Reprinted from Theoretical Population Biology
All Rights Reserved by Academic Press, New York and London

Vol. 46, No. 2, October 1994
Printed in Belgium

Simulating Probability Distributions in the Coalescent*

R. C. Griffiths

Monash University, Department of Mathematics, Clayton 3168, Australia

AND

Simon Tavaré

University of Southern California, Departments of Mathematics and Biological Sciences, Los Angeles, California 90089-1113

Received January 7, 1993

We describe some computational algorithms for computing probability distributions for sample configurations from the finite-sites models in population genetics. One particular interest is the development of computational methods for estimating substitution rates for DNA sequence data using likelihood techniques. The approach uses a recursion satisfied by the sampling probabilities to construct a Markov chain with a set of absorbing states in such a way that the required sampling distribution is the mean of a functional of the process up to the absorption time. This provides a conceptually simple framework for simulating the likelihood of the data for a set of parameter values. The method is particularly attractive in practice: it is simple to program and can be extended to cover other features of interest such as the infinitely-many-sites process, recombination, selection, and variable population size.
© 1994 Academic Press, Inc.

1. Introduction

In this paper we describe some computational algorithms for computing probability distributions for sample configurations from the finite-sites models in population genetics. Our main interest is the development of computational methods for estimating substitution rates for DNA sequence data using maximum likelihood. In broad terms the method we propose has two main aspects: The first is the derivation of a recursion satisfied by the sampling distribution of the data—the distribution of the number of each allele observed in the sample. This uses the structure of the coalescent that describes the ancestral relationships among the individuals in the sample. The recursion is, typically, too large a system to solve by direct numerical methods. We adopt instead a simulation approach. This leads to the

* The authors were supported in part by NSF grant DMS 90-05833. This research was started while Bob Griffiths was visiting the Mathematics Department at the University of Southern California. He thanks them for their hospitality.

0040-5809/94 $6.00

second aspect, the construction of a Markov chain with a set of absorbing states in such a way that the required sampling distribution is the mean of a functional of the process up to the absorption time. This provides a conceptually simple framework for simulating the likelihood of the data for given parameter values, or, more generally, for a set of parameter values.

The method is particularly attractive in practice: it is simple to program, and can be extended to cover other features of interest such as the infinitely-many-sites process, recombination, selection, and variable population size.

The overview of this paper is as follows. In Section 2 we describe the coalescent model of neutral mutation and the basic recurrence relation for the sampling distribution. In Section 3 we construct the associated Markov process and show how it can be used to simulate observations from the recursion. Section 4 studies some examples for the K-allele model. For some of these models, explicit sampling formulas are known that may be used to calibrate and check the simulation algorithm. Some variance reduction techniques are introduced too. In Section 5, we specialize to the important case in which the alleles are DNA sequences. The primary emphasis is on the development of variance reduction techniques and parsimonious methods for speeding up the simulations. Among the theoretical results in this section is a study of the joint distribution of the sample and the number of mutations in the ancestral tree that led to the sample. These results, of interest in their own right, provide one approach to reducing simulation time, by aborting runs that have "too many" mutations. Section 6 gives some examples of the simulation method for sequences. Section 7 generalizes the basic simulation approach by describing a method for simulating a likelihood surface from a single Markov chain. This technique is very useful in the context of maximum likelihood estimation, as it avoids the simulation of a surface by independent realizations of a process at different parameter values. The simultaneous method is to be preferred if the cost of evaluating functionals is cheap compared to the cost of simulating the underlying Markov chain. This is indeed the case for sequence data. The methods are illustrated by some numerical examples. The concluding Section 8 discusses the computational aspects of the algorithm, examples of other areas in which the technique might be useful, and directions for future developments in the population genetics setting.

A computer implementation of the algorithms in this paper is available from the authors on request.

2. The Coalescent Process

The coalescent process introduced by Kingman (1982a, b) describes the ancestral tree of a sample of n genes. Review articles are provided by

Tavaré (1984) and Hudson (1991). Ancestral lines backwards in time *coalesce* when two ancestors of the sample themselves share a common ancestor. A tree is then constructed with leaves representing the n sample genes and vertices where ancestral lines coalesce. The root of the tree is the common ancestor of the sample. The tree is binary in the time scale used. Let $T_n, \ldots, T_2$ be the lengths of time that there are $n, n-1, \ldots, 2$ distinct ancestors of the sample. These are mutually independent exponential random variables with rate parameters $n(n-1)/2, \ldots, 2(2-1)/2$. The tree has j edges of length T_j, $j=n, \ldots, 2$. Measuring time backwards, the number of ancestors of the sample is a death process. The quadratic death rates imply that the process is well defined beginning at the entrance boundary $n=\infty$, interpreted as the ancestral process of the entire population. The process can be derived as a limit from a Wright–Fisher model, with a population size of $2N$ genes, when time is measured in units of $2N$ generations and $N \to \infty$. This time scaling is the one used in the diffusion process approximation of the Wright–Fisher model.

Mutations occur in a Poisson process of rate $\theta/2$ along the edges of the tree, mutations in different edges being conditionally independent given the length of the edges. A general mutation scheme is one where there are d possible allele types $1, \ldots, d$ and when mutation occurs a transition is made from type i to j according to the entry p_{ij} in a transition matrix P. It is convenient to let entries on the diagonal of P be possibly non-zero, thereby allowing for differing total rates away from different alleles. The mutation rates in the model are uniquely determined by the generator matrix

$$R=(r_{ij}) \equiv \frac{\theta}{2}(P-I). \tag{1}$$

As a limit from the Wright–Fisher model, if the probability of an offspring of a type i parent being of type j is up_{ij}, $i \neq j$, then $\theta = 4Nu$ is held constant as $N \to \infty$.

The configuration of types in the sample is determined by the mutations in the tree from the root to the leaves. It will be assumed that P is a regular matrix with a stationary distribution π. If the common ancestor is chosen from a stationary population, then her type has the distribution π. Here are some examples for the gene state space.

(i) Two types, labelled 1 and 2.

$$P=\begin{pmatrix} 1-u & u \\ v & 1-v \end{pmatrix}, \qquad \pi=\left(\frac{v}{u+v}, \frac{u}{u+v}\right).$$

(ii) Four types, labelled 1, 2, 3, and 4, representing nucleotides A, C, G, T in a DNA sequence. P is determined by whatever model for base changes is appropriate.

(iii) $4^3 = 64$ types representing codons. P might be determined by a model which takes into account silent and non-silent sites in the codon, for example.

(iv) K types, representing small regions of DNA for example. P is an arbitrary mutation matrix.

(v) 4^s types representing a DNA sequence of length s bases. We assume that mutations cause just a single base change, and that the sites are completely linked. The jth site has a transition matrix $P_j, j = 1, ...s$, with stationary distribution π^j, and the probabilities of where the change happens are $h_j, j = 1, 2, ..., s$. Then

$$P = \sum_{j=1}^{s} h_j I \otimes I \otimes \cdots \otimes P_j \otimes \cdots \otimes I, \tag{2}$$

where $\otimes$ denotes direct product, I is the identity matrix, and

$$\pi = \pi^1 \otimes \cdots \otimes \pi^s.$$

Types are denoted by sequences $(i_1, ..., i_s)$ with entries in $\{1, 2, 3, 4\}$.

(vi) As in example (v), but with more general mutation structure, reflecting the fact that mutations might not produce just single base substitutions.

Let $q(\mathbf{n})$ be the probability that a sample of n genes has a type configuration of $\mathbf{n} = (n_1, ..., n_d)$. A fundamental recursion is

$$q(\mathbf{n}) = \frac{\theta}{n+\theta-1}\left(\sum_{i=1}^{d} \frac{n_i}{n} p_{ii} q(\mathbf{n}) + \sum_{i,j \in \{1, ..., d\},\, n_j > 0,\, i \neq j} \frac{n_i + 1}{n} p_{ij} q(\mathbf{n} + \mathbf{e}_i - \mathbf{e}_j)\right)$$
$$+ \frac{n-1}{n+\theta-1} \sum_{j \in \{1, ..., d\},\, n_j > 0} \frac{n_j - 1}{n-1} q(\mathbf{n} - \mathbf{e}_j), \tag{3}$$

where $\{\mathbf{e}_i\}$ are the d unit vectors. Boundary conditions are required to determine the solution to (3). These have the form

$$q(\mathbf{e}_i) = \pi_i^*, \qquad i = 1, ..., d, \tag{4}$$

where π_i^* is the probability that the most recent common ancestor is of type i. It is common to assume that

$$\pi_i^* = \pi_i, \qquad i = 1, ..., d, \tag{5}$$

where $\pi = (\pi_1, ..., \pi_d)$ is the stationary distribution of P. With this assumption, $q(\mathbf{n})$ is the stationary sampling distribution.

Equation (3) has been derived by a number of authors in various forms, among them Sawyer, *et al.* (1987), and Lundstrom (1990). Lundstrom *et al.* (1992b) use a coalescent approach and point out that there is a unique solution to these equations. Recursion in (3) is on n, the sample size. Given $\{q(\mathbf{m}); m<n\}$, simultaneous equations for the $\binom{n+d-1}{d-1}$ unknown probabilities $\{q(\mathbf{m}); m=n\}$ are non-singular, and in theory can be solved. In practice a numerical solution is difficult because of the large number of equations.

To derive (3) consider the first event back in time that happened in the ancestral tree. Relative rates of mutation and coalescence for n genes are $n\theta/2 : n(n-1)/2$, so the probability that the first event is a mutation is $\theta/(n+\theta-1)$. To obtain a configuration of $\mathbf{n}$ after mutation the configuration before must be either $\mathbf{n}$, and a transition $i \to i$ takes place for some $i=1, ..., d$ (the mutation resulted in no observable change), or $\mathbf{n}+\mathbf{e}_i-\mathbf{e}_j$, $i, j \in \{1, ..., d\}$, $n_j>0$, $i \neq j$ and a transition $i \to j$ takes place. If a coalescence was the first event back in time, then to obtain a configuration $\mathbf{n}$ the configuration must be $\mathbf{n}-\mathbf{e}_j$ for some $j \in \{1, ..., d\}$ with $n_j>0$ and the ancestral lines involved in the coalescence must be of type j.

It is worth emphasizing that the probability $q(\mathbf{n})$ satisfying (3) is determined solely by the rate matrix R given in (1). Indeed, (3) can be written in the form

$$q(\mathbf{n})=\frac{2}{n(n-1)}\left(\sum_{i=1}^{d} n_i r_{ii} q(\mathbf{n})+\sum_{i,j \in \{1, ..., d\},\, n_j>0,\, i\neq j} (n_i+1)\, r_{ij} q(\mathbf{n}+\mathbf{e}_i-\mathbf{e}_j)\right)$$
$$+\frac{1}{n-1}\sum_{j \in \{1, ..., d\},\, n_j>0} (n_j-1)\, q(\mathbf{n}-\mathbf{e}_j). \tag{6}$$

The point here is that different combinations of θ and P can give rise to the same R matrix. Nonetheless, we prefer to think of the model in terms of an overall rate θ and a matrix of substitution probabilities P. In practice, we often assume that P is known, and the aim is then to estimate the single parameter θ, which reflects both the effective population size N and the mutation probability u.

With initial distribution (5), the quantity $q(\mathbf{n})$ also has an interpretation as being the sampling distribution of a sample of n genes taken from a population with allele frequencies $(X_1, ..., X_d)$ distributed according to the stationary distribution in a diffusion process with state space $\{\mathbf{x} \in [0, 1]^d : \sum_1^d x_i=1\}$ and generator

$$L=\frac{1}{2}\sum_{i=1}^{d}\sum_{j=1}^{d} x_i(\delta_{ij}-x_j)\frac{\partial^2}{\partial x_i \partial x_j}+\sum_{j=1}^{d}\left(\sum_{i=1}^{d} x_i r_{ij}\right)\frac{\partial}{\partial x_j}. \tag{7}$$

That is,

$$q(\mathbf{n})=\frac{n!}{n_1! \cdots n_d!}\,\mathbb{E}(X_1^{n_1} \cdots X_d^{n_d}). \tag{8}$$

It is straightforward to show that

$$\mathbb{E}(LX_1^{n_1}\cdots X_d^{n_d})=0 \tag{9}$$

leads to the recursion (3).

The stationary distribution corresponding to L is known only when P has identical rows, $\mathbf{p}$. Defining $\boldsymbol{\varepsilon}=\theta\mathbf{p}$, Wright (1949) showed that the stationary distribution is Dirichlet, with density

$$\frac{\Gamma(\theta)}{\Gamma(\varepsilon_1)\cdots\Gamma(\varepsilon_d)}x_1^{\varepsilon_1-1}\cdots x_d^{\varepsilon_d-1} \tag{10}$$

on the set $0<x_1, ..., x_d<1, x_1+\cdots+x_d=1$, and

$$q(\mathbf{n})=\frac{n!}{n_1!\cdots n_d!}\frac{(\varepsilon_1)_{(n_1)}\cdots(\varepsilon_d)_{(n_d)}}{(\theta)_{(n)}}, \tag{11}$$

where $a_{(b)}=a(a+1)\cdots(a+b-1)$ for a non-negative integer b. If $d=2$, the distribution $q(\mathbf{n})$ can always be written in the form (11), possibly after rescaling P and θ.

3. Calculating Sample Probabilities by Simulation

A straightforward way to simulate a stationary sample of n genes is the following method, developed in Ethier and Griffiths (1987), Ethier and Kurtz (1992), and Griffiths (1989).

(i) Begin with two genes of identical type chosen from the stationary distribution π.

(ii) When there are m genes, $2\leqslant m\leqslant n$, in the intermediate simulation process choose one at random from the m. Then, with probability $(m-1)/(m+\theta-1)$ add another gene of the chosen type making $m+1$ genes, or with probability $\theta/(m+\theta-1)$, if the type chosen is i, make a transition of type j with probability p_{ij}.

(iii) Stop when there are $n+1$ genes, and delete the last duplicated gene to form a sample of n.

To simulate samples with an arbitrary type for the common ancestor, replace π in step (i) above with the appropriate initial distribution. The coalescent process in this form has been used to simulate the sampling distribution of sequence statistics. See Lundstrom (1990), Lundstrom *et al.* (1992a, b) for applications to the analysis of mitochondrial DNA sequence data. However, it is difficult to calculate the probability of a particular sample configuration in a multiple allele model by naive simulation of the coalescent, primarily because of the large state space. Exact calculation is possible using the recursion (3) and Lundstrom (1990) gives a useful computational method for this. It is very computer intensive and again not practical for larger sample sizes or larger state spaces.

The method presented here provides a new simulation technique for direct calculation of the probability of a sample configuration by simulation backwards along the sample paths of the coalescent.

Scaling the transition probabilities for the coalescent process gives a representation for $q(\mathbf{n})$ moving back along the tree. Let

$$\begin{aligned}
a(\mathbf{n}) &= \frac{\theta}{n(n+\theta-1)} \sum_{i,j\in\{1,\ldots,d\},\, n_j>0,\, i\neq j} (n_i+1)p_{ij} \\
&\quad + \frac{1}{n+\theta-1} \sum_{j\in\{1,\ldots,d\},\, n_j>0} (n_j-1), \\
b(\mathbf{n}) &= 1-\frac{\theta}{n(n+\theta-1)} \sum_{i=1}^{d} n_i p_{ii}, \\
f(\mathbf{n}) &= a(\mathbf{n})/b(\mathbf{n}), \qquad n\geqslant 2, \\
f(\mathbf{e}_i) &= q(\mathbf{e}_i), \qquad i=1,2,\ldots,d, \\
\lambda_{ij}(\mathbf{n}) &= \frac{\theta(n_i+1)}{n(n+\theta-1)\,a(\mathbf{n})} p_{ij}, \qquad i,j\in\{1,\ldots,d\}, \quad n_j>0, \quad i\neq j, \\
\mu_j(\mathbf{n}) &= \frac{n_j-1}{(n+\theta-1)\,a(\mathbf{n})}, \qquad j\in\{1,\ldots,d\}, \quad n_j>0.
\end{aligned} \tag{12}$$

The recursive equations (3) for the distribution $q(\mathbf{n})$ can now be written in the form

$$\begin{aligned}
q(\mathbf{n}) = f(\mathbf{n}) \Bigg(& \sum_{i,j\in\{1,\ldots,d\},\, n_j>0,\, i\neq j} \lambda_{ij}(\mathbf{n})\, q(\mathbf{n}+\mathbf{e}_i-\mathbf{e}_j) \\
& + \sum_{j\in\{1,\ldots,d\},\, n_j>0} \mu_j(\mathbf{n})\, q(\mathbf{n}-\mathbf{e}_j) \Bigg),
\end{aligned} \tag{13}$$

for $n\geqslant 2$, with boundary conditions (4).

In order to exploit this form of the recursion, we consider a Markov chain $\{\mathrm{N}(k), k\geqslant 0\}$ whose state space is $S=\{0,1,\ldots\}^d$, and whose transition mechanism is determined by

$$\begin{aligned}
&\mathbf{n}\to\mathbf{n}+\mathbf{e}_i-\mathbf{e}_j, \text{ with probability } \lambda_{ij}(\mathbf{n}), \quad i,j\in\{1,\ldots,d\}, \quad n_j>0, \quad i\neq j, \\
&\mathbf{n}\to\mathbf{n}-\mathbf{e}_j, \text{ with probability } \mu_j(\mathbf{n}), \quad j\in\{1,\ldots,d\}, \quad n_j>0,
\end{aligned} \tag{14}$$

when $n\geqslant 2$. Note that by construction for all $\mathbf{m}\in S$

$$\sum_{i,j\in\{1,\ldots,d\},\, m_j>0,\, i\neq j} \lambda_{ij}(\mathbf{m}) + \sum_{j\in\{1,\ldots,d\},\, m_j>0} \mu_j(\mathbf{m}) = 1.$$

Denoting the degree of $\mathbf{m}$ by $|\mathbf{m}| = \sum_{j=1}^{d} m_j$, we see that $\{|\mathbf{N}(k)|, k \geqslant 0\}$ is non-increasing. For each $m \geqslant 1$, define the set of states

$$S_m = \{\mathbf{m} \in S : |\mathbf{m}| = m\}.$$

For any $m \geqslant 2$, S_m is finite. If at least one element in $\mathbf{m} \in S_m$ is greater than unity, then the probability of exit to S_{m-1} from $\mathbf{m}$ is non-zero. If at least two entries in $\mathbf{m}$ are unity, then because P is regular and S_m finite, with probability one the process will hit a state in S_m, $m \geqslant 2$ with at least one entry greater than 1. It follows that from any state in S_m, $m \geqslant 2$ the process will exit into S_{m-1} with probability one. Thus the states $\{\mathbf{e}_i\}$ are the only absorbing states.

Let $\{\mathbf{N}(0), ..., \mathbf{N}(\tau)\}$ be the sample path of the process beginning at $\mathbf{N}(0) = \mathbf{n}$, until absorption in the set $A = \{\mathbf{e}_i\}$ at the random time τ. Then

$$q(\mathbf{n}) = \mathbb{E}_n \prod_{j=0}^{\tau} f(\mathbf{N}(j)), \tag{15}$$

the expected value of the product along the sample path. In order to verify (15), we use the following elementary result about Markov chains.

LEMMA 1. *Let $\{X_k; k \geqslant 0\}$ be a Markov chain with state space S and transition matrix P. Let A be a set of states for which the hitting time*

$$\eta \equiv \eta_A = \inf\{k \geqslant 0 : X_k \in A\}$$

is finite with probability one starting from any state $x \in T \equiv S \backslash A$. Let $f \geqslant 0$ be a function on S, and define

$$u_x(f) = \mathbb{E}_x \prod_{k=0}^{\eta} f(X_k) \tag{16}$$

for all $X_0 = x \in S$, so that

$$u_x(f) = f(x), \; x \in A.$$

Then for all $x \in T$

$$u_x(f) = f(x)\left(\sum_{y \in T} p_{xy} u_y(f) + \sum_{y \in A} p_{xy} f(y)\right) \equiv f(x) \sum_{y \in S} p_{xy} u_y(f). \tag{17}$$

Proof.

$$\begin{aligned} u_x(f) &= \mathbb{E}_x\left(\prod_{k=0}^{\eta} f(X_k)\right) \\ &= f(x)\, \mathbb{E}_x\left(\prod_{k=1}^{\eta} f(X_k)\right) \end{aligned}$$

$$
\begin{aligned}
&= f(x)\,\mathbb{E}_x\left(\mathbb{E}_x\left(\prod_{k=1}^{\eta} f(X_k)\right)\Big| X_1\right)\\
&= f(x)\,\mathbb{E}_x\left(\mathbb{E}_{X_1}\left(\prod_{k=1}^{\eta} f(X_k)\right)\right) \qquad \text{(by the Markov property)}\\
&= f(x)\left(\sum_{y\in T} p_{xy}u_y(f) + \sum_{y\in A} p_{xy}f(y)\right)\\
&= f(x)\sum_{y\in S} p_{xy}u_y(f).
\end{aligned}
$$

If $u_x(f)=\infty$ then both sides of (17) are infinite. ▌

The representation (15) follows from Lemma 1 by taking $A=S_1=\{\mathbf{m}\in S : |\mathbf{m}|=1\}$ and $\eta=\tau$, the hitting time of S_1. This representation provides an easy way to calculate an estimate $\hat{q}(\mathbf{n})$ by simulating the process $\{\mathrm{N}(k), k\geqslant 0\}$ repeatedly starting at $\mathrm{N}(0)=\mathbf{n}$ and taking the estimate to be the average of $\prod_{j=0}^{\tau} f(\mathrm{N}(j))$ over the realizations.

In the context of (12)–(15), $\{X_k; k\geqslant 0\}$ is the Markov chain $\{\mathrm{N}(k); k\geqslant 0\}$, and P in the lemma is constructed from $\{\{\lambda_{ij}(\mathbf{m})\}, \{\mu_j(\mathbf{m})\} : \mathbf{m}\in S\}$. The sum over absorbing states A in (17) only arises in (13) when $\mathbf{n}=2\mathbf{e}_j$ for some $j=1, ..., d$. Since $f(\mathbf{e}_j)=q(\mathbf{e}_j)$,

$$
q(2\mathbf{e}_j)=f(2\mathbf{e}_j)\left(\sum_{i\in\{1,...,d\},\, i\neq j} \lambda_{ij}(2\mathbf{e}_j)\, q(\mathbf{e}_i+\mathbf{e}_j)+\mu_j(2\mathbf{e}_j) f(\mathbf{e}_j)\right).
$$

The uniqueness of the solution to (3) and so (13) implies that (15) is a correct representation with finite expectation.

4. Some K-Allele Examples

To illustrate the simulation method, we begin by studying the simplest case, the so-called K-allele model. This section serves as motivation for the techniques developed in later sections.

4.1. *Variance Reduction*

It is well known that variance reduction techniques are an important aspect of any simulation study. One obvious method of variance reduction is to replace quantities that might be estimated by their exact values. This idea can be applied in the present context by simulation of the process $\{\mathrm{N}(k); k\geqslant 0\}$ until the hitting time η of $S_2=\{\mathbf{n}\in S : |\mathbf{n}|=2\}$, and then by direct calculation of $q(\mathbf{N}(\eta))$. Lemma 1 provides a representation equivalent to (15) in the form

$$q(\mathbf{n}) = \mathbb{E}_{\mathbf{n}}\left(\prod_{j=0}^{\eta-1} f(\mathbf{N}(j))\right) q(\mathbf{N}(\eta)). \tag{18}$$

An estimate $\hat{q}(\mathbf{n})$ is then found by averaging $q(\mathbf{N}(\eta)) \prod_{j=0}^{\eta-1} f(\mathbf{N}(j))$ over repeated runs. The same idea can be used to stop at the earlier time η when S_r is first visited. The cases $r=2$ and $r=3$ are particularly useful in practice.

It remains to calculate the probabilities $q(\mathbf{n})$ for small values of $r=|\mathbf{n}|$. A numerical integration method for doing this when $r=2$ or $r=3$ is described in detail in Section 5.1. To illustrate the methods, the next section considers a particular example in which the appropriate values of $q(\mathbf{n})$ are known explicitly from Wright's formula (11).

4.2. *A 4-Allele Model*

If P has identical rows $(d^{-1}, ..., d^{-1})$, the sampling distribution is known to be (11) with $\varepsilon_i = \theta/d$, $i = 1, ..., d$. Table I provides a comparison of exact and simulation results for the naive simulation method (corresponding to $\eta =$ time to hit S_1 in (18)) for the case $d=4$. The 95% confidence intervals are based on the simulation variance, each estimate using 20,000 runs. Notice that all of the intervals include the true sample probability.

The next simulations test the variance reduction approach described in the previous section. We apply the method by stopping the simulations at $r=2$ and $r=3$, and using the explicit formula (11) for the requisite values of $q(\mathbf{n})$. The results are given in Tables II and III, all runs being based on 20,000 replicates. The estimated standard errors are, by and large, decreasing as r increases.

TABLE I

Simulation for $d=4$: Equal Mutation Rates

θ	$\mathbf{n}$	$q(\mathbf{n})$	$\hat{q}(\mathbf{n})$	95% C.I.	Scale
0.5	(20, 15, 10, 5)	8.45	8.73	(7.78, 9.67)	10^{-7}
0.5	(50, 30, 15, 5)	2.06	1.92	(1.75, 2.10)	10^{-7}
1.0	(20, 15, 10, 5)	4.10	4.07	(3.93, 4.21)	10^{-6}
1.0	(50, 30, 15, 5)	9.14	8.96	(8.62, 9.30)	10^{-7}
2.0	(20, 15, 10, 5)	1.54	1.57	(1.54, 1.59)	10^{-5}
2.0	(50, 30, 15, 5)	2.87	2.90	(2.85, 2.96)	10^{-6}
5.0	(20, 15, 10, 5)	5.52	5.54	(5.41, 5.66)	10^{-5}
5.0	(50, 30, 15, 5)	6.19	6.32	(6.17, 6.46)	10^{-6}
10.0	(20, 15, 10, 5)	9.61	9.64	(8.33, 10.94)	10^{-5}
10.0	(50, 30, 15, 5)	4.92	4.67	(4.09, 5.26)	10^{-6}

TABLE II

Simulated Probabilities for Sample (20, 15, 10, 5)

θ	$q(\mathbf{n})$	r	$\hat{q}(\mathbf{n})$	SE.	Scale
0.5	8.45	1	8.31	0.299	10^{-7}
		2	8.61	0.315	
		3	8.14	0.219	
1.0	4.10	1	3.94	0.062	10^{-6}
		2	4.09	0.059	
		3	4.12	0.059	
2.0	1.54	1	1.55	0.013	10^{-5}
		2	1.52	0.010	
		3	1.53	0.009	
5.0	5.52	1	5.50	0.064	10^{-5}
		2	5.56	0.052	
		3	5.46	0.041	
10.0	9.61	1	9.68	0.480	10^{-5}
		2	9.42	0.351	
		3	9.52	0.283	

TABLE III

Simulated Probabilities for Sample (50, 30, 15, 5)

θ	$q(\mathbf{n})$	r	$\hat{q}(\mathbf{n})$	SE	Scale
0.5	2.06	1	2.18	0.126	10^{-7}
		2	1.85	0.067	
		3	2.09	0.078	
1.0	9.14	1	8.81	0.178	10^{-7}
		2	9.19	0.165	
		3	8.93	0.149	
2.0	2.87	1	2.83	0.027	10^{-6}
		2	2.90	0.022	
		3	2.85	0.019	
5.0	6.19	1	6.43	0.075	10^{-6}
		2	6.09	0.058	
		3	6.26	0.050	
10.0	4.92	1	5.31	0.453	10^{-6}
		2	5.67	0.777	
		3	4.81	0.228	

5. The Finite-Sites model

It is of particular importance to develop the simulation method for a sample of n sequences of length s, as in example (v) of Section 2. The large state space can cause problems with a long waiting time until absorption in S_1 for the process $\{\mathbf{N}(k); k \geqslant 0\}$. This section describes some techniques which can be used to provide variance reduction, and sometimes reduce simulation time as well.

5.1. *Variance Reduction*

Empirically, the waiting times in states satisfying $|\mathbf{N}(k)| = m$ increase as m decreases. As an illustration, we consider the case of a sample of two sequences. Suppose that the sites are homogeneous, there are two possible types at each site, and the transition matrix P for base changes has identical elements of 0.5.

Let $D(k)$ be the number of segregating sites between the two sequences at step k of the Markov process governed by the transition probabilities (12), with the convention that when $|\mathbf{N}(k)| = 1$, $D(k) = \Delta$, a cemetery state.

$\{D(k)\}$ is a modified random walk with transitions $d \to d+1$, $d \to d-1$, $(d \geqslant 1)$ having respective probabilities, $1 - d/s$, d/s, and transitions $0 \to 1$, $0 \to \Delta$ having respective probabilities $\theta/(2+\theta)$, $2/(2+\theta)$. Let μ_d be the expected waiting time from $D(0) = d$ until absorption into Δ. It follows that $\mu_0 = \theta 2^{s-1} + 1$, and for $d \geqslant 1$,

$$\mu_d = \theta 2^{s-1} + 1 + \sum_{j=1}^{d} \binom{s-1}{j-1}^{-1} \sum_{i=j}^{s} \binom{s}{i}.$$

Clearly these expected waiting times are very long for reasonable values of the sequence length s.

Long waiting times in $S_r = \{\mathbf{n} \in S : |\mathbf{n}| = r\}$ can be avoided by simulation of the process $\{\mathbf{N}(k); k \geqslant 0\}$ until the hitting time η of S_r, and then by direct calculation of $q(\mathbf{N}(\eta))$, as described in Eq. (18). For the sequence case, calculation of the probabilities $q(\mathbf{n})$ for $|\mathbf{n}| = r$ is only practical for small r. For $r = 2, 3$ and initial conditions (5), they may be calculated in the following way.

Let $X_k(t)$ be the type of site k at time t, measured from the common ancestor of the n sequences in the coalescent tree. For $k = 1, ..., s$, $\{X_k(t), t \geqslant 0\}$ is a Markov process with an infinitesimal matrix $(\theta/2)\, h_k(P_k - I)$ in the notation of example (iv) of Section 2. Denote the transition probabilities of $\{X_k(t), t \geqslant 0\}$ by $\{p_{ij}^k(t)\}$, and the stationary distribution by π^k. The stationary distribution can be calculated from P_k.

The mutation processes at the sites 1, ..., s are independent, conditional on the structure of the coalescent tree. This can be used to calculate the probabilities $q(\mathbf{n})$ for small $|\mathbf{n}|$ by considering all possible trees. Suppose $|\mathbf{n}| = 2$, and the two sequences are of types $\mathbf{i} = (i_1, ..., i_s)$, $\mathbf{j} = (j_1, ..., j_s)$. By considering that the common ancestor of site k has type a with probability π_a^k, and mixing over the time to coalescence, we obtain

$$q(\mathbf{n}) = (2 - \delta_{\mathbf{ij}}) \int_0^\infty \exp(-t) \prod_{k=1}^{s} \sum_a \pi_a^k p_{ai_k}^k(t) p_{aj_k}^k(t) dt. \tag{19}$$

If $P_1, ..., P_s$ are reversible, (19) simplifies to

$$q(\mathbf{n}) = (2 - \delta_{\mathbf{ij}}) \int_0^\infty \exp(-t) \prod_{k=1}^{s} \pi_{i_k}^k p_{i_k j_k}^k(2t) dt. \tag{20}$$

The integrals in (19) and (20) can be evaluated by numerical integration.

Similar formulae can be derived for $|\mathbf{n}| = 3$ for sequences of type $\mathbf{i}, \mathbf{j}, \mathbf{m}$. The times while there are two and three ancestors of the sample are exponential with rates 1 and 3 respectively. Let $r(\mathbf{i}, \mathbf{j}, \mathbf{m})$ be the probability of an ordered sample of the three sequences, where sequences $\mathbf{j}$ and $\mathbf{m}$ coalesce first. Then

$$r(\mathbf{i}, \mathbf{j}, \mathbf{m}) = \int_0^\infty \int_0^\infty 3e^{-s-3t} \prod_{k=1}^{s} \sum_a \pi_a^k p_{ai_k}^k(s+t) \times \sum_b p_{ab}^k(s) p_{bj_k}^k(t) p_{bm_k}^k(t) ds dt. \tag{21}$$

If $P_1, ..., P_s$ are reversible, then (21) simplifies to

$$r(\mathbf{i}, \mathbf{j}, \mathbf{m}) = \int_0^\infty \int_0^\infty 3e^{-s-3t} \prod_{k=1}^{s} \pi_{i_k}^k \times \sum_b p_{i_k b}^k(2s+t) p_{bj_k}^k(t) p_{bm_k}^k(t)\, ds\, dt. \tag{22}$$

The probability of an unordered sample of the three sequences is then

$$q(\mathbf{n}) = \begin{cases} r(\mathbf{i}, \mathbf{j}, \mathbf{m}) & \mathbf{i} = \mathbf{j} = \mathbf{m} \\ r(\mathbf{i}, \mathbf{j}, \mathbf{j}) + 2r(\mathbf{j}, \mathbf{j}, \mathbf{i}) & \mathbf{i} \neq \mathbf{j} = \mathbf{m} \\ 2(r(\mathbf{i}, \mathbf{j}, \mathbf{m}) + r(\mathbf{j}, \mathbf{i}, \mathbf{m}) + r(\mathbf{m}, \mathbf{j}, \mathbf{i})) & \mathbf{i} \neq \mathbf{j} \neq \mathbf{m}. \end{cases} \tag{23}$$

In practice, using the form (19) greatly reduces run times in the simulation, and also reduces the variance of the estimate $\hat{q}(\mathbf{n})$, compared to not doing any calculation. Use of the form (21) also provides a useful variance reduction technique, but the time taken by the algorithm can be dramatically longer due to the numerical integration required. Some examples are given in Section 6.1.

The results in (19) and (21) have obvious analogs under the initial conditions (4). In particular, for a fixed ancestor of type $(a_1, a_2, ..., a_s)$, we have

$$q(\mathbf{n}) = (2-\delta_{\mathbf{ij}})\int_0^\infty \exp(-t)\prod_{k=1}^{s} p^k_{a_k i_k}(t)\,p^k_{a_k j_k}(t)\,dt, \tag{24}$$

and

$$\begin{aligned} r(\mathbf{i}, \mathbf{j}, \mathbf{m}) = \int_0^\infty \int_0^\infty 3e^{-s-3t} \prod_{k=1}^{s} p^k_{a_k i_k}(s+t) \\ \times \sum_b p^k_{a_k b}(s)\,p^k_{b j_k}(t)\,p^k_{b m_k}(t)\,dsdt. \end{aligned} \tag{25}$$

These equations do not simplify with reversibility of the transition matrix P.

With a fixed ancestral type, it is more efficient to simulate to a sample of size two and then calculate, since every run will then give an estimate from that ancestral type.

5.2. *The Number of Mutations in the Data*

It is of interest to compute the probability $\tilde{q}(\mathbf{n}, m)$ that a sample of size n has configuration $\mathbf{n}$ and exactly m mutations that change the state of a gene in the line back to the common ancestor. The analog of Eq. (3) for $\tilde{q}$ is the recursion

$$\begin{aligned} \tilde{q}(\mathbf{n}, m) = \frac{\theta}{\theta+n-1}\sum_{i=1}^{d}\frac{n_i}{n}p_{ii}\tilde{q}(\mathbf{n}, m) \\ + \frac{\theta}{\theta+n-1}\sum_{i,j\,:\,n_j>0,\,i\neq j}\frac{n_i+1}{n}p_{ij}\tilde{q}(\mathbf{n}+\mathbf{e}_i-\mathbf{e}_j, m-1) \\ + \frac{n-1}{\theta+n-1}\sum_{j\,:\,n_j>0}\frac{n_j-1}{n-1}\tilde{q}(\mathbf{n}-\mathbf{e}_j, m). \end{aligned} \tag{26}$$

The distribution

$$q(\mathbf{n}, b) = \sum_{m=0}^{b}\tilde{q}(\mathbf{n}, m), \qquad b = 0, 1, ...$$

satisfies the recursion (26), with boundary conditions

$$q(\mathbf{n}, 0) = 0 \qquad \text{unless} \quad \mathbf{n} = n\mathbf{e}_j \text{ for some } j. \tag{27}$$

The initial conditions are (cf. (4))

$$q(\mathbf{e}_k, b) = \pi_k^*, \qquad k = 1, 2, ..., d; \quad b = 0, 1, ...$$

It is elementary to prove from (26) that

$$q(n\mathbf{e}_k, 0) = \left[\prod_{l=1}^{n-1} \frac{l}{l + \theta(1 - p_{kk})}\right] q(\mathbf{e}_k, 0). \tag{28}$$

The recursion in (26) can be solved by constructing a new process $\{(\mathbf{N}(k), B(k)), k = 0, 1, ...\}$ which, recalling (14), makes transitions as follows:

$$\begin{aligned} &(\mathbf{n}, b) \to (\mathbf{n} + \mathbf{e}_i - \mathbf{e}_j, b - 1), \text{ with probability } \lambda_{ij}(\mathbf{n}), \\ &\qquad 1 \leqslant i, j \leqslant d, n_j > 0, i \neq j, \\ &(\mathbf{n}, b) \to (\mathbf{n} - \mathbf{e}_j, b), \text{ with probability } \mu_j(\mathbf{n}), \qquad 1 \leqslant j \leqslant d, n_j > 0, \end{aligned} \tag{29}$$

when $n \geqslant 2, b \geqslant 1$. If γ denotes the time taken to reach a state of the form $(\mathbf{e}_i, a), a \geqslant 0$; $(m\mathbf{e}_j, 0)$ for some $j \in \{1, 2, ..., d\}$ and $m \geqslant 2$; or $(\mathbf{n}_0, 0)$, where $\mathbf{n}_0$ is not of the form $m\mathbf{e}_j$ and $q(\mathbf{n}_0, 0) = 0$, then the appropriate functional to evaluate is given by

$$q(\mathbf{n}, b) = \mathbb{E}_{\mathbf{n}, b}\left[\prod_{k=0}^{\gamma-1} f(\mathbf{N}(k), B(k))\right] q(\mathbf{N}(\gamma), B(\gamma)), \tag{30}$$

where the $f(\mathbf{n}, m) \equiv f(\mathbf{n})$ for all m. Fortunately, there is a simply way to avoid simulating this new process. Using the boundary conditions in (27) and (28), we have the equivalent representation

$$q(\mathbf{n}, b) = \mathbb{E}_{\mathbf{n}}\left[\prod_{j=0}^{\tau} f(\mathbf{N}(j))\right] I\{M \leqslant b\}, \tag{31}$$

where M is the number of times the process $\{\mathbf{N}(k), k = 0, 1, ...\}$ makes a move of the type $\mathbf{n} \to \mathbf{n} + \mathbf{e}_i - \mathbf{e}_j, i \neq j$, and τ is the time taken to hit S_1.

Aside from its intrinsic interest, this result also provides a way to reduce the simulation time required to estimate $q(\mathbf{n})$. A technique to avoid the problem of long runs is to abort a run if the number M of mutation events exceeds a bound b, and return a value $q(\mathbf{n}, b) = 0$ for that run. The average over all runs, including the zeros, will then be an estimate of $q(\mathbf{n}, b)$. If b is of reasonable size, then the tail probability $\sum_{m=b+1}^{\infty} \tilde{q}(\mathbf{n}, m) = q(\mathbf{n}) - q(\mathbf{n}, b)$ makes a negligible contribution to $q(\mathbf{n})$. The probability of observing greater than b mutation events in a sample path of $\{\mathbf{N}(k), k = 0, 1, ...\}$ can be quite high, while $q(\mathbf{n}) - q(\mathbf{n}, b)$, being the expected value of a functional on the sample paths, is negligible.

Ideally, we would like to choose b so that the relative error

$$\frac{q(\mathbf{n})-q(\mathbf{n}, b)}{q(\mathbf{n})}=\mathbb{P}(M>b \mid \mathbf{n})$$

is small. If there are a huge number of segregating sites, then choosing a reasonable bound b may also be impossible.

The simulation can also be modified by considering the number of birth-death events until $|\mathbf{N}(\eta)|=r$, terminating runs when this number is too large, then calculating $q(\mathbf{N}(\eta))$ for runs which are not aborted in a similar way to (18).

6. Testing the Simulation Method for Sequences

6.1. *A Variance Reduction Method*

Section 5.1 provides a method for combining simulation along a sample path with explicit evaluation of certain integrals. This example verifies that this does indeed provide a useful variance reduction technique. The 10 sequences of length 10 were simulated from a model with $\theta=2.0$ and transition matrix

$$P=\begin{pmatrix} 0.5 & 0.5 \\ 0.1 & 0.9 \end{pmatrix}.$$

The sequences and their frequencies are:

2 2 2 2 2 2 2 2 2 2	(2)
2 2 2 1 2 2 2 2 2 2	(6)
2 2 2 1 2 2 2 1 2 2	(2)

TABLE IV

Variance Reduction

Method	Estimated probability	SE
1[a]	7.447×10^{-6}	1.058×10^{-6}
2[b]	7.386×10^{-6}	8.274×10^{-7}
3[c]	7.041×10^{-6}	6.592×10^{-7}

[a] No integration.

[b] Numerical integration from two ancestors.

[c] Numerical integration from three ancestors.

In Table IV, we compare the effects of the explicit integration approach applied when there are two and three sequences remaining in the simulation, and the naive approach which simulates back to a single common ancestor. The simulations, each based on 20,000 replicates, confirm that the estimated variance is indeed reduced. The case $r=3$ takes considerably longer than the cases $r=1$ and 2 because of the time to perform the numerical integration in (22).

6.2. *Mutations in the Backwards Process*

To illustrate the effects of truncating runs that have "too many" mutations in their paths back to the common ancestor, we use the data from Section 6.1 once more. In Table V are the estimates and relative timings for 20,000 runs of the (otherwise naive) simulation method described in (31), for different values of b. In all cases, the value of θ is 2.0.

As expected, the method reduces simulation time dramatically. The variability of the estimates appears to be increasing as the cut-off level b decreases, as would be expected. Notice, though, that the 95%confidence intervals for the underlying probability overlap each other.

The estimated probabilities in Table V are not monotone in the value of b, as they should be from the representation (31). This is because the runs for different values of b are not coupled. To investigate the way in which the estimates depend on the value of b, we kept all runs and looked through the results for various values of b. The examples illustrate what happens for different values of θ. The first is based on the data of Section 6.1 where $\theta=2.0$, the second on (simulated) data with $\theta=10.0$ and sequences

1 2 2 2 2 2 2 2 2 1	(1)
1 2 2 2 1 2 2 2 2 1	(7)
1 2 2 1 1 2 2 2 2 1	(1)
1 2 2 2 1 2 2 2 1 2	(1)

TABLE V

Truncating Runs with Many Mutations

Truncation level b	Estimated probability	SE	Relative time
5	1.047×10^{-5}	1.960×10^{-6}	1.0
10	9.589×10^{-6}	1.877×10^{-6}	1.6
20	5.202×10^{-6}	0.907×10^{-6}	2.3
50	7.464×10^{-6}	1.140×10^{-6}	4.8
∞	7.885×10^{-6}	1.304×10^{-6}	7.5

and the third on $\theta = 15.0$ and sequences

2 1 2 2 2 2 1 1 1 2 (2)
1 1 2 2 1 2 2 2 1 2 (1)
1 1 2 2 1 2 2 2 2 2 (6)
2 2 2 2 2 2 1 1 1 2 (1)

In Table VI are the coupled estimates of $q(\mathbf{n}, b)$ for different values of b, together with the number of simulation runs (out of 20,000) that did not exceed b mutations.

TABLE VI

Coupled Estimates of $q(\mathbf{n}, b)$

	Runs with $\leqslant b$ mutations	$\hat{q}(\mathbf{n}, b)$	SE
b	$\theta = 2.0$	$(\times 10^6)$	$(\times 10^6)$
2	91	6.05	1.22
5	147	7.33	1.35
7	155	7.33	1.35
10	166	7.34	1.35
20	285	7.34	1.35
50	3943	7.34	1.35
25000	20000	7.34	1.35
	$\theta = 10.0$	$(\times 10^9)$	$(\times 10^9)$
2	0	0.00	0.00
4	4	6.53	3.51
6	6	6.75	3.52
10	9	6.96	3.52
20	22	6.96	3.52
50	1273	6.96	3.52
25000	20000	6.96	3.52
	$\theta = 15.0$	$(\times 10^{19})$	$(\times 10^{19})$
20	0	0.0	0.0
30	32	2.40	1.89
40	166	2.41	1.89
70	2093	2.41	1.89
150	13039	2.41	1.89
200	17126	2.41	1.89
25000	20000	2.41	1.89

In the first example we are interested in the case $b=2$, which gives the probability of getting the two segregating sites from only two mutations and no further back mutations. The estimated conditional probability that the sequences were formed by just two mutations is $6.05/7.34=0.82$. In the second example, $b=4$ is the minimal number of mutations required to produce the four segregating sites. Many other possible mutational paths can produce the data, but only 9 of the 20,000 runs are effectively contributing to the estimate of $q(\mathbf{n})$! In the final example, a huge number of mutations can occur, but only 166 contribute to the estimate of $q(\mathbf{n})$. While the *coupled* simulations are not effective in reducing simulation time in the algorithm, they do give insight into the distribution of mutational events that produce the sample.

7. Simulating Likelihood Surfaces

One of the main statistical uses for the distribution $q(\mathbf{n})$ is estimation of parameters using maximum likelihood methods. One way to do this is to simulate the likelihood *independently* at a grid of points, and examine the shape of the resulting surface. In practice, this can be a very time consuming approach, particularly in the sequence case. In this section, we describe an approach, akin to importance sampling, for estimating a likelihood surface at a grid of points using just one run of the simulation algorithm.

The method uses the following lemma in the spirit of Lemma 1.

Lemma 2. *Let $\{X_k; k \geqslant 0\}$ be a Markov chain with state space S and transition matrix P. Let A be a set of states for which the hitting time*

$$\eta \equiv \eta_A \equiv \inf\{k \geqslant 0 : X_k \in A\}$$

is finite with probability one starting from any state $x \in T \equiv S - A$. Let $h \geqslant 0$ be a given function on A, let $f \geqslant 0$ be a function on $S \times S$ and define

$$u_x(f) = \mathbb{E}_x h(X_\eta) \prod_{k=0}^{\eta-1} f(X_k, X_{k+1}) \tag{32}$$

for all $X_0 = x \in S$, so that

$$u_x(f) = h(x), \qquad x \in A.$$

Then for all $x \in T$

$$u_x(f) = \sum_{y \in T} f(x, y)\, p_{xy} u_y(f) + \sum_{y \in A} f(x, y)\, p_{xy} h(y) \tag{33}$$

$$= \sum_{y \in S} f(x, y)\, p_{xy} u_y(f). \tag{34}$$

Proof. The proof follows that of Lemma 1 and is omitted. ▮

Remark. When $f(x, y) = f(x)$ for all y, and $h(x) = f(x)$, $x \in A$, Lemma 2 reduces to Lemma 1.

In Section 3 we applied Lemma 1 with $\{X_k, k \geqslant 0\}$ being the Markov chain $\{\mathbf{N}(k), k \geqslant 0\}$ whose transition matrix was determined by (12) and (14) and η was the time taken to reach the set $S_r \equiv \{\mathbf{m} \in S : |\mathbf{m}| = r\}$. We are interested in calculating the probability $q(\mathbf{n})$ for different values of the parameter $\Theta \equiv \{\theta; p_{ij}, 1 \leqslant i, j \leqslant d\}$ in (12). Note that for a particular value of Θ we solve a system of the form

$$q(\mathbf{n}) = \sum_{\mathbf{m}} f_\Theta(\mathbf{n})\, p_\Theta(\mathbf{n}, \mathbf{m})\, q(\mathbf{m}) \tag{35}$$

where the transition matrix $p_\Theta(\mathbf{n}, \mathbf{m})$ is determined by (14), and $f_\Theta(\mathbf{n})$ is given in (12). Now suppose that Θ_0 is a particular set of parameters that satisfies

$$\{(\mathbf{n}, \mathbf{m}) : p_{\Theta_0}(\mathbf{n}, \mathbf{m}) = 0\} \subseteq \{(\mathbf{n}, \mathbf{m}) : p_\Theta(\mathbf{n}, \mathbf{m}) = 0\}.$$

We can write the equations (35) in the form

$$q(\mathbf{n}) = \sum_{\mathbf{m}} f_\Theta(\mathbf{n}) \frac{p_\Theta(\mathbf{n}, \mathbf{m})}{p_{\Theta_0}(\mathbf{n}, \mathbf{m})} p_{\Theta_0}(\mathbf{n}, \mathbf{m})\, q(\mathbf{m}) \tag{36}$$

so that from Lemma 2

$$q(\mathbf{n}) = \mathbb{E}_{\mathbf{n}}\, q(\mathbf{N}(\eta)) \prod_{j=0}^{\eta-1} f(\mathbf{N}(j), \mathbf{N}(j+1)), \tag{37}$$

where $\{\mathbf{N}(k), k \geqslant 0\}$ is the Markov chain with parameters Θ_0 and

$$f(\mathbf{n}, \mathbf{m}) = f_\Theta(\mathbf{n}) \frac{p_\Theta(\mathbf{n}, \mathbf{m})}{p_{\Theta_0}(\mathbf{n}, \mathbf{m})}. \tag{38}$$

It follows that $q(\mathbf{n})$ can be calculated from the realizations of a single Markov chain, by choosing a sensible value of Θ_0 to drive the simulations, and evaluating the functional $q(\mathrm{N}(\eta))\prod_{j=0}^{\eta-1} f(\mathrm{N}(j), \mathrm{N}(j+1))$ along the sample path for each of the different values of Θ of interest.

7.1. *Estimating the Mutation Rate θ*

One particularly important special case of this procedure involves estimation of θ, the mutation rate. Recall that $\theta = 4Nu$ where u is the mutation probability per gene per generation and N is the effective diploid population size. If u can be estimated from other information then N can be estimated, and vice-versa. θ is one of the standard parameters in population genetics, and there are numerous ways to estimate it for different models of mutation. Recent references, using methods different from the present ones, include Lundstrom et al. (1992a, b) and Felsenstein (1992a).

If the mutation matrix $\{p_{ij}\}$ in (12) is assumed known, then θ is the only parameter of interest, and we shall write $\Theta = \theta$, $\Theta_0 = \theta_0$ in what follows. From Eqs. (12) and (38) we see that for $|\mathbf{n}| = n$

$$\begin{aligned} f(\mathbf{n}, \mathbf{m}) &= \frac{a_{\theta_0}(\mathbf{n})\, \theta(n+\theta_0-1)}{b_\theta(\mathbf{n})\, \theta_0(n+\theta-1)} \quad \text{if} \quad \mathbf{m} = \mathbf{n} + \mathbf{e}_i - \mathbf{e}_j \\ &= \frac{a_{\theta_0}(\mathbf{n})(n+\theta_0-1)}{b_\theta(\mathbf{n})(n+\theta-1)} \quad \text{if} \quad \mathbf{m} = \mathbf{n} - \mathbf{e}_j. \end{aligned} \tag{39}$$

7.2. *A K-Allele Example*

Suppose then that we want to estimate the mutation rate θ. There are several issues that need to be addressed, among them sensible choices for θ_0 and an assessment of the variability and time dependence of the method. In this section, we use the surface method for the four allele model discussed in Section 4.2 for this purpose. We concentrate on the test data described in Tables I and III. The data for 100 genes had a configuration of $\mathbf{n} = (50, 30, 15, 5)$. In Table VII we give the results of 20,000 naive simulation runs to estimate $q(\mathbf{n})$ for $\theta = 0.5$, 1.0, 2.0 for different values of θ_0.

It can be seen from Table VII that the accuracy of the method worsens as θ moves away from θ_0. In particular the estimate for the combination

TABLE VII

Surface Simulation of Probabilities for Sample (50, 30, 15, 5)

	$q(\mathbf{n})$	$\theta_0 = 0.5$ $\hat{q}(\mathbf{n})$(SE)	$\theta_0 = 1.0$ $\hat{q}(\mathbf{n})$(SE)	$\theta_0 = 2.0$ $\hat{q}(\mathbf{n})$(SE)	Scale
$\theta = 0.5$	2.06	2.25 (0.139)	1.95 (0.058)	2.26 (0.178)	10^{-7}
$\theta = 1.0$	9.14	9.17 (0.606)	9.01 (0.182)	9.67 (0.298)	10^{-7}
$\theta = 2.0$	2.87	2.11 (0.168)	2.76 (0.090)	2.91 (0.027)	10^{-6}

TABLE VIII

Surface Simulation of Log-Likelihoods for Sample (50, 30, 15, 5)

θ	$\log q(\mathbf{n})$	$\log \hat{q}(\mathbf{n})$	θ	$\log q(\mathbf{n})$	$\log \hat{q}(\mathbf{n})$	θ	$\log q(\mathbf{n})$	$\log \hat{q}(\mathbf{n})$
1.0	−13.91	−14.15	4.0	−12.08	−12.09	5.5	−11.975	−11.967
2.0	−12.76	−12.73	4.5	−12.03	−12.05	5.6	−11.973	−11.964
3.0	−12.30	−12.28	5.0	−11.99	−12.02	5.7	−11.973	−11.963
4.0	−12.08	−12.09	5.5	−11.98	−12.00	5.8	−11.972	−11.962
5.0	−11.99	−12.00	6.0	−11.97	−11.98	5.9	−11.972	−11.961
6.0	−11.98	−11.98	6.5	−11.98	−12.00	6.0	−11.973	−11.962
7.0	−12.00	−12.00	7.0	−12.00	−12.01	6.1	−11.973	−11.962
8.0	−12.05	−12.04	7.5	−12.02	−12.03	6.2	−11.975	−11.963
9.0	−12.13	−12.08	8.0	−12.05	−12.06	6.3	−11.978	−11.965
10.0	−12.22	−12.11	8.5	−12.09	−12.10	6.4	−11.978	−11.967

Note. All results based on 50,000 simulations, with $\theta_0 = 6.0$.

$\theta = 2.0$, $\theta_0 = 0.5$ seems to be quite biased. The diagonal entries of Table VII should be compared to the corresponding entries for $r = 1$ in Table III. In these examples, the simultaneous estimation method takes about 40% of the time of the independent method. This difference in timing will become much more pronounced in cases (such as sequence data) where the time taken to calculate the appropriate functionals is low compared to the time taken to simulate the random trajectory. On the other hand, there seems to be a trade off in the accuracy of the estimates. A more accurate surface should result from longer runs.

The method seems to work well when one is trying to estimate probabilities in a neighborhood of the special value θ_0. This is precisely the case when trying to find maximum likelihood estimates. As an example, we have simulated the log-likelihood surface for the data set $\mathbf{n} = (50, 30, 15, 5)$ using 50,000 simulations of the method that stops simulating when S_3 is reached, and compared it to the true surface. The results are given in Table VIII

The results seem particularly encouraging. In addition to taking approximately 25% of the time for independent simulations, the method produces a maximum likelihood estimate of $\hat{\theta} = 6.0$ for the simulations along the grids 4.0(0.5)8.5 and 1(1)10 with $\theta_0 = 6.0$ in both cases. The more refined grid of 5.5(0.1)6.4 produced the estimate $\hat{\theta} = 5.9$, which is indeed the true MLE.

7.3. *A Sequence Example*

This section uses the simulated sequence data from Section 6.1. The surface simulation based on 50,000 runs on the grid of θ values 1.0(0.5)18.0 with $\theta_0 = 1.0, 9.0, 18.0$ gave the surfaces in Fig. 1.

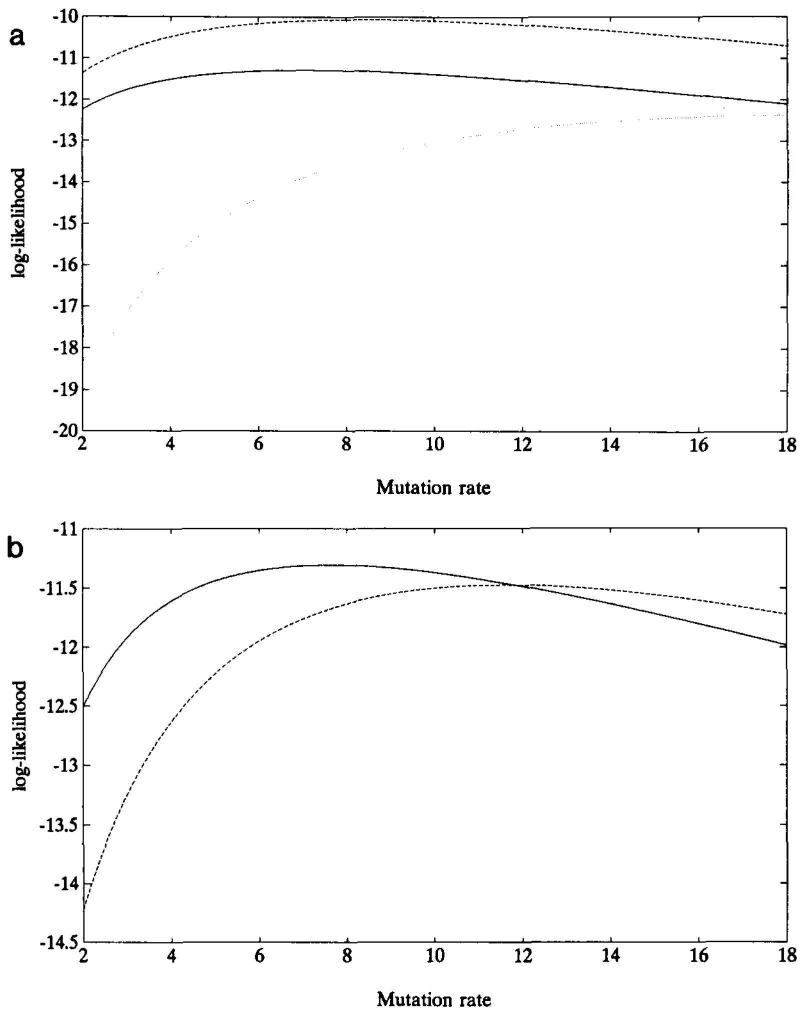

FIG. 1. Log-likelihoods in θ for different values of θ_0. (a) Solid line $\theta_0 = 1.0$; dashed line $\theta_0 = 9.0$; dotted line $\theta_0 = 18.0$. (b) dashed line formed from average at each grid point. Solid line formed from weighted average at each grid point.

In Fig. 1a, we plot the three estimates of the likelihood surface for different test values θ_0. Each of these values of θ_0 provides an unbiased estimator of the true underlying log-likelihood at the grid of θ-values, but the variances of the estimates depend markedly on θ_0. There are several ways in which these curves might be combined to produce a composite estimate of the true underlying log-likelihood curve. For a given value of θ, we weighted the estimates for each value of θ_0 inversely proportional to their (estimated) variances. This method should work well when correlations among estimates for different values of θ are not too high, as is indeed the case here. These weighted averages were used to construct the composite curve plotted in Fig. 1b. Using this curve, we find that the MLE of θ is $\hat{\theta} \approx 7.0$.

Approximate confidence intervals for parameter estimates can be found in several ways. A parametric bootstrap approach would simulate many samples using the estimated parameter values as the true parameters, and look at the observed distribution of the resulting estimates. This approach can be extremely time consuming. An alternative is to use the likelihood surface directly: include θ^* in the interval if one cannot reject the hypothesis that $\theta = \theta^*$ at the appropriate level. This test is based on the (putative, asymptotic in n) χ^2 distribution of the likelihood ratio test statistic. For the example at hand, the log-likelihood at $\theta^* = 2$ is -12.5, compared to the value of about -11.3 at $\hat{\theta}$, so we can include $\theta = 2$ (the value used to generate the data) in the confidence interval.

7.4. *A Simulated Sample of Sequences*

A simulation of 50 sequences of length 20 with two possible bases at each site, $\theta = 1.0$, and the probability of change 0.5 for each base provided the following sequences:

1 2 1 1 2 1 2 1 2 1 1 2 1 2 1 2 1 1 2 2 (13)
1 2 1 1 2 1 2 1 2 1 1 2 1 1 1 2 1 1 2 2 (21)
1 2 1 1 2 1 2 1 2 1 1 2 1 1 1 1 1 1 2 2 (16).

The results below show the behaviour of the surface method when the runs are truncated when more than 30 mutations occur in the backwards process. The estimates in Table IX are based on 10,000 runs of the naive method, with a cut-off of $b = 30$, and $\theta_0 = 1.0$. From the table, we see that the MLE of θ is $\hat{\theta} = 0.80$.

TABLE IX

Log-Likelihoods Found by Truncating at $b = 30$

θ	Likelihood	SE
0.2	1.02×10^{-13}	2.42×10^{-14}
0.4	2.47×10^{-13}	5.92×10^{-14}
0.6	3.45×10^{-13}	8.28×10^{-14}
0.8	3.86×10^{-13}	9.29×10^{-14}
1.0	3.85×10^{-13}	9.28×10^{-14}
1.2	3.58×10^{-13}	8.64×10^{-14}
1.4	3.19×10^{-13}	7.70×10^{-14}
1.6	2.75×10^{-13}	6.64×10^{-14}
1.8	2.32×10^{-13}	5.61×10^{-14}

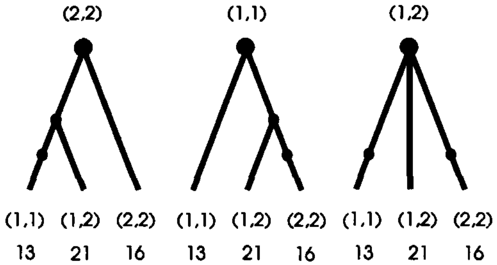

FIG. 2. Possible trees for the example in Section 7.4.

There are only two segregating sites in the sequences. The sample is consistent with being from the infinitely-many-sites model where there is no back-mutation. If this is so, there are only three possible mutation trees for the two segregating sites, as shown in Fig. 2. The third tree is the most likely.

The vertices represent mutations in these trees. Probabilities of the trees under the infinitely-many-sites model can be calculated from a computer implementation of a recursion of Griffiths (1989), and are shown in Table X. A comparison value of $\theta/2$ is chosen because in the finitely-many-sites model from which the simulated data come, the rate of base changes is $\theta/4$, compared to the infinitely-many-sites model where it is $\theta/2$. In Table X, the column headed L_{inf} is the sum of the tree probabilities in the preceding three columns, using rate $\theta/2$, whereas the column headed $L_{\text{fin}}(\theta)$ is $20 \times 19 \times 2^{20} \times$ likelihood for the finitely-many-sites model with rate θ. Thus $L_{\text{fin}}(\theta)$ is the probability of an unordered, unlabelled configuration of sites. The maximum likelihood under the infinitely-many-sites model occurs at $\theta/2 = 0.4$, at exactly the same point $\theta = 0.8$ as in the finite-sites model.

TABLE X

Calculated Tree Probabilities

$\theta/2$	Tree 1	Tree 2	Tree 3	L_{inf}	$L_{\text{fin}}(\theta)$
0.1	5.08×10^{-6}	3.81×10^{-6}	1.92×10^{-5}	2.81×10^{-5}	4.06×10^{-5}
0.2	1.20×10^{-5}	9.02×10^{-6}	4.64×10^{-5}	6.78×10^{-5}	9.84×10^{-5}
0.3	1.63×10^{-5}	1.22×10^{-5}	6.40×10^{-5}	9.25×10^{-5}	1.37×10^{-4}
0.4	1.77×10^{-5}	1.32×10^{-5}	7.09×10^{-5}	1.02×10^{-4}	1.54×10^{-4}
0.5	1.72×10^{-5}	1.28×10^{-5}	7.00×10^{-5}	1.00×10^{-4}	1.53×10^{-4}
0.6	1.56×10^{-5}	1.15×10^{-5}	6.44×10^{-5}	9.15×10^{-5}	1.43×10^{-4}
0.7	1.36×10^{-5}	9.98×10^{-6}	5.67×10^{-5}	8.03×10^{-5}	1.27×10^{-4}
0.8	1.14×10^{-5}	8.39×10^{-6}	4.84×10^{-5}	6.82×10^{-5}	1.10×10^{-4}
0.9	9.45×10^{-6}	6.90×10^{-6}	4.04×10^{-5}	5.68×10^{-5}	9.24×10^{-5}

8. Discussion

There are now many examples of Monte Carlo methods designed for the approximation of probabilities deriving from complex stochastic systems. These often come under the heading of Markov chain Monte-Carlo (MCMC) techniques, which are designed to simulate observations from densities known up to a normalising constant. Convenient references include Hastings (1970), Geyer and Thompson (1992), Smith and Roberts (1993) and Besag and Green (1993); see in particular the discussion of the last three papers. In contrast, our sampling probabilities are not known just up to a normalising constant. Rather, we have based our method for computing likelihoods in the coalescent on simulation of Markov chains whose transition mechanism is determined by recurrence equations satisfied by certain sampling probabilities. The idea behind our method is rather different from "conventional" MCMC, but, in common with MCMC, it has a long history. The technique of using simulation in a Markov chain with an absorbing state to solve a system of linear equations is one of the oldest techniques in the Monte-Carlo field, dating back at least to Forsythe and Leibler (1950). The method is described, together with a "matrix multiplication" proof of the analog of Lemma 1, in Section 5.1.3 of Rubinstein (1981) and Section 7.3 of Ripley (1987) for example.

8.1. *Computational Aspects*

A computer implementation of the simulation methods described in this paper is available, in portable C code, from the authors. The program SEQUENCE computes the likelihood of a sample of sequences when there is a possibly inhomogeneous mutation structure along bases, with a given transition matrix for base changes. This corresponds to the model in (2) with possibly different h_j, but identical P_j. The distribution of allele frequencies at a single locus is the special case of one base.

The implementation also includes options for

- simulation back to the most recent common ancestor;
- simulation back to 2 or 3 ancestors, then calculation of the sample probability. (Calculation is restricted to reversible transition matrices);
- setting a bound on the number of mutations;
- computing likelihoods when the most recent common ancestor is of a given type;
- variable rates at different positions (with fixed P matrix);
- simulation of likelihood surfaces, with the parameter set Θ_0 containing values of either θ, the mutation parameter; or P, the mutation matrix; or $\{h_j\}$, the collection of site mutation probabilities.

A computational problem with small likelihoods is underflow, so in each run instead of the product in (15), $\sum_{j=0}^{\tau} \log(f(\mathbf{N}(j)))$ is calculated, then a simple extended precision routine is used to find the mean of the exponentials of the log sums. In calculating likelihoods for sequences many runs contribute effectively zero to the mean. This is expected because many of the large number of possible sample paths have a tiny probability in the actual coalescent process.

There are several computational issues that we are working on to increase the speed with which evolutionary parameters can be estimated from very large DNA sequence data sets. The approach based on stopping runs with "too many" mutations seems particularly promising. We are also assessing the behavior of the algorithm for use on supercomputers and we are developing efficient methods for running the surface simulation algorithm. One possible approach to the search for a maximum likelihood estimator is an adaptive search method, in which the test value Θ_0 is changed during the course of the simulation. Another approach that may prove useful is to use the recursions to derive further recursions satisfied by appropriate derivatives of the sampling probabilities, and estimate these simultaneously with the sampling probabilities themselves.

We are also exploring the uses of recursions for *renormalized* sampling probabilities of the form $\tilde{q}(\mathbf{n}) = q(\mathbf{n})\, c(\mathbf{n})$ for some sequence of constants $c(\mathbf{n})$. This provides a way to avoid combinatorial evaluations, and may also be useful in reducing the variance of the estimators.

8.2. *Other Applications*

The methods developed in this paper can be applied to many other sampling problems arising in population genetics. The key ingredient in the application of these techniques is the derivation of a sampling equation analogous to (3). These may be derived by coalescent methods or directly from diffusion equations, as illustrated in (7)–(9). For the infinitely-many-sites process, the analog of Eq. (3) is given by Ethier and Griffiths (1987), and exploited further by Griffiths (1989). In Griffiths and Tavaré (1993), we have developed the sampling theory for the infinitely-many-sites process when the ancestral type of each site is unknown, and provided a method for true likelihood estimation of θ. Another important example is the two locus model with recombination, for which the basic equations are derived by Ethier and Griffiths (1990a, b) and Griffiths (1991). The technique deseribed here may then be used to find maximum likelihood estimators of the recombination fraction for a variety of different mutation models.

Other cases include models with selection (where the diffusion method comes more fully into play), and processes that incorporate the effects of variable population size, migration, and subdivision. Modifications of the basic scheme may also be used to study problems in ancestral inference. For example, it provides a way to estimate the mean time to the most recent common ancestor of a sample, conditional on the types observed in the sample. This is a quantity of some current interest in the study of human evolution. Finally, it should be clear that the method is not restricted to applications in population genetics, but rather provides another computer-intensive approach to the study of likelihoods that arise from many other Markov processes.

REFERENCES

BESAG, J., AND GREEN, P. J., (1993). Spatial statistics and Bayesian computation, *J. R. Statist. Soc. B* **55**, 25–38.

ETHIER, S. N., AND GRIFFITHS, R. C. (1987). The infinitely-many-sites model as a measure valued diffusion, *Ann. Probab.* **15**, 515–545.

ETHIER, S. N., AND GRIFFITHS, R. C. (1990a). The neutral two-locus model as a measure-valued diffusion, *Adv. Appl. Prob.* **22**, 773–786.

ETHIER, S. N., AND GRIFFITHS, R. C. (1990b). On the two locus sampling distribution. *J. Math. Biol.* **29**, 131–159.

ETHIER, S. N., AND KURTZ, T. G. (1992). On the stationary distribution of the neutral diffusion model in population genetics, *Ann. Appl. Prob.* **2**, 24–35.

EWENS, W. J. (1972). The sampling theory of selectively neutral alleles, *Theoret. Popul. Biol.* **3**, 87–112.

EWENS, W. J. (1979). "Mathematical Population Genetics," Springer-Verlag, New York.

FELSENSTEIN, J. (1992a) Estimating effective population size from samples of sequences: inefficiency of pairwise and segrating sites as compared to phylogenetic estimates, *Genet. Res. Camb.* **59**, 139–147.

FELSENSTEIN, J. (1992b). Estimating effective population size from samples of sequences: A bootstrap Monte Carlo approach, *Genet. Res. Camb.* **60**, 209–220.

FORSYTHE, S. E., AND LEIBLER, R. A. (1950). Matrix inversion by a Monte Carlo method, *Math. Tables other Aids Comput.* **4**, 127–129.

GEYER, C. J., AND THOMPSON, E. A. (1992). Constrained Monte Carlo maximum likelihood for dependent data (with discussion), *J. R. Statist. Soc. B* **54**, 657–700.

GRIFFITHS, R. C. (1989). Genealogical-tree probabilities in the infinitely-many site model, *J. Math. Biol.* **27**, 667–680.

GRIFFITHS, R. C. (1991). The two-locus ancestral graph, *in* "Selected Proceedings of the Symposium on Applied Probability, Sheffield, 1989," IMS Lecture Notes, Monographs Series (I. V. Basawa and R. L. Taylor, Eds.), Vol. 18, pp. 100–117.

GRIFFITHS, R. C., AND TAVARÉ, S. (1993). Inference for the infinitely-many-sites model, submitted.

HASTINGS, W. K. (1970). Monte Carlo sampling methods using Markov chains and their applications, *Biometrika* **57**, 97–109.

HUDSON, R. R. (1991). Gene genealogies and the coalescent process, *in* "Oxford Surveys in Evolutionary Biology" (D. Futuyama and J. Antonovics, Eds.), Vol. 7, pp. 1–44.

KINGMAN, J. F. C. (1982a). On the genealogy of large populations, *J. Appl. Prob. A* **19**, 27–43.

KINGMAN, J. F. C. (1982b). The coalescent, *Stochastic Processes Appl.* **13**, 235–248.

LUNDSTROM, R. (1990). "Stochastic models and statistical methods for DNA sequence data," Ph.D. thesis, Department of Mathematics, University of Utah.

LUNDSTROM, R., TAVARÉ, S., AND WARD, R. H. (1992a). Estimating mutation rates from molecular data using the coalescent, *Proc. Natl. Acad. Sci. USA* **89**, 5961–5965.

LUNDSTROM, R., TAVARÉ, S., AND WARD, R. H. (1992b). Modelling the evolution of the human mitochondrial genome, *Math. Biosci.* **112**, 319–335.

RIPLEY, B. D. (1987). "Stochastic Simulation," Wiley, New York.

RUBENSTEIN, R. Y. (1981). "Simulation and the Monte-Carlo Method," Wiley, New York.

SAWYER, S., DYKHUIZEN, D., AND HARTL, D. (1987). Confidence interval for the number of selectively neutral amino acid polymorphisms, *Proc. Natl. Acad. Sci. USA* **84**, 6225–6228.

SMITH, A. F. M., AND ROBERTS, G. O. (1993). Bayesian computation via the Gibbs sampler and related Markov chain Monte Carlo methods, *J. R. Statist. Soc. B* **55**, 3–24.

TAVARÉ, S. (1984). Line-of-descent and genealogical processes, and their applications in population genetics models, *Theoret. Popul. Biol.* **26**, 119–164.

WRIGHT, S. (1949). Adaptation and selection, *in* "Genetics, Paleontology, and Evolution" (G. L. Jepson, E. Mayr, and G. G. Simpson, Eds.), pp. 365–389. Princeton Univ. Press, Princeton, NJ.

Printed by Catherine Press, Ltd., Tempelhof 41, B-8000 Brugge, Belgium