PLOS ONE

RESEARCH ARTICLE

# Demographic inference from multiple whole genomes using a particle filter for continuous Markov jump processes

**Donna Henderson**[1☯], **Sha (Joe) Zhu**[1,2☯¤], **Christopher B. Cole**[3☯], **Gerton Lunter**[3,4*]

**1** Wellcome Centre for Human Genetics, Oxford, United Kingdom, **2** Big Data Institute, Oxford, United Kingdom, **3** MRC Weatherall Institute of Molecular Medicine, John Radcliffe Hospital, Headington, Oxford, United Kingdom, **4** Department of Epidemiology, University Medical Center Groningen, University of Groningen, Groningen, The Netherlands

☯ These authors contributed equally to this work.
¤ Current address: TaiChi AI Ltd, London, United Kingdom
* g.a.lunter@umcg.nl

## Abstract

Demographic events shape a population's genetic diversity, a process described by the coalescent-with-recombination model that relates demography and genetics by an unobserved sequence of genealogies along the genome. As the space of genealogies over genomes is large and complex, inference under this model is challenging. Formulating the coalescent-with-recombination model as a continuous-time and -space Markov jump process, we develop a particle filter for such processes, and use waypoints that under appropriate conditions allow the problem to be reduced to the discrete-time case. To improve inference, we generalise the Auxiliary Particle Filter for discrete-time models, and use Variational Bayes to model the uncertainty in parameter estimates for rare events, avoiding biases seen with Expectation Maximization. Using real and simulated genomes, we show that past population sizes can be accurately inferred over a larger range of epochs than was previously possible, opening the possibility of jointly analyzing multiple genomes under complex demographic models. Code is available at https://github.com/luntergroup/smcsmc.

OPEN ACCESS

**Citation:** Henderson D, Zhu S(Joe), Cole CB, Lunter G (2021) Demographic inference from multiple whole genomes using a particle filter for continuous Markov jump processes. PLoS ONE 16(3): e0247647. https://doi.org/10.1371/journal.pone.0247647

**Editor:** Inés P. Mariño, Universidad Rey Juan Carlos, SPAIN

**Received:** May 4, 2020

**Accepted:** February 10, 2021

**Published:** March 2, 2021

**Data Availability Statement:** The data analysed in this study are publicly available from the 1000 Genomes Project (ftp://ftp.1000genomes.ebi.ac.uk/vol1/ftp/phase3/data/*/high_converage_alignment; sample identifiers, CEU: NA06984, NA11918, NA12272, NA12760; CHB: NA18525, NA18620, NA18749, NA18566; YRI: NA18486, NA19129, NA19197, NA18881) and the Simons Genome Diversity Project (https://sharehost.hms.harvard.edu/genetics/reich_lab/sgdp/phased_data/; sample identifiers: S_French-1, S_French-2, S_Yoruba-1, S_Yoruba-2). Potential low-quality regions were

## Introduction

The demographic history of a species has a profound impact on its genetic diversity. Changes in population size, migration and admixture events, and population splits and mergers, shape the genealogies describing how individuals in a population are related, which in turn shape the pattern and frequency of observed genetic variants in extant genomes. By modeling this process and integrating out the unobserved genealogies, it is possible to infer the population's demographic history from the observed variants. However, in practice this is challenging, as individual mutations provide limited information about tree topologies and branch lengths. If many mutations were available to infer these genealogies this would not be problematic, but the expected number of observed mutations increases only logarithmically with the number of

masked using accessible genome masks from the 1000 Genomes Project (ftp://ftp.1000genomes.ebi.ac.uk/vol1/ftp/release/20130502/supporting/accessible_genome_masks).

**Funding:** The research was supported by the Wellcome Trust grants 090532/Z/09/Z and 102423/Z/13/Z, as well as the Wellcome Trust Core Award Grant Number 203141/Z/16/Z with additional support from the NIHR Oxford BRC. The views expressed are those of the author(s) and not necessarily those of the NHS, the NIHR or the Department of Health.

**Competing interests:** The authors have declared that no competing interests exist.

observed genomes, and recombination causes genealogies to change along the genome at a rate proportional to the mutation rate. As a result there is considerable uncertainty about the genealogies underlying a sample of genomes, and because the space of genealogies across the genome is vast, integrating out this latent variable is hard.

A number of approaches have been proposed to tackle this problem [reviewed in 1]. A common approximation is to treat recombination events as known and assume unlinked loci, either by treating each mutation as independent [2–7], or by first identifying tracts of genetic material unbroken by recombination [8–12]. To account for recombination while retaining power to infer earlier demographic events, it is necessary to model the genealogy directly. ARGWeaver [13] uses Markov chain Monte Carlo (MCMC) for inference, but does not allow the use of a complex demographic model, and since mutations are only weakly informative about genealogies this leaves the inferred trees biased towards the prior model and less suitable for inferring demography. Restricting itself to single diploid genomes, the Pairwise Sequentially Markovian Coalescent (PSMC) model [14] uses an elegant and efficient inference method, but with limited power to detect recent changes in population size or complex demographic events. Several other approaches exist that improve on PSMC in various ways [15–18], but they remain limited particularly in their ability to infer migration.

We here focus on the general problem of inferring demography from several whole-genome sequences, which is informative about demographic events in all but the most recent epochs [13, 14, 16]. A promising approach which so far has not been applied to this problem is to use a particle filter. Particle filters have many desireable properties [19–22], and applications to a range of problems in computational biology have started to appear [23–26]. Like MCMC methods, particle filters converge to the exact solution in the limit of infinite computational resources, are computationally efficient by focusing on realisations that are supported by the data, do not require the underlying model to be approximated, and generate explicit samples from the posterior distribution of the latent variable. Unlike MCMC, particle filters do not operate on complete realisations of the model, but construct samples sequentially, which is helpful since full genealogies over genomes are cumbersome to deal with.

To use particle filters, we use a formulation of the coalescent model in which the state is a genealogical tree at a particular genome locus, which “evolves” sequentially along the genome, rather than in evolutionary time. To avoid confusion, in this paper “time” by itself refers to the variable along which the model evolves, while evolutionary (coalescent, recombination) time refers to an actual time in the past on a genealogical tree.

Originally, particle filters were introduced for models with discrete time evolution and with either discrete or continuous state variables [19, 27]. In this paper, the latent variable is a piecewise constant sequence of genealogical trees along the genome, with trees changing only after recombination events that, in mammals, occur once every several hundred nucleotides. The observations of the model are genetic variants, which are similarly sparse. Realizations of the discrete-time model of this process (where “time” is the genome locus) are therefore stationary (remain in the same state) and silent (do not produce an observation) at most transitions, leading to inefficient algorithms. Instead, it seems natural to model the system as a Markov jump process (or purely discontinuous Markov process, [28]), a continuous-time stochastic process with as realisations piecewise constant functions $x : [1, L] \mapsto \mathbb{T}$, where $\mathbb{T}$ is the state space of the Markov process (the space of genealogical trees over a given number of genomes) and $L$ the length over which observations are made (here the genome size).

Particle filters have been generalised to continuous-time diffusions [29–31], as well as to Markov jump processes on discrete state spaces [32, 33], and hybrids of the two [34, 35], as well as to piecewise deterministic processes [36]; for a general treatment see [37, 38]. Here we focus on Markov jump processes that are continuous in both time and state space; to our knowledge the method has not been extended to this case. The algorithm we propose relies on Radon-Nikodym derivatives [see e.g. 31], and we establish criteria for choosing a finite set of "waypoints" that makes it possible to reduce the problem to the discrete-time case, while ensuring that particle degeneracy remains under control.

Although the algorithm generally works well, we found that for the CwR model we obtain biased inferences for some parameters. For example, coalescent rates for recent epochs are associated with tree nodes that persist across long genomic segments (the model exhibits "long forgetting times"), because their short descendant branches attract few recombinations. They have few informative mutations as well, and collecting these mutations therefore require long lags in the fixed-lag smoothing procedure, in turn resulting in increased particle degeneracy [39]. For discrete-time models the Auxiliary Particle Filter [40] addresses a related problem by "guiding" the particle filter towards states that are likely to be relevant in future iterations, using an approximate likelihood that depends on data one step ahead. This approach does not work well for some continuous-time models, including ours, that have no single preferred time scale. Instead we introduce an algorithm that shapes the resampling process by an approximate "lookahead likelihood" that can depend on data at arbitrary distances ahead. Using simulations we show that this substantially reduces the bias.

The particle filter generates samples from the posterior distribution of the latent variable, here the sequence of genealogies along the genome, and we infer the model parameters from this sample. One strategy is to use stochastic expectation-maximization [SEM; 41]. However, such approaches yield point estimates, ignoring any uncertainty in the inferred parameters. Combined with the bias due to self-normalized importance sampling which cause particle filters to under-sample low-rate events, this result in a non-zero probability of inferring zero event rates, which are fixed points of any SEM procedure. In principle this can be avoided by using an appropriate prior on the rate parameters. To implement this we use Variational Bayes to estimate an approximate joint posterior distribution over parameters and latent variables, partially accounting for the uncertainty in the inferred parameters, as well as providing way to explicitly include a prior. In this way zero-rate estimates are avoided, and more generally we show that this approach further reduces the bias in parameter estimates.

Applying these ideas to the coalescent-with-recombination (CwR) model, we find that the combination of lookahead filter and Variational Bayes inference enables us to analyze four diploid human genomes simultaneously, and infer demographic parameters across epochs spanning more than 3 orders of magnitude, without making model approximations beyond passing to a continuous-locus model.

The remainder of the paper is structured as follows. We first introduce the particle filter, generalise it to continuous-time and -space Markov jump processes, describe how to choose waypoints, introduce the lookahead filter, and describe the Variational Bayes procedure for parameter inference. In the results section we first introduce the continuous-locus CwR process, then discuss the lookahead likelihood, choice of waypoints and parameter inference for this model, before applying the model to simulated data, and finally show the results of analyzing sets of four diploid genomes of individuals from three human populations. A discussion concludes the paper.

## Methods

### The sequential coalescent with recombination model

The coalescent-with-recombination (CwR) process, and the graph structures that are the realisations of the process, was first described by Hudson [42], and was given an elegant mathematical description by Griffiths [43], who named the resulting structure the Ancestral

Recombination Graph (ARG). Like the coalescent process, these models run backwards in evolutionary time and consider the entire sequence at once, making it difficult to use them for inference on whole genomes. The first model of the CwR process that evolves sequentially rather than in the evolutionary time direction was introduced by Wiuf and Hein [44], opening up the possibility of inference over very long sequences. Like Griffiths' process, the Wiuf-Hein algorithm operates on an ARG-like graph, but it is more efficient as it does not include many of the non-observable recombination events included in Griffiths' process. The Sequential Coalescent with Recombination Model (SCRM) [45] further improved efficiency by modifying Wiuf and Hein's algorithm to operate on a local genealogy rather than an ARG-like structure. Besides the "local" tree over the observed samples, this genealogy includes branches to non-contemporaneous tips that correspond to recombination events encountered earlier in the sequence. Recombinations on these "non-local" branches can be postponed until they affect observed sequences, and can sometimes be ignored altogether, leading to further efficiency gains while the resulting sample still follows the exact CwR process. An even more efficient but approximate algorithm is obtained by culling some non-local branches. In the extreme case of culling *all* non-local branches the SCRM approximation is equivalent to the SMC' model [46, 47]. With a suitable definition of "current state" (i.e., the local tree including all non-local branches) these are all Markov processes, and can all be used in the Markov jump particle filter; here we use the SCRM model with tunable accuracy as implemented in [45].

The state space $\mathbb{T}$ of the Markov process is the set of all possible genealogies at a given locus. The probability measure of a complete realisation $x$ can be written as

$$\pi_x(x) = \exp\left\{ -\int B(x_s)\rho(s)\mathrm{d}s \right\} \left[ \prod_{j=1}^{|x|} \exp\left\{ -\int_{v_j}^{\tau_j} b_u(x_{s_j}) C(u)\mathrm{d}u \right\} \rho(s_j) C(\tau_j) \right] (\mathrm{d}s)^{|x|} (\mathrm{d}u)^{2|x|}. \quad (1)$$

Here $x$ is the sequence of genealogies along the genome; $|x|$ is the number of recombinations that occurred on $x$; $b_u(x_s)$ is the number of branches in the genealogy at locus $s$ at evolutionary time $u$; $B(x_s) = \int_{u=0}^{\mathrm{root}(x_s)} b_u(x_s)\mathrm{d}u$ is the total branch length of $x_s$; $\rho(s)$ is the recombination rate per nucleotide and per generation at locus $s$, so that $\rho(s)B(x_s)$ is the exit rate of the Markov process in state $x_s$; $(s_j, v_j)$ is the locus and recombination time of the $j$th recombination event; $\tau_j > v_j$ is the coalescence time of the corresponding coalescence event; and $C(u) = 1/2N_e(u)$ is the coalescence rate in generation $u$. See Appendix ("The sequential coalescent with recombination process") for more details. The distribution $\pi_x(x)$ has a density with respect to the Lebesgue measure $(\mathrm{d}s)^{|x|}(\mathrm{d}u)^{2|x|}$, because each of the $|x|$ recombination events is associated with a sequence locus, a recombination time, and a coalescent time.

Mutations follow a Poisson process whose rate at $s$ depends on the state $x_s$ via $\mu(s)B(x_s)$ where $\mu(s)$ is the mutation rate at $s$ per nucleotide and per generation. Mutations are not observed directly, but their descendants are; a complete observation is represented by a set $y = \{(s_j, A_j)\}_{j=1,\ldots,|y|} \in \mathcal{Y}$ where $s_j \in [1, L)$ is the locus of mutation $j$, and $A_j \in \{0, 1\}^S$ are the wildtype (0) and alternative (1) alleles observed in the $S$ samples. The conditional probability measure of the observations $y$ given a realisation $x$ is

$$\pi(y|X = x) = \frac{1}{|y|!} \exp\left\{ -\int B(x_s)\mu(s)\mathrm{d}s \right\} \left[ \prod_{j=1}^{|y|} P(A_j | x_{s_j}, \mu(s_j)) \right] (\mathrm{d}s)^{|y|} \qquad (2)$$

where $P(A|x_s, \mu)$ is the probability of observing the allelic pattern $A$ given a genealogy $x_s$ and a mutation rate $\mu$ per nucleotide and per generation; this probability is calculated using Felsenstein's peeling algorithm [48]. Note that $B(x_s)\mu(s) = \sum_{A \neq (0,\ldots,0)} P(A|x_s, \mu(s))$.

## Particle filters

Particle filters methods, also known as Sequential Monte Carlo (SMC) [22], generate samples from complex probability distributions with high-dimensional latent variables. An SMC method uses importance sampling (IS) to approximate a target distribution using weighted random samples (particles) drawn from a tractable distribution. We briefly review the discrete-time case. Suppose that particles $\{(x^{(i)}, w^{(i)})\}_{i=1,\ldots,N}$, approximate a distribution with density $p(x)$, such that

$$E_p[f(X)] = \int f(x)p(x)\mathrm{d}x \approx \frac{1}{\sum_{i=1}^{N} w^{(i)}} \sum_{i=1}^{N} w^{(i)} f(x^{(i)}) \tag{3}$$

for any bounded continuous function $f$, where $X \sim p(x)dx$. Here and in the remainder, we use "approximate" and $\approx$ to mean that $X_N \sim \Sigma w^{(i)}\, \delta_{x^{(i)}}(x)$ converges in distribution to $X \sim p(x)\mathrm{d}x$ and equality holds in (3) as $N \to \infty$; and summations without an index are over $N$ particles indexed by $i$. Under mild conditions (i.e., $q(x)/p(x)$ must exist almost everywhere and be absolutely continuous) we can use IS to obtain particles approximating another distribution $q(x)\mathrm{d}x$:

$$E_q[f(X)] = \int f(x)\frac{q(x)}{p(x)}p(x)\mathrm{d}x \approx \frac{1}{\sum w^{(i)}} \sum w^{(i)} \frac{q(x^{(i)})}{p(x^{(i)})} f(x^{(i)}) \approx \frac{1}{\sum \tilde{w}^{(i)}} \sum \tilde{w}^{(i)} f(x^{(i)}),$$

where $\tilde{w}^{(i)} := w^{(i)} q(x^{(i)})/p(x^{(i)})$, and the last step holds because $\sum \tilde{w}^{(i)} / \sum w^{(i)} \approx E_p[q/p] = E_q[1] = 1$. This shows that $\{(x^{(i)}, \tilde{w}^{(i)})\}$ approximate $q(x)\mathrm{d}x$. The normalisation ensures that any constant factor in $w^{(i)}$ drops out, so that it is sufficient to know the ratio $q(x)/p(x)$ up to a constant. A particle filter builds the desired distribution sequentially, making it suited to hidden Markov models, for which the joint distribution of latent variables $X$ and observations $Y$ has the form

$$P(X = x_{1\cdot\cdot s}) = p(x_1)p(x_2|x_1)\cdots p(x_s|x_{s-1}) \tag{4}$$

$$P(Y = y_{1\cdot\cdot s}|X = x_{1\cdot\cdot s}) = g(y_1|x_1)\cdots g(y_s|x_s) \tag{5}$$

Here $1 \cdot \cdot\, s$ denotes the set $\{1, 2, \ldots, s\}$, and $x = x_{1\cdot\cdot s} = (x_1, x_2, \ldots, x_s)$ and $y = y_{1\cdot\cdot s}$ are vectors. Let $\{(x^{(i)}, w^{(i)})\}$ be particles approximating the target distribution $P(X_{1\cdot\cdot s} = x_{1\cdot\cdot s}|Y_{1\cdot\cdot s} = y_{1\cdot\cdot s})$, which for brevity we write as $P(x_{1\cdot\cdot s}|y_{1\cdot\cdot s})$. If $\tilde{x}^{(i)}$ is the vector obtained by extending $x^{(i)}$ with a sample from $P(x_{s+1}|x_s^{(i)})$, then from (4) and (5) it follows that $\{(\tilde{x}^{(i)}, w^{(i)})\}$ approximate $P(x_{1\cdot\cdot s+1}|y_{1\cdot\cdot s}) \propto P(x_{1\cdot\cdot s+1}, y_{1\cdot\cdot s})$. Now, $P(x_{1\cdot\cdot s+1}|y_{1\cdot\cdot s+1}) \propto P(x_{1\cdot\cdot s+1}, y_{1\cdot\cdot s+1}) = P(x_{1\cdot\cdot s+1}, y_{1\cdot\cdot s})g(y_{s+1}|x_{s+1})$, so that using IS and setting

$$\tilde{w}^{(i)} = w^{(i)} g(y_{s+1}|\tilde{x}_{s+1}^{(i)}) \tag{6}$$

we obtain particles $\{(\tilde{x}^{(i)}, \tilde{w}^{(i)})\}$ that approximate $P(x_{1\cdot\cdot s+1}|y_{1\cdot\cdot s+1})$. This shows how to sequentially construct particles that approximate the target distribution $P(x_{1\cdot\cdot L}|y_{1\cdot\cdot L})$. Instead of sampling from $p(x_{s+1}|x_s^{(i)})$, any proposal distribution $q(x_{s+1}|x_s^{(i)}, y_{1\cdot\cdot L})$ (subject to conditions) can be used, which is advantageous if $q$ is easier to sample from, is closer to the target distribution, or has heavier tails than $p$. Again, IS accounts for the change in sampling distribution, resulting in

$$\tilde{w}^{(i)} = w^{(i)} g(y_{s+1}|\tilde{x}_{s+1}^{(i)}) \frac{p(\tilde{x}_{s+1}^{(i)}|x_s^{(i)})}{q(\tilde{x}_{s+1}^{(i)}|x_s^{(i)})}. \tag{7}$$

For now we will choose $q$ to be independent of $y$. Because samples from $q$ do not follow the desired target $P(x|y)$, the fraction of particles close to the target's mode diminishes exponentially at each iteration until (3) fails altogether. To address this, we occasionally draw samples from the approximating distribution itself, assigning each resampled particle weight $1/N$—interestingly, if we interpret fitness as (proportional to) the likelihood $g(y_{s+1}|\tilde{x}_{s+1}^{(i)})$, this is the same process that is used in the Wright-Fisher model with selection to describe how fitness differences shape an evolving constant-size population [49]. Doing this tends to remove particles that have drifted from the mode of the target and have low weight, and duplicates particles with large weights, while (3) remains valid. Although resampling substantially decreases the future variance of (3), it increases the variance at the current iteration. To avoid increasing this variance unnecessarily, resampling is performed only when the estimated sample size, defined as $ESS = (\Sigma w^{(i)})^2/\Sigma(w^{(i)})^2$, drops below a threshold, e.g. $N/2$. In addition, we use systematic resampling to minimize the variance that is introduced when resampling is performed [50]. This leads to Algorithm 1 [19].

Note that the algorithm can be seen as a recipe to transform a sample from $P(X)$ to a sample from $P(X)P(Y|X)/P(Y) = P(X|Y)$, that is, an application of Bayes' theorem. Following this interpretation we will refer to $P(X)$ as the prior distribution, and $P(X|Y)$ as the posterior.

The algorithm generates an approximation to $P(x_{1{\cdot\cdot}s}|y_{1{\cdot\cdot}s})$ rather than $P(x_s|y_{1{\cdot\cdot}s})$, but we follow [22] in calling it a particle filter algorithm instead of a smoothing algorithm (although our use of fixed-lag distributions for parameter estimation is a partial smoothing operation).

The marginal likelihood can be estimated (although with high variance, see [51]) by setting the weights to $N^{-1}\sum_i w_s^{(i)}$ rather than $N^{-1}$ when particles are resampled. This makes the weights asymptotically normalized, so that (3) becomes $E_{P(X,Y=y)}[f] \approx \Sigma_i\, w^{(i)} f(x^{(i)})$, and $P(Y = y) = \int P(x, y)\mathrm{d}x = E_{P(X,Y=y)}[1] \approx \sum_i w_L^{(i)}$.

**Algorithm 1** Particle filter

```
Input: $y_{1{\cdot\cdot}L}$
Output: Particles $\{(x_{1{\cdot\cdot}L}^{(i)}, w_L^{(i)})\}$ approximating $P(x_{1{\cdot\cdot}L}|y_{1{\cdot\cdot}L})$
  $w_0^{(i)} \leftarrow 1/N$, $x_0^{(i)} \leftarrow \emptyset (i = 1, \ldots, N)$
  For $s$ from 0 to $L$ - 1
    Loop invariant: $\{(x_{1{\cdot\cdot}s}^{(i)}, w_s^{(i)})\} \sim p(x_{1{\cdot\cdot}s}|y_{1{\cdot\cdot}s})$
    If $ESS$ < $N$/2:
      Resample, with replacement, $\{x_{1{\cdot\cdot}s}^{(i)}\}$ proportional to $\{w_s^{(i)}\}$
      $w_s^{(i)} \leftarrow N^{-1}(i = 1, \ldots, N)$
    For $i$ from 1 to $N$:
      Sample $x_{s+1}^{(i)} \sim q(x_{s+1}|x_s^{(i)})$
      $w_{s+1}^{(i)} \leftarrow w_s^{(i)} \frac{p(x_{s+1}^{(i)}|x_s^{(i)})}{q(x_{s+1}^{(i)}|x_s^{(i)})} g(y_{s+1}|x_{s+1}^{(i)})$.
```

## Continuous-time and -space Markov jump processes

For the hidden process we now consider Markov jump processes, which have as realisations piecewise constant functions $x : [1, L) \mapsto \mathbb{T}$ where $\mathbb{T}$ is the state space of the Markov process. Recall that in the model we consider, $\mathbb{T}$ is the space of rooted genealogical trees with branch lengths. Let $(\mathcal{X}, \mathcal{F}_x, \pi_x)$ be a probability space, where $\mathcal{X} = \mathbb{T}^{[1,L)}$ is the space of possible realisations of the hidden stochastic process $X = \{X_s\}_{s \in [1,\, L)}$, $\mathcal{F}_x \subset \mathcal{P}(\mathcal{X})$ is the $\sigma$-algebra of events, and $\pi_x(X)$ is the probability measure on $\mathcal{X}$ induced by the stochastic process $X$. See the Appendix ("Conditional distributions and the Markov property") for some remarks on how to define a Markov model when the phase space $\mathbb{T}$ is uncountable.

The complete model is defined by specifying the observation process. We consider models where observations $Y$ are generated by a Poisson process whose intensity at time (i.e. locus) $s$

depends on $X_s$ [a Cox process, see e.g. 52]. The space of observations $\mathcal{Y}$ consists of finite subsets of $[1, L) \times M$, where $M$ is a discrete set of potential events, each of which may occur at some $s \in [1, L)$. For a full observation $y = ((\tilde{s}_1, m_1), \ldots, (\tilde{s}_k, m_k)) \in \mathcal{Y}$ we write $|y| \coloneqq k$ for the number of events in $y$. Writing $\lambda(y)$ for the Lebesgue measure $(\mathrm{d}s)^{|y|}$, the emission distribution $\pi(Y|X = x)$ has a density $r(y|x)$ relative to $\lambda(y)$. For Cox processes this density has the form

$$r(y|x) = \frac{1}{|y|!}\exp\left(-\int_{s=1}^{L} r(x_s)\right)\prod_{i=1}^{|y|} r(\tilde{s}_i, m_i|x_{\tilde{s}_i}) \tag{8}$$

where $r(s, m|x_s)$ is the rate at which event $m$ occurs at time $s$ conditional on $X_s = x_s$ and

$$r(x_s) \coloneqq \sum_{m\in M} r(s, m|x_s)\mathrm{d}s \tag{9}$$

is the intensity of the emission Poisson process at $s$ conditional on $X_s = x_s$. The probability space for the joint process is $(\mathcal{X} \times \mathcal{Y}, \mathcal{F}, \pi)$, and the posterior distribution of interest is $\pi$ conditioned on an observation $y \in \mathcal{Y}$, written as $\pi(X|Y = y)$.

The *absence* of events in an interval $s \in [a, b)$ is also informative about the latent variable through the exponential factor in (8). In practice however, not all intervals may have been observed, so that events may or may not have occurred in these intervals. Assuming that the "observation process" is independent of the Markov jump process $X$, such unobserved intervals can simply be left out of integral (8).

Some more notation is needed to describe the Markov jump process version of algorithm 1. As above $\pi_x$ denotes the prior distribution of the latent variable $X$, and $\xi_x$ denotes the proposal distribution, both Markov processes on $\mathcal{X}$, playing the role of $p(x)$ and $q(x)$ in the discrete case. We write $a{:}b$ for the interval $[a, b) \subset \mathbb{R}$, and $\alpha^{a:\,b}$ for the restriction of a measure or function $\alpha$ to $a{:}b$; similarly $y_{a:b} \coloneqq y \cap([a, b) \times M)$ and $X_{a:b} \coloneqq \{X_s\}_{s\in[a,b)}$. The particle filter algorithm uses the notation $(\mathrm{d}\alpha/\mathrm{d}\beta)(x)$ for distributions $\alpha$ and $\beta$ to denote their Radon-Nikodym derivative: the ratio of their density functions with respect to a common reference measure, evaluated at $x$. To simplify notation we write the Radon-Nikodym derivative of two conditional distributions $\alpha(X|\mathcal{G})$ and $\beta(X|\mathcal{G})$ at $x$ as $(\mathrm{d}\alpha/\mathrm{d}\beta)(x|\mathcal{G})$, and we also do not explicitly restrict distributions to their appropriate intervals when this is clear from the context, so that we write for example $(\mathrm{d}\pi/\mathrm{d}\lambda)(y_{s_j:s_{j+1}}|X_{s_j:s_{j+1}} = x)$ instead of $(\mathrm{d}\pi^{s_j:s_{j+1}}(Y|X_{s_j:s_{j+1}} = x)/\mathrm{d}\lambda^{s_j:s_{j+1}})(y_{s_j:s_{j+1}})$. With this notation we can formulate Algorithm 2.

**Algorithm 2** Particle filter for Markov jump processes

```
Input: $y_{1:L} \in \mathcal{Y}$; waypoints 1 = $s_0$ < $s_1$ < ... < $s_K$ = $L$.
Output: Particles $\{(x^{(i)}_{1:L}, w^{(i)}_L)\}$ approximating the posterior distribution
$\pi(X|Y = y_{1:L})$
    $w^{(i)}_1 \leftarrow N^{-1}$, $x^{(i)}_{1:1} \leftarrow \emptyset$ ($i$ = 1, ..., $N$)
  For $j$ from 0 to $K$ - 1
    Loop invariant: $\{(x^{(i)}_{1:s_j}, w^{(i)}_{s_j})\} \approx \pi(X_{1:s_j}|Y_{1:s_j} = y_{1:s_j})$
    If $ESS(\{w^{(i)}_{s_j}\}) < N/2$:
      Resample $\{x^{(i)}_{1:s_j}\}$ with probabilities proportional to $\{w^{(i)}_{s_j}\}$
      $w^{(i)}_{s_j} \leftarrow N^{-1}$ ($i$ = 1, ..., $N$)
    For $i$ from 1 to $N$:
      Sample $x^{(i)}_{s_j:s_{j+1}} \sim \xi_x(X_{s_j:s_{j+1}}|X_{s_j} = x^{(i)}_{s_j})$
      $w^{(i)}_{s_{j+1}} \leftarrow w^{(i)}_{s_j} \frac{\mathrm{d}\pi_x}{\mathrm{d}\xi_x}(x^{(i)}_{s_j:s_{j+1}}|X_{s_j} = x^{(i)}_{s_j}) \frac{\mathrm{d}\pi}{\mathrm{d}\lambda}(y_{s_j:s_{j+1}}|X_{s_j:s_{j+1}} = x^{(i)}_{s_j:s_{j+1}})$.
```

The choice of waypoints $s_1, \ldots, s_K$ is discussed below; in particular they need not be the same as the event loci $\tilde{s}_1, \ldots, \tilde{s}_{|y|}$ of the observation $y$. Note that there is no initialization step;

instead, initially $x_{1:1}^{(i)} = \emptyset$, and the first sample will be drawn from $\xi$ conditioned on an empty set, i.e. the unconditional distribution. The loop invariant holds when $j = 0$ since $1{:}s_0 = \emptyset$. As with Algorithm 1 it is possible to estimate the likelihood density $\pi_\theta(y_{1:L})$ by replacing the factors $N^{-1}$ with $N^{-1}\sum_i w_{s_j}^{(i)}$; then the likelihood density w.r.t. $\lambda(\mathrm{d}y) = (\mathrm{d}s)^{|y|}$ is approximated by $\sum_i w_L^{(i)}$.

Note that by the nature of Markov jump processes, particles that start with identical latent variables have a positive probability of remaining identical after a finite time. Combined with resampling, this causes a considerable number of particles to have one or more identical siblings. For computational efficiency we represent such particles once, and keep track of their multiplicity $k$. When evolving a particle with multiplicity $k > 1$, we increase the exit rate $k$-fold, and when an event occurs one particle is spawned off while the remaining $k - 1$ continue unchanged.

## Using lookahead to improve the particle filter

At the $j$th iteration, Algorithm 2 uses data up to waypoint $s_j$ to build particles approximating $\pi(X_{1:s_j} | Y_{1:s_j} = y_{1:s_j})$. This is reasonable as $\pi(X_{1:s_j} | y_{1:s_j})$ is independent of data beyond $s_j$. However, not all particles are equally important for approximating subsequent posteriors, which suggests to emphasise particles that will be relevant in future at the expense of those relevant only to $\pi(X_{1:s_j} | y_{1:s_j})$. This echoes the justification of resampling: although resampling increases the variance of the approximation to the current partial posterior, the variance at subsequent iterations by increasing the number of particles that are likely to contribute to future distributions. For discrete-time models $p(X_{1:n}|y_{1:n})$, the Auxiliary Particle Filter (APF) [40] implements this intuition by targeting a resampling distribution [53], which includes a "lookahead" factor $\tilde{p}(y_{i+1}|x_i)$ approximating the probability of observing data $y_{i+1}$ given the current state $x_i$. Importance sampling is used to keep track of the desired distribution $p(X_{1:i}|y_{1:i})$.

In the continuous-time context it is natural to look an arbitrary distance ahead. Similar to APF, the lookahead distribution can be conditioned on the current state only, and must be an approximation of the true distribution. It should be heavy-tailed with respect to the true distribution to ensure that the variance of the estimator remains finite [22], which implies that the distribution should not depend on data too far beyond $s$; what is "too far" depends on how well the lookahead distribution approximates the true distribution.

The lookahead distribution is only evaluated on a fixed observation $y$, and is used to quantify the plausibility of a current state $x_s^{(i)}$, rather than to define a distribution over $y$. For this reason we call it a lookahead *likelihood*. In fact, for correctness of the algorithm it is not necessary that this likelihood derives from a probability distribution. We define the lookahead likelihood as a family of functions $h^s(y_{s:L}|x_s) : \mathcal{Y}_{s:L} \times \mathbb{T} \to \mathbb{R}$, and an associated family of unnormalized distributions $\tilde{\pi}^s(x_{1:s}, y_{1:L}) = \pi^{1:s}(x_{1:s}, y_{1:s}) h^s(y_{s:L}|x_s) \lambda^{s:L}(y_{s:L})$ on $\mathcal{X}_{1:s} \times \mathcal{Y}$. The functions $h^s$ can be chosen arbitrarily, except that $h^s(\cdot, x_s)\lambda^{s:L}$ must be absolutely continuous w.r.t. $\pi^{s:L}(\cdot|X_s = x_s)$ to ensure that importance sampling is justified. The lookahead Algorithm 3 keeps track of two sets of weights, which together with a single set of samples form two sets of particles that approximate the resampling and target distributions.

**Algorithm 3** Markov-jump particle filter with lookahead

```
Input: $y_{1:L} \in \mathcal{Y}$; waypoints $1 = s_0 < s_1 < \ldots < s_K = L$.
Output: Particles $\{(x_{1:L}^{(i)}, w_L^{(i)})\}$ approximating $\pi(X | Y_{1:L} = y_{1:L})$
  $w_1^{(i)} \leftarrow 1/N$, $v_1^{(i)} \leftarrow 1/N$, $x_{1:1}^{(i)} \leftarrow \emptyset$ ($i = 1, \ldots, N$)
  For $j$ from 0 to $K - 1$
    Loop invariant: $\{(x_{1:s_j}^{(i)}, w_{s_j}^{(i)})\} \approx \pi^{1:s_j}(X_{1:s_j} | Y_{1:s_j} = y_{1:s_j})$
Loop invariant: $\{(x^{(i)}_{1:s_j}, v^{(i)}_{s_j})\} \approx \tilde{\pi}^{s_j}(X_{1:s_j} | Y = y_{1:L})$
If $ESS(\{v^{(i)}_{s_j}\}) < N/2$:
  Resample $\{x^{(i)}_{1:s_j}\}$ with probabilities proportional to $\{v^{(i)}_{s_j}\}$
  $w^{(i)}_{s_j} \leftarrow N^{-1} w^{(i)}_{s_j} / v^{(i)}_{s_j}$ (i = 1, ..., N)
  $v^{(i)}_{s_j} \leftarrow N^{-1}$ (i = 1, ..., N)
For i from 1 to N:
  Sample $x^{(i)}_{s_j:s_{j+1}} \sim \xi_x(X_{s_j:s_{j+1}} | X_{s_j} = x^{(i)}_{s_j})$
  $w^{(i)}_{s_{j+1}} \leftarrow w^{(i)}_{s_j} \frac{d\pi_x}{d\xi_x}(x^{(i)}_{s_j:s_{j+1}} | X_{s_j} = x^{(i)}_{s_j}) \frac{d\pi}{d\lambda}(y_{s_j:s_{j+1}} | X_{s_j:s_{j+1}} = x^{(i)}_{s_j:s_{j+1}})$
  $v^{(i)}_{s_{j+1}} \leftarrow v^{(i)}_{s_j} \frac{d\pi_x}{d\xi_x}(x^{(i)}_{s_j:s_{j+1}} | X_{s_j} = x^{(i)}_{s_j}) \frac{d\tilde{\pi}^{s_{j+1}}}{d\tilde{\pi}^{s_j}}(y_{s_j:L} | X_{s_j:s_{j+1}} = x^{(i)}_{s_j:s_{j+1}})$
```

(see Appendix, "Proof of Algorithm 3".) To implement the lookahead particle filter we need a tractable approximate likelihood of future data given a current genealogy. To do this we simplify the full likelihood, and ignore all data except for a digest of singletons and doubletons that are informative of the topology and branch lengths near the tips of the genealogy—in particular, singletons are informative of terminal branch lengths, and doubletons identify the existence of nodes with precisely two descendants ("cherries"). This digest consists of the distance $s_i$ to the nearest future singleton for each haploid sequence, and $\leq n/2$ mutually consistent cherries $c_k = (a_k, b_k)$ identified by their two descendants $a_k$, $b_k$, together with loci $s'_k \leq s''_k$ where their first and last supporting doubleton were observed (Fig 1a). Under some simplifying assumptions we derive an approximation of the likelihood $h^s(\{t_i\}, \{c_k, s'_k, s''_k\} | x_s)$ of the current genealogy given these data; see Appendix ("A lookahead likelihood") for details.

## Choosing waypoints

The choice of waypoints $s_j$ can significantly impact the performance of the algorithm: choosing too few increases the variance of the approximation, and choosing too many slows down the algorithm without increasing its accuracy. Waypoints determine where the algorithms might perform a resampling step. A high density of waypoints is therefore always acceptable, but a low density may result in particle degeneracy. Choosing a waypoint at every event ensures that any weight variance induced at these sites is mitigated, but there is still the opportunity for weight variance to build up between events.

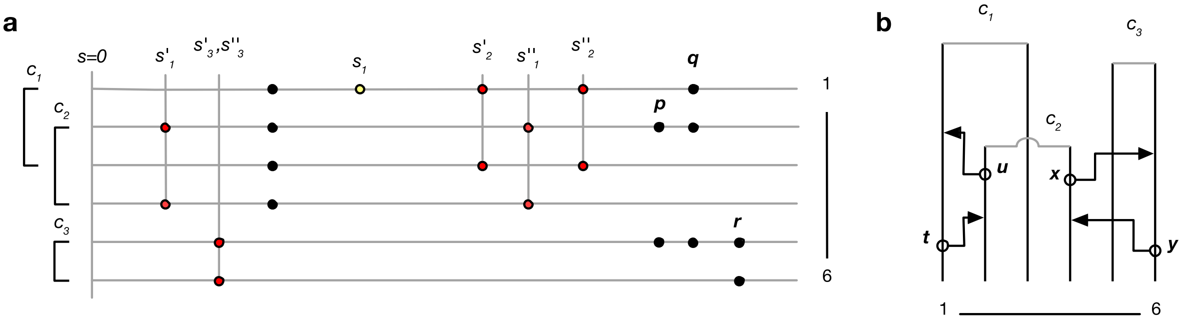

**Fig 1.** a. Example of data digest. Lines represent genomes of 6 lineages, circles observed genetic variants. Of the data shown, one singleton (yellow) and five doubletons (red) contribute to the digest. Cherry $c_3$ is supported by a single doubleton; $r$ does not contribute because the mutation patterns $p$ and $q$ are incompatible with $c_3$. Similarly, $p$ does not contribute because it is incompatible with $c_2$ and $c_3$. **b**. Partial genealogy (unbroken lines) over 6 lineages. Open circles and arrowheads represent potential recombination and coalescence events that would change the terminal branch length for lineage 1 ($t$,$u$), and remove cherry $c_3$ ($x$,$y$).

https://doi.org/10.1371/journal.pone.0247647.g001

If $\xi_x = \pi_x$, particle weights diverge only because different particles $(x_{1:s}^{(i)}, w_s^{(i)})$ experience a different total intensity $r(x_s^{(i)})$ of observed events. If $ESS_0$ is the current estimated sample size, then under some assumptions, along an interval of length $L$ where no events occur we have

$$ESS \geq ESS_0 e^{-\sigma^2 L^2} \tag{10}$$

(see Appendix, "Particle weight variance"), where $\sigma^2$ the variance of the total event intensity $r(x_s)$ (9) under the prior $\pi_x(x)$. Therefore, if we choose waypoints at every event, adding additional waypoints so that they are never more than a distance $1/\sqrt{2\sigma^2}$ apart, the ESS will not drop more than a factor $\sqrt{1/e} \approx 0.6$ between waypoints, and particle degeneracy is avoided.

To apply this to our situation, assume a panmictic population with constant diploid effective population size $N_e$. The variance of the total coalescent branch length in a sample of $n$ individuals is $(4N_e)^2 \sum_{i=1}^{n-1} i^{-2}$ [54]. The variance of total mutation intensity $\sigma^2$ is obtained by multiplying this by $\mu^2$, since the rate of mutations on the coalescent tree is $\mu$ times the total branch length. Rewriting this in terms of the heterozygosity $\theta = 4N_e\,\mu$, and approximating the sum with $\sum_{i=1}^{\infty} i^{-2} = \pi^2/6$ gives $\sigma^2 = \theta^2\, \pi^2/6$, and a minimum waypoint distance of $1/\sqrt{2\sigma^2} = \sqrt{3}/\pi\theta \approx 1/2\theta$.

Because the assumptions mentioned above are in practice only met approximately, this minimum waypoint density should be taken as a guide; breakdown of the assumptions can be monitored by tracking the *ESS*, increasing the density of waypoints if necessary.

## Parameter inference

Parameters can be inferred by stochastic expectation maximization (SEM), which involves maximizing the expected log likelihood over the posterior distribution of the latent variable. The probability density for a Poisson process is $\frac{1}{c!}\theta^c e^{-q\theta}$, where $c$ is the event count, and $\theta$ is the rate of events per unit of "opportunity" $q$, measured in units of time or space or some combination of them. The expected log likelihood $c \log \theta - q\theta$ (ignoring constants) is maximized for $\theta = c/q$, where $c$ and $q$ are the *expected* event count and opportunity. We consider Markov jump processes $X_s$ with parameters $\theta$ and distribution

$$\pi_x(x|\theta) = \prod_i \frac{1}{|x|_i!} \exp\{-\theta_i Q_i(x)\} \theta_i^{|x|_i} \mathrm{d}x, \tag{11}$$

where $|x|_i$ is the event count and $Q_i(x)$ is the total opportunity for events of type $i$ in realisation $x$; both can be random variables. Similar to the Poisson case, the parameters maximizing the expected log likelihood are

$$\theta'_{i,\mathrm{EM}} = \frac{\mathbb{E}_{\pi(x|y,\theta)}[\ |x|_i\ ]}{\mathbb{E}_{\pi(x|y,\theta)}[\ Q_i(x)\ ]} \tag{12}$$

The expectations can be computed by using samples over $x \sim \pi(x|y, \theta)$ as approximated by Algorithm 3.

To evaluate the expectations above we do not use the complete set of events in the full realisations $x$, since resampling causes early parts of $x$ to become degenerate due to "coalescences" of the particle's history of events along the sequence, which would lead to high variance of the estimates. Using only the most recent events is also problematic as these have not been shaped by many observations and mostly follow the prior $\pi_x(x|\theta)$, resulting in highly biased estimates. Smoothing techniques such as two-filter smoothing [55] cannot be used here since finite-time transition probabilities are intractable. For discrete-time models fixed-lag smoothing is often effective [39]. For our model the optimal lag depends on the epoch, as the age of tree nodes strongly influence their correlation distance. For each epoch we determine the correlation distance empirically, and for the lag we use this distance multiplied by a factor $\alpha$; we obtain good results with $\alpha = 1$.

Particularly in cases where some event types are rare, Variational Bayes can improve on EM by iteratively estimating posterior distributions rather than point estimates of $\theta$. A tractable algorithm is obtained if the joint posterior $\pi(x, \theta|y)\mathrm{d}x\mathrm{d}\theta$ is approximated as a product of two independent distributions over $x$ and $\theta$, and an appropriate prior over $\theta$ is chosen. For the Poisson example above, combining a $\Gamma(\theta|\alpha_0, \beta_0)$ prior with the likelihood $\theta^c\, e^{-q\theta}$ results in a $\Gamma(\theta|\alpha_0 + c, \beta_0 + q)$ posterior. Similarly, with this choice the Variational Bayes approximation results in an inferred posterior distribution of the form

$$\theta'_{i,\mathsf{VB}} \sim \Gamma(\alpha_0 + \mathbb{E}[\ |x|_i\ ], \beta_0 + \mathbb{E}[\ Q_i(x)\ ]) \tag{13}$$

where expectations are taken over $x \sim \int \pi(x|y, \theta)\pi(\theta)\mathrm{d}\theta$, and $\pi(\theta)$ is the current posterior over $\theta$. It would appear that obtaining samples $x$ from this distribution is intractable. However, if $\pi(\theta)$ is a Gamma distribution, $\theta$ can be integrated out analytically in the likelihood $\pi(x, y|\theta)\Gamma(\theta|\alpha, \beta)$, resulting in an expression that is identical to the likelihood of the point estimate $\theta_i = \alpha_i/\beta_i$ except for an additional scaling factor $e^{\psi(\alpha}{}_{i})/\alpha_i$ for each event of type $i$ in $x$, where $\psi$ is the digamma function. These scaling factors render the normalization constant of the likelihood intractable, but fortunately SMC algorithms only require densities to be defined up to normalization. As a result, Algorithm 3 can be used to generate samples from this distribution at no additional computational cost. See the Appendix ("Variational Bayes for Markov Jump processes") for more details.

Explicitly, for model (1) the parameters $\theta' = (\rho_{\mathsf{EM}}, C_{\mathsf{EM}})$ maximising $\mathbb{E}[\log \pi_x(x|\theta')]$, where the expectation is taken over the posterior $x \sim \pi(x|y, \theta)\mathrm{d}x$ as approximated by Algorithm 3, is

$$\rho'_{\mathsf{EM}} = \frac{\mathbb{E}[\ |x|\ ]}{\mathbb{E}[\int B(x_s)\mathrm{d}s]} \qquad \text{and} \qquad C'_{\mathsf{EM}} = \frac{\mathbb{E}[\ |x|\ ]}{\mathbb{E}[\sum_{j=1}^{|x|}\int_{v_j}^{\tau_j} b_u(x_{s_j})]}, \tag{14}$$

where $\theta = (\rho, C)$ is the vector of current parameter estimates. Note that $C'_{\mathsf{EM}}$ in (14) is constant in evolutionary time. In practice we maximize (1) with respect to piecewise constant functions $C'_{\mathsf{EM}}(t)$, which yields

$$C'_{\mathsf{EM}}(t) = \frac{\mathbb{E}[\ |x|_{v,\tau}\ ]}{\mathbb{E}[\sum_{j=1}^{|x|}\int_{u\in[v_j,\tau_j)\cap[v,\tau)} b_u(x_{s_j})\mathrm{d}u]} \tag{15}$$

for $t \in [v, \tau)$, where $|x|_{v,\tau}$ denotes the number of coalescent events in $x$ that occur in the epoch $[v, \tau)$. Similarly, a Variational Bayes inference procedure uses

$$\rho'_{\mathsf{VB}} \sim \Gamma(\alpha_\rho + \mathbb{E}[\ |x|\ ], \beta_\rho + \mathbb{E}[\int B(x_s)\mathrm{d}s]) \tag{16}$$

$$C'_{\mathsf{VB}} \sim \Gamma(\alpha_C + \mathbb{E}[\ |x|\ ], \beta_C + \mathbb{E}[\sum_{j=1}^{|x|}\int_{v_j}^{\tau_j} b_u(x_{s_j})]) \tag{17}$$

where expectations are taken over $x \sim \int \pi(x|y, \theta)p(\theta)\mathrm{d}\theta$, where $p(\theta)$ is the posterior parameter distribution (16 and 17) of the previous iteration, and $\alpha_\rho, \beta_\rho, \alpha_C, \beta_C$ parameterize the prior distributions $\rho \sim \Gamma(\alpha_\rho, \beta_\rho)$ and $C \sim \Gamma(\alpha_C, \beta_C)$.

# Results

## Simulation study

We implemented the model and algorithm above in a Python/C++ program `SMCSMC` (Sequential Monte Carlo for the Sequentially Markovian Coalescent) and assessed it on simulated and real data.

To investigate the effect of the lookahead particle filter, we simulated four 50 megabase (Mb) diploid genomes under a constant population-size model ($N_e = 10,000$, $\mu = 2.5 \times 10^{-8}$ and $\rho = 10^{-8}$, both per generation and per site, generation time $g = 30$ years). We inferred population sizes $N_e$ through evolutionary time, defined as the inverse of twice the instantaneous coalescent rate, as a piecewise constant function across 9 epochs (with boundaries at 400, 800, 1200, 2$k$, 4$k$, 8$k$, 20$k$, 40$k$ and 60$k$ generations) using particle filters Algorithms 2 and 3, as well as a recombination rate, which was taken to be constant through evolutionary time (and along the genome). Although recombination rate can be inferred, we here focus on the accuracy of the inferred $N_e$ through evolutionary time. Observations are often available as unphased genotypes, and we assessed both algorithms using phased and unphased data, using the same simulations for both. Experiments were run for 15 EM iterations and repeated 15 times (Fig 2a).

On phased data (Fig 2a, top rows), $N_e$ values inferred without lookahead show a strong positive bias in recent epochs, corresponding to a negative bias in the inferred coalescence rate. Increasing the number of particles reduces this bias somewhat. By contrast, the lookahead filter shows no discernable bias on these data, even for as little as 1, 000 particles. On unphased data (Fig 2a, bottom rows), the default particle filter continues to work reasonably well; in fact the bias appears somewhat reduced compared to phased data analyses, presumably because integrating over the phase makes the likelihood surface smoother, reducing particle degeneracy. By contrast, the lookahead particle filter shows an increased bias on these data compared to the default implementation. This is presumably because of the reliance of the lookahead likelihood on the distance to the next singleton; this statistic is much less informative for

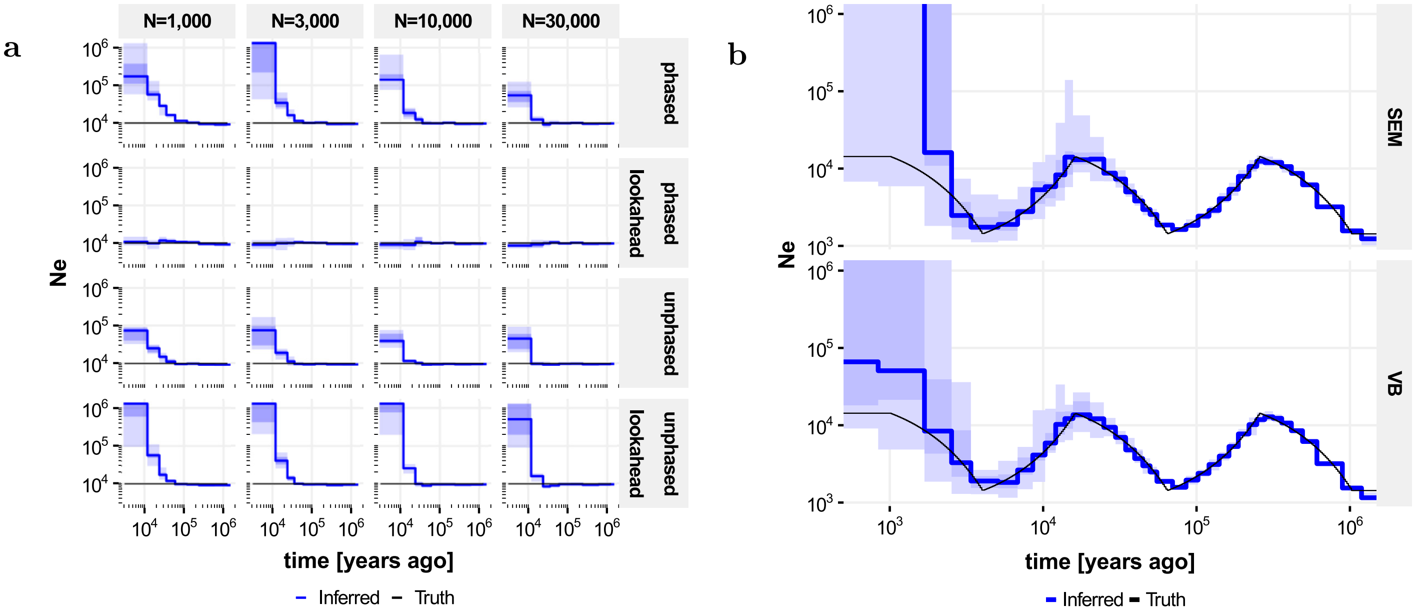

**Fig 2. Accuracy of population size inferences in simulated data.** Shown are true population sizes (black) and median inferred population sizes across 15 independent runs (blue); shaded areas denote quartiles and full extent. **a** Impact of lookahead, phasing and number of particles on the bias in population size estimates for recent epochs, for data simulated under a constant population size model. **b** Inference in the "zigzag" model on phased data using lookahead and 30, 000 particles, comparing inference using stochastic Expectation Maximization (SEM) and Variational Bayes (VB).

https://doi.org/10.1371/journal.pone.0247647.g002

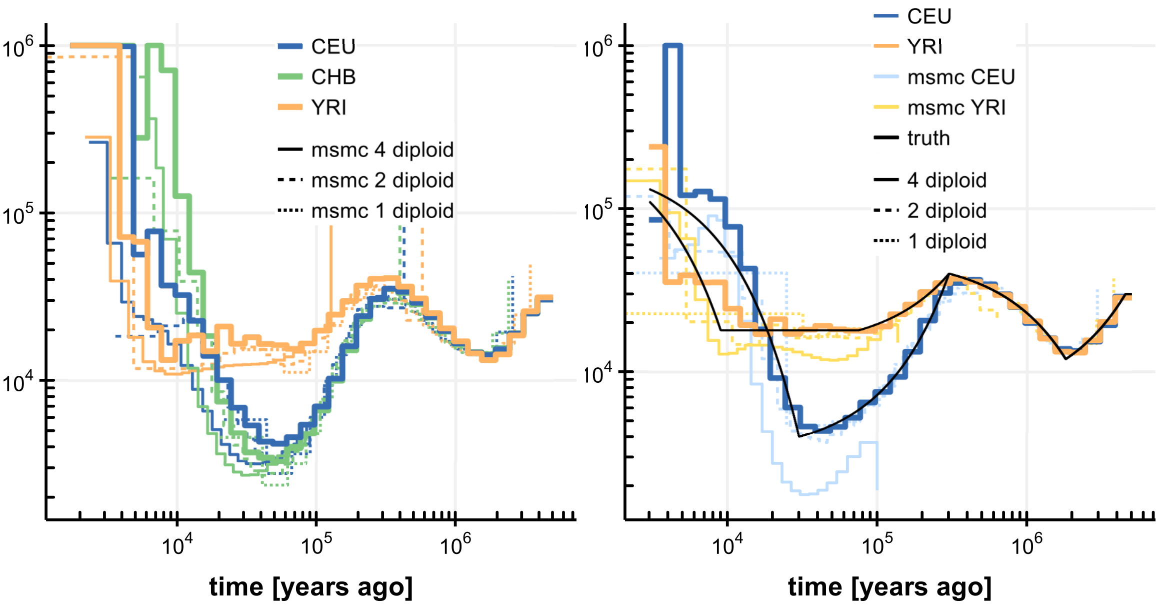

**Fig 3. Population size inferences by SMCSMC on four diploid samples.** Left, three human populations (CEU, CHB, YRI), together with inferences from msmc using 1, 2 and 4 diploid samples. Right, simulated populations resembling CEU and YRI population histories. All inferences (SMCSMC, msmc) were run for 20 iterations.

https://doi.org/10.1371/journal.pone.0247647.g003

unphased data, making the lookahead procedure less effective, and even counterproductive for early epochs.

We next investigated the impact of using Variational Bayes instead of stochastic EM, using the lookahead filter on phased data. We simulated four 2 gigabase (Gb) diploid genomes using human-like evolutionary parameters ($\mu = 1.25 \times 10^{-8}$, $\rho = 3.5 \times 10^{-8}$, $g = 29$, $N_e(0) = 14312$) under a “zigzag” model similar to that used in [16] and [18], and inferred $N_e$ across 37 approximately exponentially spaced epochs; see Appendix (“Implementation Details”). Both approaches give accurate $N_e$ inferences from 2, 000 years up to 1 million years ago (Mya); other experiments indicate that population sizes can be inferred up to 10 Mya (but see Fig 3b). The upwards bias in the most recent epochs is reduced considerably by the Variational Bayes approach compared to SEM (Fig 2b), although some bias remains.

## Inference on human subpopulations

We applied SMCSMC to three sets of four phased diploid samples, of Northern European (CEU), Han Chinese (CHB) and Yoruban (YRI) ancestry respectively, from the 1000 Genomes project. For comparison we also ran msmc [16] inferring on the same data, and on subsets of 2 and 1 diploid samples. Inferences show good agreement where msmchas power (Fig 3). Since the inferences show some variation particularly in more recent epochs, we simulated data under a demographic model closely resembling CEU and YRI ancestry as inferred by multiple methods (see Appendix, “Implementation Details”), and we inferred population sizes using SMCSMC and msmc as before. This confirmed the accuracy of SMCSMC inferences from about 5,000 to 5 million years ago, while inferences in more recent epochs show more variability. A representative comparison of run times is provided in Table 1.

## Discussion

Motivated by the problem of recovering a population’s demographic history from the genomes of a sample of its individuals [1], we have introduced a continuous-locus approximation of the

**Table 1. Runtimes (total CPU time, hours) for analyzing one or two diploid human genomes using `msmc` (40 EM iterations), and `SMCSMC` (15 Variational Bayes iterations).** Table lists means ± one standard deviation across 10 independent runs in a high performance compute environment. Note that due to parallel execution of SMCSMC (146 genomic chunks) and msmc (8 cores), wall clock time was considerably less than the total CPU time.

| Algorithm | 2 haploids | 4 haploids |
|---|---|---|
| msmc | 5.2±0.5 | 107.3±18.7 |
| SMCSMC 5,000 particles | 109.2±5.7 | 277±15 |
| SMCSMC 10,000 particles | 219±11 | 673±32 |

https://doi.org/10.1371/journal.pone.0247647.t001

CwR model, and developed a particle filter algorithm for continuous-time Markov jump processes with a continuous phase space, by evaluating the doubly-continuous process at a suitably chosen set of “waypoints”, and applying a standard particle filter to the resulting discrete-time continuous-state process. It however proved very challenging to obtain reliable parameter inferences for our intended application using this approach. To overcome this challenge we have extended the standard particle filter algorithm in two ways. First, we have generalized the Auxiliary Particle Filter of Pitt and Shephard [40] from a discrete-time one-step-lookahead algorithm to a continuous-time unbounded-lookahead method. This helped to address a challenging feature of the CwR model, namely that recent demographic events induce “sticky” model states with very long forgetting times. With an appropriate lookahead likelihood function (and phased genotype data), we showed that the unbounded-lookahead algorithm mitigates the bias that is otherwise observed in the inferred parameters associated with these recent demographic events. Some bias however remained, particularly for very early epochs. We reduced this remaining bias by a Variational Bayes alternative to stochastic expectation maximization (SEM), which explicitly models part of the uncertainty in the inferred parameters, and avoid zero rate estimates which are fixed points for the SEM procedure. The combination of a continuous-time particle filter, the unbounded-lookahead method, and VB inference, allowed us to infer demographic parameters from up to four diploid genomes across many epochs, without making model approximations beyond passing to the continuous-locus limit.

On three sets of four diploid genomes, from individuals of central European, Han Chinese and Yoruban (Nigeria) ancestry respectively, we obtain inferences of effective population size over epochs ranging from 5,000 years to 5 million years ago. These inferences agree well with those made with other methods [14–18], and show higher precision across a wider range of epochs than was previously achievable by a single method. Despite the improvements from the unbounded-lookahead particle filter and the Variational Bayes inference procedure, the proposed method still struggles in very recent epochs (more recent than a few thousand years ago), and haplotype-based methods [e.g., 12] remain more suitable in this regime. In addition, methods focusing on recent demography benefit from the larger number of recent evolutionary event present in larger samples of individuals, and the proposed model will not scale well to such data, unless model approximations such as those proposed in [18] are used.

A key advantage of particle filters is that they are fundamentally simulation-based. This allowed us to perform inference under the full CwR model without having to resort to model approximations, such as requiring coalescences to occur at certain evolutionary times only, that characterizes most other approaches. The same approach will make it possible to analyze complex demographic models, as long as forward simulation (along the sequential variable) is tractable. The proposed particle filter is based on the sequential coalescent simulator SCRM [45], which already implements complex models of demography that include migration, population splits and mergers, and admixture events. Although not the focus of this paper, it should therefore be straightforward to infer the associated model parameters, including directional migration rates. In addition, several aspects of the standard CwR model are known to be unrealistic. For instance, gene conversions and doublet mutations are common [56, 57], and background selection profoundly shapes the heterozygosity in the human genome [58]. These features are absent from current models aimed at inferring demography, but impact patterns of heterozygosity and may well bias inferences of demography if not included in the model. As long as it is possible to include such features into a coalescent simulator, a particle filter can model such effects, reducing the biases otherwise expected in other parameter due to model misspecification. Because a particle filter produces an estimate of the likelihood, any improved model fit resulting from adding any of these features can in principle be quantified, if these likelihoods can be estimated with sufficiently small variance. However, even improved models will capture only a fraction of relevant features of a population's evolution, and the inferred effective population sizes will continue to have a complex relationship with census population due to population substructure, variation in family size, and many other aspects [59].

A further advantage of a particle filter is that it provides a sample from the posterior distribution of ancestral recombination graphs (ARGs). Such explicit samples simplify the estimation of the age of mutations and recombinations, and explicit identification of sequence tracks with particular evolutionary histories, for instance tracts arising from admixture by a secondary population. In contrast to MCMC-based approaches [13], a particle filter can provide only one self-consistent sample of an ARG per run. However, for marginal statistics such as the expected age of a mutation or the expected number of recombinations in a sequence segment, a particle filter can provide weighted samples from the posterior in a single run.

The algorithm presented here scales in practice to about 4 diploid genomes, but requires increasingly large numbers of particles as larger numbers of genomes are analyzed jointly. This is because the space of possible tree topologies increases exponentially with the number of genomes observed, while the number of informative mutations grows much more slowly, resulting in increasing uncertainty in the topology given observed mutations. This uncertainty is further compounded by uncertainty in branch lengths. Nevertheless, the many effectively independent genealogies underlying even a single genome provide considerable information about past demographic events [14], and a joint analysis of even modest numbers of genomes under demographic models involving migration and admixture events enable more complex demographic scenarios to be investigated. Our results show that particle filters are a viable approach to demographic inference from whole-genome sequences, and the ability to handle complex model without having to resort to approximations opens possibilities for further model improvements, hopefully leading to more insight in our species' recent demographic history.

# Appendix

## Conditional distributions and the Markov property

Here we outline how to define a conditional distribution $\pi(\cdot|\mathcal{G})$ given a distribution $\pi$ on $\mathcal{X}$ and a conditioning subset $\mathcal{G} \subset \mathcal{X}$ of measure 0. Suppose $\mathcal{G}_\tau$ is a family of subsets of $\mathcal{X}$ so that $\cup_\tau \mathcal{G}_\tau = \mathcal{X}$. A particular subset $\mathcal{G}_\tau$ for a fixed $\tau$ plays the role of the conditioning event $B$ in the standard definition $P(A|B) = P(A \cap B)/P(B)$. It can be shown that, under some conditions, there exists an essentially unique family of measures $\pi_{\mathcal{G}_\tau}$ and a measure $\mu$ so that $\pi_{\mathcal{G}_\tau}$ is concentrated on $\mathcal{G}_\tau$, $\pi_{\mathcal{G}_\tau}(\mathcal{X}) = 1$ for all $\tau$, and $E_\pi[f] = \iint f(x)\pi_{\mathcal{G}_\tau}(\mathrm{d}x)\mu(\mathrm{d}\tau)$ for well-behaved functions $f$ [60], making it possible to define the conditional expectation as $E_\pi[f|\mathcal{G}] = E_{\pi_\mathcal{G}}[f] = \int f(x)\pi_\mathcal{G}(\mathrm{d}x)$.

Using this, the Markov property of $\pi$ can be expressed in terms of conditional expectations:

$$E_\pi[f(x_t)|x_{s_1} = \tau_1, \ldots, x_{s_k} = \tau_k] = E_\pi[f(x_t)|x_{s_k} = \tau_k] \tag{18}$$

for loci $s_1 < s_2 < \ldots < s_k < t$ and any well-behaved function $f$.

## Proof of Algorithm 3

The algorithm is proved by induction on $j$. For $j = 0$ the loop invariant holds, while for $j = K$ it implies the output condition. Suppose the loop invariant is true for some $j$. If $ESS < N/2$, assume w.l.o.g. that $v^{(i)} = v^{(i)}_{s_j}$ are normalized, let $i_k$ be the index of the $k$th new particle, $\hat{v}^{(k)} = N^{-1}$ and $\hat{w}^{(k)} = N^{-1}w^{(i_k)}/v^{(i_k)}$ be its weights, and write $\pi^j = \pi(X_{1:s_j}|Y_{1:s_j} = y_{1:s_j})$, $\tilde{\pi}^j = \tilde{\pi}^{s_j}(X_{1:s_j}|Y_{1:s_j} = y_{1:s_j})$, then

$$E_{\{v^{(i)}\}}[\sum_{k=1}^{N}\hat{v}^{(k)}f(X^{(i_k)})] = \frac{1}{N}\sum_{k=1}^{N}E_{\{v^{(i)}\}}[f(X^{(i_k)})] = \frac{1}{N}\sum_{k=1}^{N}\sum_{i=1}^{N}v^{(i)}f(X^{(i)}) \approx E_{\tilde{\pi}^j}[f(X)],$$

and

$$E_{\{v^{(i)}\}}[\sum_{k=1}^{N}\hat{w}^{(k)}f(X^{(i_k)})] = \frac{1}{N}\sum_{k=1}^{N}E_{\{v^{(i)}\}}[\frac{w^{(i_k)}}{v^{(k)}}f(X^{(i_k)})] = \frac{1}{N}\sum_{k=1}^{N}\sum_{i=1}^{N}w^{(i)}f(X^{(i)}) \approx E_{\pi^j}[f(X)],$$

so that the loop invariant continues to hold after the optional resampling step.

After sampling $x^{(i)}_{s_j:s_{j+1}} \sim \xi_x(X_{s_j:s_{j+1}}|X_{s_j} = x^{(i)}_{s_j})$, the particles $\{(x^{(i)}_{1:s_j}, w^{(i)}_{s_j})\}$ approximate $\pi(X_{1:s_j}|Y = y_{1:s_j})\xi_x(X_{s_j:s_{j+1}}|X_{s_j})$. To make this distribution absolutely continuous w.r.t. $\pi(X_{1:s_{j+1}}, Y_{s_j:s_{j+1}}|Y_{1:s_j})$, multiply it with the constant measure $\lambda_{s_j:s_{j+1}}(y_{s_j:s_{j+1}})$; any measure will do as long as it has a density w.r.t. $\lambda_{s_j:s_{j+1}}$ and is independent of $X$. Taking the Radon-Nikodym derivative of these two distributions gives

$$\begin{aligned}
&\frac{\mathrm{d}\pi^{1:s_{j+1}}(X_{1:s+1}, Y_{s_j:s_{j+1}}|Y_{1:s_j})}{\mathrm{d}[\pi^{1:s_j}(X_{1:s_j}|Y_{1:s_j})\xi_x^{s_j:s_{j+1}}(X_{s_j:s_{j+1}}|X_{s_j})\lambda^{s_j:s_{j+1}}(Y_{s_j:s_{j+1}})]}(x_{1:s_{j+1}}, y_{s_j:s_{j+1}})\\
&= \frac{\mathrm{d}[\pi^{1:s_j}(X_{1:s_j}|Y_{1:s_j})\pi_x^{s_j:s_{j+1}}(X_{s_j:s_{j+1}}|X_{s_j})\pi^{s_j:s_{j+1}}(Y_{s_j:s_{j+1}}|X_{s_j:s_{j+1}})]}{\mathrm{d}[\pi^{1:s_j}(X_{1:s_j}|Y_{1:s_j})\xi_x^{s_j:s_{j+1}}(X_{s_j:s_{j+1}}|X_{s_j})\lambda^{s_j:s_{j+1}}(Y_{s_j:s_{j+1}})]}(x_{1:s_{j+1}}, y_{s_j:s_{j+1}})\\
&= \frac{\mathrm{d}\pi_x}{\mathrm{d}\xi_x}(x_{s_j:s_{j+1}}|X_{s_j} = x_{s_j})\frac{\mathrm{d}\pi}{\mathrm{d}\lambda}(y_{s_j:s_{j+1}}|X_{s_j:s_{j+1}} = x_{s_j:s_{j+1}})
\end{aligned}$$

This shows that $\{w^{(i)}_{s_{j+1}}, X^{(i)}_{1:s_{j+1}}\}$ form particles approximating $\pi^{1:s_{j+1}}(X_{1:s+1}, y_{s_j:s_{j+1}}|y_{1:s_j})$, and since $\pi(x_{1:s_{j+1}}, y_{s_j:s_{j+1}}|Y_{1:s_j}= y_{1:s_j}) \propto \pi(x_{1:s_{j+1}}|Y_{1:s_{j+1}}= y_{1:s_{j+1}})\lambda_{s_j:s_{j+1}}(y_{s_j:s_{j+1}})$ they also approximate $\pi(x_{1:s_{j+1}}|Y_{1:s_{j+1}} = y_{1:s_{j+1}})$. The argument showing that $(v^{(i)}_{s_{j+1}}, X^{(i)}_{1:s_{j+1}}) \approx \tilde{\pi}^{s_j}(x_{1:s_{j+1}}|Y= y_{1:L})$ is analogous. This proves the loop invariant for $j + 1$, and the algorithm.

## Particle weight variance

To derive a criterion on the waypoints that limits the effect of weight variance build-up, let $R(s) = f(X_s)$ be the stochastic variable that measures the instantaneous rate of occurrence of emission events for a particular (random) particle $X$, and let $W(s) = W_0\exp(-\int_0^s R(u)\mathrm{d}u)$ be that particle's time-dependent weight; the dependence on $W$ on $X$ is not written explicitly. Note that the expression for $W(s)$ is valid as long as no events have occurred in the interval $[0, L)$. We assume that $R(s)$ is time-homogeneous, that it can be approximated by a Gaussian process, that particles are drawn from the equilibrium distribution, and that $W_0$ and $R(s)$ are independent. Write $\langle V(X)\rangle \coloneqq \int V(X)d\pi(X)$ for the expectation of $V$ over $\pi(X)$. Writing $R(s) = \mu + \tilde{R}(s)$ where $\mu = \langle R(s)\rangle$ is the mean event rate (which is independent of $s$ by assumption), then

$$\langle W(L)\rangle = \langle W_0 e^{-\int_0^L \mu+\tilde{R}(s)\mathrm{d}s}\rangle = \langle W_0\rangle e^{-\mu L}\langle\prod_{i=1}^{k}(1-\tilde{R}(s_i)\Delta s)\rangle \tag{19}$$

as $k \to \infty$, where $\Delta s = L/k$ and $s_i = i\Delta s$. The last expectation becomes

$$\begin{aligned}
&\langle\sum_{n=0}^{\infty}(-1)^n \int\limits_{0<s_1<\cdots<s_n<L} \tilde{R}(s_1)\cdots\tilde{R}(s_n)\mathrm{d}s_1\cdots\mathrm{d}s_n\rangle \\
=\ & \sum_{n=0}^{\infty}\frac{(-1)^n}{n!}\int\limits_{s_1,\ldots,s_n=0}^{L} \langle\tilde{R}(s_1)\cdots\tilde{R}(s_n)\rangle\mathrm{d}s_1\cdots\mathrm{d}s_n \\
=\ & \sum_{m=0}^{\infty}\frac{1}{(2m)!}\frac{(2m-1)!}{2^{m-1}(m-1)!}\left(\int_{s_1,s_2=0}^{L} K(s_1,s_2)\mathrm{d}s_1\mathrm{d}s_2\right)^m = \sum_{m=0}^{\infty}\frac{1}{2^m m!}C^m = e^{\frac{1}{2}C}
\end{aligned}$$

where in the second equality we used the formula for higher moments of a Gaussian distribution, $K$ is the covariance function of the Gaussian process $\tilde{R}(t)$, and $C$ is the integral $\int_{s_1,s_2=0}^{L} K(s_1,s_2)\mathrm{d}s_1\mathrm{d}s_2$. Now define $\sigma^2 \coloneqq K(s, s)$ and assume that the covariance function satisfies $0 \le K(s_1, s_2) \le \sigma^2$, then $0 \le C \le \sigma^2 L^2$ and

$$\langle W\rangle \quad = \langle W_0\rangle e^{-\mu L+\frac{1}{2}C} \ge \langle W_0\rangle e^{-\mu L}, \tag{20}$$

$$\langle W^2\rangle \quad = \langle W_0^2\rangle e^{-2\mu L+C} \le \langle W_0^2\rangle e^{-2\mu L+\sigma^2 L^2}, \tag{21}$$

so that across an interval $[0, L)$ where no events occur,

$$ESS = \frac{(\sum_{i=1}^{N} w^{(i)})^2}{\sum_{i=1}^{N}(w^{(i)})^2} \approx \frac{N^2\langle W\rangle^2}{N\langle W^2\rangle} \ge \frac{N\langle W_0\rangle^2 e^{-2\mu L}}{\langle W_0^2\rangle e^{-2\mu L+\sigma^2 L^2}} = ESS_0 e^{-\sigma^2 L^2} \tag{22}$$

where $ESS_0$ is the expected sample size at $s = 0$, and $\approx$ denotes convergence in distribution as $N \to \infty$ as before.

In practice particles will not be drawn from the equilibrium distribution $\pi_x(X)$, but from the joint distribution on $X$ and $Y$ conditioned on observations $y$. However, for most likelihoods conditioning will reduce the variance of $R$ as observations tend to constrain the distribution of likely particles, making this a conservative assumption. The other assumption that is likely not met is that $R(t)$ is a Gaussian process; it is less clear whether making this approximation will in practice be conservative.

## The sequential coalescent with recombination process

In formula (1), if $s$ is a recombination point, $x_s$ is the genealogy just *left* of the recombination point and includes the infinite branch from the root, so that $b_u(x_s) = 1$ for $u$ above the root.

The measure (1) describes the CwR process exactly as long as $x$ encodes both the local genealogy and the non-local branches used by the SCRM algorithm. In practice the SCRM algorithm prunes some of these branches, and we use (1) on the pruned $x$.

Note that we take the view that the realisation $x$ encodes not only the sequence of genealogies $x_s$ but also the number of recombinations $|x|$ (some of which may not change the tree), their loci $s_j = s_j^x$, and the recombination and coalescence times $v_j^x$ and $\tau_j^x$. This information is also kept in the implementation of the algorithm, and is used to calculate the sufficient statistics required for inference of the coalescence and recombination rates.

## Variational Bayes for Markov jump processes

We consider hidden Markov models where the latent variable follows a Markov jump process over $x \in \mathcal{X}$, that with respect to a suitable measure d$x$d$y$ admits a probability density of the form

$$\pi_{xy}(x, y|\theta)\mathrm{d}x\mathrm{d}y = \pi_y(y|x)\prod_i \exp\{-\theta_i B_i(x)\}\theta_i^{|x|_i}\mathrm{d}x\mathrm{d}y. \tag{23}$$

Here, $|x|_i$ is the event count for events of type $i$ in realisation $x$, and $B_i(x)$ is the total opportunity for events of that type in $x$. For example, in our case

$$B_U^R(x) = \int_s \int_{u\in U} b_u(x_s)\mathrm{d}u\mathrm{d}s; \qquad B_U^C(x) = \sum_{j=1}^{|x|}\int_{u\in[v_j,\tau_j]\cap U} b_u(x_{s_j})\mathrm{d}u, \tag{24}$$

and $|x|_U^R = \#\{j : v_j \in U\}$, $|x|_U^C = \#\{j : \tau_j \in U\}$, for recombinations and coalescence opportunities and counts occurring in an epoch $U \subset [0, \infty)$.

A Variational Bayes approach approximates the true joint posterior density $\pi(x, \theta|y) \propto \pi_{xy}(x, y|\theta)\pi_\theta(\theta)$, where $\pi_\theta$ is a prior on the parameters, with a probability density $\phi(x, \theta)$ that is easier to work with (here the constant of proportionality implied by "$\propto$" hides a constant density $\lambda(y)$). Following Hinton and van Camp [61] and MacKay [62], we choose to constrain $\phi$ by requiring it to factorize as $\phi(x, \theta) = \phi_x(x)\phi_\theta(\theta)$, and we choose to optimize it by minimizing the Kullback-Leibler divergence $KL(\phi||\pi)$, also referred to as the variational free energy [63],

$$F(\phi) = -\int_x \int_\theta \phi(x, \theta)\log\left[\frac{\pi(x, \theta|y)}{\phi(x, \theta)}\right]\mathrm{d}\theta\mathrm{d}x. \tag{25}$$

To optimize $\phi_\theta(\theta)$ we write $F(\phi)$ as a function of $\phi_\theta$ with $\phi_x$ fixed, as

$$F(\phi) \quad = -\int \phi_x(x)\phi_\theta(\theta)\left\{\sum_i |x|_i \log\theta_i \ - \ B_i(x)\theta_i \ + \ \log\pi_\theta(\theta) \ - \ \log\phi_\theta(\theta)\right\}\mathrm{d}\theta\mathrm{d}x + const \tag{26}$$

$$= \int \phi_\theta(\theta)\log\frac{\phi_\theta(\theta)}{\pi_\theta(\theta)\prod_i \theta_i^{\mathbb{E}_{\phi_x}[|x|_i]}\exp\{-\mathbb{E}_{\phi_x}[B_i(x)]\theta_i\}}\mathrm{d}\theta + const \tag{27}$$

This is minimized by setting log$\phi_\theta(\theta)$ equal to the log of the denominator. We can still choose the prior $\pi_\theta(\theta)$; a product of Gamma distributions $\prod_i \Gamma(\alpha_i, \beta_i)(\theta_i) \propto \prod_i \theta_i^{\alpha_i - 1}\exp\{-\beta_i\theta_i\}$ is suitable as it is conjugate to the factors appearing in the denominator. The result is that

$$\phi_\theta(\theta) = \prod_i \Gamma(\alpha_i', \beta_i') \tag{28}$$

with $\alpha_i' = \alpha_i + \mathbb{E}_{\phi_x}[|x|_i]$ and $\beta_i' = \beta_i + \mathbb{E}_{\phi_x}[B_i(x)]$. Next, to optimize $\phi_x(x)$ we write $F(\phi)$ as a function of $\phi_x$ with $\phi_\theta$ fixed,

$$\begin{aligned} F(\phi) &= -\int \phi_x(x)\phi_\theta(\theta)\left\{\sum_i |x|_i \log\theta_i - B_i(x)\theta_i + \log\pi_y(y|x) - \log\phi_x(x)\right\}\mathrm{d}\theta\mathrm{d}x + const \\ &= \int \phi_x(x)\log\frac{\phi_x(x)}{\pi_y(y|x)\prod_i \exp\{|x|_i \mathbb{E}_{\phi_\theta}[\log\theta_i] - B_i(x)\mathbb{E}_{\phi_\theta}[\theta_i]\}}\mathrm{d}x + const \end{aligned}$$

Define $\bar{\theta}_i := \mathbb{E}_{\phi_\theta}[\theta_i]$ and $\theta_i^* := \exp\{\mathbb{E}_{\phi_\theta}[\log\theta_i]\}$, then using properties of the Gamma distribution we get $\bar{\theta} = \alpha_i'/\beta_i'$ and $\theta_i^* = \exp\{\psi(\alpha_i') - \log\beta_i'\}$ where $\psi$ is the digamma function. Again, $F(\phi)$ is minimized if the numerator and denominator are proportional, which happens for

$$\phi_x(x) \quad \propto \pi_y(y|x)\prod_i \exp\{-\bar{\theta}_i B_i(x)\}\,(\theta_i^*)^{|x|_i} \propto \pi(x|y,\bar{\theta})\prod_i\left(\frac{\theta_i^*}{\bar{\theta}_i}\right)^{|x|_i} = \pi(x|y,\bar{\theta})\prod_i \eta_i^{|x|_i} \quad (29)$$

where $\eta_i := \theta_i^*/\bar{\theta}_i = \exp\{\psi(\alpha_i')\}/\alpha_i'$. As given, the algorithms in this paper sample from a distribution of the form $\pi(x|y,\bar{\theta})$, but they can easily be modified to sample from $\phi_x(x)$ instead by including an additional factor $\eta_i$ in a particle's weight for every event of type $i$ that occurs.

## A lookahead likelihood

Let $s_i$ denote the distance along the genome to the nearest future singleton in each sequence, and let $c_k = (a_k, b_k)$ be $\leq n/2$ mutually consistent cherries with loci $s_k' \leq s_k''$ of their first and last supporting doubleton. To simplify notation we assume that the current locus is 0 (Fig 1a).

Note that recombinations result in a change of a terminal branch length (TBL) if either the recombination occurred in the branch itself and the new lineage does not coalesce back into it, or the recombination occurred outside the branch and the new lineage coalesces into it (Fig 1b). To compute the likelihood that the first singleton in lineage $i$ occurs at locus $s_i$, we assume that all TBLs are equal to $l_i$, and that coalescences occur before $l_i$. Then, the total rate of events that change the TBL $i$ is

$$\rho_i := l_i\rho\frac{n-1}{n} + (n-1)l_i\rho\frac{1}{n} = 2l_i\rho\frac{n-1}{n}.$$

Define $\mu_i \coloneqq \mu l_i$ to be the total mutation rate on branch $i$, and assume that when a TBL changes, it reverts deterministically to some length $l_i'$. If a terminal branch with length $l_i$ changes at $u$ to $l_i'$, which happens with probability $e^{-\rho}{}_i\, u\, \rho_i\mathrm{d}u$, the likelihood that the first singleton occurs at distance $s_i$ is $e^{-\mu_i u}e^{-\mu_i'(s_i-u)}\mu_i'\mathrm{d}t$, where $\mu_i' := \mu l_i'$. Conversely, if that branch does not change along $[0, s_i)$, which happens with probability $e^{-\rho_i s_i}$, the likelihood is $e^{-\mu_i s_i}\mu_i\mathrm{d}t$. Combining these possibilities and marginalizing over $u \in [0, s_i)$ gives (using Mathematica to evaluate the integral)

$$p(\text{1st singleton in } i \text{ at } s_i|l_i, l_i') = \frac{1}{\rho_i + \mu_i - \mu_i'}\left(\rho_i\mu_i' e^{-\mu_i' s_i} + (\mu_i - \mu_i')(\rho_i + \mu_i)e^{-(\rho_i+\mu_i)s_i}\right)\mathrm{d}t.$$

In the case that no singleton is observed up until $s_i$ but data was missing thereafter, the same probability densities apply except for the factors $\mu_i$ and $\mu_i'$ in the likelihood, so that

$$p(\text{no singleton in } i \text{ until } s_i|l_i, l_i') = \frac{1}{\rho_i + \mu_i - \mu_i'}\left(\rho_i e^{-\mu_i' s_i} + (\mu_i - \mu_i')e^{-(\rho_i+\mu_i)s_i}\right)\mathrm{d}t. \qquad (38)$$

We account for the uncertainty in $l_i'$ by marginalizing over the empirical distribution of TBLs for sequence $i$.

To approximate the likelihood of the doubleton data, note that a node $c$ with precisely two descendants $(a, b)$ (a "cherry") at height $l$ changes if a recombination occurs in either branch $a$ or $b$ and the new lineage coalesces out, or a recombination occurs outside of $a$ and $b$ and coalesces into either (Fig 1b). Again assuming that all TBLs are $l$ and coalescences occur before $l$, the total rate of change is $2l\rho\frac{n-2}{n} + (n-2)l\rho\frac{2}{n} = 4l\rho\frac{n-2}{n} := \rho_C$. When a cherry changes, we assume that the new cherry is drawn from the equilibrium distribution. To calculate the probablity of observing $c = (a, b)$ at equilibrium, assume that a tree supports $1 \leq k \leq n/2$ cherries. The branches of $c$ are among the $2k$ branches subtended by the tree's $k$ cherries with probability $\frac{2k}{n}\frac{2k-1}{n-1}$, and $a$ is paired with $b$ with probability $\frac{1}{2k-1}$. Since $k$ has mean $n/3$ if $n \geq 3$ [64], the probability of observing $(a, b)$ at equilibrium is $\frac{2}{3(n-1)}$. We approximate the likelihood of a doubleton by 0 if the $c$ is not in the tree, and by 1 if it is. Then, the likelihood of observing $c_k = (a_k, b_k)$ at the last known locus $s''_k$ conditional on the tree currently containing $c_k$ is

$$p(a_k, b_k, s'_k, s''_k | (a_k, b_k; l) \in \tau) = e^{-\rho_C s''_k} + \frac{2}{3(n-1)}(1 - e^{-\rho_C s''_k}), \tag{30}$$

where $(a_k, b_k;l) \in \tau$ expresses that $\tau$ contains cherry $c_k = (a_k, b_k)$ at height $l$. Now suppose $c_k \notin \tau$ and let $\bar{l}$ be the average TBL in $\tau$. Under similar assumptions, cherries are created at a rate $(n-1)\rho\bar{l}$ and assuming that new cherries are drawn from the equilibrium distribution, the likelihood of observing $c_k$ at the first known locus $s'_k$ is

$$p(a_k, b_k, s'_k, s''_k | \bar{l}, (a_k, b_k) \notin \tau) = \frac{2}{3(n-1)}(1 - e^{-\rho'_C s'_k}), \tag{31}$$

where $\rho'_C = (n-1)\bar{l}\rho$ is the effective rate of recombinations that potentially result in the creation of $c_k$. Note that (30 and 31) are likelihoods for $\tau$ supporting $c_k$ at the given locus, rather than for a doubleton mutation actually occurring.

These likelihoods show good performance, but result in some negative bias in inferred population size for recent epochs. We traced this to the lack of correlation between $l_i$ and $l'_i$, requiring a single very recent coalescence to explain a long segment devoid of singletons, rather than allowing for the possibility of several correlated coalescences each in slightly earlier epochs. To model correlations, we averaged the likelihood above over $\rho' = \rho$ and $\rho' = \rho/2$ each weighted with probability 1/2. This effectively removed the negative bias.

To deal with missing data, we reduce $\mu$ proportionally to the missing segment length and the number of lineages missing. For unphased mutation data, singletons and doubletons can still be extracted, and are greedily assigned to compatible lineages. The likelihoods are also similarly calculated, by greedily assigning cherries to observed doubletons. Unphased singletons can result from mutations on either of the individual's alleles; the likelihood term uses the sum of the two branch lengths for that individual to calculate the expected rate of unphased singletons.

## Implementation details

While $x_{1:s}$ refers to the entire sequence of genealogies along the sequence segment 1: $s$, storing this sequence would require too much memory. Instead we only store the most recent genealogy $x_s$ (including non-local branches where appropriate), which is sufficient to simulate subsequent genealogies using the SCRM algorithm. To implement epoch-dependent lags when harvesting sufficient statistics, we do store records of the events (recombinations, coalescences and migrations) that changed $x$ along the sequence, as well as the associated opportunities, for each particle and each epoch; this implicitly stores the full ARG. To avoid making copies of

Table 2. Commands to generate simulation data.

| Experiment | Command |
|---|---|
| zigzag | `scrm 8 1 -N0 14312 -t 1431200 -r 400736 2000000000 -eN 0 1 -eG 0.000582262 1318.18 -eG 0.00232905 -329.546 -eG 0.00931619 82.3865 -eG 0.0372648 -20.5966 -eG 0.149059 5.14916 -eN 0.596236 0.1 -seed 1 -T -L -p 10 -l 300000` |
| CEU | `scrm 8 1 -N0 14312 -t 1789000 -r 500920 2500000000 -eN 0 10.4807 -eG 0.00120468 214.8965 -eG 0.0180702 -14.15827 -eG 0.180702 1.33255 -eG 1.084212 -0.563414 -eN 2.71053 2.096143 -seed 1 -T -L -p 10 -l 300000` |
| YRI | `scrm 8 1 -N0 14312 -t 1789000 -r 500920 2500000000 -eN 0 10.4807 -eG 0.00120468 502.8635 -eG 0.00542106 0 -eG 0.0451755 -5.89189 -eG 0.180702 1.33255 -eG 1.084212 -0.563414 -eN 2.71053 2.096143 -seed 1 -T -L -p 10 -l 300000` |

https://doi.org/10.1371/journal.pone.0247647.t002

potentially many event records when particles are duplicated at resampling, these are stored in a linked list, and are shared by duplicated particles where appropriate, forming a tree structure. Records are removed dynamically after contributing to the summary statistics, and when particles fail to be resampled, ensuring that memory usage is bounded.

The likelihood calculations involve many evaluations of the exponential function, often for small exponents. We use the continued-fraction approximation $e^x \approx 1 + 2x/\left(2 - x + \frac{1}{6}x^2\right)$ for $|x| < 0.03$, with relative error bounded by $10^{-10}$ [65].

Table 2 shows the commands to generate the data for the three simulation experiments. Epoch boundaries for $N_e$ inference in generations for the zigzag experiment were defined by taking interval boundaries $-14312\log(1-i/256)/2$, $i = 0, \ldots, 255$, merging intervals according to the pattern $4 * 1 + 7 * 2 + 8 * 5 + 7 * 13 + 1 * 15 + 8 * 11 + 1 * 3$ (37 epochs; see [14]). For the real data experiments, epochs boundaries for the 32 epochs were logarithmically spaced from 133 to 133016 generations ago, using generation time $g = 29$ years, without merging intervals (command line option `-P 133 133016 31*1`).

## Acknowledgments

We thank Arnaud Doucet for helpful discussions, and Paul Staab for implementing the SCRM library.

## Author Contributions

**Conceptualization:** Gerton Lunter.

**Formal analysis:** Donna Henderson, Sha (Joe) Zhu, Christopher B. Cole, Gerton Lunter.

**Software:** Donna Henderson, Sha (Joe) Zhu, Christopher B. Cole, Gerton Lunter.

**Supervision:** Gerton Lunter.

**Writing – original draft:** Donna Henderson, Sha (Joe) Zhu, Christopher B. Cole, Gerton Lunter.

**Writing – review & editing:** Donna Henderson, Sha (Joe) Zhu, Christopher B. Cole, Gerton Lunter.

## References

1. Schraiber JG, Akey JM. Methods and models for unravelling human evolutionary history. Nature Reviews Genetics. 2015;. https://doi.org/10.1038/nrg4005 PMID: 26553329

2. Beaumont M. Detecting Population Expansion and Decline Using Microsatellites. Genetics. 1999; 153 (4):2013–2029. PMID: 10581303

3. Pritchard JK, Stephens M, Donnelly P. Inference of Population Structure Using Multilocus Genotype Data. Genetics. 2000; 155(2):945–959. PMID: 10835412

4. Beaumont M, Zhang W, Balding DJ. Approximate Bayesian Computation in Population Genetics. Genetics. 2002; 162(4):2025–2035. PMID: 12524368

5. Gutenkunst RN, Hernandez RD, Williamson SH, Bustamante CD. Inferring the joint demographic history of multiple populations from multidimensional SNP frequency data. PLoS Genet. 2009; 5(10): e1000695. https://doi.org/10.1371/journal.pgen.1000695 PMID: 19851460

6. Gronau I, Hubisz MJ, Gulko B, Danko CG, Siepel A. Bayesian inference of ancient human demography from individual genome sequences. Nature genetics. 2011; 43(10):1031–1034. https://doi.org/10.1038/ng.937 PMID: 21926973

7. Excoffier L, Dupanloup I, Huerta-Sánchez E, Sousa VC, Foll M. Robust demographic inference from genomic and SNP data. PLoS Genet. 2013; 9(10):e1003905. https://doi.org/10.1371/journal.pgen.1003905 PMID: 24204310

8. Drummond AJ, Rambaut A. BEAST: Bayesian evolutionary analysis by sampling trees. BMC Evolutionary Biology. 2007; 7:214. https://doi.org/10.1186/1471-2148-7-214 PMID: 17996036

9. Browning SR, Browning BL. A fast, powerful method for detecting identity by descent. American Journal of Human Genetics. 2011; 88(2):173–182. https://doi.org/10.1016/j.ajhg.2011.01.010 PMID: 21310274

10. Palamara PF, Lencz T, Darvasi A, Peér I. Length distributions of identity by descent reveal fine-scale demographic history. Am J Hum Genet. 2012; 91(5):809–22. https://doi.org/10.1016/j.ajhg.2012.08.030 PMID: 23103233

11. Harris K, Nielsen R. Inferring demographic history from a spectrum of shared haplotype lengths. PLoS Genet. 2013; 9(6):e1003521. https://doi.org/10.1371/journal.pgen.1003521 PMID: 23754952

12. Hellenthal G, Busby GB, Band G, Wilson JF, Capelli C, Falush D, et al. A genetic atlas of human admixture history. Science. 2014; 343(6172):747–751. https://doi.org/10.1126/science.1243518 PMID: 24531965

13. Rasmussen MD, Hubisz MJ, Gronau I, Siepel A. Genome-wide inference of ancestral recombination graphs. PLoS Genet. 2014; 10(5):e1004342. https://doi.org/10.1371/journal.pgen.1004342 PMID: 24831947

14. Li H, Durbin R. Inference of human population history from individual whole-genome sequences. Nature. 2011; 475(7357):493–496. https://doi.org/10.1038/nature10231 PMID: 21753753

15. Sheehan S, Harris K, Song YS. Estimating variable effective population sizes from multiple genomes: a sequentially Markov conditional sampling distribution approach. Genetics. 2013; 194(3):647–662. https://doi.org/10.1534/genetics.112.149096 PMID: 23608192

16. Schiffels S, Durbin R. Inferring human population size and separation history from multiple genome sequences. Nature genetics. 2014; 46(8):919–925. https://doi.org/10.1038/ng.3015 PMID: 24952747

17. Steinrücken M, Kamm J, Spence JP, Song YS. Inference of complex population histories using whole-genome sequences from multiple populations. Proceedings of the National Academy of Sciences. 2019; p. 17115–20. https://doi.org/10.1073/pnas.1905060116 PMID: 31387977

18. Terhorst J, Kamm JA, Song YS. Robust and scalable inference of population history from hundreds of unphased whole genomes. Nature Genetics. 2017; 49(2):303–309. https://doi.org/10.1038/ng.3748 PMID: 28024154

19. Gordon NJ, Salmond DJ, Smith AFM. Novel approach to nonlinear/non-Gaussian Bayesian state estimation. IEEE Proceedings F, Radar and Signal Processing. 1993; 140(2):107–113. https://doi.org/10.1049/ip-f-2.1993.0015

20. Doucet A, Godsill S, Andrieu C. On sequential Monte Carlo sampling methods for Bayesian filtering. Statistics and computing. 2000; 10(3):197–208. https://doi.org/10.1023/A:1008935410038

21. Arulampalam MS, Maskell S, Gordon N, Clapp T. A Tutorial on Particle Filters for Online Nonlinear/Non-Gaussian Bayesian Tracking. IEEE Trans Signal Processing. 2002; 50(2):174–188. https://doi.org/10.1109/78.978374

22. Doucet A, Johansen AM. A tutorial on particle filtering and smoothing: Fifteen years later. Handbook of nonlinear filtering. 2011; 12(3):656–704.

23. Taylor S, Ridall G, Sherlock C, Fearnhead P. Particle learning approach to Bayesian model selecion: An application from neurology. In: Springer Proceedings in Mathematics and Statistics. vol. 63; 2014. p. 165–167.

24. Smith RA, Ionides EL, King AA. Infectious Disease Dynamics Inferred from Genetic Data via Sequential Monte Carlo. Molecular Biology and Evolution. 2017;. https://doi.org/10.1093/molbev/msx124 PMID: 28402447

**25.** Fourment M, Claywell BC, McCoy C, Matsen FA Iv, Darling AE. Effective Online Bayesian Phylogenetics via Sequential Monte Carlo with Guided Proposals. Systematic Biology. 2018;. https://doi.org/10.1093/sysbio/syx090 PMID: 29186587

**26.** Wang L, Wang S, Bouchard-Côté A. An Annealed Sequential Monte Carlo Method for Bayesian Phylogenetics. Systematic Biology. 2019.

**27.** Rosenbluth MN, Rosenbluth AW. Monte Carlo Calculation of the Average Extension of Molecular Chains. J Chem Phys. 1955; 23(356):356–359. https://doi.org/10.1063/1.1741967

**28.** Feller W. On the Integro-Differential Equations of Purely Discontinuous Markoff Processes. Transactions of the American Mathematical Society. 1940; 48(3):488–515. https://doi.org/10.1090/S0002-9947-1940-0002697-3

**29.** Del Moral P, Jacob J, Protter P. The Monte Carlo method for filtering with discrete-time observations. Probability Theory and Related Fields. 2002; 120:346–368. https://doi.org/10.1007/PL00008786

**30.** Golightly A, Wilkinson DJ. Bayesian sequential inference for nonlinear multivariate diffusions. Statistics and Computing. 2006; 16(4):323–338. https://doi.org/10.1007/s11222-006-9392-x

**31.** Fearnhead P, Papaspiliopoulos O, Roberts GO. Particle Filters for Partially Observed Diffusions. Journal of the Royal Statistical Society: Series B. 2008; 70(4):755–777. https://doi.org/10.1111/j.1467-9868.2008.00661.x

**32.** Nodelman U, Shelton CR, Koller D. Continuous Time Bayesian Networks. In: Proceedings of the UAI; 2002.

**33.** Ng B, Pfeffer A, Dearden R. Continuous Time Particle Filtering. In: Proceedings of the IJCAI; 2005. p. 1360–1365.

**34.** Doucet A, Gordon NJ, Krishnamurthy V. Particle Filters for State Estimation of Jump Markov Linear Systems. IEEE Transactions on Signal Processing. 2001; 49(3):613–624. https://doi.org/10.1109/78.905890

**35.** Sherlock C, Golightly A, Gillespie CS. Bayesian Inference for Hybrid Discrete-Continuous Systems Biology Models. Inverse Problems. 2014; 30:114005. https://doi.org/10.1088/0266-5611/30/11/114005

**36.** Witeley N, Johansen AM, Godsill S. Monte Carlo Filtering of Piecewise Deterministic Processes. Journal of Computational and Graphical Statistics. 2011; 20(1):119–139. https://doi.org/10.1198/jcgs.2009.08052

**37.** Del Moral P, Miclo L. Branching and interacting particle systems. Approximations of Feynman-Kac formulae with applications to non-linear filtering. Séminaire de probabilités (Strasbourg). 2000; p. 1–145.

**38.** Del Moral P. Mean Field Simulation for Monte Carlo Integration. Chapman and Hall/CRC; 2016.

**39.** Olsson J, Capp'e O, Douc R, Moulines É. Sequential Monte Carlo smoothing with application to parameter estimation in nonlinear state space models. Bernoulli. 2008; 14(1):155–179. https://doi.org/10.3150/07-BEJ6150

**40.** Pitt MK, Shephard N. Filtering via Simulation: Auxiliary Particle Filters. Journal of the American Statistical Association. 1999; 94(446):590–599. https://doi.org/10.1080/01621459.1999.10474153

**41.** Nielsen SF. The stochastic EM algorithm: estimation and asymptotic results. Bernoulli. 2000; 6(3):457–489. https://doi.org/10.2307/3318671

**42.** Hudson RR. Properties of a neutral allele model with intragenic recombination. Theoretical Population Biology. 1983; 23(2):183–201. https://doi.org/10.1016/0040-5809(83)90013-8 PMID: 6612631

**43.** Griffiths RC, Marjoram P. An ancestral recombination graph. In: Donnelly P, Tavaré S, editors. Progress in Population Genetics and Human Evolution. Springer-Verlag; 1997. p. 257–270.

**44.** Wiuf C, Hein J. Recombination as a Point Process along Sequences. Theoretical Population Biology. 1999; 55:248–259. https://doi.org/10.1006/tpbi.1998.1403 PMID: 10366550

**45.** 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. https://doi.org/10.1093/bioinformatics/btu861 PMID: 25596205

**46.** McVean GA, Cardin NJ. Approximating the coalescent with recombination. Philos Trans R Soc Lond B Biol Sci. 2005; 360(1459):1387–1393. https://doi.org/10.1098/rstb.2005.1673 PMID: 16048782

**47.** Marjoram P, Wall JD. Fast "coalescent" simulation. BMC Genetics. 2006; 7(16). https://doi.org/10.1186/1471-2156-7-16 PMID: 16539698

**48.** Felsenstein J. Evolutionary trees from DNA sequences: A maximum likelihood approach. Journal of Molecular Evolution. 1981; 17(6):368–376. https://doi.org/10.1007/BF01734359 PMID: 7288891

**49.** Heyde CC. The effect of selection on genetic balance when the population size is varying. Th Pop Biol. 1977; 11:249–251. https://doi.org/10.1016/0040-5809(77)90027-2 PMID: 867288

**50.** Carpenter J, Clifford P, Fearnhead P. Improved particle filter for nonlinear problems. IEE Proceedings—Radar, Sonar and Navigation. 1999; 146:2–7. https://doi.org/10.1049/ip-rsn:19990255

51. Bérard J, Del Moral P, Doucet A. A lognormal central limit theorem for particle approximations of normalizing constants. Electron J Probab. 2014; 19(94):1–28.

52. Kou SC, Xie XS, Liu JS. Bayesian analysis of single-molecule experimental data. Journal of the Royal Statistical Society Series C. 2005; 54:496–506. https://doi.org/10.1111/j.1467-9876.2005.00509.x

53. Johansen AM, Doucet A. A note on auxiliary particle filters. Statistics and Probability Letters. 2008; 78 (12):1498–1504. https://doi.org/10.1016/j.spl.2008.01.032

54. Tavaré S. Line-of-descent and genealogical processes, and their application in population genetic models. Theoretical Population Biology. 1984; 26:119–164. https://doi.org/10.1016/0040-5809(84)90027-3

55. Briers M, Doucet A, Maskell S. Smoothing algorithms for state-space models. Annals of the Institute of Statistical Mathematics. 2009; 62:61–89. https://doi.org/10.1007/s10463-009-0236-2

56. Harpak A, Lan X, Gao Z, Pritchard JK. Frequent nonallelic gene conversion on the human lineage and its effect on the divergence of gene duplicates. Proc Nat Acad Sci. 2017; 114(48):12779–12784. https://doi.org/10.1073/pnas.1708151114 PMID: 29138319

57. Whelan S, Goldman N. Estimating the Frequency of Events That Cause Multiple-Nucleotide Changes. Genetics. 2004; 167(4):2027–2043. https://doi.org/10.1534/genetics.103.023226 PMID: 15342538

58. McVicker G, Gordon D, Davis P Colleen Green. Widespread Genomic Signatures of Natural Selection in Hominid Evolution. PLoS Genet. 2009; 5(5):e1000471. https://doi.org/10.1371/journal.pgen.1000471 PMID: 19424416

59. Frankham R. Effective population size/adult population size ratios in wildlife: a review. Genetics Research. 1995; 66(2):95–107. https://doi.org/10.1017/S0016672300034455

60. Chang JT, Pollard D. Conditioning as disintegration. Statistica Neerlandica. 1997; 51(3):287–317. https://doi.org/10.1111/1467-9574.00056

61. Hinton G, van Camp D. Keeping neural networks simple by minimizing the description length of their weights. In: Proceedings of the COLT'93; 1993. p. 5–13.

62. Mackay D. Ensemble learning for hidden Markov models; 1997. Available from: www.inference.org.uk/mackay/ensemblePaper.pdf.

63. Feynman RP. Statistical Mechanics: A Set Of Lectures. Reading, Mass: W. A. Benjamin; 1972.

64. McKenzie A, Steel M. Distributions of cherries for two models of trees. Math Biosci. 2000; 164(1):81–92. https://doi.org/10.1016/S0025-5564(99)00060-7 PMID: 10704639

65. Lorentzen L, Waadeland H. Continued Fractions. In: Chui CK, editor. Atlantis Studies in Mathematics for Engineering and Science 1. Atlantis Press / World Scientific; 2008.