Published in final edited form as:

J Comput Graph Stat. 2017 ; 26(1): 182–194. doi:10.1080/10618600.2016.1159212.

Efficient computation of the joint sample frequency spectra for multiple populations

John A. Kamm,

Department of Statistics, University of California, Berkeley

Jonathan Terhorst, and

Department of Statistics, University of California, Berkeley

Yun S. Song

Departments of EECS, Statistics, and Integrative Biology, University of California, Berkeley

Abstract

A wide range of studies in population genetics have employed the sample frequency spectrum (SFS), a summary statistic which describes the distribution of mutant alleles at a polymorphic site in a sample of DNA sequences and provides a highly efficient dimensional reduction of large-scale population genomic variation data. Recently, there has been much interest in analyzing the joint SFS data from multiple populations to infer parameters of complex demographic histories, including variable population sizes, population split times, migration rates, admixture proportions, and so on. SFS-based inference methods require accurate computation of the expected SFS under a given demographic model. Although much methodological progress has been made, existing methods suffer from numerical instability and high computational complexity when multiple populations are involved and the sample size is large. In this paper, we present new analytic formulas and algorithms that enable accurate, efficient computation of the expected joint SFS for thousands of individuals sampled from hundreds of populations related by a complex demographic model with arbitrary population size histories (including piecewise-exponential growth). Our results are implemented in a new software package called *mom* (MOran Models for Inference). Through an empirical study we demonstrate our improvements to numerical stability and computational complexity.

Keywords

coalescent; population genetics; demographic inference; sum-product algorithm

1 Introduction

Massive increases in the availability of DNA sequence data are creating exciting new opportunities for discovery in evolutionary genetics. With those opportunities come new challenges, as algorithms and statistical models originally designed to analyze small amounts of DNA from a limited number of individuals are now called on to study thousands to tens of thousands of individuals at a time. As in other branches of statistics, the arrival of “big data” has led to a new focus on scalability, accuracy, speed, and related computational issues within statistical genetics.

In this paper, we tackle this challenge by presenting a new method for computing the expected joint sample frequency spectrum (SFS), a summary statistic on DNA sequences which lies at the core of a large number of empirical investigations and inference procedures in population genetics (Bhaskar et al., 2015; Coventry et al., 2010; Excoffier et al., 2013; Gazave et al., 2014; Gravel et al., 2011; Griffiths and Tavaré, 1998; Gutenkunst et al., 2009; Jenkins et al., 2014; Nelson et al., 2012; Nielsen, 2000; Wakeley and Hey, 1997). As we discuss in greater technical detail below, the joint SFS is of interest because it maps complex demographic models involving population size changes, population splits, migration, and admixture to a low-dimensional vector. This dimensionality reduction is useful for performing inference using sampled DNA sequences. Inferring population demographic histories is not only innately interesting, for example in dating events such as the out-of-Africa migration of modern humans (Gutenkunst et al., 2009; Schaffner et al., 2005), but is also important for biological applications, such as distinguishing between the effects of natural selection and demography (Beaumont and Nichols, 1996; Boyko et al., 2008).

We argue both theoretically and by simulation that existing methods in population genetics cannot scale to the problem sizes encountered in modern genetic analyses. Our primary contribution is a novel algorithm (and accompanying open-source software package) which is several orders of magnitude faster (in the number of sampled individuals n) than existing approaches. Moreover, by careful algorithmic design we are able to mitigate certain numerical issues (e.g., underflow and catastrophic cancellation) that often arise when applying these models to data. The combined effect of these innovations is to permit the analysis of much larger data sets, which will lead to improved inference in population genetics.

The rest of the paper is organized as follows: In Section 2, we describe the problem more precisely, survey related work, and summarize our main results. Section 3 presents the theoretical results that lead to the improved algorithm described in Section 4. Runtime and numerical accuracy results are discussed in Section 5. Mathematical proofs of our theoretical results are deferred to Section 6.

Software availability

The algorithms presented in this paper are implemented in a new software package called *momi* (MOran Models for Inference), which is freely available at <https://github.com/popgenmethods/momi>.

2 Background and summary

In this section, we briefly review some terminology and concepts from population genetics that are relevant to our problem. We then discuss existing work and summarize our main results.

2.1 Motivation

To simplify the exposition, suppose that the genomes of n individuals have been randomly sampled from a single population. (We will shortly consider the general case where genomes are sampled from multiple related populations, which is substantially more challenging to analyze and is the main focus of this paper.) The positions in the genome where at least one,

but fewer than all, of the sampled individuals bears a mutant allele (or genetic type) are called *segregating* sites. Assume that all such sites are dimorphic, meaning that we can label each sampled individual at each segregating site as either “ancestral” or “derived”. The *sample frequency spectrum* is defined to be the count vector $\hat{\xi} \in \mathbb{Z}_+^{n-1}$, of which the i th entry $\hat{\xi}_i$ records the number of segregating sites that have exactly i copies of the mutant allele in the sample. (Thus, $\|\hat{\xi}\|_1$ equals the total number of segregating sites found in the sampled genomes.)

The SFS is of interest because its distribution encodes past demographic events experienced by the population(s) from which the samples were obtained. In particular, suppose we are analyzing a single population and let $\alpha^{-1}(t) : \mathbb{R}_{\geq 0} \rightarrow \mathbb{R}_{>0}$ be a strictly positive function such that $\alpha^{-1}(t)$ equals the *effective* size (Ewens, 2004) of the population at t time units back in the past. (The inverse exponent is included because the function $\alpha(t)$ is used below to describe the rate at which a pair of individuals in the population find a common ancestor or *coalesce*; intuitively, this is more probable when the population is small.) There is a well-defined map

$$\alpha^{-1} \mapsto \xi(\alpha^{-1}) = \mathbb{E}_{\alpha^{-1}} \hat{\xi} \quad (1)$$

which takes population size histories to their expected frequency spectra. Here, the notation $\mathbb{E}_{\alpha^{-1}}$ denotes expectation with respect to the distribution on genomes induced by α^{-1} . (For illustrative purposes we prefer to define this map abstractly for now; a concrete instantiation is given in Section 3.3).

Although the above map is not injective for an arbitrary function (Myers et al., 2008), a general sufficient condition for injectivity has been found recently (Bhaskar and Song, 2014). For most classes of size functions of interest to population geneticists (for example, piecewise-exponential), the map is injective provided that the sample size n is sufficiently large. Hence, this leads to the inverse problem of estimating a size function $\hat{\alpha}^{-1}(\cdot)$ given the sample statistic $\hat{\xi}$. This machinery may be generalized to more complex demographic scenarios involving multiple populations which split or merged from one another, migration between populations, and other features, although we note that identifiability for these more general models has yet to be established.

The map (1) cannot in general be analytically inverted, so numerical methods are required to recover α^{-1} . In order to numerically solve this inverse problem, it is necessary to be able to evaluate (1) rapidly and accurately.

2.2 Existing work

Two approaches exist for computing the forward mapping (1). One is based on the so-called Wright-Fisher diffusion (Ewens, 2004), a stochastic process which describes the frequency trajectory of a mutant allele segregating in a population as time moves forward. The diffusion framework has the advantage of being applicable to arbitrary demographic models, but its computational complexity grows exponentially with the number of populations. Also, it requires numerically solving a system of partial differential equations, which can be

difficult in practice. For these reasons, current implementations (Gravel et al., 2011; Gutenkunst et al., 2009; Luki and Hey, 2012) of the diffusion approach are limited to analyzing no more than four populations at once.

An alternative class of methods, which includes ours, relies on a backward-in-time stochastic process called the coalescent (Kingman, 1982a,b,c), which is dual to the Wright-Fisher diffusion. Intuitively, the coalescent describes the genealogical structure of a sample of chromosomes; below we provide a more detailed description of the coalescent since it is an integral part of our method. Using the coalescent, one computes the expected SFS by integrating over all genealogies underlying the sample. This can be done either via Monte Carlo or analytically. Monte Carlo integration (Nielsen, 2000) can effectively handle arbitrary demographic histories with a large number of populations, and Excoffier et al. (2013) have recently developed a useful implementation. However, when the number $\mathcal{D}$ of populations (or demes) is moderate to large, most of the $O(n^{\mathcal{D}})$ SFS entries will be unobserved in simulations, and thus the Monte Carlo integral may naively assign a probability of 0 to observed SNPs. Monte Carlo computation of the expected SFS thus requires careful regularization techniques to avoid degeneracy.

An alternative to the Monte Carlo approach is to compute the expected SFS exactly via analytic integration over coalescent genealogies (Griffiths and Tavaré, 1998; Wakeley and Hey, 1997). For a demography involving multiple populations, this can be done by a dynamic program (Chen, 2012, 2013). This algorithm is more complicated and less general than both the Monte Carlo and diffusion approaches: while it can handle population splits, merges, size changes, and instantaneous gene flow, it is difficult to include continuous gene flow between populations. However, it scales well to a large number $\mathcal{D}$ of populations, since it only computes entries of the SFS that are observed in the data, and ignores the $O(n^{\mathcal{D}})$ SFS entries that are not observed.

2.3 Summary of our main results

Unfortunately, existing coalescent-based algorithms (Chen, 2012, 2013; Wakeley and Hey, 1997) do not scale well to large sample size n , either in terms of running time or numerical stability. These algorithms rely on large alternating sums that explode with n and exhibit catastrophic cancellation. To circumvent these problems, we obtain new results for computing the *truncated SFS*, a key quantity needed to compute the joint SFS for multiple populations. For a fixed time τ , the truncated SFS gives the expected number of mutations arising in the time interval $[0, \tau)$ that are found in $k = 1, 2, \dots, n$ individuals sampled at time 0. We provide an algorithm for computing this quantity efficiently and in a numerically stable manner.

For general demographic histories, the complexity of the dynamic program devised by Chen (2012, 2013) is $O(n^5 V + WL)$, where V is the number of populations (vertices) throughout the history, L is the number of distinct SFS entries to be computed, and W is a term that depends on n and the graph structure of the demography; in this paper, we improve this to $O(n^2 V + WL)$. For the special case of a tree-shaped demography without migration or admixture, Chen's algorithm gives $W = O(n^4 V)$, for a total cost of $O(n^5 V + n^4 VL)$. In this special case (without admixture), we introduce an additional speedup that improves the

runtime to $O(n^3 V + n^2 VL)$. (See Section 4.2.2). We show through an empirical study that our algorithm is not only orders of magnitude faster, but also more numerically stable.

Lastly, we note that our algorithm relies on ideas similar to those found in Bryant et al. (2012) and De Maio et al. (2013), but with a different focus. Those methods aim to compute a “phylogenetic SFS”, in which mutations are allowed to be recurrent and time scales are sufficiently long that population size can be assumed to be constant. In contrast, our method considers an infinite sites model (Kimura, 1969) without recurrent mutation, and can handle arbitrary population size change functions. These features make it more appropriate for use in population genetics.

3 Theoretical results on the truncated SFS

In this section we study the truncated sample frequency spectrum, which is the key object needed to compute the joint SFS in our algorithm. For readers who are unfamiliar with this area, we first give a brief introduction to the key concepts in order to make our results more understandable.

3.1 Background on the coalescent and the SFS

Kingman’s *coalescent* (Kingman, 1982a,b,c) is a partition-valued stochastic process which describes shared ancestry of a group of individuals sampled randomly from a population.

More precisely, the coalescent $\{\mathcal{C}_t^n\}_{t \geq 0}$ on n leaves is the backward-in-time Markov jump process whose value at time t is a partition of $\{1, \dots, n\}$, and at time t , each pairs of blocks

in $\mathcal{C}_t^n$ merge with rate $\alpha(t)$. We also call $\frac{1}{\alpha(t)}$ the *population size history function*. We often drop the dependence on n , and write $\mathcal{C}_t = \mathcal{C}_t^n$. We prefer to denote a dependence on n through the probability $\mathbb{P}_n$ and the expectation $\mathbb{E}_n$. So if $X(\mathcal{C}^n)$ denotes a random variable of the process $\mathcal{C}^n$, we usually write $\mathbb{E}_n[X]$ instead of $\mathbb{E}[X(\mathcal{C}^n)]$.

A sample path of $\{\mathcal{C}_t^n\}_{t \geq 0}$ can be viewed as a rooted ultrametric binary tree with n leaves, labeled $1, \dots, n$, corresponding to sampled individuals. The tree extends backwards in time and each branch represents a partition block of the process such that $\mathcal{C}_t^n$ corresponds to the partition induced on $\{1, \dots, n\}$ by cutting the tree at height t . Merging a pair of blocks after a random amount of time results in a corresponding merger of tree lineages. This process repeats until all individuals have “coalesced” into a single common ancestor.

The coalescent may be used to model the mutation patterns arising in sampled DNA. The basic idea is as follows: each branch in the coalescent tree represents a succession of individuals who are ancestral to the present-day individuals (leaves) they subtend. Now suppose one of those ancestral individuals experiences a mutation at a certain position in her genome. Then, barring mutations at that position in all other parts of the tree, all of her present-day descendants will bear the mutant allele at that position, while all other members of the sample will not. If we suppose that the probability of a mutation occurring on a particular branch of a coalescent tree is proportional to the length of that branch, then this

creates a link between observed mutation data and the distribution of the coalescent tree, and hence between the observed data and $\alpha(t)$.

It turns out that this assumption is accurate for selectively neutral mutations: conditional on the underlying coalescent tree, they are distributed roughly as a Poisson point process occurring at some rate $\theta/2 \ll 1$ on the tree. (The scaling factor $1/2$ is a convention in the literature.) This implies that, given that a mutation is segregating within the sample, it falls uniformly with respect to branch lengths on the coalescent tree of the sampled individuals. Now let $\mathcal{M} \subsetneq \{1, \dots, n\}$ be the set of leaves carrying the mutant allele (we only consider mutations beneath the root, so by assumption $\mathcal{M} \subsetneq \{1, \dots, n\}$). Then we define the sample frequency spectrum $f_n(k)$, for $0 < k < n$, as the first order Taylor series coefficient of $\mathbb{P}_n(|\mathcal{M}| = k)$ in the mutation rate,

$$\mathbb{P}_n(|\mathcal{M}| = k) = \frac{\theta}{2} f_n(k) + o(\theta).$$

We will generally refer to $f_n(k)$ as the sample frequency spectrum (SFS). We also note two alternative definitions of the SFS. First, $f_n(k)$ is the expected number of mutations with k

descendants when $\frac{\theta}{2} = 1$. Second, $\frac{1}{\binom{n}{|K|}} f_n(|K|)$ is the expected length of the branch whose leaf set is $K \subset \{1, \dots, n\}$. More specifically, let $\mathbb{I}$ denote the indicator function, and define $\mathcal{L}_K := \int_0^\infty \mathbb{I}_{K \in \mathcal{C}_t} dt$. Then

$$\frac{1}{\binom{n}{|K|}} f_n(|K|) = \mathbb{E}_n[\mathcal{L}_K].$$

The equivalence of these alternate definitions follows from previous results in Bhaskar et al. (2012); Griffiths and Tavaré (1998); Jenkins and Song (2011).

We now consider truncating the coalescent with mutation at time τ , as illustrated in Figure 1. Let $\mathcal{M}^\tau$ denote the set of leaves under mutations occurring in the time interval $[0, \tau)$. We define the *truncated* SFS $f_n^\tau(k)$ according to

$$\mathbb{P}_n(|\mathcal{M}^\tau| = k) = \frac{\theta}{2} f_n^\tau(k) + o(\theta),$$

where $f_n^\tau(k)$ corresponds to the total expected length of all branches in the time interval $[0, \tau)$ each of which subtends k leaves. Using the truncated SFS $f_n^\tau(k)$ for each population appearing in a demographic history, where τ denotes the length of time a particular population exists, it is possible to compute the joint SFS for multiple related populations (Chen, 2012). In Section 4.1, we describe a dynamic program algorithm for computing the joint SFS for multiple populations related by a complex demography, and the way in which this algorithm uses the truncated SFS $f_n^\tau(k)$.

3.2 Previous work on the truncated SFS

The key challenge is to compute the truncated SFS $f_n^\tau(k)$. Let $A_t^\mathcal{C}$ be the random variable which equals the number of coalescent lineages at time t ancestral to the sample. This is a pure death process which decreases by one each time a coalescence event occurs, until finally reaching the absorbing state 1 when all individuals have found a common ancestor.

More precisely, $A_t^\mathcal{C} = |\mathcal{C}_t|$ and the rate of transition from m to $m-1$ is given by

$\lambda_{m,m-1}^\mathcal{C}(t) = \binom{m}{2} \alpha(t)$. We define a conditional version of the SFS via $f_n^\tau(k|A_\tau^\mathcal{C}=m)$ according to

$$\mathbb{P}_n(|\mathcal{M}^\tau|=k|A_\tau^\mathcal{C}=m) = \frac{\theta}{2} f_n^\tau(k|A_\tau^\mathcal{C}=m) + o(\theta). \quad (2)$$

Here, $f_n^\tau(k|A_\tau^\mathcal{C}=m)$ is the total expected length of all branches each subtending k leaves given that there are m ancestors at time τ .

In what follows we will introduce a recursive method for calculating the truncated SFS. The recursion is in terms of the sample size n so we let $m \leq n$ denote a generic sample size from this point forward. The method of Chen (2012, eq. 5) computes the truncated SFS by summing over the number m of ancestors remaining at time τ :

$$f_n^\tau(k) = \sum_{m=1}^{n-k+1} \mathbb{P}_n(A_\tau^\mathcal{C}=m) f_n^\tau(k|A_\tau^\mathcal{C}=m). \quad (3)$$

The first term in the summand, $\mathbb{P}_n(A_\tau^\mathcal{C}=m)$, can be computed in at least three ways: by numerically exponentiating the rate matrix of $A^\mathcal{C}$, by computing an alternating sum with $O(n)$ terms (Tavaré, 1984), or by solving a recursion described in Section 6.1. We note that the recursion described in Section 6.1 has the advantage of computing all values of

$\mathbb{P}_n(A_\tau^\mathcal{C}=m)$, $m \leq n$, in $O(n^2)$ time.

The second term $f_n^\tau(k|A_\tau^\mathcal{C}=m)$ in the summand of (3) is computed in Chen (2012, eq. 4) as

$$f_n^\tau(k|A_\tau^\mathcal{C}=m) = \sum_{i=m}^n ip_{\nu,i}^{k,1} \mathbb{E}_\nu[T_i^\tau|A_\tau^\mathcal{C}=m], \quad (4)$$

where

$$p_{\nu,i}^{k,j} := \begin{cases} \frac{\binom{k-1}{j-1} \binom{\nu-k-1}{i-j-1}}{\binom{\nu-1}{i-1}}, & \text{if } k \geq j > 0 \text{ and } \nu - k \geq -i - j > 0, \\ 1, & \text{if } j=k=0 \text{ or } i-j=\nu-k=0, \\ 0, & \text{else,} \end{cases}$$

is the transition probability of the Pólya urn model, starting with $i-j$ white balls and j black balls, and ending with $\nu-k$ white balls and k black balls (Johnson and Kotz, 1977), and

$$T_i^\tau := \int_0^\tau \mathbb{I}_{A_t^\mathcal{C}=i} dt$$

is the length of time in $[0, \tau)$ where there are i ancestral lineages to the sample, as illustrated in Figure 1. Chen (2012, eq. 3) provides a formula for the conditional expectation

$\mathbb{E}_\nu[T_i^\tau | A_\tau^\mathcal{C}=m]$ for the case of constant population size, which he later extends (Chen, 2013) to the case of an exponentially growing population. However, these formulas involve an alternating sum with $O(\nu^2)$ terms. Thus, computing $\mathbb{E}_\nu[T_i^\tau | A_\tau^\mathcal{C}=m]$ for every value of i, m, ν , as required to compute $\{f_\nu^\tau(k)\}_{k \leq \nu \leq n}$ with (3) and (4), takes $O(n^5)$ time with these formulas. In addition, large alternating sums are numerically unstable due to catastrophic cancellation (Higham, 2002), and so these formulas require the use of high-precision numerical libraries, further increasing runtime.

3.3 A fast, stable algorithm for computing the truncated SFS

Here, we present a numerically stable algorithm to compute $\{f_\nu^\tau(k) | 1 \leq k \leq \nu \leq n\}$ in $O(n^2)$ time instead of $O(n^5)$ time. Our approach utilizes the following two lemmas:

Lemma 1—*The entry $f_n^\tau(n)$ of the truncated SFS is given by*

$$f_n^\tau(n) = \tau - \sum_{k=1}^{n-1} \frac{k}{n} f_n^\tau(k). \quad (5)$$

Lemma 2—*For all $1 \leq k \leq \nu$, the truncated SFS $f_\nu^\tau(k)$ satisfies the linear recurrence*

$$f_\nu^\tau(k) = \frac{\nu-k+1}{\nu+1} f_{\nu+1}^\tau(k) + \frac{k+1}{\nu+1} f_{\nu+1}^\tau(k+1). \quad (6)$$

We prove Lemma 1 in Section 6.2. We note here that our proof also yields the identity

$\mathbb{E}[T_{\text{MRCA}}] = \sum_{k=1}^{n-1} \frac{k}{n} f_n^\tau(k)$, where T_{MRCA} denotes the time to the most recent common ancestor of the sample; to our knowledge, this formula is new. A proof of Lemma 2 is provided in Section 6.3.

We now sketch our algorithm. For a given n , we show below that all values of $f_n^\tau(k)$, for $1 \leq k < n$, can be computed in $O(n^2)$ time. We then compute $f_n^\tau(n)$ using Lemma 1 in $O(n)$ time. Finally, using $f_n^\tau(k)$ for $1 \leq k < n$ as boundary conditions, Lemma 2 allows us to compute all $f_\nu^\tau(k)$, for $\nu = n-1, n-2, \dots, 2$ and $k = 1, \dots, \nu$, in $O(n^2)$ time.

We now describe how to compute the aforementioned terms $f_n^\tau(k)$, for all $k < n$, in $O(n^2)$ time. We first recall the result of Polanski and Kimmel (2003) which represents the untruncated SFS $f_n(k)$, for $1 \leq k < n-1$, as

$$f_n(k) = \sum_{m=2}^n W_{n,k,m} c_m, \quad (7)$$

where

$$c_m := \mathbb{E}_m[T_m] = \int_0^\infty t \binom{m}{2} \alpha(t) \exp \left[- \binom{m}{2} \int_0^t \alpha(x) dx \right] dt = \int_0^\infty \exp \left[- \binom{m}{2} \int_0^t \alpha(x) dx \right] dt \quad (8)$$

denotes the waiting time to the first coalescence for a sample of size m , and $W_{n,k,m}$ are universal constants that are efficiently computable using the following recursions (Polanski and Kimmel, 2003):

$$W_{n,k,2} = \frac{6}{n+1},$$

$$W_{n,k,3} = 30 \frac{(n-2k)}{(n+1)(n+2)},$$

$$W_{n,k,m+2} = - \frac{(1+m)(3+2m)(n-m)}{m(2m-1)(n+m+1)} W_{n,k,m} + \frac{(3+2m)(n-2k)}{m(n+m+1)} W_{n,k,m+1}, \quad (9)$$

for $2 \leq m < n-2$. The key observation is to note that, in a similar vein as (7), we have:

Lemma 3—*The truncated SFS $f_n^\tau(k)$, for $1 \leq k < n-1$, can be written as*

$$f_n^\tau(k) = \sum_{m=2}^n W_{n,k,m} c_m^\tau, \quad (10)$$

where c_m^τ is a truncated version of (8):

$$c_m^\tau := \mathbb{E}_m[T_m^\tau] = \int_0^\tau \exp \left[- \binom{m}{2} \int_0^t \alpha(x) dx \right] dt. \quad (11)$$

We prove Lemma 3 in Section 6.4. For piecewise-exponential $\alpha(t)$, c_m^τ can be computed explicitly using formulas from Bhaskar et al. (2015). Using (9), we can compute all values of $W_{n,k,m}$ for $1 \leq k \leq n$ and $2 \leq m \leq n$, in $\mathcal{O}(n^2)$ time. Then, using (10), all values of $f_n^\tau(k)$, for $1 \leq k \leq n-1$ can be computed in $\mathcal{O}(n^2)$ time.

Note that the above algorithm not only significantly improves computational complexity, but also resolves numerical issues, since it allows us to avoid computing the expected times

$\mathbb{E}_\nu[T_i^\tau | A_\tau^\mathcal{C} = m]$, which are alternating sums of $\mathcal{O}(n^2)$ terms and are numerically unstable to evaluate for large values of n (say, $n > 100$).

3.4 An alternative formula for piecewise-constant subpopulation sizes

For demographic scenarios with piecewise-constant subpopulation sizes, we present an alternative formula for computing the truncated SFS within a constant piece. This formula has the same sample computational complexity as that described in the previous section.

Let $\mathcal{K}_t$ denote the *coalescent with killing*, a stochastic process that is closely related to the Chinese restaurant process, Hoppe's urn, and Ewens' sampling formula (Aldous, 1985; Hoppe, 1984). In particular, the coalescent with killing $\{\mathcal{K}_t\}_{t \geq 0}$ is a stochastic process whose value at time t is a *marked* partition of $\{1, \dots, n\}$, where each partition block is marked as “killed” or “unkilled”. We obtain the partition for $\mathcal{K}_t$ by dropping mutations onto the

coalescent tree as a Poisson point process with rate $\frac{\theta}{2}$, and then defining an equivalence relation on $\{1, \dots, n\}$, where $i \sim j$ if and only if i, j have coalesced by time t and there are no mutations on the branches between i and j (i.e., i and j are identical by descent). We furthermore mark the equivalence classes (i.e. partition blocks) of $\mathcal{K}_t$ that are descended from a mutation in $[0, t)$ as “killed”. See Figure 2 for an illustration. The process $\mathcal{K}_\tau$ can also be obtained by running Hoppe's urn, or equivalently the Chinese restaurant process, forward in time (Durrett, 2008, Theorem 1.9).

Let $A_\tau^{\mathcal{K}}$ be the number of unkilld blocks in $\mathcal{K}_\tau$, so that $A_\tau^{\mathcal{K}}$ is a pure death process with

transition rate $\lambda_{i,i-1}^{\mathcal{K}}(t) = \binom{i}{2} \alpha(t) + \frac{i\theta}{2}$ (the rate of coalescence is the number of unkilld pairs $\binom{i}{2} \alpha(t)$, and the rate of killing due to mutation is $\frac{i\theta}{2}$). Our next proposition gives a formula for the truncated conditional sample frequency spectrum given $A_\tau^{\mathcal{K}}$, i.e., $f_n^\tau(k | A_\tau^{\mathcal{K}} = m)$.

Proposition 1—Consider the constant population size history $\frac{1}{\alpha(t)} = \frac{1}{\alpha}$ for $t \in [0, \tau]$, and let $m > 0$ and $0 < k \leq n - m$. The joint probability that the number of derived mutants is k and the number of unkilld ancestral lineages is m , when truncating at time τ , is given by

$$\mathbb{P}_n(|\mathcal{M}^\tau| = k, A_\tau^\mathcal{K} = m) = \frac{\theta}{2} f_n^\tau(k | A_\tau^\mathcal{K} = m) \mathbb{P}(A_\tau^\mathcal{C} = m) + o(\theta),$$

where

$$f_n^\tau(k | A_\tau^\mathcal{K} = m) = \frac{2}{\alpha k} \frac{\binom{n-m}{k}}{\binom{n-1}{k}}. \quad (12)$$

We prove Proposition 1 in Section 6.5. Note that this equation does not hold for the case $k = n, m = 0$, but fortunately we do not need to consider that case in what follows below.

We can use Proposition 1 to stably and efficiently compute the terms $f_n^\tau(k)$, for $k \leq n$, as follows. We first compute the case $k \leq n$. Note that

$$\mathbb{P}_n(|\mathcal{M}^\tau| = K) = \sum_m \mathbb{P}_n(|\mathcal{M}^\tau| = K, A_\tau^\mathcal{K} = m). \text{ So for } k \leq n, \text{ by Proposition 1}$$

$$f_n^\tau(k) = \sum_{m=1}^n f_n^\tau(k | A_\tau^\mathcal{K} = m) \mathbb{P}_n(A_\tau^\mathcal{C} = m) = \sum_{m=1}^n \frac{2}{\alpha k} \frac{\binom{n-m}{k}}{\binom{n-1}{k}} \mathbb{P}_n(A_\tau^\mathcal{C} = m). \quad (13)$$

The sum in (13) contains $O(n)$ terms, so it costs $O(n^2)$ to compute $f_n^\tau(k)$ for all $k \leq n$. After this, we use Lemma 1 to compute $f_n^\tau(n)$, and then use Lemma 2 to compute $f_n^\tau(k)$ for all $1 \leq k \leq n$. Since there are $O(n^2)$ such terms, this also takes $O(n^2)$ time.

4 The joint SFS for multiple populations

In this section we discuss an algorithm for computing the multi-population SFS (Chen, 2012, 2013; Wakeley and Hey, 1997). We describe the algorithm in Section 4.1, and note how the results from Section 3 improve the time complexity of this algorithm. In Section 4.2, we focus on the special case of tree-shaped demographies, and introduce a further algorithmic speedup by replacing the coalescent with a Moran model.

Let V be the number of subpopulations in the demographic history, n the total sample size, and L the number of SFS entries to compute. Then the results from Section 3 improve the computational complexity of the SFS from $O(n^5 V + WL)$ to $O(n^2 V + WL)$, where W is a term that depends on n and the structure of the demographic history. In the special case of

tree-shaped demographies, the algorithm from Chen (2012) gives $WL = O(n^4 VL)$. The Moran-based speedup introduced in Section 4.2 improves the runtime of tree-shaped demographies from $O(n^4 VL)$ to $O(n^3 V + n^2 VL)$.

The Moran-based speedup can be generalized to non-tree demographies, but the notation, implementation, and analysis of computational complexity becomes substantially more complicated. We thus leave its generalization to future work.

4.1 A coalescent-based dynamic program

Suppose at the present we have $\mathcal{D}$ populations, and in the i th population we observe n_i alleles. For a single point mutation, let $\mathbf{x} = (x_1, \dots, x_{\mathcal{D}})$ denote the number of alleles that are

derived in each population. We wish to compute $f(\mathbf{x})$, where $\frac{\theta}{2}f(\mathbf{x})$ is the expected number of point mutations with derived counts $\mathbf{x}$.

For demographic histories consisting of population size changes, population splits, population mergers, and pulse admixture events, Chen (2012) gave an algorithm to compute $f(\mathbf{x})$ using the truncated SFS $f_n^\tau(k)$ that we defined in Section 3.

We describe this algorithm to compute $f(\mathbf{x})$. We start by representing the population history as a directed acyclic graph (DAG), where each vertex v represents a subpopulation (Figure 3). We draw a directed edge from v to v' if there is gene flow from the bottom-most part of v to the top-most part of v' , where “down” is the present and “up” is the ancient past. Thus, the leaf vertices correspond to the subpopulations at the present. For a vertex v in the population history graph, let $\tau_v \in (0, \infty)$ denote the length of time the corresponding population persists, and let $\alpha_v : [0, \tau_v) \rightarrow \mathbb{R}^+$ denote the inverse population size history of v . So going backwards in time from the present, $\alpha_v(t)$ gives the instantaneous rate at which two particular lineages in v coalesce, after v has existed for time t . We use $f_n^v(k)$ to denote the truncated SFS for the coalescent embedded in v , i.e., $f_n^v(k) = f_n^{\tau_v}(k)$ for a coalescent with coalescence rate $\alpha_v(t)$. Then we have

$$f(\mathbf{x}) = \sum_v \sum_{m_0^v, k_0^v} f_{m_0^v}^v(k_0^v) \mathbb{P}(\mathbf{x} | k_0^v, m_0^v) \mathbb{P}(m_0^v) \quad (14)$$

where m_0^v denotes the number of lineages at the bottom of v that are ancestral to the initial sample, and k_0^v denotes the number of these lineages with a derived allele.

In order to use (14), we must compute $f_{m_0^v}^v(k_0^v)$ for every population v , and every value of m_0^v and k_0^v . If n is the total sample size and V the total number of vertices, then this takes $O(n^5 V)$ time using the formulas of Chen (2012). Our results from Section 3 improve this to $O(n^2 V)$.

To use (14), we must also compute the terms $\mathbb{P}(\mathbf{x} | k_0^v, m_0^v) \mathbb{P}(m_0^v)$, for which Chen (2012) constructs a dynamic program, starting at the leaf vertices and moving up the graph. This dynamic program essentially consists of setting up a Bayesian graphical model with random

variables m_0^v, k_0^v and performing belief propagation, which can be done via the sum-product algorithm (“tree-peeling”) if the population graph is a tree (Felsenstein, 1981; Pearl, 1982), or via a junction tree algorithm if not (Lauritzen and Spiegelhalter, 1988).

The time complexity of the algorithm thus depends on the topological structure of the population graph. In the special case where the demographic history is a binary tree, the tree-peeling algorithm computes the values $\mathbb{P}(\mathbf{x}|k_0^v, m_0^v)\mathbb{P}(m_0^v)$ in $\mathcal{O}(n^4 V)$ time, since the vertex v has $\mathcal{O}(n^2)$ possible states (k_0^v, m_0^v) , so summing over the transitions between every pair of states costs $\mathcal{O}(n^4)$. Note that Chen (2012) mistakenly states that the computation takes $\mathcal{O}(n^3 V)$ time.

To summarize, let W be the time it takes to compute (14) after the terms $f_m^v(k)$ have been precomputed, and let L be the number of distinct entries $\mathbf{x}$ for which we wish to compute $f(\mathbf{x})$. Then our results from Section 3 improve the computational complexity from $\mathcal{O}(n^5 V + WL)$ to $\mathcal{O}(n^2 V + WL)$. In the case of a binary tree the original algorithm of Chen (2012) gives $WL = \mathcal{O}(n^4 VL)$. In the following section, we improve this to $\mathcal{O}(n^3 V + n^2 VL)$.

4.2 A Moran-based dynamic program

Here, we describe a new dynamic program that improves the computational complexity of computing $f(\mathbf{x})$ for tree-shaped demographies. The main idea is to replace the backwards-in-time coalescent with a forwards-in-time Moran model.

4.2.1 Algorithm description—We assume the $\mathcal{D}$ populations at the present are related by a binary rooted tree with $\mathcal{D}$ leaves, where each leaf represents a population at the present, and at each internal vertex, a parent population splits into two child populations. (Note that a non-binary tree can be represented as a binary tree, with additional vertices of height 0).

Instead of working with the multi-population coalescent directly, we will consider a multi-population Moran model, in which the coalescent is embedded (Moran, 1958). In particular, let $\mathcal{L}(v)$ denote the leaf populations descended from the population v , and let $n_v = |\mathcal{L}(v)|$ be the number of present-day alleles with ancestry in v . For each population v (except the root), we construct a Moran model going *forward* in time, i.e. starting at τ_v and ending at 0. The Moran model consists of n_v lineages, each with either an ancestral or derived allele.

Going forward in time, every lineage copies itself onto every other lineage at rate $\frac{1}{2}\alpha_v(t)$.

Thus, the total rate of copying events is $\binom{n_v}{2}\alpha_v(t)$. Let μ_t^v denote the number of derived alleles at time t in population v . Then the transition rate of μ_t^v when $\mu_t^v = x$, is

$$\lambda_{x \rightarrow x+1}(t) = \lambda_{x \rightarrow x-1}(t) = \frac{x(n_v - x)}{2}\alpha_v(t) \text{ since there are } x(n_v - x) \text{ pairs of lineages with different alleles.}$$

The coalescent is embedded within the Moran model, because if we trace the ancestry of genetic material backwards in time in the Moran model, we obtain a genealogy with the same distribution as under the coalescent (Durrett, 2008, Theorem 1.30). Thus, we can

obtain the expected number of mutations with derived counts $\mathbf{x}$, by summing over the population v in which the mutation occurred:

$$f(\mathbf{x}) = \sum_v \sum_{k=1}^{n_v} f_{n_v}^v(k) \mathbb{P}(\mathbf{x} | \mu_0^v = k, \mu_{\tau_v}^v = 0). \quad (15)$$

Let $\mathbf{x}_v = \{x_i : i \in \mathcal{L}(v)\}$ denote the subsample of derived allele counts in the populations descended from v . Similarly, let $\mathbf{x}_v^c = \{x_i : i \notin \mathcal{L}(v)\}$. Then for $k \geq 1$,

$$\mathbb{P}(\mathbf{x} | \mu_0^v = k, \mu_{\tau_v}^v = 0) = \begin{cases} \mathbb{P}(\mathbf{x}_v | \mu_0^v = k), & \text{if } \mathbf{x}_v^c = \mathbf{0}, \\ 0, & \text{if } \mathbf{x}_v^c \neq \mathbf{0}. \end{cases} \quad (16)$$

So it suffices to compute $\mathbb{P}(\mathbf{x}_v | \mu_0^v = k)$ for all v and k . If v is the i th leaf population, then $\mathbb{P}(\mathbf{x}_v | \mu_0^v = k) = \mathbb{I}_{k=x_i}$. On the other hand, if v is an interior vertex with children v_1 and v_2 , then

$$\mathbb{P}(\mathbf{x}_v | \mu_0^v = k) = \sum_{k_1=0}^{n_{v_1}} \frac{\binom{n_{v_1}}{k_1} \binom{n_{v_2}}{k-k_1}}{\binom{n_v}{k}} \mathbb{P}(\mathbf{x}_{v_1} | \mu_{\tau_{v_1}}^{v_1} = k_1) \mathbb{P}(\mathbf{x}_{v_2} | \mu_{\tau_{v_2}}^{v_2} = k - k_1), \quad (17)$$

where $\mathbb{P}(\mathbf{x}_{v_i} | \mu_{\tau_{v_i}}^{v_i})$ can be computed from

$$\mathbb{P}(\mathbf{x}_v | \mu_{\tau_v}^v = k) = \sum_{j=0}^{n_v} \mathbb{P}(\mathbf{x}_v | \mu_0^v = j) \mathbb{P}(\mu_0^v = j | \mu_{\tau_v}^v = k). \quad (18)$$

To compute the transition probability $\mathbb{P}(\mu_0^v = j | \mu_{\tau_v}^v = k)$, note that the transition rate matrix of μ_t^v can be written as $Q^{(n_v)} \alpha(t)$, where $Q^{(n_v)} = (q_{ij}^{(n_v)})_{0 \leq i, j \leq n_v}$ is a $(n_v + 1) \times (n_v + 1)$ matrix with

$$q_{ij}^{(n_v)} = \begin{cases} -i(n_v - i), & \text{if } i=j, \\ \frac{1}{2}i(n_v - i), & \text{if } |j - i|=1, \\ 0, & \text{else,} \end{cases}$$

so then the transition probability is given by the matrix exponential

$$\mathbb{P}(\mu_0^v = j | \mu_{\tau_v}^v = k) = [e^{Q^{(n_v)} \int_0^{\tau_v} \alpha_v(t) dt}]_{k,j}. \quad (19)$$

Thus, the joint SFS $f(\mathbf{x})$ can be computed using (15) and (16), with $\mathbb{P}(\mathbf{x}_v | \mu_0^v = k)$ given by recursively computing (17), (18), and (19), in a depth-first search on the population tree (i.e., Felsenstein's tree-peeling algorithm, or the sum-product algorithm for belief propagation).

4.2.2 Computational complexity of Moran approach—We now consider the computational complexity associated with (17), (18), and (19) for each vertex v . For fixed configuration $\mathbf{x}$, (17) and (18) must be computed for $O(n_v)$ values, and are each sums with $O(n_v)$ terms; they thus contribute $O(n_v^2 L)$ to the total runtime, where L is the number of distinct values of $\mathbf{x}$. The matrix exponential $e^{(Q^{(nv)} \int_0^{\tau_v} \alpha_v(t) dt)}$ in (19) can be computed in several ways, including spectral decomposition or scaling-and-squaring, and costs $O(n_v^3)$ time (Moler and Van Loan, 2003). The computational complexity associated with a single vertex v is thus $O(n_v^3 + n_v^2 L)$. Therefore, for a binary population tree with V nodes, arbitrary population size functions, and no migration, the total cost of computing the observed SFS entries is $O(n^3 V + n^2 VL)$.

The time complexity can be further improved with techniques applied by Bryant et al. (2012), but in practice we found that the decreased running time was offset by other problems such as numerical instability or large hidden time costs. In particular, let

$\ell_t^v(k) = \mathbb{P}(\mathbf{x}_v | \mu_t^v = k)$, and $\tilde{\ell}_t^v(k) = \binom{n_v}{k} \ell_t^v(k)$. Then (17) can be written as a convolution

$$\tilde{\ell}_0^v = \tilde{\ell}_{\tau_{v1}}^{v1} * \tilde{\ell}_{\tau_{v2}}^{v2}, \quad (20)$$

which can be computed via the FFT (Cooley and Tukey, 1965), reducing the complexity of (17) from $O(n_v^2 L)$ to $O(n_v \log(n_v) L)$. Another potential speedup is to rewrite (18) as

$$\ell_{\tau_v}^v = e^{(Q^{(nv)} \int_0^{\tau_v} \alpha_v(t) dt)} \ell_0^v \quad (21)$$

and utilize the sparsity of $Q^{(nv)}$ (Al-Mohy and Higham, 2011). In particular, (21) can be computed by $\mathcal{T}$ sparse matrix-vector products, where $\mathcal{T}$ depends on $Q^{(nv)}$ and the desired level of precision. This reduces the cost of (18) and (19) from $O(n_v^3 + n_v^2 L)$ to $O(n_v \mathcal{T} L)$.

These two speedups were applied by Bryant et al. (2012) to reduce the complexity of their coalescent-based approach from $O(n_v^6 + n_v^4 L)$ to $O(n_v^2 L (\log(n_v) + \mathcal{T}))$. When applied in our Moran-based approach, they reduce the complexity of (17), (18), and (19) from

$O(n_v^3 + n_v^2 L)$ to $O(n_v L (\log(n_v) + \mathcal{T}))$. In practice, however, we found $\mathcal{T}$ to be quite large, and it was faster to use the naive approach to compute and multiply $e^{(Q^{(nv)} \int_0^{\tau_v} \alpha_v(t) dt)}$. Furthermore, computing (20) via the FFT can be very numerically unstable. Taking the Fourier transform introduces cancellation errors, due to multiplying and adding terms like

e^{-ix} , and we found that converting from $\tilde{\ell}_0^v$ back to ℓ_0^v can cause these errors to blow up, due to the combinatorial factors.

5 Runtime and accuracy results

5.1 Comparison with Chen (2012)

We implemented our formulas and algorithm in Python, using the Python packages *numpy* and *scipy*. We also implemented the formulas from Chen (2012), and compared the performance of the two algorithms on simulated data.

We simulated datasets with $n \in \{2, 4, 8, \dots, 256\}$ lineages and $\mathcal{D} \in \{2, 4, 8, \dots, n\}$

populations at present, each containing $\frac{n}{\mathcal{D}}$ lineages. For each value of $n, \mathcal{D}$, we used the program *scrm* (Staab et al., 2015) to generate 20 random datasets, each with a demographic history that is a random binary tree.

In Figure 4, we compare the running time of the original algorithm of Chen (2012) against our new algorithm that utilizes the formulas for $f_n^\tau(k)$ presented in Section 3 and our new Moran-based approach described in Section 4.2. We find our algorithm to be orders of magnitude faster; the difference is especially pronounced as the number n of lineages grows. Note that, due to the increased running time of Chen's algorithm, we did not finish running his method for $n = 256$ and $\mathcal{D} = 32$.

In Figure 5, we compare the accuracy of the two algorithms. The figure compares the SFS entries returned by the two methods across a subset of the simulations depicted in Figure 4. The line $y = x$ is also plotted; points falling on the line depict the SFS entries where both methods agreed. All negative return values represent numerical errors. For $n = 64$ the two methods generally agree, but for larger n Chen's algorithm displays considerable numerical instability, returning extremely large positive and negative numbers.

5.2 Comparison with *a i*

We also compared our method with the popular program *a i* (Gutenkunst et al., 2009). We note that our method has several key differences from *a i*, and the two methods have strengths in distinct use cases. *a i* computes the joint SFS by numerically integrating a PDE. This PDE can be easily modified to include effects such as natural selection and continuous migration, which gives *a i* more flexibility than the coalescent or Moran-based approaches discussed in this paper. However, numerical integration of multidimensional PDEs is a challenging problem; thus *a i* can handle only a small number of populations (up to 3), and may occasionally encounter numerical instability.

We compared our method (*mom*) with *a i* on a *modified* version of the three-population out-of-Africa demography inferred by Gutenkunst et al. (2009). In this history, the Eurasian population initially splits off from the African population, and then splits into separate European and Asian populations. The populations experience several piecewise constant size changes, and the European and Asia populations experience exponential growth in the recent past. The original, unmodified history also contains continuous migration between the

separated populations; since our implementation does not currently support migration, we modified the demographic history to have no migration. (Support for migration will be added to our method in the future.)

We consider sample sizes of $n = 16, 32, 64, 128$ per population; for a_i , we consider discretization grid sizes of $G = 16, 32, \dots, 512, 1024$ points per population, with $G \propto n$. We compare the values returned by *mom*i and a_i in Figure 6. *mom*i and a_i mostly agree, but

some entries of a_i are off by a factor of 10 or 100, especially when $\frac{G}{n}$ is small. In general the value computed by a_i converges to that of *mom*i as G increases. However, for $G = 1024$ and $n = 64, 128$, a_i appears to have some numerical instability. In particular, a_i diverges in some of the smaller entries, and returns some negative numbers.

We compare the runtime of *mom*i and a_i in Figure 7, which shows the time to compute all entries of the SFS. The two methods are roughly comparable, depending on the number G of grid points in a_i . We note that a_i computes all entries of the SFS together, whereas *mom*i and the method of Chen (2012, 2013) can easily compute a subset of the SFS. This is a key advantage for the latter methods when the number of populations $\mathcal{D}$ is large, as the size of the SFS grows exponentially with $\mathcal{D}$.

6 Proofs

In this section, we provide proofs of the mathematical results presented in earlier sections.

6.1 A recursion for efficiently computing $\mathbb{P}_\nu(A_\tau^{\mathcal{C}} = m)$

We describe how to compute $\mathbb{P}_\nu(A_\tau^{\mathcal{C}} = m)$, for all values of $m \leq n$, in $O(n^2)$ time. First, note that

$$\begin{aligned} \mathbb{P}_{\nu-1}(A_\tau^{\mathcal{C}} = m) &= \mathbb{P}_\nu(A_\tau^{\mathcal{C}} = m+1, \{\nu\} \in \mathcal{C}_\tau) + \mathbb{P}_\nu(A_\tau^{\mathcal{C}} = m, \{\nu\} \notin \mathcal{C}_\tau) \\ &= \frac{(m+1)p_{\nu,m+1}^{1,1}}{\binom{\nu}{1}} \mathbb{P}_\nu(A_\tau^{\mathcal{C}} = m+1) + \left(1 - \frac{mp_{\nu,m}^{1,1}}{\binom{\nu}{1}}\right) \mathbb{P}_\nu(A_\tau^{\mathcal{C}} = m) \\ &= \frac{(m+1)(m)}{\nu(\nu-1)} \mathbb{P}_\nu(A_\tau^{\mathcal{C}} = m+1) + \left(1 - \frac{m(m-1)}{\nu(\nu-1)}\right) \mathbb{P}_\nu(A_\tau^{\mathcal{C}} = m). \end{aligned}$$

Rearranging, we get the recursion

$$\mathbb{P}_\nu(A_\tau^{\mathcal{C}} = m) = \frac{1}{1 - \frac{m(m-1)}{\nu(\nu-1)}} \left[\mathbb{P}_{\nu-1}(A_\tau^{\mathcal{C}} = m) - \frac{(m+1)(m)}{\nu(\nu-1)} \mathbb{P}_\nu(A_\tau^{\mathcal{C}} = m+1) \right] \quad (22)$$

with base cases

$$\mathbb{P}_\nu(A_\tau^{\mathcal{C}} = \nu) = e^{-\binom{\nu}{2} \int_0^\tau \alpha(t) dt}.$$

So after solving $\int_0^\tau \alpha(t) dt$, we can use the recursion and memoization to solve for all of the $O(n^2)$ terms $\mathbb{P}_\nu(A_\tau^{\mathcal{C}} = m)$ in $O(n^2)$ time. In particular, in the case of constant population size, $\alpha(t) = \alpha$, the base case is given by

$$\mathbb{P}_\nu(A_\tau^{\mathcal{C}} = \nu) = e^{-\binom{\nu}{2} \alpha \tau},$$

and in the case of an exponentially growing population size, $\alpha(t) = \alpha(\tau) e^{\beta(\tau-t)}$, the base case is given by

$$\mathbb{P}_\nu(A_\tau^{\mathcal{C}} = \nu) = e^{-\binom{\nu}{2} \alpha(\tau) (e^{\beta\tau} - \frac{1}{\beta})}.$$

6.2 Proof of Lemma 1

Let T_{MRCA} denote the time to the most recent common ancestor of the sample. We first note that

$$f_n^\tau(n) = \tau - \mathbb{E}_n[T_{\text{MRCA}} \wedge \tau],$$

since the branch length subtending the whole sample is the time between τ and T_{MRCA} .

Next, note that $\frac{\theta}{2} \mathbb{E}_n[T_{\text{MRCA}} \wedge \tau]$ is equal to the number of polymorphic mutations in $[0, \tau]$ where the individual “1” is derived. This is because, as we trace the ancestry of “1” backwards in time, all mutations hitting the lineage below T_{MRCA} are polymorphic, while all mutations hitting above T_{MRCA} are monomorphic.

The expected number of polymorphic mutations with “1” derived is also equal to

$\frac{\theta}{2} \sum_{k=1}^{n-1} \frac{k}{n} f_n^\tau(k)$, since if a mutation has k derived leaves, the chance that “1” is in the derived set is $\frac{k}{n}$. Thus

$$\mathbb{E}_n[T_{\text{MRCA}} \wedge \tau] = \sum_{k=1}^{n-1} \frac{k}{n} f_n^\tau(k),$$

which completes the proof.

6.3 Proof of Lemma 2

We first note that

$$\mathbb{P}_n(\mathcal{M}^\tau = \{1, \dots, k\}) = \mathbb{P}_{n+1}(\mathcal{M}^\tau = \{1, \dots, k\}) + \mathbb{P}_{n+1}(\mathcal{M}^\tau = \{1, \dots, k, n+1\}).$$

$$\mathbb{P}_n(\mathcal{M}^\tau = K) = \frac{\theta}{2} \frac{f_n^\tau(|K|)}{\binom{n}{|K|}} + o(\theta)$$

By exchangeability, we have for all $K \subseteq \{1, \dots, n\}$, so

$$\frac{1}{\binom{n}{k}} f_n^\tau(k) = \frac{1}{\binom{n+1}{k}} f_{n+1}^\tau(k) + \frac{1}{\binom{n+1}{k+1}} f_{n+1}^\tau(k+1).$$

Multiplying both sides by $\binom{n}{k}$ gives

$$f_n^\tau(k) = \frac{n-k+1}{n+1} f_{n+1}^\tau(k) + \frac{k+1}{n+1} f_{n+1}^\tau(k+1).$$

6.4 Proof of Lemma 3

Let $\alpha^*(t)$ denote the inverse population size history given by

$$\alpha^*(t) = \begin{cases} \alpha(t) & \text{if } t < \tau \\ \infty & \text{if } t \geq \tau. \end{cases}$$

So the demographic history with population size $\frac{1}{\alpha^*(t)}$ agrees with the original history up to time τ , at which point the population size drops to 0, and all lineages instantly coalesce into a single lineage with probability 1.

Let $T_{m,*}$ denote the amount of time there are m ancestral lineages for the coalescent with

size history $\frac{1}{\alpha^*(t)}$. Similarly, let $f_{n,*}(k)$ denote the SFS under the size history $\frac{1}{\alpha^*(t)}$. Then from the result of Polanski and Kimmel (2003),

$$f_{n,*}(k) = \sum_{m=2}^n W_{n,k,m} \mathbb{E}_m[T_{m,*}].$$

Note that for $m > 1$, we almost surely have $T_{m,*} = T_{m,*}^\tau$, i.e. the intercoalescence time equals its truncated version, since all lineages coalesce instantly at τ with probability 1. Thus,

$\mathbb{E}_m[T_{m,*}] = \mathbb{E}_m[T_{m,*}^\tau]$. Similarly, for $k < n$, $f_{n,*}(k) = f_{n,*}^\tau(k)$, i.e. the SFS equals the truncated SFS, because the probability of a polymorphic mutation occurring in $[\tau, \infty)$ is 0.

Finally, note that $\mathbb{E}_m[T_{m,*}^\tau] = \mathbb{E}_m[T_m^\tau]$ and $f_{n,*}^\tau(k) = f_n^\tau(k)$, because $\alpha(t)$ and $\alpha^*(t)$ are identical on $[0, \tau]$.

6.5 Proof of Proposition 1

We start by showing that $\mathbb{P}_n(A_\tau^\mathcal{K} = m) = \mathbb{P}_n(A_\tau^\mathcal{C} = m) + O(\theta)$. Let $T_i^\tau(\mathcal{K}) = \int_0^\tau \mathbb{I}_{A_t^\mathcal{K} = i} dt$ denote the amount of time where $\mathcal{K}$ has i unkilld lineages. Let p denote the probability density function. For $(t_n, \dots, t_m)$ with $t_i = \tau$, we have

$$\begin{aligned} p(T_n^\tau(\mathcal{K}) = t_n, \dots, T_m^\tau(\mathcal{K}) = t_m) \\ &= e^{-\lambda_{m,m-1}^\mathcal{K} t_m} \prod_{i=m+1}^n \lambda_{i,i-1}^\mathcal{K} e^{-\lambda_{i,i-1}^\mathcal{K} t_i} \\ &= e^{-\binom{m}{2} \alpha + \frac{m\theta}{2} t_m} \prod_{i=m+1}^n \left(\binom{i}{2} \alpha + \frac{i\theta}{2} \right) e^{-\binom{i}{2} \alpha + \frac{i\theta}{2} t_i} \\ &= e^{-\binom{m}{2} \alpha t_m} \prod_{i=m+1}^n \binom{i}{2} \alpha e^{-\binom{i}{2} \alpha t_i} + O(\theta) \\ &= p(T_n^\tau = t_n, \dots, T_m^\tau = t_m) + O(\theta), \end{aligned}$$

and so

$$\lim_{\theta \rightarrow 0} \mathbb{P}_n(A_\tau^\mathcal{K} = m) = \lim_{\theta \rightarrow 0} \int_{\sum t_i = \tau} p(T_n^\tau(\mathcal{K}) = t_n, \dots, T_m^\tau(\mathcal{K}) = t_m) dt = \int_{\sum t_i = \tau} p(T_n^\tau = t_n, \dots, T_m^\tau = t_m) dt = \mathbb{P}_n(A_\tau^\mathcal{C} = m).$$

where we can exchange the limit and the integral by the Bounded Convergence Theorem,

$$\text{because } p(T_n^\tau(\mathcal{K}) = t_n, \dots, T_m^\tau(\mathcal{K}) = t_m) \leq \prod_{i=m+1}^n \left(\binom{i}{2} \alpha + \frac{i}{2} \right)$$

Thus we have

$$\begin{aligned} \mathbb{P}_n(|\mathcal{M}^\tau| = k, A_\tau^\mathcal{K} = m) &= \mathbb{P}_n(|\mathcal{M}^\tau| = k | A_\tau^\mathcal{K} = m) \mathbb{P}_n(A_\tau^\mathcal{K} = m) \\ &= \left(\frac{\theta}{2} f_n^\tau(k | A_\tau^\mathcal{K} = m) o(\theta) \right) (\mathbb{P}_n(A_\tau^\mathcal{C} = m) + O(\theta)) \\ &= \frac{\theta}{2} f_n^\tau(k | A_\tau^\mathcal{K} = m) \mathbb{P}_n(A_\tau^\mathcal{C} = m) + o(\theta), \end{aligned}$$

which proves the first part of the proposition.

We next solve for $f_n^\tau(k | A_\tau^\mathcal{K} = m)$, the first order Taylor series coefficient for

$$\mathbb{P}_n(|\mathcal{M}^\tau| = k | A_\tau^\mathcal{K} = m) \text{ in the mutation rate } \frac{\theta}{2}.$$

When there are i unkilld lineages, the probability that the next event is a killing event is

$\frac{\theta}{\alpha(i-1)+\theta} = \frac{\theta}{\alpha(i-1)} + O(\theta)$. Given that the event is a killing, the chance that the killed lineage has k leaf descendants is $p_{n,i}^{k,1}$. So summing over i , and dividing out the mutation rate $\frac{\theta}{2}$, we get

$$\begin{aligned}
 f_n^\tau(k|A_\tau^\mathcal{K}=m) &= \frac{2^{n-k+1}}{\alpha} \sum_{i=m+1}^n \frac{1}{i-1} p_{n,i}^{k,1} \\
 &= \frac{2^{n-k+1}}{\alpha} \sum_{i=m+1}^n \frac{1}{i-1} \frac{\binom{n-k-1}{i-2}}{\binom{n-1}{i-1}} \\
 &= \frac{2^{n-k+1}}{\alpha} \sum_{i=m+1}^n \frac{1}{i-1} \frac{(n-k-1)!(i-1)!(n-i)!}{(i-2)!(n-k-i+1)!(n-1)!} \\
 &= \frac{2(n-k-1)!}{\alpha(n-1)!} \sum_{i=m+1}^n \frac{(n-i)!}{(n-k-i+1)!} \\
 &= \frac{2(n-k-1)!}{\alpha(n-1)!} \sum_{j=0}^{n-k-m} \frac{(j+k-1)!}{j!} \\
 &= \frac{2}{\alpha k \binom{n-1}{k}} \sum_{j=0}^{n-k-m} \binom{j+k-1}{j} \\
 &= \frac{2}{\alpha k} \frac{\binom{n-m}{k}}{\binom{n-1}{k}},
 \end{aligned}$$

where we made the change of variables $j = n - k - i + 1$, and where the final line follows

from repeated application of the combinatorial identity $\binom{a}{b} = \binom{a-1}{b} + \binom{a-1}{b-1}$.

6.5.1 Alternative proof for $f_n^\tau(k|A_\tau^\mathcal{K}=m)$ via the Chinese Restaurant Process—

We sketch an alternative proof of the expression for $f_n^\tau(k|A_\tau^\mathcal{K}=m)$, using the Chinese Restaurant Process.

Consider the coalescent with killing going forward in time (towards the present), and only looking at it when the number of individuals increases. Then when there are i lineages, a

new mutation occurs with probability $\frac{\theta}{\alpha i + \theta} = \frac{\theta/\alpha}{i + \theta/\alpha}$, and each lineage branches with

probability $\frac{\alpha}{\alpha i + \theta} = \frac{1}{i + \theta/\alpha}$. Thus, conditional on $A_\tau^\mathcal{K}=m$, the distribution on $\mathcal{K}_\tau$ is given by

a Chinese Restaurant Process (Aldous, 1985), starting with m tables each with 1 person, and with new tables founded with parameter θ/α .

Let $(x)_{i\uparrow} = x(x+1) \cdots (x+i-1)$ denote the rising factorial. If there is a single mutation with k descendants, then there are $\binom{n-m}{k}$ ways to pick which of the $n-m$ events involve mutant lineages. The probability of a particular such ordering is

$$\frac{\theta (1)_{k\uparrow} (m)_{n-k-m\uparrow}}{\alpha (m+\theta/\alpha)_{n-m\uparrow}} = \frac{\theta (k-1)!(n-k-1)!/m!}{\alpha (n-1)!/m!} + o(\theta).$$

Summing over all $\binom{n-m}{k}$ orderings, and dividing by $\frac{\theta}{2}$, yields

$$f_n^\tau(k|A_\tau^{\mathcal{K}}=m) = \frac{2}{\alpha} \binom{n-m}{k} \frac{(k-1)!(n-k-1)!/m!}{(n-1)!/m!}.$$

Acknowledgments

This research is supported in part by NIH grants R01-GM109454 and R01-GM108805, a Packard Fellowship for Science and Engineering, a Miller Research Professorship, and a Citadel Graduate Fellowship.

References

- Al-Mohy AH, Higham NJ. Computing the action of the matrix exponential, with an application to exponential integrators. *SIAM Journal on Scientific Computing*. 2011; 33(2):488–511.
- Aldous, DJ. Exchangeability and related topics. In: Hennequin, P., editor. *École d' Été de Probabilités de Saint-Flour XIII — 1983*, volume 1117 of *Lecture Notes in Mathematics*. Berlin Heidelberg: Springer; 1985. p. 1-198.
- Beaumont MA, Nichols RA. Evaluating loci for use in the genetic analysis of population structure. *Proceedings of the Royal Society of London. Series B: Biological Sciences*. 1996; 263(1377):1619–1626.
- Bhaskar A, Kamm JA, Song YS. Approximate sampling formulae for general finite-alleles models of mutation. *Advances in Applied Probability*. 2012; 44:408–428. (PMC3953561). [PubMed: 24634516]
- Bhaskar A, Song YS. Descartes' rule of signs and the identifiability of population demographic models from genomic variation data. *Annals of Statistics*. 2014; 42(6):2469–2493. [PubMed: 28018011]
- Bhaskar A, Wang YXR, Song YS. Efficient inference of population size histories and locus-specific mutation rates from large-sample genomic variation data. *Genome Research*. 2015; 25(2):268–279. [PubMed: 25564017]
- Boyko AR, Williamson SH, Indap AR, Degenhardt JD, Hernandez RD, Lohmueller KE, Adams MD, Schmidt S, Sninsky JJ, Sunyaev SR, et al. Assessing the evolutionary impact of amino acid mutations in the human genome. *PLoS Genetics*. 2008; 4(5):e1000083. [PubMed: 18516229]
- Bryant D, Bouckaert R, Felsenstein J, Rosenberg NA, RoyChoudhury A. Inferring species trees directly from biallelic genetic markers: bypassing gene trees in a full coalescent analysis. *Molecular Biology and Evolution*. 2012; 29(8):1917–1932. [PubMed: 22422763]
- Chen H. The joint allele frequency spectrum of multiple populations: A coalescent theory approach. *Theoretical Population Biology*. 2012; 81(2):179–195. [PubMed: 22155588]

- Chen H. Intercoalescence time distribution of incomplete gene genealogies in temporally varying populations, and applications in population genetic inference. *Annals of Human Genetics*. 2013; 77(2):158–173. [PubMed: 23369132]
- Cooley JW, Tukey JW. An algorithm for the machine calculation of complex fourier series. *Mathematics of Computation*. 1965; 19(90):297–301.
- Coventry A, Bull-Otterson LM, Liu X, Clark AG, Maxwell TJ, Crosby J, Hixson JE, Rea TJ, Muzny DM, Lewis LR, et al. Deep resequencing reveals excess rare recent variants consistent with explosive population growth. *Nature Communications*. 2010; 1:131.
- De Maio N, Schlötterer C, Kosiol C. Linking great apes genome evolution across time scales using polymorphism-aware phylogenetic models. *Molecular biology and evolution*. 2013; 30(10):2249–2262. [PubMed: 23906727]
- Durrett, R. *Probability Models for DNA Sequence Evolution*. 2nd. New York: Springer; 2008.
- Ewens, WJ. *Mathematical Population Genetics: I. Theoretical Introduction*. New York: Springer Science+Business Media, Inc.; 2004.
- Excoffier L, Dupanloup I, Huerta-Sánchez E, Sousa VC, Foll M. Robust demographic inference from genomic and SNP data. *PLoS Genetics*. 2013; 9(10):e1003905. [PubMed: 24204310]
- Felsenstein J. Evolutionary trees from DNA sequences: a maximum likelihood approach. *Journal of Molecular Evolution*. 1981; 17:368–376. [PubMed: 7288891]
- Gazave E, Ma L, Chang D, Coventry A, Gao F, Muzny D, Boerwinkle E, Gibbs RA, Sing CF, Clark AG, et al. Neutral genomic regions refine models of recent rapid human population growth. *Proceedings of the National Academy of Sciences*. 2014; 111(2):757–762.
- Gravel S, Henn BM, Gutenkunst RN, Indap AR, Marth GT, Clark AG, Yu F, Gibbs RA, Bustamante CD, Altshuler DL, et al. Demographic history and rare allele sharing among human populations. *Proceedings of the National Academy of Sciences*. 2011; 108(29):11983–11988.
- Griffiths R, Tavaré S. The age of a mutation in a general coalescent tree. *Communications in Statistics. Stochastic Models*. 1998; 14(1–2):273–295.
- Gutenkunst RN, Hernandez RD, Williamson SH, Bustamante CD. Inferring the joint demographic history of multiple populations from multidimensional SNP frequency data. *PLoS Genetics*. 2009; 5(10):e1000695. [PubMed: 19851460]
- Higham, NJ. *Accuracy and Stability of Numerical Algorithms*. 2nd. SIAM: Society for Industrial and Applied Mathematics; 2002.
- Hoppe F. Pólya-like urns and the Ewens' sampling formula. *J. Math. Biol*. 1984; 20:91–94.
- Jenkins PA, Mueller JW, Song YS. General triallelic frequency spectrum under demographic models with variable population size. *Genetics*. 2014; 196(1):295–311. (PMC3872192). [PubMed: 24214345]
- Jenkins PA, Song YS. The effect of recurrent mutation on the frequency spectrum of a segregating site and the age of an allele. *Theor. Popul. Biol*. 2011; 80(2):158–173. (PMC3143209). [PubMed: 21550359]
- Johnson, NL., Kotz, S. *Urn Models and Their Application: An Approach to Modern Discrete Probability Theory*. New York: Wiley; 1977.
- Kimura M. The number of heterozygous nucleotide sites maintained in a finite population due to steady flux of mutations. *Genetics*. 1969; 61(4):893. [PubMed: 5364968]
- Kingman JFC. The coalescent. *Stoch. Process. Appl*. 1982a; 13:235–248.
- Kingman, JFC. Exchangeability and the evolution of large populations. In: Koch, G., Spizzichino, F., editors. *Exchangeability in Probability and Statistics*. North-Holland Publishing Company; 1982b. p. 97–112.
- Kingman JFC. On the genealogy of large populations. *J. Appl. Prob*. 1982c; 19A:27–43.
- Lauritzen SL, Spiegelhalter DJ. Local computations with probabilities on graphical structures and their application to expert systems. *Journal of the Royal Statistical Society. Series B (Methodological)*. 1988; 50(2):157–224.
- Luki S, Hey J. Demographic inference using spectral methods on SNP data, with an analysis of the human out-of-Africa expansion. *Genetics*. 2012; 192(2):619–639. [PubMed: 22865734]

- Moler C, Van Loan C. Nineteen dubious ways to compute the exponential of a matrix, twenty-five years later. *SIAM review*. 2003; 45(1):3–49.
- Moran P. Random processes in genetics. *Mathematical Proceedings of the Cambridge Philosophical Society*. 1958; 54:60–71.
- Myers S, Fefferman C, Patterson N. Can one learn history from the allelic spectrum? *Theoretical Population Biology*. 2008; 73(3):342–348. [PubMed: 18321552]
- Nelson MR, Wegmann D, Ehm MG, Kessner D, Jean PS, Verzilli C, Shen J, Tang Z, Bacanu S-A, Fraser D, et al. An abundance of rare functional variants in 202 drug target genes sequenced in 14,002 people. *Science*. 2012; 337(6090):100–104. [PubMed: 22604722]
- Nielsen R. Estimation of population parameters and recombination rates from single nucleotide polymorphisms. *Genetics*. 2000; 154(2):931–942. [PubMed: 10655242]
- Pearl, J. Reverend Bayes on inference engines: a distributed hierarchical approach; *Proceedings of the National Conference on Artificial Intelligence*; 1982. p. 133-136.
- Polanski A, Kimmel M. New explicit expressions for relative frequencies of single-nucleotide polymorphisms with application to statistical inference on population growth. *Genetics*. 2003; 165(1):427–436. [PubMed: 14504247]
- Schaffner SF, Foo C, Gabriel S, Reich D, Daly WJ, Altshuler D. Calibrating a coalescent simulation of human genome sequence variation. *Genome Res*. 2005; 15:1576–1583. [PubMed: 16251467]
- Staab PR, Zhu S, Metzler D, Lunter G. scrm: efficiently simulating long sequences using the approximated coalescent with recombination. *Bioinformatics*. 2015; 31(10):1680–1682. [PubMed: 25596205]
- Tavaré S. Line-of-descent and genealogical processes, and their applications in population genetics models. *Theor. Popul. Biol.* 1984; 26:119–164. [PubMed: 6505980]
- Wakeley J, Hey J. Estimating ancestral population parameters. *Genetics*. 1997; 145(3):847–855. [PubMed: 9055093]

Figure 1.

A sample path of the coalescent truncated at time τ . Star symbols denote mutations, while $\mathcal{M}^\tau$ denotes the set of leaves under those mutations. T_k^τ denotes the waiting time in the interval $[0, \tau)$ while there are k lineages.

Figure 2.

The coalescent with killing for the genealogy in Figure 1. Note that $\mathcal{K}_\tau$ is a marked partition, with the blocks killed by mutations in $[0, \tau)$ being specially marked.

Figure 3.

A demographic history with a pulse migration event (left), and its corresponding directed graph (right).

Figure 4.

Average computation time of the joint SFS. For each combination of the sample size n and

the number $\mathcal{D}$ of populations (with $\frac{n}{\mathcal{D}}$ samples per population), we generated 20 random datasets, each under a demographic history that is a random binary tree. The expected joint SFS for the resulting segregating sites were then computed using our method (*momi*) and that of Chen (2012). In the top row, we plot the average runtime per joint SFS entry, and in the bottom row, the average amount of time needed to precompute the truncated SFS for every subpopulation within each demographic history. (a) Runtime results plotted separately for each method in a linear scale. Note the y-axis is on a different scale for each row. (b) Runtime results with the axes on a log-log scale, so that shorter runtimes are visible.

Figure 5.

Numerical stability of the two algorithms. The plot compares the numerical values returned by our method (*momi*) and Chen's method, for the simulations described in Figure 4. The dashed red line represents the identity $y = x$. To adequately illustrate the full numerical range (both positive and negative) of values encountered in the simulations, we applied the transformation $z \mapsto \text{sign}(z) \log(1 + |z|)$ to the values of each method in order to produce the scatter plot. The two methods agree for $n \leq 64$, while Chen's method is extremely unstable for larger n .

Figure 6.

Comparison of SFS values computed by a_i and $momi$, varying the sample size n per population and the number G of grid points per population in a_i . We used the 3-population out-of-Africa demography inferred in Gutenkunst et al. (2009), but modified to have no gene flow (migration). The x -axis is $f_{momi}(\mathbf{x})$ the value computed by $momi$, the y -axis is the

absolute value of the ratio $\left| \frac{f_{a_i}(\mathbf{x})}{f_{momi}(\mathbf{x})} \right|$, and the color gives the sign of $f_{a_i}(\mathbf{x})$ (all values

returned by f_{a_i} were positive). f_{a_i} generally converges to $f_{mom}(\mathbf{x})$ as G increases, but for large values of n and G , f_{a_i} appears to diverge, and returns some negative values.

Figure 7.
Runtime of *momi* and *ai* to compute the results in Figure 6.