the local trap problem. The posterior probability distribution of trees can contain multiple energy minima. This phenomenon has been observed in applications of the maximum parsimony method (Maddison, 1991) and the maximum likelihood method (Felsenstein, 1981, 1983, 1993; Salter and Pearl, 2001). A number of authors have noticed the local trap difficulty and have applied some advanced MCMC algorithms to the problem. For example, Huelsenbeck and Ronquist (2001) and Altekar *et al.* (2004) applied parallel tempering and its parallel implementation to this problem, respectively. However, no authors have addressed the difficulty of high dimensionality.

With the development of science and technology, we frequently must deal with high-dimensional systems, particularly in biology. The traditional MCMC algorithms often suffer from a severe difficulty in convergence. One reason is multimodality. Many techniques to alleviate this problem have been proposed such as simulated tempering (Marinari and Parisi, 1992), parallel tempering (Geyer, 1991; Hukushima and Nemoto, 1996), and evolutionary Monte Carlo (Liang and Wong, 2001). However, the slow convergence is not due to the multimodality, but the curse of dimensionality (i.e., the number of samples increase exponentially with dimension to maintain a given level of accuracy).

In this chapter, we will apply SAMC to the problem and propose to make use of the sequential structure of phylogenetic trees in conjunction with SAMC (i.e, SSAMC) to overcome the curse of dimensionality. SSAMC works by simulating a sequence of systems of different dimensions. The idea is to use the information provided by the simulation from low-dimensional systems and thus alleviate high-dimensionality problems significantly. We demonstrate the phylogeny reconstruction capability of our algorithm by estimating the original tree topology using data that we generated with fixed branch lengths, substitution model, and tree topology from root data. Our method is also applied to analyze nine bacteriophage T7 and DNA sequences for 32 species of African cichlid fishes. We can expect that SSAMC will be able to escape from local traps and achieve a much faster convergence to the global likelihood maxima than other MCMC simulation-based PTR methods such as BAMBE and MrBayes.

The remaining part of this chapter is organized as follows. In Section 3.2, we describe a probability distribution on phylogenetic trees. In Section 3.3, we describe the SSAMC algorithm. Section 3.4 presents numerical results in several examples.

3.2 A Probability Distribution on Phylogenetic Trees

In this chapter, we use the same tree terminology with one described in Mau and Newton (1997). We give a brief description of the tree terminology they used.

3.2.1 A Tree Representation

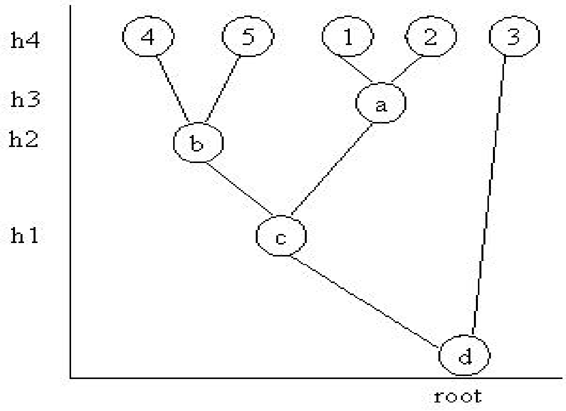


Figure 4: Graphical depiction of a sample tree with five taxa.

Let consider a tree like Figure 4. All trees will be assumed to be rooted binary, meaning that an edge splits into two children edges (i.e., three edges meet at every branch node, a *node* being an endpoint of an edge). A phylogenetic tree represents the relationship of a set of species or genetic sequences. This phylogeny tree can be viewed abstractly as a rooted binary weighted tree. Mathematically, a tree is a connected graph with node sets (terminal node and internal node) and edge set. Each edge of the tree has a certain amount of evolutionary divergence associated to it, defined by some measure of distance between sequences, or from a model of substitution of residues over the course of evolution. This is called “length”. The time separating a child from its parent is its edge weight, called its branch length ($\{h1, h2, h3, h4\}$ in Figure 4). We call nodes terminal nodes (or leaves) if they are connected through a single edge and internal nodes otherwise. A true biological phylogeny has a “root”, or ultimate ancestor of all the sequences. Some algorithms provide information, or at least a conjecture, about the location of the root. Others, including parsimony,

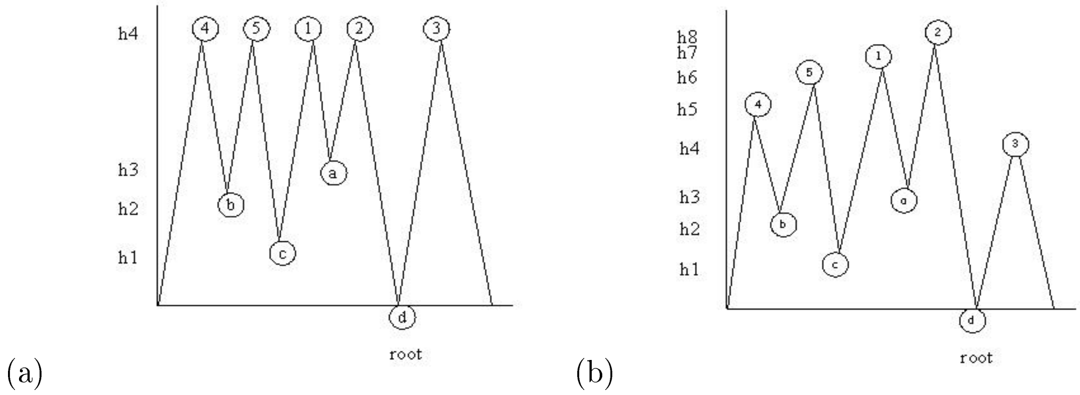


Figure 5: (a) A tree with the molecular assumption. (b) A tree without the molecular assumption.

are completely uninformative about its position and other criteria have to be used for rooting the tree. The placement of the root relative to the leaves determines the direction of time and hence ancestry. The labeled shape of the tree is called the tree topology. This is determined by which pairs of nodes coalesce. The topology can be summarized by using parentheses to indicate coalescence. For example, the tree topology in Figure 4 is (((4,5),(1,2)),3).

There are many different applications of trees. As a result, there are many different algorithms for manipulating them. However, many of the different tree algorithms have in common the characteristic that they systematically visit all the nodes in the tree (i.e., the algorithm walks through the tree data structure and performs some computation at each node in the tree). This process of walking through the tree is called a tree traversal (Durbin *et al.,* 1998). There are essentially two different methods in which to visit systematically all the nodes of a tree: depth-first traversal as a recursive traversal and breadth-first traversal as a non-recursive traversal. Certain depth-first traversal methods occur frequently enough that they are given names of their own: pre-order traversal, in-order traversal and post-order traversal (Drozdek and Simon, 1995).

A binary tree with n leaves has $n-1$ internal nodes including the root. Each taxon (leaf) and internal node appear at a peak and a valley in graph, respectively. The permutation of taxa is read across tops of the peaks. Branch lengths and tree topology are determined by $n-1$ (Figure 5(a)) or $2(n-1)$ (Figure 5(b)) valley depths. There is a left/right choice to make a subtree at each internal node. Based on these choices there is a unique post-order traversal of the tree. In this chapter, we use a post-order traversal. The permutation of leaf labels and the ordered valley depths determine the tree completely.

3.2.2 A Bayesian Approach

A Bayesian analysis requires a likelihood model for sequence evolution through a phylogenetic tree, prior distribution on trees and model parameters and data. Let a phylogenetic tree $\psi = (\tau, \beta)$ be described by its tree topology τ and associated branch lengths β. Let ϕ be a parameter vector describing rates of change among states in the Markov process for a given branch. Data on n taxa can be arranged as a $n \times N$ matrix, where N is the common number of sites, or positions, providing information for each taxon. Elements of this matrix are discrete characters from a finite set D of size d. This data are viewed as a realization of a stochastic process that has evolved along the branches of an unknown phylogeny ψ. Modeling is reduced to a single site by assuming that evolution among sites is independent. Thus, $\omega = (\psi, \phi) = (\tau, \beta, \phi)$ represents a specific choice of tree topology, branch lengths, and model parameters and we calculate the likelihood model $L(\omega|x) = L(\tau, \beta, \phi|x)$ for observed data x.

The posterior distribution of a particular tree topology τ is given by

$$p(\tau|x) = \frac{\int_B \int_\Phi L(\tau, \beta, \phi|x) p(\tau, \beta, \phi) d\phi d\beta}{p(x)}$$

where B and Φ are the sets of all possible branch lengths and model parameters, respectively. $p(\tau, \beta, \phi)$ is a prior joint distribution of different parameter values $\{\tau, \beta, \phi\}$. $L(\tau, \beta, \phi|x)$ is the likelihood function and describes the probability of different parameter values $\{\tau, \beta, \phi\}$ given data. $p(x) = \sum_{\tau} \int_{B} \int_{\Phi} L(\tau, \beta, \phi|x)p(\tau, \beta, \phi)d\phi d\beta$ is the total probability of the data over the parameter space $\Omega = (\Psi, \Phi)$ where Ψ is the sets of all possible trees. The Bayesian approach is based on this $p(\tau|x)$ called posterior distribution.

3.2.3 Nucleotide Substitution Models

All evolutionary models deal with the random substitution of one nucleotide for another at individual sites, and share the following set of underlying assumptions: Markov property, homogeneity and stationarity (Salemi and Vandamme, 2003). There are several evolutionary models: one parameter model (Jukes and Cantor, 1969), in which nucleotide substitutions have equal probabilities; an extension 2-parameter model by Kimura (1980), who allowed the different rate of transitional and transversional events; Felsenstein model (1981), which added three parameters to the Jukes-Cantor model by allowing the stationary probabilities to be different; HKY85 model (Hasegawa, Kishino, and Yano, 1985), a general stationary distribution of the nucleotides and different rates for transition and transversion events.

This chapter considers the most general model TN93 (Tamura and Nei, 1993). This model has both HKY85 and F84 (Felsenstein's PHYLIP since 1984) as special cases. Instantaneous rate matrix R is parameterized as follows:

$$\theta \left(\begin{array}{lllll} & A & G & C & T \\ A & -(\kappa\pi_G + \pi_C + \pi_T) & \kappa\pi_G & \pi_C & \pi_T \\ G & \kappa\pi_A & -(\kappa\pi_A + \pi_C + \pi_T) & \pi_C & \pi_T \\ C & \pi_A & \pi_G & -(\pi_A + \pi_G + \kappa\gamma\pi_T) & \kappa\gamma\pi_T \\ T & \pi_A & \pi_G & \kappa\gamma\pi_C & -(\pi_A + \pi_G + \kappa\gamma\pi_C) \end{array} \right)$$

where $\kappa = \dfrac{4\phi}{\gamma + 1}$, and we implicitly assume the order A, G, C, T for bases. There are seven parameters, six of which are free. The model is reversible with stationary distribution given by $\pi_A, \pi_G, \pi_C, \pi_T$ ($\sum_{i \in \{A,G,C,T\}} \pi_i = 1$). The parameter θ controls the overall mutation rate. The transition/transversion ratio is κ, and γ is the final parameter which affects the ratio of transition/transversion rates among purines and pyrimidines (Simon and Larget, 2001).

Our SSAMC approach uses transition probabilities calculated from the HKY85 model in TN93 ($\phi = \kappa/2,\ \gamma = 1$). The elements of the transition probability matrix are given by

$$Q_{ij}(t) \left(\begin{array}{ll} \pi_j + \pi_j \left(\dfrac{1}{\lambda_j} - 1 \right) e^{-\theta t} + \left(\dfrac{\lambda_j - \pi_j}{\lambda_j} \right) e^{-\theta\gamma_j t} & i = j \\ \pi_j + \pi_j \left(\dfrac{1}{\lambda_j} - 1 \right) e^{-\theta t} - \left(\dfrac{\pi_j}{\lambda_j} \right) e^{-\theta\gamma_j t} & i \neq j \ (\mathit{transitional\ event}) \\ \pi_j (1 - e^{-\theta t}) & i \neq j \ (\mathit{transversional\ event}) \end{array} \right)$$

where $\lambda_j = \pi_A + \pi_G$ if base j is a purine (A or G) and $\lambda_j = \pi_C + \pi_T$ if base j is a pyrimidine (C or T), and $\gamma_j = 1 + (\kappa - 1)\lambda_j$ (Hasegawa *et al.*, 1985; Li *et al.*, 2000).

3.2.4 A Likelihood Model

Let ρ be a root node and y_ρ be the ancestral root state. Given y_ρ and branch lengths, Markov process on each node emanates independently from the root ρ along the corresponding branches of Ψ. Let y_ν be a value when a given process reaches an internal node ν. This evolution stops when it reaches observed taxa (e.g., leaves). Conditionally on the phylogeny Ψ, the initial distribution π_0 on $D = \{A, G, C, T\}$, and transition probabilities $p(y_\nu | y_{\sigma(\nu)}, \beta_\nu, \phi)$ where $\sigma(\nu)$ is the parent node of ν, β_ν is the branch length, and ϕ is a parameter vector of substitution model, the probability of the particular realization y is given by

$$p(Y = y | \tau, \beta, \phi) \; = \; \prod_{k=1}^{N} \pi_0(y_{\rho_k}) \prod_{\nu \in \{all \;\; nodes \;\; except \;\; \rho_k\}} p(y_{\nu_k} | y_{\sigma(\nu_k)}, \beta_\nu, \phi).$$

To calculate the likelihood function from leaf data at multiple sites, we must marginalize this likelihood function over all values for all sites. Unfortunately, this straightforward computation is not feasible, but the amount of computation can be reduced considerably by the pruning method because it takes care of the Markov property of the substitution model (Felsenstein, 1983). The pruning method has a recursive relationship in a tree, starting from the leaves and working recursively to the root for each site as follows: for each leaf ν and state s, $L_\nu(s) = I(y_\nu = s)$ where $I(\cdot)$ is the indicator function and for an internal node ν and $\sigma(u) = \sigma(w) = \nu$, $L_\nu(s) = \big(\sum_{x \in D} L_u(x) p(x|s, \beta_u, \phi)\big) \times \big(\sum_{x \in D} L_w(x) p(x|s, \beta_w, \phi)\big)$. The likelihood function is given by

$$L(\tau, \beta, \phi) = \prod_{k=1}^{N} \sum_{s \in D} \pi_0(s) L_\rho^k(s).$$

3.3 The Sequential Stochastic Approximation Monte Carlo Algorithm

To overcome the curse of dimensionality, we propose the following sequential version of SAMC, the so-called sequential SAMC (SSAMC) algorithm, to reconstruct phylogenetic trees. The SSAMC algorithm consists of two steps: buildup ladder construction and SSAMC simulation.

3.3.1 Buildup Ladder Construction

A builder ladder (Wong and Liang, 1997; Liang, 2003) comprises a sequence of systems of different dimensions. Typically, we have

$$dim(\mathcal{X}_1) < dim(\mathcal{X}_2) < \cdots < dim(\mathcal{X}_m)$$

where $\mathcal{X}_i$ denotes the sample space of the i^{th} system, with an associated density/mass function $\pi_i(z_i)/Z_i$ and partition function Z_i. The principle of the buildup ladder construction is to approximate the original system by a system with a reduced dimension; the reduced system is again approximated by a system with a further reduced dimension until one reaches a system of a manageable dimension, that is, the corresponding system is able to be sampled easily by a local updating algorithm, such as the MH algorithm or the Gibbs sampler. The solution of the reduced system is then extrapolated level by level until the target system is reached.

For the phylogeny example, the buildup ladder can be constructed as follows. Intuitively, we want to first approximate the shape of the phylogenetic tree using a small number of taxa, and then add other taxa to the tree locally. To achieve this goal, the taxa can be ordered in the following manner. Let A and A^c denote the sets of ordered and not yet ordered taxa, respectively. We start with an arbitrary taxa, the next taxa added to A should be one with the maximum distance with the starting taxa, the third taxa should be the one with the maximum distance with the set A, and so on. Here we define the distance of a taxa to A as the minimum distance of the taxa in A, i.e., $min_{j \in A} d_{ij}$ for $i \in A^c$, where d_{ij} denotes the distance of the taxa i and j. The ordering procedure can be summarized as follows:

1. Calculate the pairwise distance matrix (d_{ij}) of the taxa. For example, the distance can be simply the number of differences between two sequences, or

the alignment score calculated according to the substitution scoring matrices [e.g., PAM matices (Dayhoff *et al.*, 1978) or BLOSUM matrices (Henikoff and Henikoff, 1992)]. In this chapter, we use the alignment score for the pairwise distance.

2. Order the taxa sequentially. The next taxa added to A is the taxa $k(\in A^c)$ which satisfies the condition: There exist a taxa $m \in A$ such that $d_{km} = max_{i \in A^c} min_{j \in A} d_{ij}$. If there are several taxa, all satisfying the above condition, choose one randomly.

3.3.2 Sequential SAMC Simulation

Suppose a build-up order has been constructed for a set of taxa. Let $D_1, \cdots, D_m$ denote m subsets of taxa, $D_1 \subset D_2 \subset \cdots \subset D_m$, where D_i contains the first $|D_i|$ taxa in the build-up order and D_m contains all taxa of the dataset. We then work on the following distribution:

$$f(\tau) \propto \sum_{k=1}^{m} \frac{1}{Z_k} f(\tau_k|D_k)$$

where τ_k is the phylogenetic tree constructed for the taxa in the set D_k, $f(\tau_k|D_k)/Z_k$ is the posterior distribution of the tree, and Z_k is the unknown normalizing constant. The sample space of $f(\tau)$ can be written as $\bigcup_{k=k_0}^{n} \mathcal{X}_k$, where $\mathcal{X}_k$ denotes the sample space of $f(\tau_k|D_k)$. We can then employ the following procedure to estimate the unknown normalizing constants and explore the target sample space $\mathcal{X}_m$.

3.3.2.1 Proposal Distribution

Let $Q(i \to j)$ denote the proposal probability for a transition from the level i to the level j of the buildup ladder. For example, Q can be specified as a tridiagonal matrix with elements $Q_{1,1} = Q_{n,n} = 2/3, Q_{1,2} = Q_{n,n-1} = 1/3$, and $Q_{k,k-1} = Q_{k,k} = Q_{k,k+1} =$

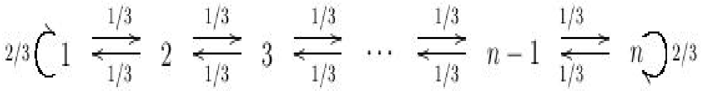
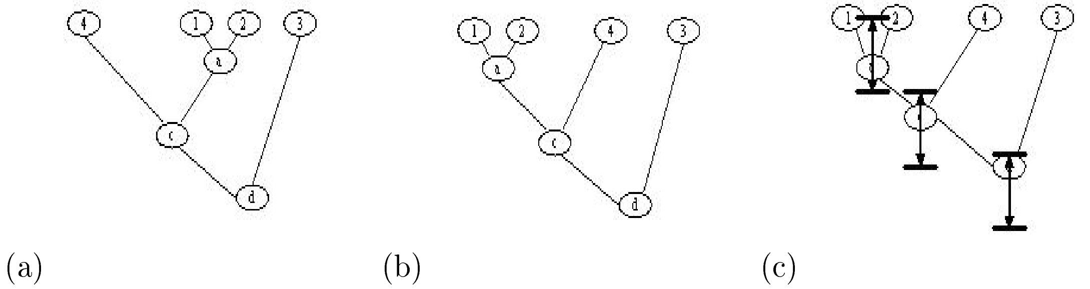


Figure 6: An example of tridiagonal matrix.

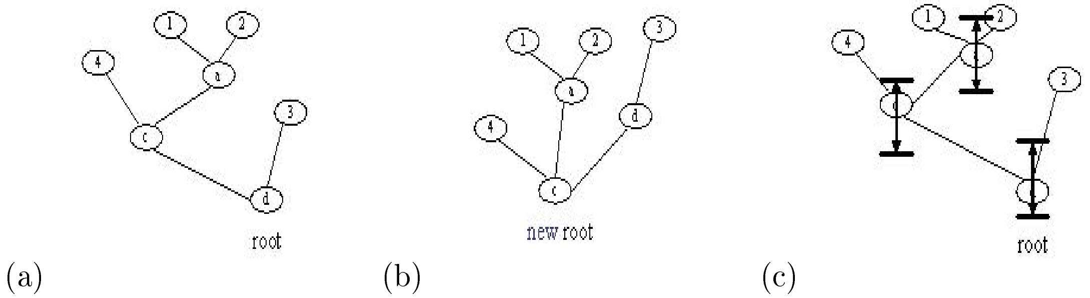
1/3 for $k = 2, \cdots, n-1$ (Figure 6). Let $T(\tau_i \to \tau_j)$ denote the proposal distribution of generating τ_j conditional on τ_i. If $i = j$ (i.e., updating the tree conditional on the same set of taxa), in this case, T can be specified as in Larget and Simon (1999). There are two main tree proposal algorithms, each with clock and non-clock as follows: global with a molecular clock, global without the molecular clock, local with a molecular clock, and local without the molecular clock. Here we give a brief description for this proposal distribution [see Larget and Simon (1999)]. In the global proposal with a molecular clock, a distance from the root in a depth first traversal of the tree gives a graph with peaks and valleys for any given collection of left/right orientations for subtrees at each internal node (Figure 7). The tree is parameterized by a permutation of the taxa as read from left to right in the representation and the valley depths from left to right. The global proposal without the molecular clock is the same as the global proposal with a molecular clock except that all peaks may be different distances from the root in the tree representation and thus each valley has two depths, the depths to its left and right peaks (Figure 8). The local proposal distribution with a molecular clock modifies the tree only in a small neighborhood of a randomly chosen internal branch, leaving the remainder of the tree unchanged. The local proposal distribution without the molecular clock acts on the unrooted tree.

If $i < j$, the transition is to extrapolate $\tau_i (\in \mathcal{X}_i)$ to $\tau_j (\in \mathcal{X}_j)$; otherwise, the transition is to project $\tau_i (\in \mathcal{X}_i)$ to $\tau_j (\in \mathcal{X}_j)$. The extrapolation and projection operators



(a) (b) (c)

Figure 7: An example of global with a molecular clock. (a) A current tree. (b) A tree selected randomly. (c) A tree perturbed each valley depth independently.



(a) (b) (c)

Figure 8: An example of global without the molecular clock. (a) A current tree. (b) A re-rooted tree. (c) A tree perturbed each valley depth independently.

should be chosen such that the pairwise move $\tau_i \leftrightarrow \tau_j$ is reversible (illustrated by Figures 9 and 10).

1. *Projection Operator*

   The projection operator is very simple; we can just remove the leaves belonging to the set $D_i \setminus D_j$ and the corresponding internal nodes. Thus, we have $T(\tau_i \to \tau_j) = 1$ for the projection operator.

2. *Extrapolation Operator*

   The extrapolation operator can be described by the following procedure. Let D^* denote the set including all taxa and nodes in D_i.

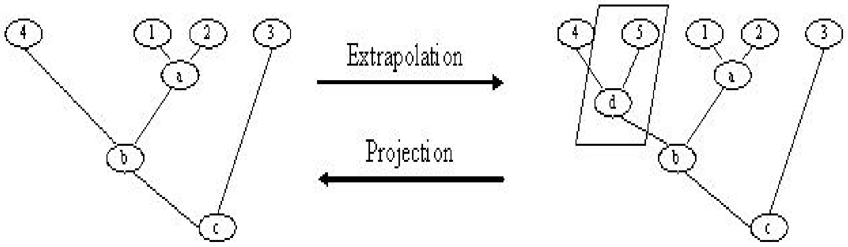


Figure 9: Illustrative graphs for extrapolation and projection operators when a new taxon ‘5’ is sampled from leaves. Extrapolation (left → right): add taxon ‘5’ and internal node ’d’ to the current tree where node ‘d’ is selected uniformly under branch length L_{4b}. Projection (right → left): delete taxon ‘5’ with corresponding internal node ‘d’ from the current tree.

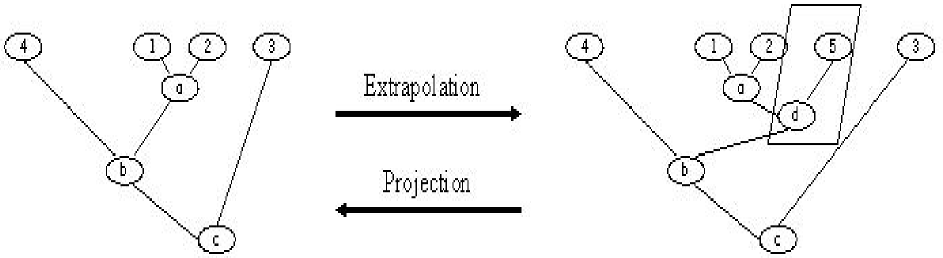


Figure 10: Illustrative graphs for extrapolation and projection operators when a new taxon ‘5’ is sampled from nodes. Extrapolation (left → right): add taxon ‘5’ and internal node ’d’ to the current tree where node ‘d’ is selected uniformly under branch length L_{ab}. Projection (right → left): delete taxon ‘5’ with corresponding internal node ‘d’ from the current tree.

- For each taxon $k \in D_j \setminus D_i$, do the following steps:

  A) Sample a leaf or node l with probability $p_{kl} = e^{-d_{kl}/t_s} / \sum_{l'=1}^{|D^*|} e^{-d_{kl'}/t_s}$, $l' \in D^*$, where t_s is called the insertion temperature. A large t_s corresponds to a random insertion, whereas a small t_s corresponds to the nearest neighbor insertion.

  B) Add the taxon k to the tree as a sister leaf (or node) of l and set $D^* \leftarrow D^* + \{k, parent\ node\ of\ k\ and\ l\}$. The position of the parent node of k and l is chosen uniformly on the branch between l and its current parent node.

- Calculate the extrapolation probability $T(\tau_i \rightarrow \tau_j) = \prod_{k \in D_j \setminus D_i} \frac{p_{kl}}{L_{lk}}$, where L_{lk} denotes the length of the branch between l and its parent node before adding taxa k to the tree.

3.3.2.2 SSAMC Algorithm

Let $k^{(t)}$ and $\tau^{(t)}$ denote the ladder level and the tree sampled at iteration t, respectively. Let $e^{\theta_{t,k}}$ denote the working estimate of Z_k, and $\theta_t = (\theta_{t,k_0}, \cdots, \theta_{t,k_n})$. One iteration of SSAMC consists of the following steps:

1. Generate level k^* according to the proposal matrix Q.

2. If $k^* = k^{(t)}$, simulate a sample τ^* from $f(\tau^{(t)}|D_{k^{(t)}})$ by a MCMC iteration and set $(k^{(t+1)}, \tau^{(t+1)}) = (k^*, \tau^*)$.

3. If $k^* \neq k^{(t)}$, generate a sample τ^* according to the proposal distribution T and Q, and accept the sample (k^*, τ^*) with probability

$$min\left\{1, \frac{e^{\theta_{t,k^{(t)}}}}{e^{\theta_{t,k^*}}} \frac{f(\tau^*|D_{k^*})}{f(\tau^{(t)}|D_{k^{(t)}})} \frac{Q(k^* \rightarrow k^{(t)})}{Q(k^{(t)} \rightarrow k^*)} \frac{T(\tau^* \rightarrow \tau^{(t)})}{T(\tau^{(t)} \rightarrow \tau^*)}\right\}$$

If it is accepted, set $(k^{(t+1)}, \tau^{(t+1)}) = (k^*, \tau^*)$; otherwise, $(k^{(t+1)}, \tau^{(t+1)}) = (k^{(t)}, \tau^{(t)})$.

4. Set $\theta^* = \theta_t + \gamma_t(e_{t+1} - \pi)$, where $e_{t+1} = (e_{t+1,1}, \cdots, e_{t+1,m})$, and $e_{t+1,i} = 1$ if $k^{(t+1)} = k_i$ and 0 otherwise. If $\theta^* \in \Theta$, set $\theta_{t+1} = \theta^*$; otherwise, set $\theta_{t+1} = \theta^* + c^*$, where c^* is chosen such that $\theta^* + c^* \in \Theta$.

The MCMC algorithm employed in step 2 of the above algorithm can be the MH algorithm, the Gibbs sampler or any other advanced MCMC algorithms, such as simulated tempering, parallel tempering, evolutionary Monte Carlo and SAMC-importance-resampling. In this paper, we used the MH algorithm.

We considered several issues for an effective implementation of SSAMC.

- Partition of the sample space. The sample space can be partitioned according to the index of a set of taxa because our aim is to eliminate the curse of dimensionality.

- Choice of N, τ and t_0. Here N is the total number of iterations of a run, and τ and t_0 determine the gain factor $\{\gamma_t\}$. In SSAMC, the desired sampling distribution has been set to be uniform over all subregions, so the convergence of the run can be diagnosed by examining the equality of the realized sampling frequencies of these subregions (see section 2.3).

3.4 Examples

In the first subsection, we demonstrated phylogeny reconstruction capability of SSAMC by showing the regeneration of tree topology based on data (e.g., taxa) which generated using fixed branch lengths and tree topology. We also showed that SSAMC is superior to other PTR methods such as BAMBE and MrBayes in finding the global likelihood maxima. In the next two subsections, we applied SSAMC to analyze nine bacteriophage T7 and DNA sequences for 32 species of cichlid fishes and present the best tree and the convergence property. As a gain factor in this paper, we set

$$\gamma_t = \left[\frac{t_0}{max(t_0, t)}\right]^{\tau}, \quad t = 0, 1, 2, \cdots,$$

for some specified value of $t_0 > 1$ and $\tau \in \left(\frac{1}{2}, 1\right]$.

3.4.1 Phylogenetic Tree Reconstruction

3.4.1.1 Tree Topology of 10 Taxa and 200 Sites with Molecular Clock

Table 4: The root data for phylogenetic tree reconstruction

ATGAACCCTT	ACATCCTAAT	AACCCTTCTT	TTCGGACTAG	GTCTAGGAAC
TACAATTACA	TTTGCAAGCT	CCCACTGACT	CCTTGCTTGA	ATAGGCCTTG
AACTAAACAC	CCTCGCTATT	ATCCCACTGA	TAGCCCAACT	CCACCACCCC
CGGGCAGTCG	AAGCTACCAC	AAAATACTTC	CTCACCCAAG	CTGCTGCCGC

The purpose of phylogenetic studies is to reconstruct the correct evolutionary relationship between organisms. In our attempt to reach this goal, we first set root data with 200 sites like Table 4. We fixed all equal branch lengths (molecular clock) which is the time of divergence between organisms and used the five-parameter HKY85 model for nucleotide substitution model. Under this HKY85 model, we set $\kappa = 2.0,\ \theta = 1.0$ and $(\pi_A, \pi_G, \pi_C, \pi_T) = \left(\frac{1}{4}, \frac{1}{4}, \frac{1}{4}, \frac{1}{4}\right)$. Based on this information, we generated 10 taxa and made a tree topology like Figure 11(a). The sample space was restricted to the taxa region $\mathcal{X} = \{1, 2, 3, 4, 5, 6, 7, 8, 9, 10\}$, and was then partitioned into 8 subregions, $E_1 = \{x \in \mathcal{X} :\ x \leq 3\}, E_2 = \{x \in \mathcal{X} :\ x = 4\}, \cdots, E_8 = \{x \in \mathcal{X} :\ x = 10\}$. The parameters were set as follows: $\tau = 0.6,\ t_0 = 1000$. SSAMC was run 10 times independently, and each run consists of $n = 10^7$ iterations. Figure 11(b) shows the estimated best tree with maximum log likelihood value -2433.5838. Table 5 shows the comparison of branch lengths between true tree and estimated best tree. It indicates that both trees have overall similar branch lengths. Thus, Figure 11 indicates that two trees are very similar each other. In addition, by Table 6 since the sampling frequency for each subregion is not less than 80% of the average sampling frequency, a SSAMC run is regarded as converged.

Table 5: Comparison of branch lengths between true tree and estimated best tree with 10 taxa and 200 sites

node	true	ssamc	node	true	ssamc	node	true	ssamc	node	true	ssamc
0	1.00	0.73	1	1.00	0.73	2	2.00	1.88	3	2.00	1.88
4	5.00	4.96	5	8.00	6.53	6	6.00	6.86	7	4.00	3.62
8	3.00	3.13	9	3.00	3.13	10	2.00	1.74	11	2.00	2.49
12	1.00	0.59	13	3.00	1.57	14	2.00	2.62	15	0.00	0.00
16	4.00	2.29	17	2.00	3.24	18	1.00	0.49			

Table 6: The relative sampling frequency of each subset in estimated best tree with 10 taxa and 200 sites

subset	frequency	subset	frequency	subset	frequency	subset	frequency
1	106.8265	2	105.3385	3	103.2719	4	101.3029
5	99.1611	6	96.6812	7	94.8926	8	92.5253

3.4.1.2 Tree Topology of 20 Taxa and 200 Sites with Molecular Non-clock

Although section (3.4.1.1) gives a good result, there is no significant difference between other softwares for phylogenetic tree reconstruction. Thus, we considered more complicated tree topology with a molecular non-clock (i.e, different branch lengths). Based on the same information with the first example with the exception of branch lengths, we generated 20 taxa and made a tree topology like Figure 12(a). The sample space was restricted to the taxa region $\mathcal{X} = \{1, 2, \cdots, 20\}$, and was then partitioned into 12 subregions, $E_1 = \{x \in \mathcal{X} : \; x \leq 4\}, E_2 = \{x \in \mathcal{X} : \; x = 5\}, \cdots, E_7 = \{x \in \mathcal{X} : \; x = 10\}, E_8 = \{x \in \mathcal{X} : \; x = \{11, 12\}\}, \cdots, E_{12} = \{x \in \mathcal{X} : \; x = \{19, 20\}\}$.

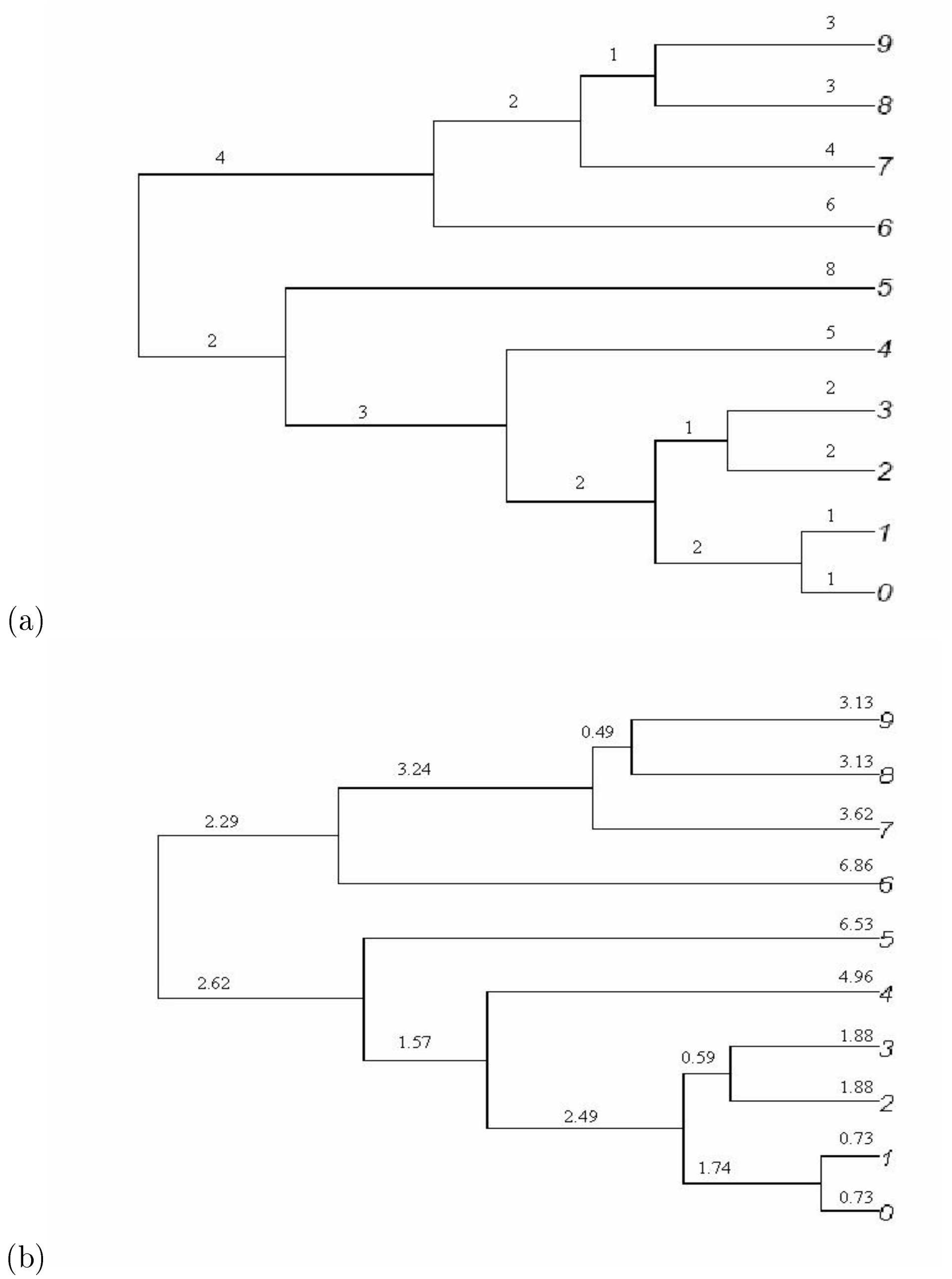


Figure 11: Comparison between true tree topology and best tree estimated by SSAMC with 10 taxa and 200 sites. (a) A true tree (log likelihood = -2438.3435). (b) A best tree estimated by SSAMC (log likelihood = -2433.5838).

Table 7: Comparison of branch lengths between true tree and estimated best tree with 20 taxa and 200 sites

node	true	ssamc	node	true	ssamc	node	true	ssamc	node	true	ssamc
0	0.10	0.11	1	0.10	0.07	2	0.20	0.04	3	0.20	0.22
4	0.20	0.27	5	0.30	0.37	6	0.10	0.06	7	0.10	0.17
8	0.40	0.26	9	0.10	0.04	10	0.10	0.07	11	0.10	0.03
12	0.20	0.21	13	0.20	0.19	14	0.10	0.08	15	0.20	0.19
16	0.20	0.14	17	0.20	0.22	18	0.20	0.21	19	0.20	0.09
20	0.30	0.23	21	0.20	0.20	22	0.20	0.13	23	0.20	0.52
24	0.20	0.03	25	0.60	2.00	26	0.20	0.25	27	0.30	0.75
28	0.30	0.28	29	0.20	0.08	30	0.40	0.27	31	0.00	0.00
32	0.10	0.08	33	0.30	0.48	34	1.60	0.13	35	0.20	1.41
36	0.20	0.75	37	0.30	0.39	38	1.40	0.56			

Table 8: The relative sampling frequency of each subset in estimated best tree with 20 taxa and 200 sites

subset	frequency	subset	frequency	subset	frequency	subset	frequency
1	100.8321	2	100.6020	3	100.4161	4	100.2512
5	99.8520	6	99.4255	7	98.9422	8	98.7080
9	100.7606	10	100.1461	11	100.4045	12	99.6597

The parameters were set as follows: $\tau = 1.0$, $t_0 = 50000$. SSAMC was run 5 times independently and each run consists of $n = 5 \times 10^6$ iterations. Figure 12(b) shows the estimated best tree with a maximum log likelihood value of -4196.0907. Table 7 shows the comparison of branch lengths between true tree and estimated best tree. It indicates that both trees have overall similar branch lengths. Therefore, Figure 12 indicates that two trees has similar structure to each other. In addition, because the sampling frequency for each subregion is not less than 80% of the average sampling frequency, a SSAMC run is regarded as converged (Table 8).

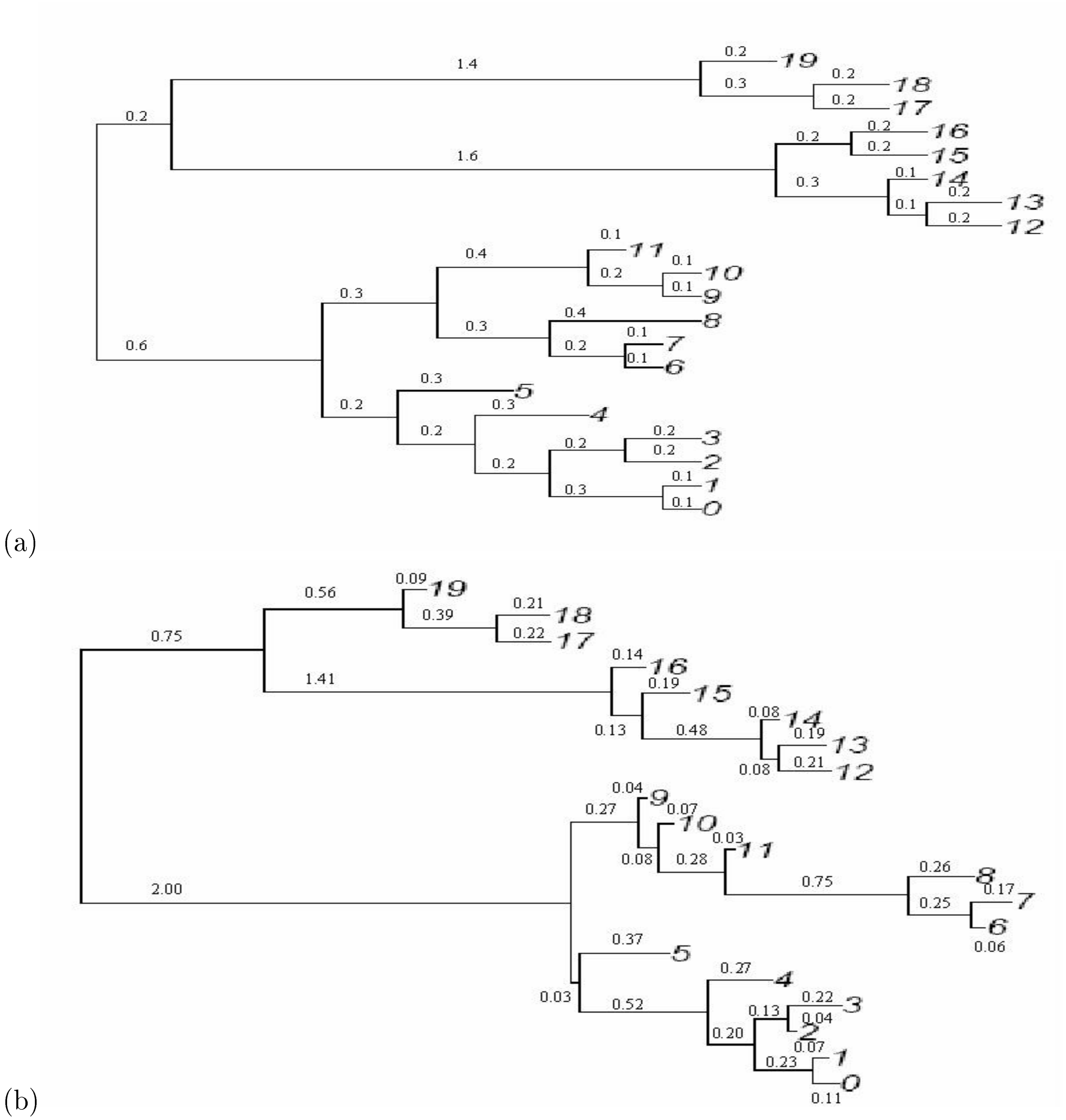


Figure 12: Comparison between true tree topology and best tree estimated by SSAMC with 20 taxa and 200 sites. (a) A true tree with 20 taxa and 200 sites (log likelihood = -4209.442723). (b) A best tree with 20 taxa and 200 sites generated by SSAMC (log likelihood = -4196.090699).

3.4.1.3 Comparison with Other PTR Methods for Phylogenetic Tree Reconstruction

In this chapter, we are interested in assessing high dimensionality and local trap problems. A number of authors considered only the local trap difficulty. The alternative method most similar as a method to alleviate local trap problems is MrBayes. We also considered BAMBE, a popular software for a consensus tree reconstruction, to compare with our method. Based on the same information in section (3.4.1.2), we ran 5 times BAMBE and MrBayes independently with 2×10^6 iterations. Figure 13 shows the comparison between SSAMC and BAMBE, and SSAMC and MrBayes.

In the early iterations, BAMBE sometimes reaches its maximum log likelihood value. However, it does not improve maximum log likelihood value. The reason is that it may not escape from local traps. In MrBayes, all results were worse than SSAMC. Here we wondered what would happen if we ran more iterations. Thus, we ran 10 times BAMBE and MrBayes independently with 2.0×10^7 iterations (Figure 14). During the simulation, we found that there were errors in BAMBE. After an indefinite time, the log likelihood value of BAMBE tends to negative infinity. Thus, results using BAMBE cannot be improved as the simulation goes on. Also, MrBayes never reaches the best value of SSAMC. Therefore, in all cases, SSAMC provides more accurate results comparing the other softwares. Figure 15 indicates that the tree topology estimated by SSAMC more closely resembles to true tree topology. Thus, the SSAMC algorithm can escape from the local trap problems and reach the optimal tree much faster.

3.4.2 Bacteriophage Example

Phylogenetic inferences are premised on the inheritance of ancestral characteristics and on the existence of an evolutionary history defined by changes in these characteristics. Phylogenetic analysis seeks to infer the evolutionary history that is most con-

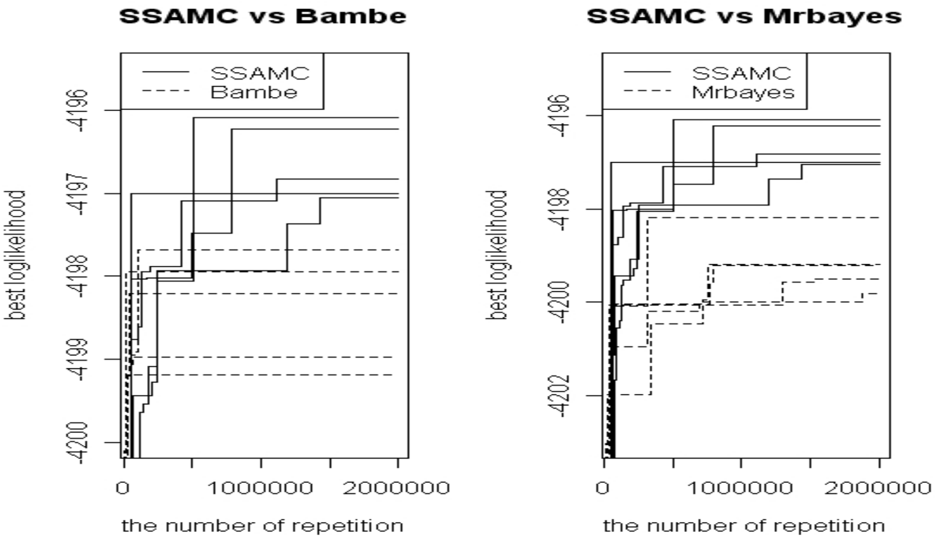


Figure 13: Comparison between estimates by SSAMC and other methods with 2×10^6 iterations. (a) The comparison between SSAMC and BAMBE. (b) The comparison between SSAMC and MrBayes.

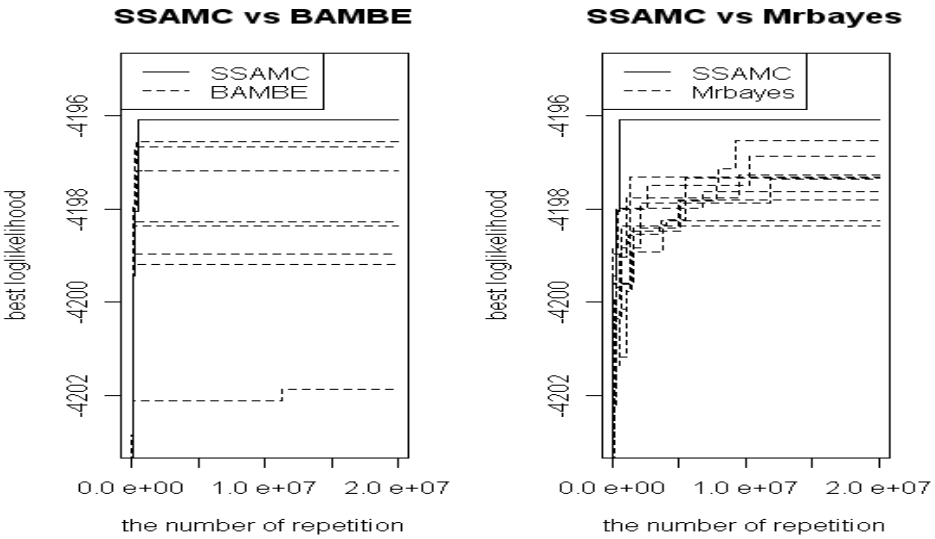
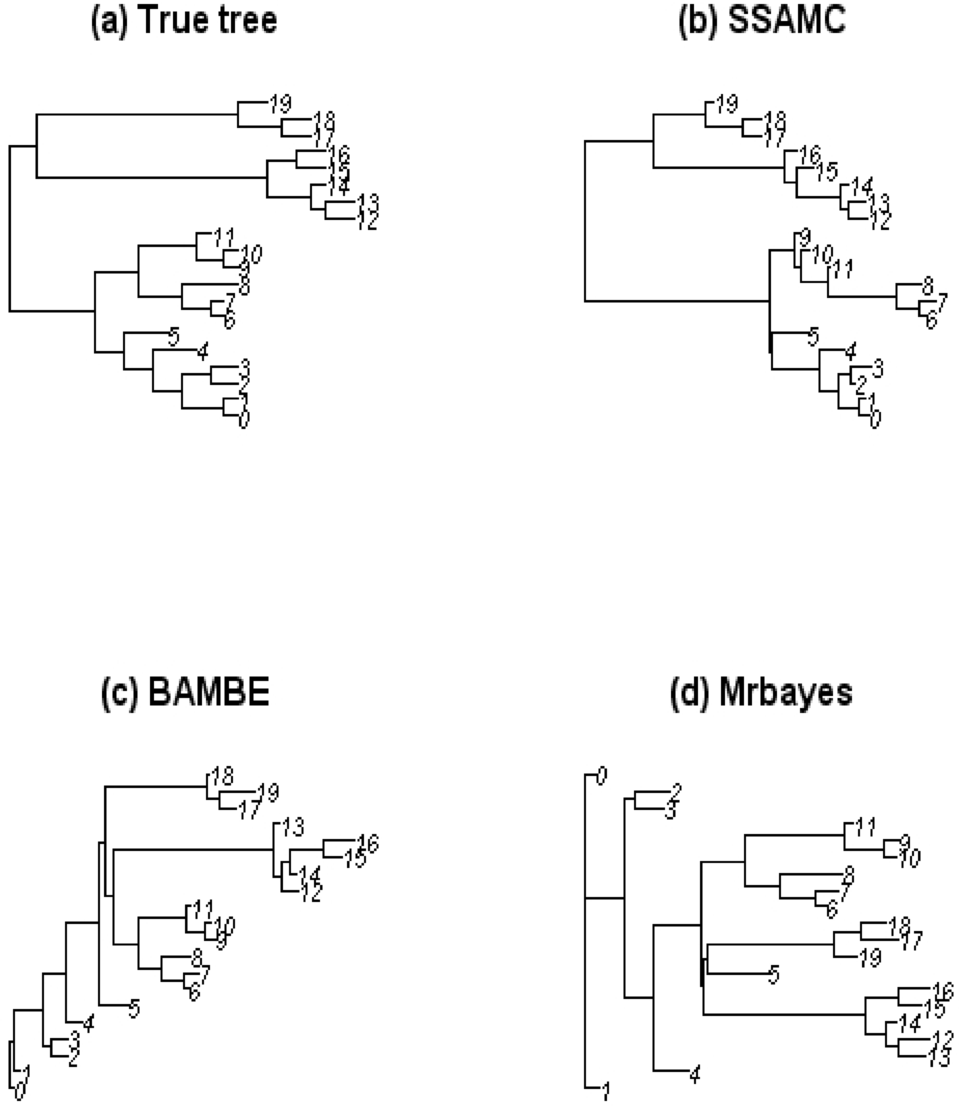


Figure 14: Comparison between estimates by SSAMC and other methods with 2×10^7 iterations. (a) The comparison between SSAMC and BAMBE. (b) The comparison between SSAMC and MrBayes.



(a) True tree

(b) SSAMC

(c) BAMBE

(d) Mrbayes

Figure 15: The comparison of tree topologies generated by SSAMC, BAMBE, and MrBayes. Maximum log likelihood = [(a) -4209.442723, (b)-4196.090699, (c) -4197.680353, (d) -4198.194959].

Table 9: The part of the aligned DNA sequences of nine bacteriophage T7

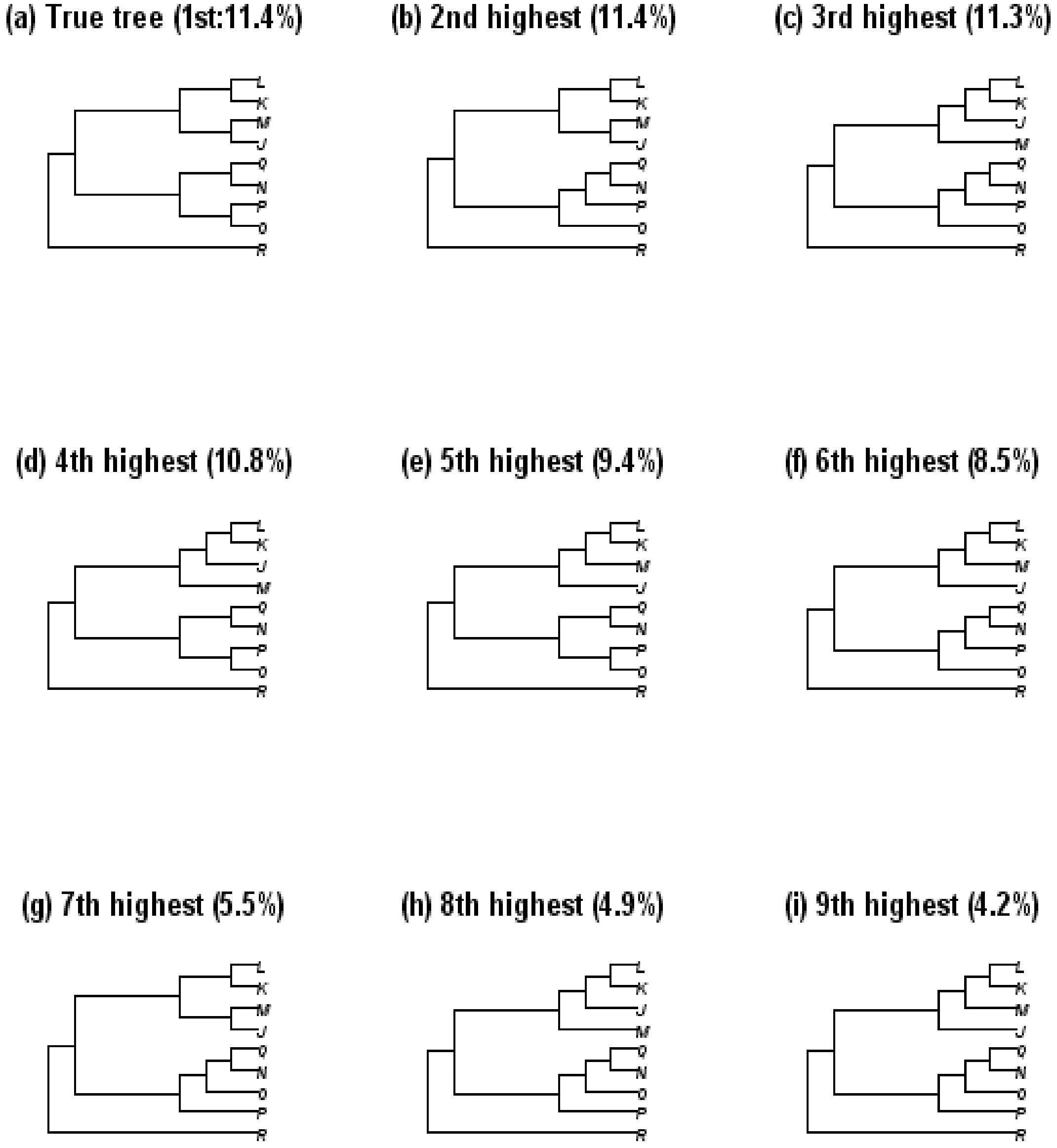
R	CCGGGCCTCG GCTGCGCACC CGCGCCCCAC TGCTGCGGCG GGTCCTCCGG GGACGCTCGG CGC
J	CCGGGCCCTA GCCGTACACC CGCGTTCCAC TGCCACGGCG GGTCCTCCGG TGGTGCCCAG CGT
K	TCGGGCCCCG GCCGCACACC CGCACTCCAC TGCCATGGCG GGGCCGCCGG TGGTGCCCAG CGT
L	TCAGGCCCCG ACCGCACATC CGCACTTCAC TGCCATAGCG GGGCCGCTGG TGGTGCCCAG CGT
M	CTGAGCCCCG GCCGTATACC CGTGCTCCAT TGCCACGGCG GGTTCTCCAA TAGTGTCCAG CGT
N	CCGGGTTCCG GTCACGCACT TACGCCCTGC CGCCGCGACA AATCCTTCGG TGGCACCTGG CAC
O	CTGGACCCCG GCCGCGCACC TGCGCCCCGC TACCGCGATG AATCTTCCGG TGGCATCCGA CAC
P	CCGGACCCCG GCCACGCCCC TGCGCCCCGC TGCCGCGACG AATCCTCCGG TGGCACCCGG TAC
Q	CCGGGTCCCG GTCACGCACT TACGCCCTGC TGTCGCGACA AATCCTTCGG TGGCACCCGG CAC

sistent with a set of observed data. In this example, we considered nine bacteriophage T7 as observed data with a known phylogeny of nine taxa. The data is shown in Table 9 (Hillis *et al.*, 1992). This data is an ideal data for phylogenetic tree construction. The data in Table 9 is part of the aligned DNA sequences of nine bacteriophage T7. This portion, 63 sites out of 1,091 total sites, is regarded as "informative" (See Li *et al.,* 2000) and thus we used only these sites in this example. Taxa R is an outgroup from bacteriophage T7. We assumed the molecular non-clock and used the five-parameter HKY85 model for nucleotide substitution model. The sample space was restricted to the taxa region $\mathcal{X} = \{1, 2, \cdots, 9\}$ and was then partitioned into 6 subregions, $E_1 = \{x \in \mathcal{X} : \; x \leq 4\}, E_2 = \{x \in \mathcal{X} : \; x = 5\}, \cdots, E_6 = \{x \in \mathcal{X} : \; x = 9\}$. The parameters were set as follows: $\tau = 1.0$, $t_0 = 10000$. SSAMC was run with $n = 1.0 \times 10^6$ iterations.

From Table 9, we see that a sequence K is very close to a sequence L, and similarly a sequence N is very close to a sequence Q. On the other hand, the sequences K and L are both far from the sequences N and Q. The nine trees with highest posterior probabilities found by our method reflect this relationship (Figure 16). Together they make up 77.3 % of the total probability. The estimated best tree with maximum log likelihood value is the same as true tree representations. Also, Figures 16(a) and 16(d) contain all 10 top trees in log likelihood value (Table 10). Therefore, SSAMC has a capability to reconstruct phylogenetic trees. In addition, for summarizing samples of substitution parameters and trees and branch lengths, we used MrBayes. Figure 17 shows a plot of the generation versus the log probability of the data (the log likelihood values). Since there is no tendency of increase or decrease over time in this plot, a SSAMC run may be at stationarity. Table 11 is a table summarizing the samples of the parameter values. For each parameter, the table lists the mean and variance of the sampled values, the lower and upper boundaries of the 95% credibility interval, and the median of the sampled values. The last column in Table 11 contains a convergence diagnostic, the Potential Scale Reduction Factor (PSRF). If we have a good sample from the posterior probability distribution, these values should be close to 1.0 (Gelman and Rubin, 1992). Since all PSRF values are close to 1.0, a SSAMC run is regarded as converged. Table 12 shows summary statistics for the taxon bipartitions. We also know that all PSRF values for branch lengths are close to 1.0. The clade credibility tree (Figure 18) gives the probability of each partition or clade in the tree, and the phylogram (Figure 19) gives the branch lengths measured in expected substitutions per site. The clade credibility tree and phylogram indicate that they give a similar result with the true tree representation.

3.4.3 A Nucleotide Sequence Example

We have analyzed aligned protein-coding mitochondrial DNA sequences obtained from 32 species of cichlid fishes (Kocher *et al.,* 1995) using the HKY85 model of nucleotide substitution. Table 13 shows a tribal classification of 32 species of African cichlid fish. Each DNA sequence contains 1044 sites that can be partitioned into three blocks of sites according to codon position, and our analysis allowed different parameter values across blocks (Mau *et al.,* 1999).



(a) True tree (1st:11.4%)

(b) 2nd highest (11.4%)

(c) 3rd highest (11.3%)

(d) 4th highest (10.8%)

(e) 5th highest (9.4%)

(f) 6th highest (8.5%)

(g) 7th highest (5.5%)

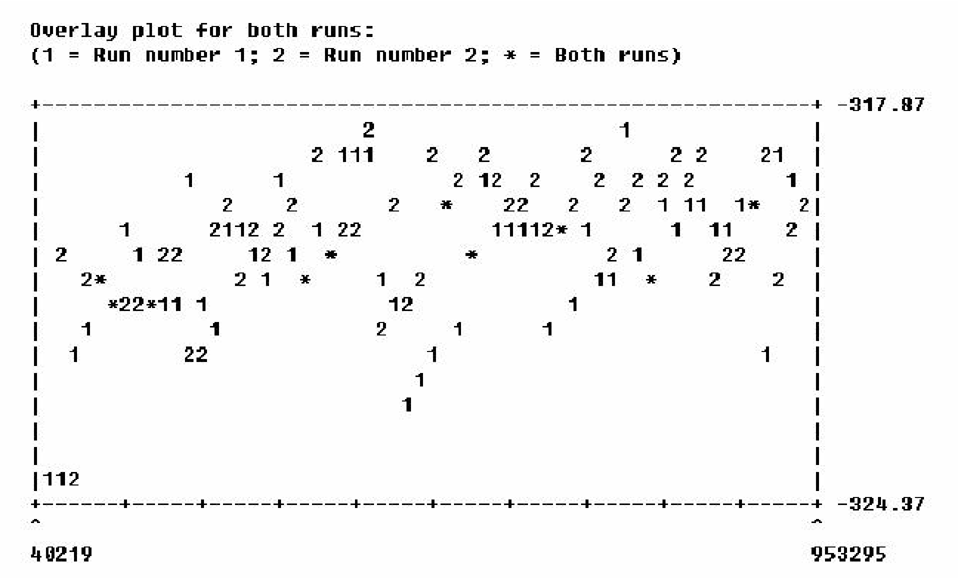
(h) 8th highest (4.9%)

(i) 9th highest (4.2%)

Figure 16: Experimental phylogeny using nine bacteriophage T7 DNA sequences of 63 sites. The topology in (a) is the true phylogenetic tree structure. The nine topologies in (a)-(i) have the highest posterior probabilities among 444,742 possibilities. Topologies (a) through (i) constitute a 77.3% credible regions.

Table 10: The top 10 tree in log likelihood value for nine bacteriophage T7

Rank	log likelihood	Tree topology
1	-310.992241	(R,(((J,M),(K,L)),((P,O),(Q,N))))
2	-311.152437	(R,(((J,M),(K,L)),((P,O),(Q,N))))
3	-311.714725	(R,(((J,M),(K,L)),((P,O),(Q,N))))
4	-311.737703	(R,(((J,M),(K,L)),((P,O),(Q,N))))
5	-311.842822	(R,(((J,M),(K,L)),((P,O),(Q,N))))
6	-311.849127	(R,(((J,M),(K,L)),((P,O),(Q,N))))
7	-312.216711	(R,(((J,M),(K,L)),((P,O),(Q,N))))
8	-312.448688	(R,(((J,M),(K,L)),((P,O),(Q,N))))
9	-312.457940	(R,(((J,(K,L)),M),((P,O),(Q,N))))
10	-312.480285	(R,(((J,(K,L)),M),((P,O),(Q,N))))



```
Overlay plot for both runs:
(1 = Run number 1; 2 = Run number 2; * = Both runs)

+------------------------------------------------------------+ -317.87
|                         2                   1              |
|                     2 111    2   2       2      2 2    21  |
|           1      1             2 12  2    2  2 2 2       1 |
|              2    2       2   *    22   2   2  1 11  1*   2|
|      1      2112 2  1 22          11112* 1      1  11    2 |
| 2     1 22     12 1  *          *          2 1      22     |
|   2*          2 1  *     1  2             11  *    2    2  |
|     *22*11 1              12            1                  |
|   1         1            2     1      1                    |
|  1        22                 1                         1   |
|                             1                              |
|                            1                               |
|                                                            |
|                                                            |
|112                                                         |
+------+-----+-----+-----+-----+-----+-----+-----+-----+-----+ -324.37
^                                                            ^
40219                                                        953295
```

Figure 17: A plot of the generation versus the log likelihood values for nine bacteriophage T7. If a chain is at stationary, this plot should look like 'white noise', that is, they should be no tendency of increase or decrease over time.

Table 11: Model parameter summaries for African cichlids

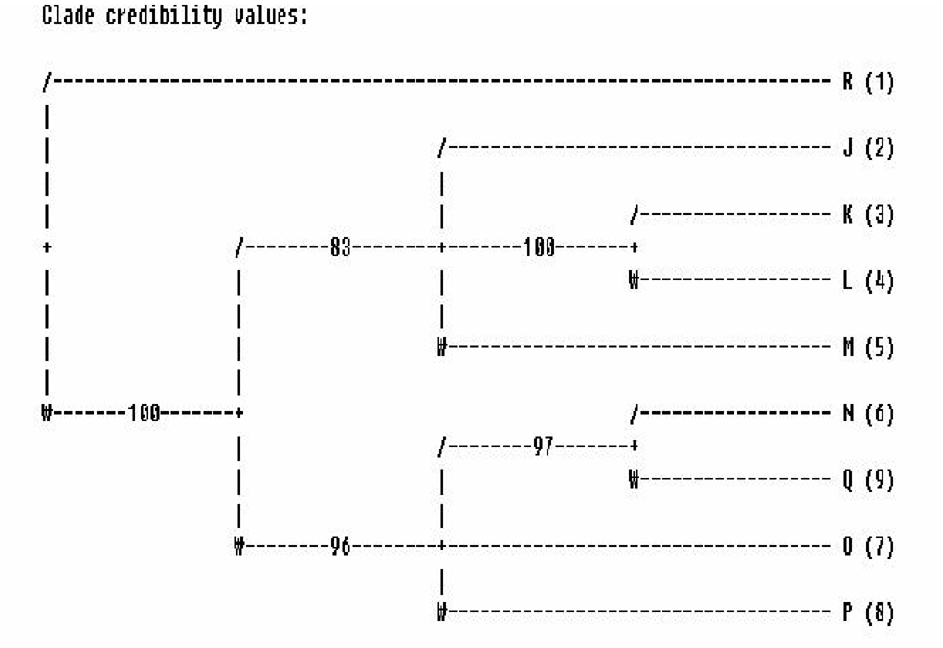
			95 % Cred. Interval			
Parameter	Mean	Variance	Lower	Upper	Median	PSRF*
Tratio	0.025952	0.001973	0.000000	0.100000	0.000000	1.000
Kappa	58.866465	494.812818	22.704060	113.103064	54.338657	1.040
pi(A)	0.131748	0.000715	0.084548	0.189567	0.129840	1.003
pi(G)	0.309034	0.001840	0.227592	0.394851	0.308259	1.000
pi(C)	0.398761	0.002136	0.309287	0.488913	0.398629	1.001
pi(T)	0.160458	0.000687	0.113570	0.216034	0.158825	1.000
Pratio	0.057618	0.002442	0.000000	0.100000	0.100000	1.000
Th	0.055388	0.000638	0.020783	0.119848	0.050985	1.044

*: Convergence diagnostic (PSRF = Potential Scale Reduction Factor[Gelman and Rubin, 1992], uncorrected) should approach 1 as runs converge. The values may be unreliable if you have a small number of samples. PSRF should only be used as a rough guide to convergence since all the assumptions that allow one to interpret it as a scale reduction factor are not met in the phylogenetic context.

Table 12: The summary statistics for taxon bipartitions, a tree with clade credibility, and a phylogram for nine bacteriophage T7

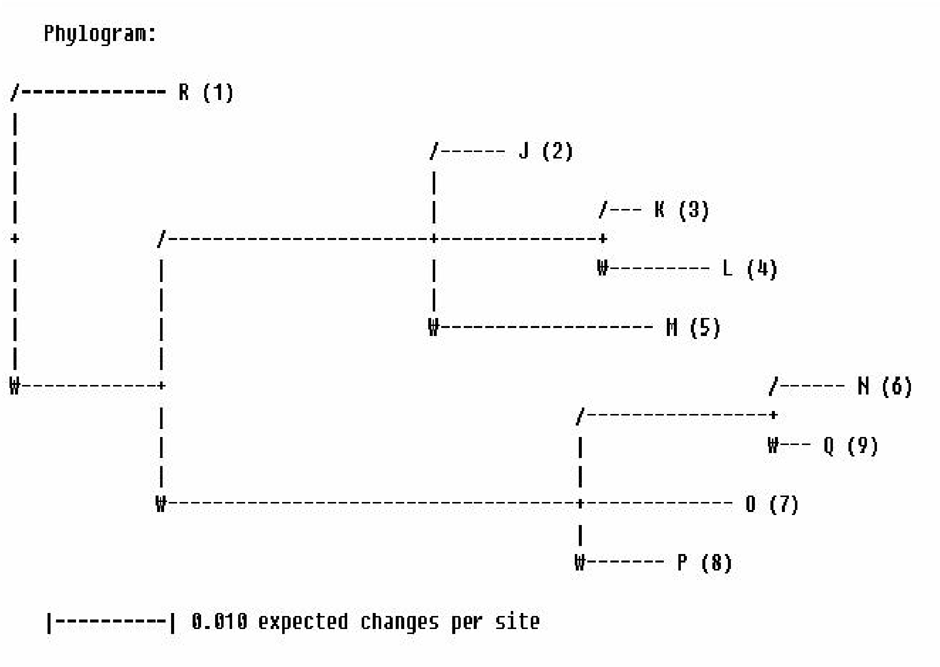
ID	Partition[a]	NUM[b]	Prob[c]	Brlen[d]	Var[e]	PSRF[f]
1	=======+=	444986	1.000000	0.006242	0.000027	1.016
2	=++++++++	444986	1.000000	0.012484	0.000095	1.011
3	======+==	444986	1.000000	0.011870	0.000063	1.022
4	=====+===	444986	1.000000	0.005418	0.000016	1.017
5	========+	444986	1.000000	0.002791	0.000008	1.030
6	====+====	444986	1.000000	0.016767	0.000107	1.037
7	==+======	444986	1.000000	0.003150	0.000012	1.007
8	=+=======	444986	1.000000	0.005370	0.000022	1.018
9	+========	444986	1.000000	0.012484	0.000095	1.011
10	===+=====	444986	1.000000	0.008149	0.000036	1.028
11	==++=====	444777	0.999530	0.013670	0.000083	1.019
12	=====+==+	433309	0.973759	0.015449	0.000097	1.034
13	=====++++	428040	0.961918	0.033980	0.000571	1.013
14	=++++====	367692	0.826300	0.021531	0.000421	1.014
15	======++=	186307	0.418681	0.006910	0.000052	1.006
16	=====+=++	172297	0.387196	0.005720	0.000035	1.012
17	=+++=====	145318	0.326568	0.005047	0.000035	1.003
18	=+==+====	144371	0.324439	0.004363	0.000022	1.014
19	==+++====	113849	0.255848	0.003657	0.000017	1.001
20	=====++=+	76499	0.171913	0.003262	0.000014	1.008
21	====+++++	26065	0.058575	0.006727	0.000041	1.000

a: The partition in the "=" and "+"; e.g., The first partition (ID 1) is the terminal branch lending to taxon 2 since it has a "+" in the 2nd position and a "=" in all other positions. b: The number of times the partition was sampled. c: The probability of the partition. d: The mean of the branch length. e: The variance of the branch length. f: The Potential Scale Reduction Factor (PSRF).



```
Clade credibility values:

/------------------------------------------------------------------------ R (1)
|
|                                    /----------------------------------- J (2)
|                                    |
|                                    |                 /----------------- K (3)
+                 /--------83--------+-------100-------+
|                 |                  |                 \----------------- L (4)
|                 |                  |
|                 |                  \----------------------------------- M (5)
|                 |
\-------100-------+                                    /----------------- N (6)
                  |                  /-------97--------+
                  |                  |                 \----------------- Q (9)
                  |                  |
                  \--------96--------+----------------------------------- O (7)
                                     |
                                     \----------------------------------- P (8)
```

Figure 18: The clade credibility tree of nine bacteriophage T7.



```
Phylogram:

/------------- R (1)
|
|                                     /------ J (2)
|                                     |
|                                     |              /--- K (3)
+            /------------------------+--------------+
|            |                        |              \--------- L (4)
|            |                        |
|            |                        \------------------- M (5)
|            |
\------------+                                                      /------ N (6)
             |                                     /----------------+
             |                                     |                \--- Q (9)
             |                                     |
             \-------------------------------------+------------- O (7)
                                                   |
                                                   \------- P (8)

   |----------| 0.010 expected changes per site
```

Figure 19: The phylogram of nine bacteriophage T7.

Table 13: The tribal classification of 32 species of African cichlid fish

Label	Species name	Tribe	Clade
1	Cichlasoma citrinellum	Central America	Outgroup
2	Pseudotropheus zebra	Malawi	A
3	Buccochromis lepturus	Malawi	A
4	Champsochromis spilorhynchus	Malawi	A
5	Lethrinops auritus	Malawi	A
6	Rhamphochromis sp.	Malawi	A
7	Lobochilotes labiatus	Tropheini	B
8	Petrochromis orthognathus	Tropheini	B
9	Gnathochromis pfefferi	Limnochromini	B
10	Tropheus moorii	Tropheini	B
11	Callochromis macrops	Ectodini	C
12	Cardiopharynx schoutedeni	Ectodini	C
13	Opthalmotilapia ventralis	Ectodini	C
14	Xenotilapia flavipinnus	Ectodini	C
15	Xenotilapia sima	Ectodini	C
16	Chalinochromis popeleni	Lamprologini	D
17	Julidochromis marlieri	Lamprologini	D
18	Telmatochromis temporalis	Lamprologini	D
19	Neolamprologus brichardi	Lamprologini	D
20	Neolamprologus tetracanthus	Lamprologini	D
21	Lamprologus callipterus	Lamprologini	D
22	Lepidiolamprologus elongatus	Lamprologini	D
23	Perissodus microlepis 1	Perissodini	E
24	Perissodus microlepis 2	Perissodini	E
25	Cyphotilapia frontosa	Tropheini	E
26	Tanganicodus irsacae	Eretmodini	Unattached
27	Limnochromis auritus	Limnochromini	E
28	Paracyprichromis brieni	Cyprichromini	E
29	Oreochromis niloticus	Tilapiini	F
30	Tylochromis polylepis	Tylochromini	F
31	Boulengerochromis microlepis	Tilapiini	F
32	Bathybates sp.	Bathybatini	F

We assumed the molecular non-clock. The sample space was restricted to the taxa region $\mathcal{X} = \{1, 2, \cdots, 32\}$ and was then partitioned into 18 subregions, $E_1 = \{x \in \mathcal{X} : \ x \leq 4\}, E_2 = \{x \in \mathcal{X} : \ x = 5\}, \cdots, E_7 = \{x \in \mathcal{X} : \ x = 10\}, E_8 = \{x \in \mathcal{X} : \ x = \{11, 12\}\}, \cdots, E_{18} = \{x \in \mathcal{X} : \ x = \{31, 32\}\}$. The parameters were set as follows: $\tau = 1.0, \ t_0 = 50000$. SSAMC was run with $n = 1.0 \times 10^6$ iterations.

Mau *et al.* (1999) showed a fair degree of similarity in the different solutions such as neighbor-joining plus parsimony, maximum likelihood and MCMC methods. Our numerical result also shows a similarity with tree topologies estimated by Mau *et* *al.* (1999) [i.e., each estimate has clades A, B, C, D, and F in common; the estimates are differ according to the elements of clade E; except for the common pair $\{23, 24\}$, these species are dispersed throughout Lake Tanganyika (Table 14)]. In particular, the estimated tree by our method looks like similar to that by the MCMC method. Like the results of Mau *et al.,* our algorithm concurs in placing the B clade closer to the Malawi flock A.

We are now interested in maximum posterior likelihood. Thus, we compared our algorithm with two popular software applications for phylogenetic tree reconstruction, BAMBE and MrBayes. Figures 20, 21, and 22 indicate that tree topology estimated by SSAMC is quite similar to both BAMBE and MrBayes, but SSAMC is much better in terms of maximum posterior log likelihood value. Tables 15 and 16 show summary statistics for the samples of the parameter values and taxon bipartitions, respectively. Since all PSRF values are close to 1.0, a SSAMC run is regarded as converged. The clade credibility tree (Figure 23) and phylogram (Figure 24) indicate that they give a similar result with the best tree estimated by our method.

Table 14: Comparison of the SSAMC estimate of the phylogeny with estimates using other methods for African cichlid fish. An subtopologies are $A_1 = ((02((0304)05))06)$, $A_2 = ((02((0305)04))06)$, $B_1 = (((0708)10)09)$, $B_2 = (((0708)09)10)$, $B_3 = ((07(0809))10)$, $C_1 = ((11(1213))(1415))$, $C_2 = (11((1213)(1415)))$, $D_1 = (((16(2021))((1718)19))22)$, $D_2 = ((16(((1718)19)(2021)))22)$, $F_1 = ((2930)(3132))$, $F_2 = (((2930)31)32)$, $F_3 = (((2930)32)31)$, $F_4 = (29(30(3132)))$.

Method	Tree topology
Neighbor-joining plus parsimony	$((((((A_2B_3)(C_2(2324)))(2527))28)(D_226))F_3)$
Maximum likelihood	$(((((A_2B_2)((((2324)25)28)27))C_2)(D_226))F_2)$
Markov chain Monte Carlo	$(((((A_1B_1)(((2324)28)(2527)))C_1)(D_126))F_1)$
SSAMC	$((((((A_1B_3)(((2324)28)(2527)))C_1)D_2)26)F_4)$

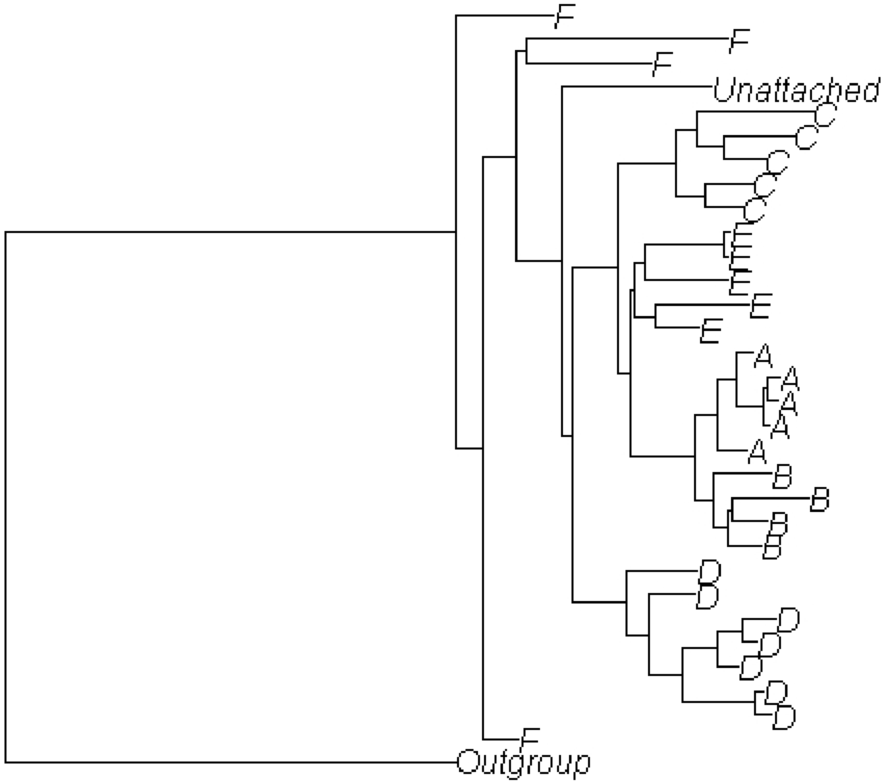


Figure 20: The SSAMC estimate for African cichlid fish (log likelihood = -7736.7444).

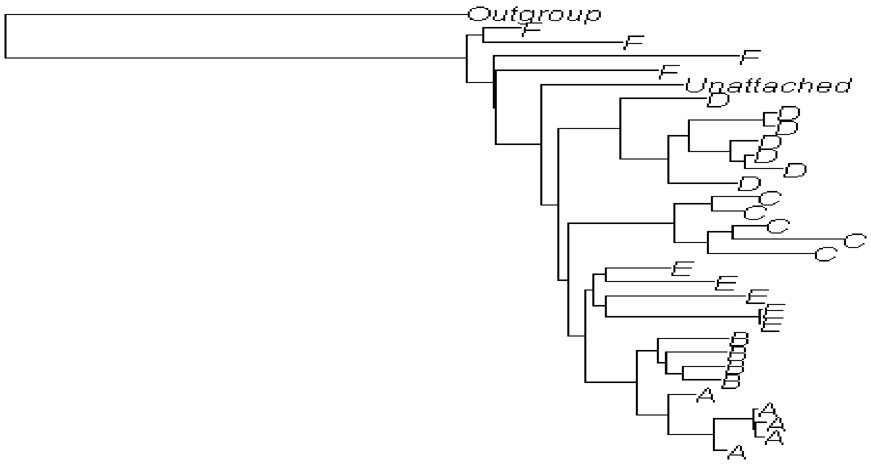


Figure 21: The BAMBE estimate for African cichlid fish (log likelihood = -7888.5659).

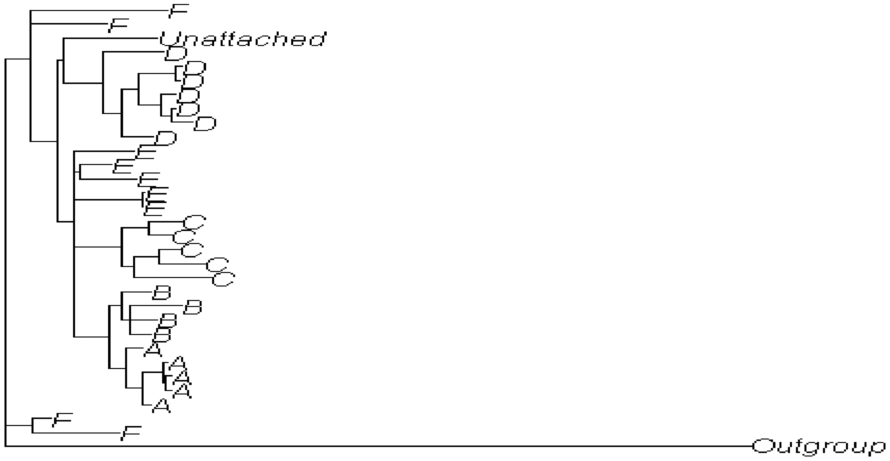


Figure 22: The MrBayes estimate for African cichlid fish (log likelihood = -7876.98).

Table 15: Model parameter summaries of African cichlid fish

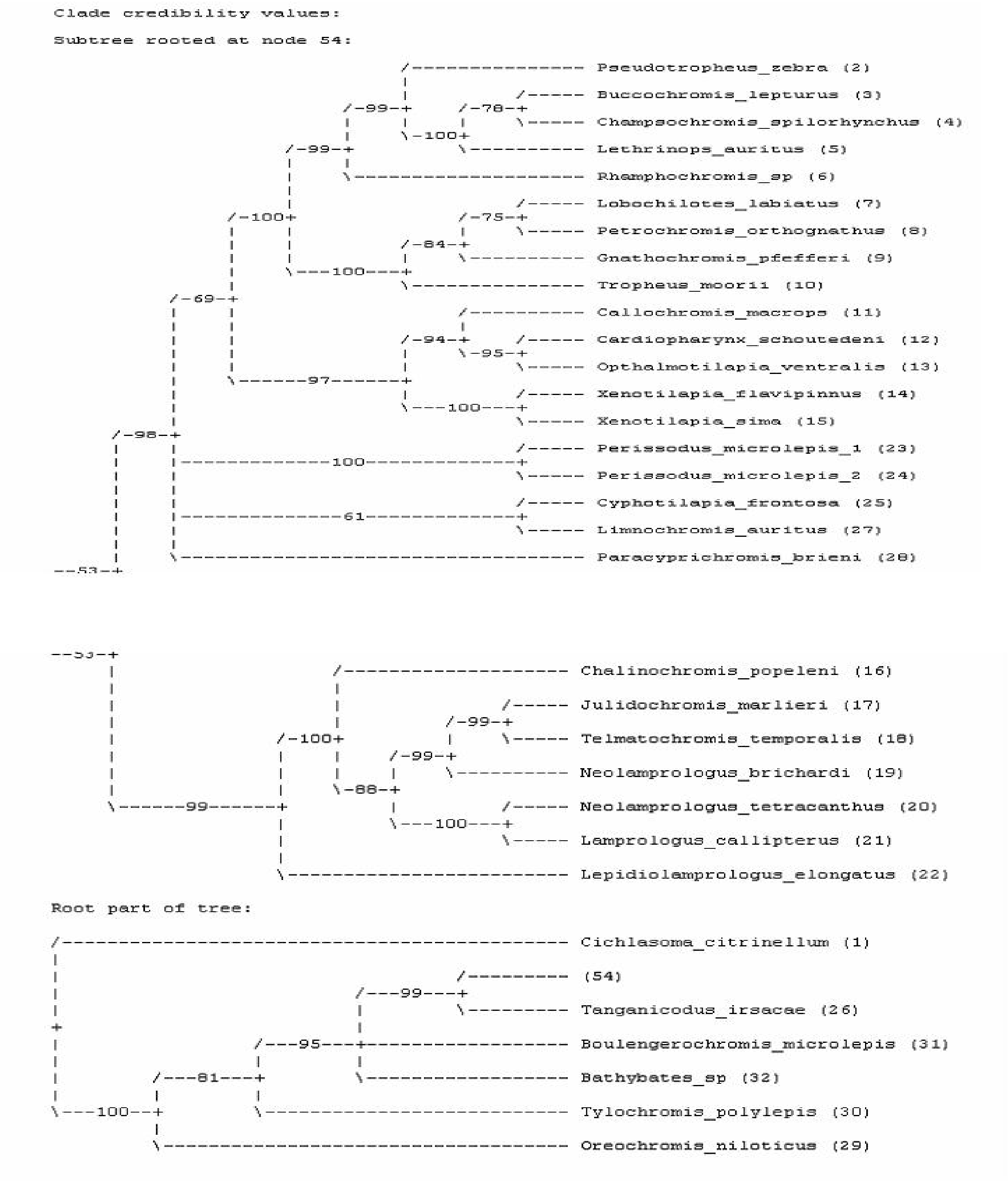
			95 % Cred. Interval			
Parameter	Mean	Variance	Lower	Upper	Median	PSRF*
Tratio	0.013417	0.001188	0.000000	0.100000	0.000000	1.000
Kappa0	5.924017	1.155349	3.982923	8.027161	5.847887	1.015
Theta0	0.547963	0.003829	0.436087	0.679446	0.544960	1.000
pi0(A)	0.284100	0.000355	0.248053	0.321887	0.283671	1.000
pi0(G)	0.200467	0.000263	0.169666	0.233156	0.200106	1.000
pi0(C)	0.325071	0.000391	0.287046	0.364589	0.324866	1.001
pi0(T)	0.190363	0.000248	0.160787	0.222290	0.189913	1.001
Kappa1	2.370734	0.352743	1.366809	3.696883	2.310012	1.000
Theta1	0.341831	0.002969	0.242743	0.456637	0.339473	1.009
pi1(A)	0.173706	0.000320	0.139973	0.209993	0.173328	1.000
pi1(G)	0.122537	0.000238	0.093684	0.154262	0.122170	1.001
pi1(C)	0.319533	0.000483	0.278068	0.364831	0.318883	1.000
pi1(T)	0.384225	0.000548	0.338623	0.431416	0.384030	1.000
Kappa2	13.465889	4.975446	8.217172	18.754512	13.304374	1.008
Theta2	2.110206	0.005745	1.951282	2.250288	2.114424	1.005
pi2(A)	0.381259	0.000387	0.341838	0.419903	0.381433	1.001
pi2(G)	0.101990	0.000047	0.089045	0.115877	0.101816	1.007
pi2(C)	0.328156	0.000200	0.301112	0.356964	0.327929	1.000
pi2(T)	0.188595	0.000099	0.169693	0.208517	0.188337	1.001
Pratio	0.025022	0.001876	0.000000	0.100000	0.000000	1.000
Th	0.115320	0.000436	0.076376	0.152878	0.116998	1.473

*: Convergence diagnostic (PSRF) should approach 1 as runs converge. The values may be unreliable if you have a small number of samples. PSRF should only be used as a rough guide to convergence since all the assumptions that allow one to interpret it as a scale reduction factor are not met in the phylogenetic context.

Table 16: The part of summary statistics for taxon bipartitions of African cichlid fish

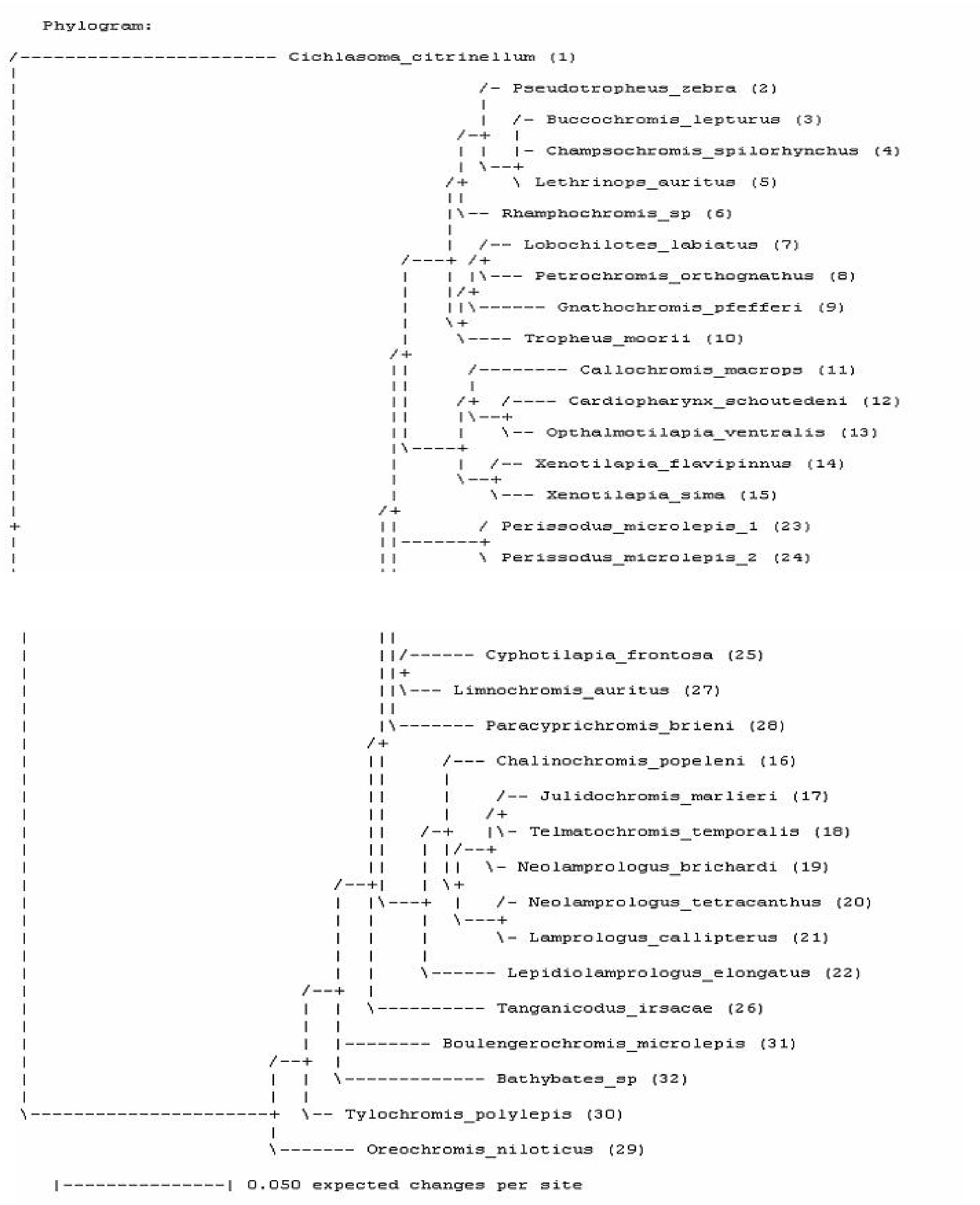
ID	Partition[a]	NUM[b]	Prob[c]	Brlen[d]	Var[e]	PSRF[f]
1	==============+=================	144622	1.0	0.010463	0.000008	1.018
2	=============+==================	144622	1.0	0.008151	0.000005	1.070
3	============+===================	144622	1.0	0.007788	0.000008	1.008
4	==========+=====================	144622	1.0	0.023060	0.000040	1.045
5	===========+====================	144622	1.0	0.013015	0.000021	1.010
6	====+===========================	144622	1.0	0.001149	0.000001	1.000
7	=====+==========================	144622	1.0	0.005761	0.000004	1.013
8	===+============================	144622	1.0	0.002079	0.000003	1.065
9	=+==============================	144622	1.0	0.003194	0.000002	1.010
10	==+=============================	144622	1.0	0.002121	0.000001	1.016
11	===========================+====	144622	1.0	0.020653	0.000022	1.054
12	==============================+=	144622	1.0	0.025331	0.000036	1.016
13	=========================+======	144622	1.0	0.031049	0.000029	1.024
14	===============================+	144622	1.0	0.042156	0.000077	1.002
15	=+++++++++++++++++++++++++++++++	144622	1.0	0.072954	0.000441	1.492
16	============================+===	144622	1.0	0.021344	0.000115	1.056
17	======================+=========	144622	1.0	0.001151	0.000003	1.000
18	=======================+========	144622	1.0	0.001606	0.000005	1.025
19	======================++========	144622	1.0	0.023365	0.000030	1.007
20	========================+=======	144622	1.0	0.019296	0.000024	1.056
21	==========================+=====	144622	1.0	0.011001	0.000009	1.000
22	===============+================	144622	1.0	0.011651	0.000006	1.000
23	===================+============	144622	1.0	0.001583	0.000001	1.112
24	=======+========================	144622	1.0	0.009800	0.000010	1.025
25	==================+=============	144622	1.0	0.004872	0.000003	1.003
26	======+=========================	144622	1.0	0.006463	0.000004	1.053
27	=====================+==========	144622	1.0	0.019778	0.000019	1.005
28	=================+==============	144622	1.0	0.002171	0.000001	1.023
29	+===============================	144622	1.0	0.072954	0.000441	1.492
30	========+=======================	144622	1.0	0.016826	0.000011	1.078
31	================+===============	144622	1.0	0.006206	0.000006	1.002
32	====================+===========	144622	1.0	0.002716	0.000011	1.030
33	=============================+==	144622	1.0	0.007850	0.000012	1.000
34	=========+======================	144622	1.0	0.010423	0.000013	1.021
35	=+++++++++======================	144621	1.0	0.011214	0.000010	1.003

a: The partition in the "=" and "+"; e.g., The first partition (ID 1) is the terminal branch lending to taxon 18 since it has a "+" in the 18th position and a "=" in all other positions. b: The number of times the partition was sampled. c: The probability of the partition. d: The mean of the branch length. e: The variance of the branch length. f: The Potential Scale Reduction Factor (PSRF).



```
Clade credibility values:

Subtree rooted at node 54:

                              /--------------- Pseudotropheus_zebra (2)
                              |
                              |         /----- Buccochromis_lepturus (3)
                         /-99-+    /-78-+
                         |    |    |    \----- Champsochromis_spilorhynchus (4)
                         |    \-100+
                    /-99-+         \---------- Lethrinops_auritus (5)
                    |    |
                    |    \-------------------- Rhamphochromis_sp (6)
                    |
                    |                   /----- Lobochilotes_labiatus (7)
               /-100+              /-75-+
               |    |              |    \----- Petrochromis_orthognathus (8)
               |    |         /-84-+
               |    |         |    \---------- Gnathochromis_pfefferi (9)
               |    \---100---+
               |              \--------------- Tropheus_moorii (10)
          /-69-+
          |    |                   /---------- Callochromis_macrops (11)
          |    |                   |
          |    |              /-94-+    /----- Cardiopharynx_schoutedeni (12)
          |    |              |    \-95-+
          |    |              |         \----- Opthalmotilapia_ventralis (13)
          |    \------97------+
          |                   |         /----- Xenotilapia_flavipinnus (14)
          |                   \---100---+
          |                             \----- Xenotilapia_sima (15)
     /-98-+
     |    |                             /----- Perissodus_microlepis_1 (23)
     |    |-------------100-------------+
     |    |                             \----- Perissodus_microlepis_2 (24)
     |    |                             /----- Cyphotilapia_frontosa (25)
     |    |--------------61-------------+
     |    |                             \----- Limnochromis_auritus (27)
     |    |
     |    \----------------------------------- Paracyprichromis_brieni (28)
--53-+

--53-+
     |                   /-------------------- Chalinochromis_popeleni (16)
     |                   |
     |                   |              /----- Julidochromis_marlieri (17)
     |                   |         /-99-+
     |              /-100+         |    \----- Telmatochromis_temporalis (18)
     |              |    |    /-99-+
     |              |    |    |    \---------- Neolamprologus_brichardi (19)
     |              |    \-88-+
     \------99------+         |         /----- Neolamprologus_tetracanthus (20)
                    |         \---100---+
                    |                   \----- Lamprologus_callipterus (21)
                    |
                    \------------------------- Lepidiolamprologus_elongatus (22)

Root part of tree:

/-------------------------------------------- Cichlasoma_citrinellum (1)
|
|                                   /-------- (54)
|                          /---99---+
|                          |        \-------- Tanganicodus_irsacae (26)
+                          |
|                 /---95---+----------------- Boulengerochromis_microlepis (31)
|                 |        |
|        /---81---+        \----------------- Bathybates_sp (32)
|        |        |
\---100--+        \-------------------------- Tylochromis_polylepis (30)
         |
         \----------------------------------- Oreochromis_niloticus (29)
```

Figure 23: The clade credibility tree of African cichlid fish.



```
   Phylogram:

/----------------------- Cichlasoma_citrinellum (1)
|
|                                         /- Pseudotropheus_zebra (2)
|                                         |
|                                         |  /- Buccochromis_lepturus (3)
|                                       /-+  |
|                                       | |  |- Champsochromis_spilorhynchus (4)
|                                       | \--+
|                                      /+    \ Lethrinops_auritus (5)
|                                      ||
|                                      |\-- Rhamphochromis_sp (6)
|                                      |
|                                      |  /-- Lobochilotes_labiatus (7)
|                                  /---+ /+
|                                  |   | |\--- Petrochromis_orthognathus (8)
|                                  |   |/+
|                                  |   ||\------ Gnathochromis_pfefferi (9)
|                                  |   \+
|                                  |    \---- Tropheus_moorii (10)
|                                 /+
|                                 ||     /-------- Callochromis_macrops (11)
|                                 ||     |
|                                 ||    /+  /---- Cardiopharynx_schoutedeni (12)
|                                 ||    |\--+
|                                 ||    |   \-- Opthalmotilapia_ventralis (13)
|                                 |\----+
|                                 |     |  /-- Xenotilapia_flavipinnus (14)
|                                 |     \--+
|                                 |        \--- Xenotilapia_sima (15)
|                                /+
+                                ||       / Perissodus_microlepis_1 (23)
|                                ||-------+
|                                ||       \ Perissodus_microlepis_2 (24)


|                                ||
|                                ||/------ Cyphotilapia_frontosa (25)
|                                ||+
|                                ||\--- Limnochromis_auritus (27)
|                                ||
|                                |\------- Paracyprichromis_brieni (28)
|                               /+
|                               ||     /--- Chalinochromis_popeleni (16)
|                               ||     |
|                               ||     |    /-- Julidochromis_marlieri (17)
|                               ||     |   /+
|                               ||   /-+   |\- Telmatochromis_temporalis (18)
|                               ||   | |/--+
|                               ||   | ||  \- Neolamprologus_brichardi (19)
|                            /--+|   | \+
|                            |  |\---+  |   /- Neolamprologus_tetracanthus (20)
|                            |  |    |  \---+
|                            |  |    |      \- Lamprologus_callipterus (21)
|                            |  |    |
|                            |  |    \------ Lepidiolamprologus_elongatus (22)
|                         /--+  |
|                         |  |  \---------- Tanganicodus_irsacae (26)
|                         |  |
|                         |  |-------- Boulengerochromis_microlepis (31)
|                      /--+  |
|                      |  |  \------------- Bathybates_sp (32)
|                      |  |
\----------------------+  \-- Tylochromis_polylepis (30)
                       |
                       \------- Oreochromis_niloticus (29)

    |---------------| 0.050 expected changes per site
```

Figure 24: The phylogram of African cichlid fish.

CHAPTER IV

SUMMARY AND CONCLUSIONS

We have shown that the ASAMC algorithm can be effectively applied to simulations of protein folding using the BLN model. In all cases it did better than the SA and Metropolis Monte Carlo method, and it found new lowest energy conformations. The numerical results showed that the ASAMC algorithm is a very promising algorithm for a general optimization task. Liang and Wong (2001) and Liang (2004) showed the ASAMC algorithm can be successfully applied to the $2D$ and $3D$ AB models and suggested that if we can incorporate some specific moves, which are designed based on some specific properties of the protein model, into the ASAMC algorithm, the performance of the ASAMC algorithm may be further improved. Thus, we have proposed one method for the use of secondary structures in protein folding. This method shows that the average minimum energy and r.m.s.d. found by the ASAMC algorithm is better than those by the SA and Metropolis Monte Carlo methods in all runs. The r.m.s.d. data of our simulations are also in the reasonable range (Moult *et al.,* 1999; Venclovas *et al.,* 1999; Lu *et al.,* 2003), although a value of 6 $\mathring{A}$ for the r.m.s.d. has been suggested as a target value for a small protein (Reva *et al.,* 1998). The improvement in accuracy of folding prediction is in effect in current *ab initio* protein folding; however, the correct protein folding is still very difficult, especially for folding predictions with a very simplified model such as the BLN model. Lu *et al.* (2003) stated that the CASP3 meeting indicated the absolute accuracy of all *ab initio* methods is still low compared with solving the structure experimentally, with over 90% of predictions for the "hard" targets having a global r.m.s.d. for $C_\alpha > 10\mathring{A}$.

A further area of interest is how to define the best energy function. In our simulation, the ASAMC algorithm is very effective for small proteins ($<$ 50 sequences). However, for somewhat large proteins ($>$ 50 sequences) our algorithm produces less reliable results including unstableness in the r.m.s.d. of the structures of folded predicted proteins versus the native structure. The use of the ASAMC algorithm will be further applied to these problems.

We have also proposed the use of the sequential structure of phylogenetic trees in conjunction with SAMC to overcome the difficulty of high dimensionality, showing that the SSAMC algorithm can be efficiently applied to simulations for phylogenetic tree reconstruction. Our proposal for investigating evolutionary histories is a package of model assumptions and movement strategies, together with a diagnostics to examine the convergence of run. As in the model used by Mau *et al.* (1999), Larget and Simon (1999), and Li *et al.* (2000), we reviewed with three examples.

A Bayesian analysis of phylogenetic trees suffers from two difficulties. First, it requires the evaluation of high-dimensional summations and integrals and thus this computation is a challenging task because of the curse of dimensionality. Second, it has a local trap problem. The SSAMC algorithm is very efficient with respect to the high-dimensional problems because it makes use of the sequential structure of phylogenetic trees to overcome the curse of dimensionality. In regard to the second problem, our algorithm is also efficient because it has the capability of controlling the sampling frequency to escape from any local traps (Liang *et al.,* 2005). The numerical results indicate that it has a capability of phylogeny tree reconstruction by alleviating local trap difficulty and the curse of dimensionality and is superior to other phylogenetic tree reconstruction methods such as BAMBE (Simon and Larget, 2001) and MrBayes (Ronquist and Huelsenbeck, 2003) in finding the global likelihood maxima. Therefore, we conclude that SSAMC is a new, promising phylogenetic tree reconstruction method which can overcome both curse of dimensionality and local optimum traps.

REFERENCES

Altekar, G., Dwarkadas, S., Huelsenbeck, J. P., and Ronquist, F. (2004), "Parallel Metropolis Coupled Markov Chain Monte Carlo for Bayesian Phylogenetic Inference," *Bioinformatics*, 20, 407–415.

Bastolla, U., Frauenkron, H., Gerstner, E., Grassberger, P., and Nadler, W. (1998), "Testing a New Monte Carlo Algorithm for Protein Folding," *Proteins: Structure, Function & Genetics*, 32, 52–66.

Berg, B. A. and Neuhaus, T. (1991), "Multicanonical Algorithms for First Order Phase Transitions," *Physics Letters B*, 267, 249–253.

——— (1992), "Multicanonical Ensemble: A New Approach to Simulate First-order Phase Transitions," *Physical Review Letters*, 68, 9–12.

Berman, H. M., Westbrook, J., Feng, Z., Gilliland, G., Bhat, T. N., Weissig, H., Shindyalov, I. N., and Bourne, P. E. (2000), "The Protein Data Bank," *Nucleic Acids Research*, 28, 235–242.

Brown, S., Fawzi, N. J., and Head-Gordon, T. (2003), "Coarse-grained Sequences for Protein Folding and Design," *Proceedings of the National Academy of Sciences of the United States of America*, 100, 10712–10717.

Creighton, T. E. (1978), "Experimental Studies of Protein Folding and Unfolding," *Progress in Biophysics and Molecular Biology*, 33, 231–297.

Dayhoff, M. O., Schwartz, R. M., and Orcutt, B. C. (1978), "A Model of Evolutionary Change in Proteins," in *Atlas of Protein Sequence and Structure*, ed. M. O. Dayhoff, Vol. 5, supplement 3, 345–352. Washington D. C.: National Biomedical Research Foundation.

Delyon, B., Lavielle, M., and Moulines, E. (1999), "Convergence of a Stochastic Approximation Version of the EM Algorithm," *Annals of Statistics*, 27, 94–128.

Deutsch, J. M. (1997), "Long Range Moves for High Density Polymer Simulations," *Journal of Chemical Physics*, 106, 8849–8854.

Dill, K. A., Fiebig, K. M., and Chan, H. S. (1993), "Cooperativity in Protein-Folding Kinetics," *Proceedings of the National Academy of Sciences of the United States of America*, 90, 1942–1946.

Drozdek, A. and Simon, D. (1995), *Data Structure in C*, Boston: PWS Publishing Company.

Durbin, R., Eddy, S., Krogh, A., and Mitchison, G. (1998), *Biological Sequence Analysis: Probabilistic Models of Proteins and Nucleic Acids*, Cambridge, England: Cambridge University Press, pp. 206–210.

Evans, S. N. and Speed, T. P. (1993), "Invariants of Some Probability Models used in Phylogenetic Inference," *Annals of Statistics*, 21, 355–377.

Felsenstein, J. (1981), "Evolutionary Trees from DNA Sequences: A Maximum Likelihood Approach," *Journal of Molecular Evolution*, 17, 368–376.

——— (1983), "Statistical Inference of Phylogenies," *Journal of the Royal Statistical Society, A*, 146, 246–272.

——— (1985), "Confidence Limits on Phylogenies: An Approach using the Bootstrap," *Evolution*, 39, 783–791.

——— (1993), “PHYLIP (Phylogenetic Inference Package),” University of Washington, Seattle.

Fitch, W. M. (1971), “Toward Defining the Course of Evolution: Minimum Change for a Specific Topology,” *Systematic Zoology*, 20, 406–416.

Fraley, C. and Raftery, A. E. (2002), “Model-based on Clustering, Discriminant Analysis, and Density Estimation,” *Journal of the American Statistical Association*, 97, 611–631.

Frauenkron, H., Bastolla, U., Gerstner, E., Grassberger, P., and Nadler, W. (1998), “New Monte Carlo Algorithm for Protein Folding,” *Physical Review Letters*, 80, 3149–3152.

Gelfand, A. E. and Banerjee, S. (1998), “Computing Marginal Posterior Modes using Stochastic Approximation,” Technical Report, University of Connecticut, Department of Statistics.

Gelman, A. and Rubin, D. B. (1992), “Inference from Iterative Simulation using Multiple Sequences,” *Statistical Science*, 7, 457–511.

Geyer, C. J. (1991), “Markov Chain Monte Carlo Maximum Likelihood,” in *Computing Science and Statistics: Proceedings of the 23rd Symposium on the Interface*, ed. E. M. Keramigas, Fairfax, VA: Interface Foundations, p.156.

Goldberg, D. E. (1989), *Genetic Algorithms in Search, Optimization, and Machine Learning*, Ann Arbor, MI: Addison-Wesley.

Grassberger, P. (1997), “Pruned-enriched Rosenbluth Method: Simulations of θ Polymers of Chain Length up to 1000000,” *Physical Review E*, 56, 3682–3693.

Gu, M. G. and Kong, F. H. (1998), "A Stochastic Approximation Algorithm with Markov Chain Monte-Carlo Method for Incomplete Data Estimation Problems," *Proceedings of the National Academy of Sciences of the United States of America*, 95, 7270–7274.

Gu, M. G. and Zhu, H. T. (2001), "Maximum Likelihood Estimation for Spatial Models by Markov Chain Monte Carlo Stochastic Approximation," *Journal of the Royal Statistical Society: Series B*, 95, 339–355.

Guo, Z., Thirumalai, D., and Honeycutt, J. D. (1992), "Folding Kinetics of Proteins: A Model Study," *Journal of Chemical Physics*, 97, 525–535.

Hasegawa, M., Kishino, H., and Yano, T. (1985), "A New Molecular Clock of Mitochondrial DNA and the Evolution of Hominoids," *Proceedings of the Japanese Academy of Sciences, B*, 60, 95–98.

Henikoff, S. and Henikoff, J. G. (1992), "Amino Acid Substitution Matrices from Protein Blocks," *Proceedings of the National Academy of Sciences of the United States of America*, 89, 10915–10919.

Hesselbo, B. and Stinchcomb, R. B. (1995), "Monte Carlo Simulation and Global Optimization without Parameters," *Physical Review Letters*, 74, 2151–2155.

Hillis, D. M., Bull, J. J., White, M. E., Badgett, M. R., and Molineux, I. J. (1992), "Experimental Phylogenetics: Generation of a Known Phylogeny," *Science*, 255, 589–592.

Holland, J. H. (1975), *Adaptation in Natural and Artificial Systems*, Ann Arbor, MI: The University of Michigan Press.

Honeycutt, J. D. and Thirumalai, D. (1990), "Metastability of the Folded States of Globular Proteins," *Proceedings of the National Academy of Sciences of the United States of America*, 87, 3526–3529.

——— (1992), "The Nature of Folded States of Globular Proteins," *Biopolymers*, 32, 695–709.

Huelsenbeck, J. P. and Ronquist, F. (2001), "MrBayes: Bayesian Inference of Phylogenetic Trees," *Bioinformatics*, 17, 754–755.

Hukushima, K. and Nemoto, K. (1996), "Exchange Monte Carlo and Application to Spin Glass Simulations," *Journal of the Physical Society of Japan*, 65, 1604–1608.

Jukes, T. H. and Cantor, C. R. (1969), "Evolution of Protein Molecules," in *Mammalian Protein Metabolism*, ed. H. N. Munro, New York: Academic Press, pp. 21–132.

Kabsch, W. and Sander, C. (1983), "Dictionary of Protein Secondary Structure: Pattern Recognition of Hydrogen-bonded and Geometrical Features," *Biopolymers*, 22, 2577–2637.

Kim, P. S. and Baldwin, R. L. (1990), "Intermediates in the Folding Reactions of Small Proteins," *Annual Review of Biochemistry*, 59, 631–660.

Kim, S. Y., Lee, S. J., and Lee, J. (2003), "Conformational Space Annealing and an Off-lattice Frustrated Model Protein," *Journal of Chemical Physics*, 119, 19, 10274–10279.

Kimura, M. (1980), "A Simple Method for Estimating Evolutionary Rates of Base Substitutions Through Comparative Studies of Nucleotide Sequences," *Journal of Molecular Evolution*, 16, 111–120.

Kirkpatrick, S., Gelatt, Jr., C. D., and Vecchi, M. P. (1983), "Optimization by Simulated Annealing," *Science*, 220, 671–680.

Kishino, H., Miyata, T., and Hasegawa, M. (1990), "Maximum Likelihood Inference of Protein Phylogeny and the Origin of Chloroplasts," *Journal of Molecular Evolution*, 31, 151–160.

Kocher, T. D., Conroy, J. A., McKaye, K. R., Stauffer, J. R., and Lockwood, S. F. (1995), "Evolution of NADH Dehydrogenase Subunit 2 in East African Cichlid Fish," *Molecular Phylogenetics and Evolution*, 4, 420–432.

Lake, J. A. (1987), "A Rate-independent Technique for Analysis of Nucleic Acid Sequences: Evolutionary Parsimony," *Molecular Biology and Evolution*, 4, 167–191.

Larget, B. and Simon, D. (1999), "Markov Chain Monte Carlo Algorithms for the Bayesian Analysis of Phylogenetic Trees," *Molecular Biology and Evolution*, 16, 750–759.

Lee, J. (1993), "New Monte Carlo Algorithm: Entropic Sampling," *Physical Review Letters*, 71, 211–214.

Levinthal, C. (1968), "Are There Pathways for Protein Folding?" *Journal de Chimie Physique et de Physico Chimie Biologique*, 65, 44–45.

Li, S., Peral, D., and Doss, H. (2000), "Phylogenetic Tree Construction Using Markov Chain Monte Carlo," *Journal of the American Statistical Association*, 95, 493–508.

Li, Z. and Scheraga, H. A. (1987), "Monte Carlo-minimization Approach to the Multiple-minima Problem in Protein Folding," *Proceedings of the National Academy of Sciences of the United States of America*, 84, 6611–6615.

Liang, F. (2003), "Use of Sequential Structure in Simulation from High Dimensional Systems," *Physical Review E*, 67, 56101–56107.

——— (2004), "Annealing Contour Monte Carlo Algorithm for Structure Optimization in an Off-lattice Protein Model," *Journal of Chemical Physics*, 120, 14, 6756–6763.

——— (2006), "Annealing Stochastic Monte Carlo for Neural Network Training," Technical Report, Texas A&M University, Department of Statistics.

Liang, F., Liu, C., and Carroll, R. (2005), "Stochastic Approximation in Monte Carlo Computation," Technical Report, Revised for JASA, Texas A&M University, Department of Statistics.

Liang, F. and Wong, W. H. (2001), "Evolutionary Monte Carlo for Protein Folding Simulations," *Journal of Chemical Physics*, 115, 7, 3374–3380.

Lu, B. Z., Wang, B. H., Chen, W. Z., and Wang, C. X. (2003), "A New Computational Approach for Real Protein Folding Prediction," *Protein Engineering*, 16, 9, 659–663.

Maddison, D. R. (1991), "The Discovery and Importance of Multiple Islands of Most Parsimonious Trees," *Systematic Zoology*, 40, 315–328.

Marinari, E. and Parisi, G. (1992), "Simulated Tempering: A New Monte Carlo Scheme," *Europhysics Letters*, 19, 451–458.

Mau, B. and Newton, M. A. (1997), "Phylogenetic Inference for Binary Data on Dendograms using Markov Chain Monte Carlo," *Journal of Computational Graphics Statistics*, 6, 122–131.

Mau, B., Newton, M. A., and Larget, B. (1999), "Bayesian Phylogenetic Inference via Markov Chain Monte Carlo," *Biometrics*, 55, 1–12.

Mengersen, K. L. and Tweedie, R. L. (1996), "Rates of Convergence of the Hastings and Metroplois Algorithms," *The Annals of Statistics*, 24, 101–121.

Miller, R., Danko, C. A., Fasolka, M. J., Balazs, A. C., Chan, H. S., and Dill, K. A. (1992), "Folding Kinetics of Proteins and Copolymers," *Journal of Chemical Physics*, 768–780.

Moult, J., Hubbard, T., Fidelis, K., and Pedersen, J. T. (1999), "Critical Assessment of Methods of Protein Structure Prediction (CASP): Round III," *Proteins: Structure, Function & Genetics*, 2–6 Supplement 3.

Moyeed, R. A. and Baddeley, A. J. (1991), "Stochastic Approximation of the MLE for a Spatial Point Pattern," *Scandinavian Journal of Statistics*, 18, 39–50.

Newton, M. A. (1996), "Bootstrapping Phylogenies: Large Deviations and Dispersion Effects," *Biometrika*, 83, 315–328.

Ramakrishnan, R., Ramachandran, B., and Pekny, J. F. (1997), "A Dynamic Monte Carlo Algorithm for Exploration of Dense Conformational Spaces in Heteropolymers," *Journal of Chemical Physics*, 106, 2418–2425.

Rannala, B. and Yang, Z. (1996), "Probability Distribution of Molecular Evolutionalry Trees: A New Method of Phylogenetic Inference," *Journal of Molecular Evolution*, 43, 304–311.

Reva, B. A., Finkelstein, A. V., and Skolnick, J. (1998), "What is the Probability of a Chance Prediction of a Protein Structure with an rmsd of 6 A?" *Folding And Design*, 3, 2, 141–147.

Ronquist, F. and Huelsenbeck, J. P. (2003), “MrBayes 3: Bayesian Phylogenetic Inference under Mixed Models,” *Bioinformatics*, 19, 1572–1574.

Rosenbluth, M. N. and Rosenbluth, A. W. (1955), “Monte Carlo Calculations of the Average Extension of Macromolecular Chains,” *Journal of Chemical Physics*, 23, 356–359.

Rost, B. and Sander, C. (1993), “Prediction of Protein Secondary Structure at Better than 70% Accuracy,” *Journal of Molecular Biology*, 232, 584–599.

Saitou, N. and Nei, M. (1987), “The Neighbor-joining Method: A New Method for Reconstructing Phylogenetic Trees,” *Molecular Biology and Evolution*, 4, 406–425.

Salemi, M. and Vandamme, A. M. (2003), *The Phylogenetic Handbook: A Practical Approach to DNA and Protein Phylogeny*, London: Cambridge Press.

Salter, L. A. and Pearl, D. K. (2001), “Stochastic Search Strategy for Estimation of Maximum Likelihood Phylogenetic Trees,” *Systematic Zoology*, 50, 7–17.

Simon, D. and Larget, B. (2001), “Bayesian Analysis in Molecular Biology and Evolution (BAMBE),” Version 2.03. Department of Mathematics and Computer Science, Duquesne University, Pittsburgh, PA.

Sinsheimer, J. S., Lake, J. A., and Little, R. J. (1996), “Bayesian Hypothesis Testing of Four-taxon Topologies using Molecular Sequence Data,” *Biometrics*, 52, 193–210.

Tamura, K. and Nei, M. (1993), “Estimation of the Number of Nucleotide Substitutions in the Control Region of Mitochondrial DNA in Humans and Chimpanze,” *Molecular Biology and Evolution*, 10, 512–526.

Thirumalai, D. and Guo, Z. (1995), “Nucleation Mechanism for Protein Folding and Theoretical Predictions for Hydrogen-Exchange Labelling Experiments,” *Biopolymers*, 35, 137–140.

Veitshans, T., Klimov, D., and Thirumalai, D. (1997), “Protein Folding Kinetics: Time Scales, Pathways, and Energy Landscapes in Terms of Sequence Dependent Properties,” *Folding And Design*, 2, 1–22.

Venclovas, C., Zemla, A., Fidelis, K., and Moult, J. (1999), “Some Measures of Comparative Performance in the Three CASPs,” *Proteins: Structure, Function & Genetics*, 37, 231–237.

Wang, F. and Landau, D. P. (2001), “Efficient Multiple-range Random Walk Algorithm to Calculate the Density of States,” *Physical Review Letters*, 86, 2050–2053.

Wetlaufer, D. B. (1973), “Nucleation, Rapid Folding, and Globular Intrachain Regions in Proteins,” *Proceedings of the National Academy of Sciences of the United States of America*, 70, 697.

Wong, W. H. and Liang, F. (1997), “Dynamic Weighting in Monte Carlo and Optimization,” *Proceedings of the National Academy of Sciences of the United States of America*, 94, 14220–14224.

Yang, Z. and Rannala, B. (1997), “Bayesian Phylogenetic Inference using DNA Sequences: A Markov Chain Monte Carlo Method,” *Molecular Biology and Evolution*, 14, 717–724.

Younes, L. (1988), “Estimation and Annealing for Gibbsian Fields,” *Annales de l’institut Henri Poincare (B) - Probabilites et Statistiques*, 24, 269–294.

——— (1999), "On the Convergence of Markovian Stochastic Algorithms with Rapidly Decreasing Ergodicity Rates," *Stochastic and Stochatic Reports*, 65, 1779–228.

VITA

Sooyoung Cheon was born in Naju, Korea. He received a Bachelor of Science degree in mathematics from Korea University in Seoul, Korea in 1994 and Master of Science degree in statistics from Korea University in Seoul, Korea, under the direction of Dr. Song, Seuck Heun in 2002. He continued his studies under the direction of Dr. Faming Liang and received a Doctor of Philosophy degree from Texas A&M University in May 2007. His permanent address is 629-2 Seheung-burak, Yongheung 2-gu, Geumjeong-meon, Yeongam-gun, Jeonnam, Korea.