ORIGINAL ARTICLE

Demographic inference using particle filters for continuous Markov jump processes

Donna Henderson[1*] | **Sha (Joe) Zhu**[2*] | **Gerton Lunter**[1]

[1]Wellcome Centre for Human Genetics, Oxford University, Oxford, UK

[2]Big Data Institute, Oxford University, Oxford, UK

Correspondence
Gerton Lunter.
Email: gerton.lunter@well.ox.ac.uk

Funding information
This work was supported by Wellcome Trust grants 090532/Z/09/Z and 102423/Z/13/Z.

Demographic events shape a population's genetic diversity, with the coalescent-with-recombination (CwR) model quantifying how the two are related through a sequence of genealogies. However, inferring demography from genome sequences is challenging because genealogies are only weakly constrained by polymorphisms, and the space of genealogies over genomes is huge. Here we show that particle filters are a promising approach to demographic inference. We introduce a continuous-sequence approximation of the CwR, resulting in a continuous-time and -space Markov jump process. We develop a particle filter algorithm for such processes, and introduce the unbounded-lookahead particle filter, generalising the unit-lookahead Auxiliary Particle Filter for discrete-time models. Using Variational Bayes for parameter inference, we can analyze up to four diploid samples simultaneously, resulting in accurate inferences of past population sizes over a larger range of epochs than was previously possible.

KEYWORDS
Continuous-time continuous-space Markov jump process, particle filter, Variational Bayes, coalescent process, demographic inference

Abbreviations: EM, expectation maximization; VB, variational Bayes; IS, importance sampling

[*] Equally contributing authors.

1 | INTRODUCTION

The demographic history of a species has a profound impact on its genetic diversity. Demographic events such as changes in population size, migration and admixture events, and population splits and mergers, shape the genealogies describing how individuals in a population are related. These in turn shape the pattern and frequency of observed genetic polymorhphisms in extant genomes that result from mutations in their ancestors. In principle, by modeling this process and integrating out the unobserved genealogies, it is possible to infer demographic history from observed polymorphisms. However, in practice this is challenging. Individual mutations provide limited information about the tree topology, and are even less informative about branch lengths. In addition, the number of mutations per tree is strictly limited, for two reasons: the expected number of mutations on a genealogy increases logarithmically with the number of observed genomes, and recombination causes genealogies to change along the genome at a rate proportional to the mutation rate. Therefore, after conditioning on observed mutations, considerable uncertainty remains about the genealogies underlying a sample of genomes. Because the space of genealogies across the genome is vast and topologically complex, integrating out this latent variable is hard.

A large number of approaches have been developed to tackle this problem (see Schraiber and Akey, 2015). A common approximation is to treat recombination events as known and assume unlinked loci, either by treating each mutation as independent (Beaumont, 1999; Pritchard et al., 2000; Beaumont et al., 2002; Gutenkunst et al., 2009; Gronau et al., 2011; Excoffier et al., 2013), or by first identifying tracts of genetic material unbroken by recombination (Drummond and Rambaut, 2007; Browning and Browning, 2011; Palamara et al., 2012; Harris and Nielsen, 2013; Hellenthal et al., 2014). To account for recombination while retaining power to infer earlier demographic events, it is necessary to model the genealogy directly. ARGWeaver (Rasmussen et al., 2014) implements the full coalescent with recombination model and uses Markov chain Monte Carlo (MCMC) for inference. While this works well, ARGWeaver does not allow the use of a complex demographic model, and since the genealogies are only weakly shaped by mutations and more strongly affected by the demographic model, the genealogies inferred by the method are strongly biased and are less useful for inferring demography. The PSMC model (Li and Durbin, 2011) limits itself to single diploid genomes, resulting in an elegant and efficient inference method, but with limited power to detect recent changes in population size or complex demographic events. Several other approaches improve on PSMC in various ways (Sheehan et al., 2013; Schiffels and Durbin, 2014; Steinrücken et al., 2015; Terhorst et al., 2017), but remain limited particularly in their ability to infer migration.

A promising approach which so far has not been applied to this problem is to use particle filters for inference. Particle filters are a general and powerful class of methods for inference in latent variable models (Gordon et al., 1993; Doucet et al., 2000; Arulampalam et al., 2002; Doucet and Johansen, 2011) with many desireable properties. Like MCMC approaches they converge to the exact solution in the limit of infinite 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 build up samples from the posterior sequentially. This is helpful since full genealogies over genomes are large data structures.

Originally, particle filters were introduced for models with discrete time evolution and with either discrete or continuous state variables. In our case the latent variable is a sequence of piecewise constant genealogical trees along the genome, with trees changing only after recombination events that, in mammals, occur once per several hundred nucleotides. The observations of the model are polymorphisms, which are similarly sparse. Realizations of the discrete-time model of this process (with "time" playing the role of genome position) are therefore stationary and silent at most transitions, leading to inefficient algorithms. Instead, it seems natural to model the system as 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, here the space of all genealogical trees. Such processes are known as Markov jump processes or purely discontinuous Markov processes (Feller, 1940).

Particle filters have been generalised to continuous-time diffusions (Del Moral et al., 2002; Golightly and Wilkinson, 2006), as well as to Markov jump processes on discrete state spaces (Nodelman et al., 2002; Ng et al., 2005), and hybrids of the two (Doucet et al., 2001; Sherlock et al., 2014), as well as to piecewise deterministic processes (Witeley et al., 2011), but to our knowledge the method has not been extended to Markov jump processes that are continuous in both time and space. Here we develop a particle filter algorithm for continuous time- and space Markov jump processes, and apply it to the coalescent-with-recombination (CwR) model in the continuous-locus limit. The algorithm is similar to the discrete-time particle filter, but instead of working with ratios of probability distributions, the importance sampling step uses Radon-Nikodym derivatives, ratios of probability densities with respect to a common (Lebesgue) measure. A second difference is that continuous-time models have no natural intermediate time steps; instead the algorithm applies importance sampling at user-provided "way points".

Although this algorithms generally works well, it gives biased inferences for recent epochs. In the CwR model, the corresponding evolutionary events introduce long-lived features in the genealogy resulting in an equally long-lived sparsity of singleton mutations. It takes many iterations of the particle filter before this sparse signal shapes the posterior, and by then the few particles carrying the relevant rare events will often have been removed by the particle filter's stochastic resampling process. The Auxiliary Particle Filter (Pitt and Shephard, 1999) addresses a related problem for discrete-time models by making the resampling process dependent on the data one step ahead. This does not work well for continuous-time models, as there is no preferred time interval. Instead we introduce an algorithm that shapes the resampling process by a "lookahead likelihood" function that can depend on data at arbitrary distances, and we show that this approach mitigates the bias seen in the standard implementation.

The particle filter generates samples of genealogies from the posterior distribution, and we infer the parameters of the demographic model from this sample. One strategy is to use stochastic expectation-maximization (SEM; Nielsen, 2000). An issue with EM is that the joint posterior distribution over parameters is represented by a point-mass, ignoring the uncertainty in parameter estimates. This has undesired consequences in the current setting: particle filters tend to under-sample low-rate events, resulting in downwardly biased estimates and a non-zero probability that a zero rate is inferred, which is a fixed point of the SEM procedure. To avoid zero-rate estimates one would like to put an appropriate prior on the parameters, but this is not possible in the standard SEM framework. To address this issue, 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 and allowing us to include an explicit prior ensuring that zero-rate estimates are avoided, and we show that this further reduces the bias. 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, and introduce the lookahead filter. We then describe the continuous-locus coalescent-with-recombination process that is the focus of this work, and end the methods section with a description of the Variational Bayes procedure for parameter inference. In the results section we show the advanges of the lookahead particle filter and Variational Bayes inference using simulated data, and then show the results of analyzing sets of four diploid genomes of individuals from three human populations. A discussion concludes the paper.

2 | METHODS

2.1 | Particle filters

Particle filters, also known as Sequential Monte Carlo (SMC) methods (Doucet and Johansen, 2011), are a class of algorithms for generating samples from a complex probability distributions with high-dimensional latent variables. SMC methods use importance sampling (IS) to approximate these distributions, using weighted random samples termed particles drawn from a tractable distribution. We briefly review the discrete-time case. Suppose that particles $\{(x^{(i)}, w^{(i)})\}_{i=1,\dots,N}$, approximate a distribution with density $p(x)$, such that for $X \sim p(x)\mathrm{d}x$,

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

for any bounded continuous function f. Throughout we use "approximate" and the symbol $\approx$ to mean that equality holds as $N \to \infty$.[1] Using IS we can obtain particles approximating another distribution $q(x)$:[2]

$$\begin{aligned} E_q[f(X)] &= \int f(x)\frac{q(x)}{p(x)}p(x)\mathrm{d}x \approx \frac{1}{\sum w^{(i)}} \sum w^{(i)} \frac{q(x^{(i)})}{p(x^{(i)})} f(x^{(i)}) \\ &= \frac{\sum \widetilde{w}^{(i)}}{\sum w^{(i)}} \frac{1}{\sum \widetilde{w}^{(i)}} \sum \widetilde{w}^{(i)} f(x^{(i)}) \approx \frac{1}{\sum \widetilde{w}^{(i)}} \sum \widetilde{w}^{(i)} f(x^{(i)}), \end{aligned} \tag{2}$$

where $\widetilde{w}^{(i)} := w^{(i)} q(x^{(i)})/p(x^{(i)})$, and the last step holds because $\sum \widetilde{w}^{(i)} / \sum w^{(i)} \approx E_p[q/p] = E_q[1] = 1$. This shows that $\{(x^{(i)}, \widetilde{w}^{(i)})\}$ approximate $q(x)$. [3] Note that any constant factor in $w^{(i)}$ drops out because of the normalisation, so that it is sufficient to determine 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 have the form

$$P(X = x_{1:s}) = p(x_1)p(x_2|x_1)\cdots p(x_s|x_{s-1}) \tag{3}$$

$$P(Y = y_{1:s}|X = x_{1:s}) = g(y_1|x_1)\cdots g(y_s|x_s) \tag{4}$$

Here $1{:}s$ denotes the set $\{1, 2, \dots, s\}$, and $x = x_{1:s} = (x_1, x_2, \dots, x_s)$ and $y = y_{1:s}$ are vectors. Let $\{(x^{(i)}, w^{(i)})\}$ be particles approximating the posterior $P(X_{1:s} = x_{1:s}|Y_{1:s} = y_{1:s})$, which for brevity we write as $p(x_{1:s}|y_{1:s})$. If $\widetilde{x}^{(i)}$ is the vector obtained by extending $x^{(i)}$ with a sample from $p(x_{s+1}|x_s^{(i)})$, then from (3) and (4) it follows that $\{(\widetilde{x}^{(i)}, w^{(i)})\}$ approximate $P(X_{1:s+1} = x_{1:s+1}|Y_{1:s} = y_{1:s}) \propto P(X_{1:s+1} = x_{1:s+1}, Y_{1:s} = y_{1:s})$. Now, $p(x_{1:s+1}|y_{1:s+1}) \propto P(X_{1:s+1} = x_{1:s+1}, Y_{1:s} = y_{1:s})g(y_{s+1}|x_{s+1})$, so that using IS and setting

$$\widetilde{w}^{(i)} = w^{(i)} g(y_{s+1}|\widetilde{x}_{s+1}^{(i)}) \tag{5}$$

we obtain particles $\{(\widetilde{x}^{(i)}, \widetilde{w}^{(i)})\}$ that approximate $p(x_{1:s+1}|y_{1:s+1})$. This shows how to sequentially construct particles that approximates the posterior distribution $p(x_{1:L}|y_{1:L})$. Instead of sampling from $p(x_{s+1}|x^{(i)})$, any proposal distribution $q(x_{s+1}|x^{(i)}, y_{1:L})$ can be used[2], which is advantageous if q is easier to sample from, is closer to the pos-

[1] The sequence of random variables X_n where $X_n \sim \sum_{i=1}^n w^{(i)} \delta_{x^{(i)}}(x)$ converges in distribution to $X \sim p(x)\mathrm{d}x$.

[2] The ratio $q(x)/p(x)$ must exist almost everywhere and be absolutely continuous.

[3] For brevity we write $q(x)$ for "the distribution with probability mass (density) function $q(x)$ (w.r.t. a suitable measure)".

terior distribution, or has heavier tails than p. Again, IS accounts for the change in sampling distribution, resulting in

$$\widetilde{w}^{(i)} = w^{(i)} g(y_{s+1}|\widetilde{x}_{s+1}^{(i)}) \frac{p(\widetilde{x}_{s+1}^{(i)}|x_s^{(i)})}{q(\widetilde{x}_{s+1}^{(i)}|x_s^{(i)})}. \tag{6}$$

For now we will choose q to be independent of y, and have dropped y from the notation. Because samples from q generally do not follow the desired posterior $p(x|y)$, the fraction of particles that is close to the posterior's mode diminishes exponentially at each iteration until (1) fails altogether. To address this, a resampling step draws samples from the approximating distribution itself, assigning each resampled particle weight $1/N$ – the same process that is used in the Wright-Fisher model with selection to describe how fitness differences shape an evolving population (Heyde, 1977). This procedure removes particles with small weight that have drifted from the mode of the posterior, and duplicates those with large weights, while preserving the validity of (1). Although resampling substantially decreases the future variance of (1), it increases the variance at the present iteration. To avoid increasing this variance unnecessarily, resampling is performed only when the estimated sample size $ESS = \sum w^{(i)} / \sum (w^{(i)})^2$ drops below a threshold, usually $N/2$. This leads to Algorithm 1 (Gordon et al., 1993). The full likelihood can be estimated by setting the weights to $N^{-1} \sum w_{1:s}^{(i)}$ rather than N^{-1} when particles are resampled. This makes the weights asymptotically normalized, so that (1) becomes $E_{p(X,Y=y)}[f] \approx \sum w^{(i)} f(x^{(i)})$, and in particular, $P(Y=y) = \int p(x,y)\mathrm{d}x = E_{p(X,Y=y)}[1] \approx \sum w_{1:L}^{(i)}$.

Algorithm 1 Particle filter

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

2.2 | Continuous-time and -space Markov jump processes

We now consider Markov jump processes, with as realisations piecewise constant functions $x : [1, L) \mapsto \mathbb{T}$ where $\mathbb{T}$ is the state space of the Markov process. Denote the space of all realisations on $[1, L)$ by $\mathcal{X} = \mathbb{T}^{[1,L)}$, and the probability measure on $\mathcal{X}$ induced by the stochastic process $\{X_s\}_{s\in[1,L)}$ by $\pi_x(X)$. The definition of a Markov model involves the concept of conditional probability, which is usually defined as $P(A|B) = P(A \cap B)/P(B)$. This strategy fails for Markov processes; for instance, the probability that a realisation x has the property $x_s = \tau$ for any state $\tau \in \mathbb{T}$ is 0. Instead, the concept of a conditional *distribution* is used, written $X \sim \pi(\cdot|\mathcal{G})$ where $\mathcal{G}$ is the set being conditioned on; see the Appendix for details. The probability distribution of a complete observation $y \in \mathcal{Y}$ is denoted by $\pi(Y|X = x)$,

where $\mathcal{Y}$ is the space of all possible observations on $[1, L)$, and π itself denotes the full hidden Markov process on $\mathcal{X} \times \mathcal{Y}$. The posterior distribution of interest is π conditioned on observed mutations $y \in \mathcal{Y}$, written as $\pi(X|Y = y)$.

In our case, $\mathbb{T}$ is the set of all possible genealogies at a given locus, and the exit rate for the Markov process on $\mathcal{X}$ at a state $x_s \in \mathbb{T}$ is $\rho(s)B(x_s)$ where $\rho(s)$ is the local recombination rate per nucleotide and per generation and $B(x_s)$ is the total branch length of genealogy x_s at locus s. Mutations follow a Poisson process whose rate at s depends on the hidden state x_s (a Cox process, see e.g. Kou et al., 2005). Here the local rate is $\mu(s)B(x_s)$ 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, $A_j \in \{0, 1\}^S$ represents the distribution of wildtype (0) and alternative (1) alleles observed in the S samples.

To describe the Markov jump process version of algorithm 1 we introduce some notation. As above π_x denotes the prior distribution of the latent variable, and ξ_x denotes the proposal distribution, both Markov processes on $\mathcal{X}$, playing the role of $p(x)$ and $q(x)$ for the discrete case. Let $a{:}b$ denote the real interval $[a, b)$, and let $\alpha^{a:b}$ denote the restriction of a measure or function α to the interval $[a, b) \subseteq [1, L)$. Let $\pi(Y|X = x)$ denote the conditional distribution over $\mathcal{Y}$ given a realisation x (the "emission distribution"), and for an element $y \in \mathcal{Y}$ with $|y|$ mutations, let $\lambda(y)$ be the Lebesgue measure $(\mathrm{d}s)^{|y|}$, so that $\pi(Y|X = x)$ has a density relative to λ. The particle filter algorithm uses the notation $(\mathrm{d}\alpha/\mathrm{d}\beta)(x)$ for distributions α and β 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 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 $(\mathrm{d}\pi^{s_j:s_{j+1}}(Y|X_{s_j:s_{j+1}} = x)/\mathrm{d}\lambda^{s_j:s_{j+1}})(y_{s_{j-1}:s_j})$. With this notation we can formulate Algorithm 2.

Algorithm 2 Particle filter for Markov jump processes

Input: $y_{1:L} \in \mathcal{Y}$; waypoints $1 = s_0 < s_1 < \ldots < s_K = L$.
Output: Particles $\{(x^{(i)}_{1:L}, w^{(i)}_L)\}$ approximating the posterior distribution $\pi(X|Y = y_{1:L})$

$w^{(i)}_1 \leftarrow N^{-1}, x^{(i)}_{1:1} \leftarrow \emptyset \; (i = 1, \ldots, N)$
For j from 0 to $K - 1$:
  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 $\{w^{(i)}_{s_j}\}$
    $w^{(i)}_{s_j} \leftarrow N^{-1} \; (i = 1, \ldots, N)$
  For i from 1 to N:
    Sample $x^{(i)}_{s_j:s_{j+1}} \sim \xi_x(X_{s_j:s_{j+1}}|X_{s_j} = x^{(i)}_{s_j})$
    $w^{(i)}_{s_{j+1}} \leftarrow w^{(i)}_{s_j} \dfrac{\mathrm{d}\pi_x}{\mathrm{d}\xi_x}(x^{(i)}_{s_j:s_{j+1}}|X_{s_j} = x^{(i)}_{s_j}) \dfrac{\mathrm{d}\pi}{\mathrm{d}\lambda}(y_{s_j:s_{j+1}}|X_{s_j:s_{j+1}} = x^{(i)}_{s_j:s_{j+1}})$

Note that there is no initialization step; instead, initially $x^{(i)}_{1:1} = \emptyset$, and the first sample will be drawn from ξ conditioned on an empty set, i.e. the unconditional distribution. The invariant is trivially true when $j = 0$ since $1{:}s_0 = \emptyset$. As with Algorithm 1 it is possible to estimate the likelihood of $y_{1:L}$ by replacing the factors N^{-1} with $N^{-1} \sum w^{(i)}_{1:s_j}$; then the likelihood density w.r.t. $\lambda(\mathrm{d}y)$ is approximated by $\sum w^{(i)}_{1:L}$, that is, the likelihood measure is approximated by $(\mathrm{d}s)^{|y|} \sum w^{(i)}_{1:L}$, where the factor $(\mathrm{d}s)^{|y|}$ appears because the probability of observing each one of the $|y|$ mutations at precisely its location $s = s_j$ is infinitesimally small.

The "waypoints" s_j can be chosen arbitrarily, but in practice their choice 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.

2.3 | A lookahead particle filter for Markov jump processes

At the jth 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 $\pi(X_{1:s_j}|y_{1:s_j})$ does not involve data beyond s_j. However, not all particles are equally important for approximating subsequent posteriors, and emphasising particles that will be relevant in future, at the expense of those that are relevant only to $\pi(X_{1:s_j}|y_{1:s_j})$, improves the overall accuracy of the algorithm. This argument echoes the justification of resampling: although resampling increases the variance of the approximation to the current partial posterior, it reduces the variance at subsequent iterations by increasing the number of particles that are likely to contribute to future distributions. The Auxiliary Particle Filter (Pitt and Shephard, 1999) implements this intuition for discrete-time models $p(X_{1:n}, Y_{1:n})$ by targeting a special resampling distribution (Johansen and Doucet, 2008). This distribution includes a "lookahead" factor $\tilde{p}(y_{i+1}|x_i)$ that approximates the probability of observing data y_{i+1} at the next time step 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, instead of using a one-step or fixed-distance lookahead. Since at intermediate stages of the algorithm no approximation of the future state is yet available, the lookahead distribution can be conditioned on the current state only, and will in practice 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 (Doucet and Johansen, 2011), which implies that the distribution should not depend on data too far beyond s; what is "too far" will depend on how well the lookahead distribution approximates the true distribution.

The lookahead distribution is only evaluated on a fixed observation y, and is not used to define a distribution over y, but to quantify the plausibility of a current state $x_s^{(i)}$. For this reason we refer to it as the lookahead *likelihood*. In fact, for the validity of the algorithm below it is not necessary for this likelihood to derive from a proper distribution. We therefore define the lookahead likelihood as an arbitrary[4] 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 $\widetilde{\pi}^s(x_{1:s}, y_{1:L}) = \pi^{1:s}(x_{1:s}, y_{1:s})h^s(y_{s:L}|x_s)\lambda^{s:L}(y_{s:L})$ on $\mathcal{X}_{1:s} \times \mathcal{Y}$.

The lookahead algorithm 3 keeps track of two sets of weights and one set of samples, which form two sets of particles that approximate the resampling and target distributions. The resampling distribution alone controls the resampling process, and ensures that samples that are likely to contribute to subsequent posteriors, as quantified by the lookahead likelihood, are likely to be kept even if they have low likelihood under the current target distribution.

2.4 | The sequential coalescent with recombination model

The coalescent-with-recombination (CwR) process, and the graph structure that results from it, was first described by Hudson (1983), and was given an elegant mathematical description by Griffiths and Marjoram (1997), who named the resulting structure the Ancestral Recombination Graph (ARG). Like the coalescent process, these models run backwards in 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 spatially rather than in the time direction was introduced by Wiuf and Hein (1999), 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)

[4] To ensure that importance sampling is justified, $h^s(\cdot, x_s)\lambda^{s:L}$ must be absolutely continuous w.r.t. $\pi^{s:L}(\cdot|X_s = x_s)$.

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 $K-1$:

Invariant: $\{(x_{1:s_j}^{(i)}, w_{s_j}^{(i)})\} \approx \pi^{1:s_j}(X_{1:s_j}|Y_{1:s_j} = y_{1:s_j})$

Invariant: $\{(x_{1:s_j}^{(i)}, v_{s_j}^{(i)})\} \approx \widetilde{\pi}^{s_j}(X_{1:s_j}|Y = y_{1:L})$

If $ESS(\{v_{s_j}^{(i)}\}) < N/2$

Resample $\{x_{1:s_j}^{(i)}\}$ with probabilities proportional to $\{v_{s_j}^{(i)}\}$

$w_{s_j}^{(i)} \leftarrow N^{-1} w_{s_j}^{(i)} / v_{s_j}^{(i)} \; (i = 1, \ldots, N)$

$v_{s_j}^{(i)} \leftarrow N^{-1} \; (i = 1, \ldots, N)$

For i from 1 to N:

Sample $x_{s_j:s_{j+1}}^{(i)} \sim \xi_x(X_{s_j:s_{j+1}}|X_{s_j} = x_{s_j}^{(i)})$

$w_{s_{j+1}}^{(i)} \leftarrow w_{s_j}^{(i)} \dfrac{\mathrm{d}\pi_x}{\mathrm{d}\xi_x}(x_{s_j:s_{j+1}}^{(i)}|X_{s_j} = x_{s_j}^{(i)}) \dfrac{\mathrm{d}\pi}{\mathrm{d}\lambda}(y_{s_j:s_{j+1}}|X_{s_j:s_{j+1}} = x_{s_j:s_{j+1}}^{(i)})$

$v_{s_{j+1}}^{(i)} \leftarrow v_{s_j}^{(i)} \dfrac{\mathrm{d}\pi_x}{\mathrm{d}\xi_x}(x_{s_j:s_{j+1}}^{(i)}|X_{s_j} = x_{s_j}^{(i)}) \dfrac{\mathrm{d}\widetilde{\pi}^{s_{j+1}}}{\mathrm{d}\widetilde{\pi}^{s_j}}(y_{s_j:L}|X_{s_j:s_{j+1}} = x_{s_j:s_{j+1}}^{(i)})$

developed by Staab et al. (2015) 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 (McVean and Cardin, 2005; Marjoram and Wall, 2006). 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 Staab et al. (2015).

The probability measure of a realisation x can be written as

$$\pi_x(x|\rho, C) = \exp\left\{-\int B(x_s)\rho(s)\mathrm{d}s\right\} \left[\prod_{j=1}^{|x|} \exp\left\{-\int_{\nu_j}^{\tau_j} b_u(x_{s_j})C(u)\mathrm{d}u\right\} \rho(s_j)C(\tau_j)\right] (\mathrm{d}s)^{|x|}(\mathrm{d}u)^{2|x|}. \quad (7)$$

Here $x \in \mathcal{X}$ is the sequence of genealogies, $|x|$ is the number of recombinations that occurred on x, $b_u(x_s)$ is the number of branches in the genealogy at position s at 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, (s_j, ν_j) is the locus and time of the jth recombination event, $\tau_j > \nu_j$ is the time of the corresponding coalescence event, and $C(u) = 1/2N_e(u)$ is the coalescence rate in generation u. To make the formula work, 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 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. The measure (7) 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 (7) 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 ν_j^x and τ_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.

The conditional probability measure of the observations $y = \{(s_j, A_j)\}_{j=1}^{|y|}$ given a realisation x becomes

$$\pi(y|X = x) = \exp\left\{-\int B(x_s)\mu(s)\mathrm{d}s\right\}\left[\prod_{j=1}^{|y|} P(A_j|x_{s_j}, \mu(s_j))\right](\mathrm{d}s)^{|y|} \tag{8}$$

where $P(A|x_s, \mu)$ is the probability of observing the allelic pattern A given a genealogy x_s and a mutation rate μ per nucleotide and per generation; this probably is calculated using Felsenstein's peeling algorithm (Felsenstein, 1981).

2.5 | Parameter inference

To infer parameters using stochastic expectation maximization (SEM), note that the expected log likelihood for a Poisson process with rate ρ is $c \log \rho - t\rho$ where c and t are the expected event count and opportunity, which is maximized for $\rho = c/t$. For Markov jump processes both the events count and the opportunity can be random variables. From (7-8), the parameters that maximize the expected log of $\pi(x, y|\theta) = \pi_x(x|\theta)\pi(y|x)$ are

$$\rho'_{\mathsf{EM}} = \frac{\mathbb{E}[\,|x|\,]}{\mathbb{E}[\int B(x_s)\mathrm{d}s]} \quad \text{and} \quad C'_{\mathsf{EM}} = \frac{\mathbb{E}[\,|x|\,]}{\mathbb{E}[\sum_{j=1}^{|x|}\int_{\nu_j}^{\tau_j} b_u(x_{s_j})]}, \tag{9}$$

where $\theta = (\rho, C)$ is the vector of current parameter estimates, and expectations are taken over $x \sim \pi(x|y, \theta)$ as approximated by algorithm 3. Note that C'_{EM} in (9) is constant in time. In practice we maximize (7-8) with respect to piecewise constant functions $C'_{\mathsf{EM}}(t)$, which yields

$$C'_{\mathsf{EM}}(t) = \frac{\mathbb{E}[\,|x|_{\nu,\tau}\,]}{\mathbb{E}[\sum_{j=1}^{|x|}\int_{u\in[\nu_j,\tau_j)\cap[\nu,\tau)} b_u(x_{s_j})\mathrm{d}u]} \tag{10}$$

for $t \in [\nu, \tau)$, where we wrote $|x|_{\nu,\tau}$ for the number of coalescent events in x that occur in the epoch $[\nu, \tau)$.

To develop an equivalent Variational Bayes (VB) inference procedure, note that for the Poisson example above, a $\Gamma(\rho|\alpha, \beta)$ prior would result in a $\Gamma(\rho|\alpha', \beta')$ posterior, where $\alpha' = \alpha + c$, $\beta' = \beta + t$, and c, t are the expected event count and opportunity given the data and the current posterior distribution on ρ. With this choice of prior ρ can be integrated out analytically in the likelihood $\pi(x, y|\theta) = \int_\rho \pi(x, y|\rho)\Gamma(\rho|\theta)\mathrm{d}\rho$, where $\theta = (\alpha', \beta')$ are the parameters of the Gamma prior, resulting in an expression that is identical to the likelihood under the point estimate $\rho = \alpha'/\beta'$ except for additional weighting factors for each point event in x; see the Appendix for details. These weighting factors render the normalization constant of the likelihood intractable, but fortunately SMC algorithms only require likelihoods up to normalization so that algorithm 3 can be used to calculate the required expectations at no additional computational cost. The inferred posterior distributions of recombination and coalescent rates take the form

$$\rho'_{\mathsf{VB}} \sim \Gamma(\alpha_\rho + \mathbb{E}[\,|x|\,], \beta_\rho + \mathbb{E}[\int B(x_s)\mathrm{d}s]) \tag{11}$$

$$C'_{\mathsf{VB}} \sim \Gamma(\alpha_C + \mathbb{E}[\,|x|\,], \beta_C + \mathbb{E}[\sum_{j=1}^{|x|}\int_{\nu_j}^{\tau_j} b_u(x_{s_j})]) \tag{12}$$

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 (11-12) of the previous iteration.

To estimate the expectations (9-12) we do not use full realisations x since resampling causes early parts of x to become degenerate due to "coalescences" of the particle's sampling history along the sequence, resulting in high variance. Using the most recently sampled events is also problematic as these have not been shaped by many observations and mostly follow the prior $\pi_x(\theta)$, resulting in highly biased estimates. Smoothing techniques such as two-filter smoothing (Briers et al., 2009) are difficult to apply here since finite-time transition probabilities are intractable. For discrete-time models fixed-lag smoothing is often used (Doucet and Sénécal, 2004). 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 α; we obtain good results with $\alpha = 1$.

3 | RESULTS

We implemented the algorithms outlined above in a C++ program `SMC`^{2}, and assessed it on simulated and real data.

3.1 | Simulation study

To investigate the effect of the lookahead particle filter, we simulated four 50 megabase (Mb) diploid genomes under a constant population-size model ($N_e = 10,000$, $\mu = 2.5 \times 10^{-8}$ / year, $\rho = 10^{-8}$ / year, generation time $g = 30$ years). We inferred population sizes N_e through time using particle filters Alg. 2 and Alg. 3. Observations are often available as unphased genotypes, and we assessed both algorithms using phased and unphased data, from the same simulated data. Experiments were run for 15 EM iterations and repeated 15 times (Fig. 1a). Without lookahead, inferences of N_e show a strong positive bias in recent epochs, corresponding to a negative bias in the inferred coalescence rate. This bias is less pronounced for unphased data, presumably because these data cause the likelihood surface to be smoother, making it less challenging for the particle filter to track. Increasing the number of particles reduces this bias somewhat. The lookahead filter shows no discernable bias on phased data, even for $1,000$ particles. The lookahead likelihood uses the distance to the next singleton to guide the particle filter towards very recent coalescences, and this is less effective for unphased data. Indeed, although for unphased data the bias does decrease as the number of particles increases, it remains considerable for the most recent epoch (here, less than $10,000$ years ago).

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 Schiffels and Durbin (2014) and Terhorst et al. (2017). Both approaches give accurate inferences from $2,000$ years up to 1 million years ago (Mya); other experiments suggest that popuation sizes can be inferred up to 10 Mya (data not shown). Inferences in the most recent epochs remain biased, which however is mitigated considerably by the Variational Bayes appraoch compared to SEM (Fig. 1b).

3.2 | Inference on human subpopulations

We applied `SMC`^{2} to three sets of four phased diploid samples of Central European (CEU), Han Chinese (CHB) and Yoruban (YRI) ancestry from the 1000 Genomes project. For comparison we also ran `msmc` on the same data, and on

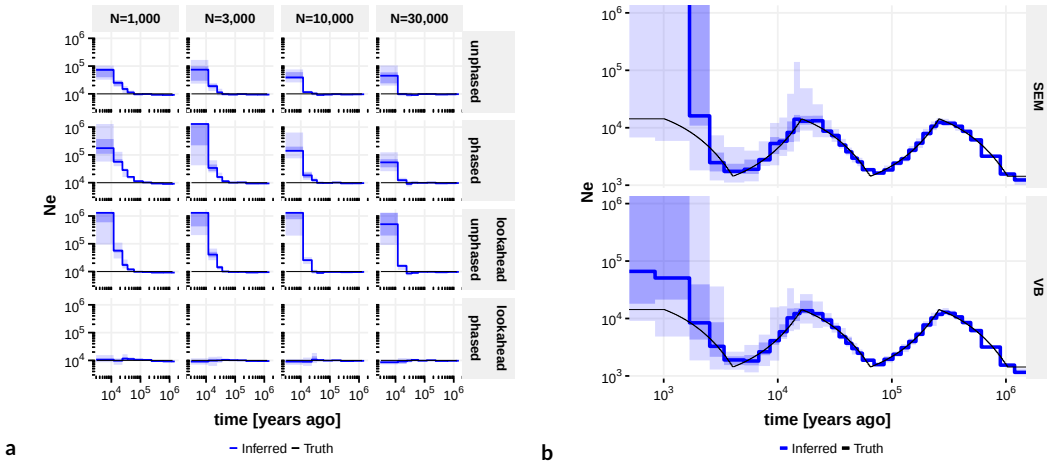

FIGURE 1 True population sizes (black), and median inferred population sizes across 15 independent runs (blue); shaded areas denote quartiles and full extent. **a** Impact of lookahead, phasing and number of particles on the bias in population size estimates for recent epochs, for data simulated under 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).

subsets of 2 and 1 diploid samples. Inferences show good agreement where `msmc` has power. 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 section 5.5), and inferred population sizes using $\texttt{SMC}^2$ and `msmc` as before. This confirmed the accuracy of $\texttt{SMC}^2$ inferences from about 5,000 to 5 million years ago, while inferences in more recent epochs show more variability.

4 | DISCUSSION

Motivated by the challenging inference problem of recovering a population's demographic history from the genomes of a sample of its individuals (Schraiber and Akey, 2015), we have here developed a particle filter algorithm. This algorithm allowed us to perform inference under the full coalescent-with-recombination model, without having to resort to model approximations that characterizes most other approaches. The model however proved very challenging, and to overcome these we have generalised the standard particle filter algorithm in two ways. First, we have extended the method to Markov jump processes that are continuous both in the "sequential" variable (usually time, but here genome locus) as well as in the state space variables. Second, we have generalized the Auxiliary Particle Filter of Pitt and Shephard (1999) from a discrete-time one-step-lookahead algorithm to a continuous-time unbounded-lookahead method. A feature of the coalescent-with-recombination model that recent demographic events induce induce"sticky" model states with very long forgetting times, posing particular problems for inference. With an appropriate "lookahead likelihood" function, we showed that the unbounded-lookahead algorithm mitigates the bias that is otherwise observed in the inferred parameters associated with these recent demographic events. We then showed that the remaining bias

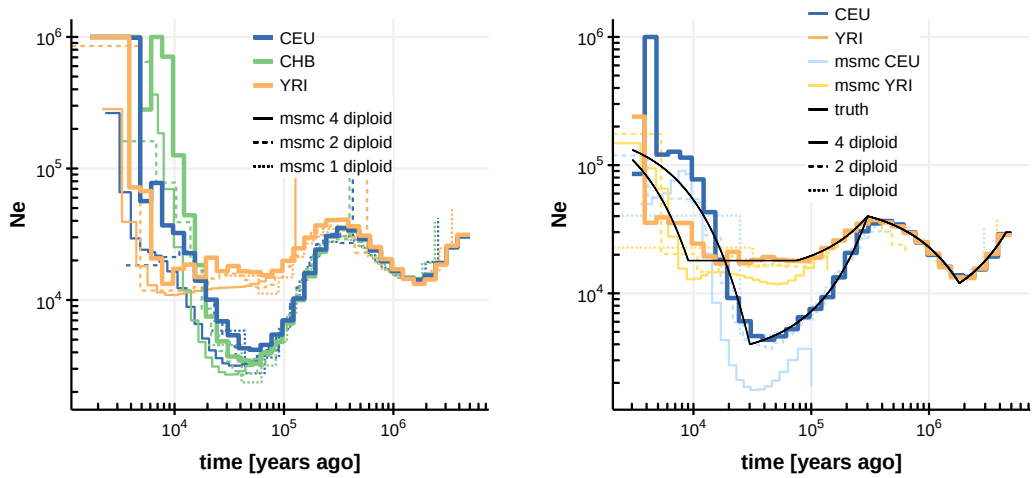

FIGURE 2 Population size inferences by SMC^2 on four diploid samples. Left, three human populations (CEU, CHB, YRI), together with inferences from msmc using 1, 2 and 4 diploid samples. Right, two simulated populations resembling CEU and YRI population histories. All inferences (SMC^2, msmc) were run for 20 VB or EM iterations.

was further reduced by a Variational Bayes alternative to stochastic expectation maximization, which explicitly models part of the uncertainty in the inferred parameters. In combination, the algorithm 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 repectively, 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 (Li and Durbin, 2011; Sheehan et al., 2013; Schiffels and Durbin, 2014; Steinrücken et al., 2015; Terhorst et al., 2017), 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., Hellenthal et al., 2014) 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 Terhorst et al. (2017) are used.

A key advantage of particle filters is that they are fundamentally simulation-based. This makes it possible to analyze complex models, as long as forward simulation (along the sequential variable) is tractable. The proposed particle filter is based on the sequential coalescent simulator SCRM (Staab et al., 2015), 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 will therefore be straightforward to infer the associated model parameters, including directional migration rates. In addition, several aspects of the standard coalescent-with-recombination model are known to be unrealistic. For instance, gene conversions and doublet mutations are common (Harpak et al., 2017; Whelan and Goldman, 2004), and background selection profoundly shapes the heterozygousity in the human genome (McVicker et al., 2009). 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 account for the bias they may otherwise incur in inferences. Because a particle filter produces an estimate of the data likelihood, any improved model fit resulting from adding any of these features can easily be quantified. A further advantage of a particle filter is that it provides a sample from the posterior distribution of ancestral recombination graphs (ARGs). This gives straightforward access to estimates of the age of mutations and recombinations, and allows 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 (Rasmussen et al., 2014), 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.

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, perhaps leading to more insight in our species' recent demographic history.

5 | APPENDIX

5.1 | Conditional distributions and the Markov property

Here we outline how to define a conditional distribution $\pi(\cdot|\mathcal{G})$ given a distribution π 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 τ 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 μ so that $\pi_{\mathcal{G}_\tau}$ is concentrated on $\mathcal{G}_\tau$, $\pi_{\mathcal{G}_\tau}(\mathcal{X}) = 1$ for all τ, and $E_\pi[f] = \iint f(x)\pi_{\mathcal{G}_\tau}(\mathrm{d}x)\mu(\mathrm{d}\tau)$ for well-behaved functions f (Chang and Pollard, 1997), 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 π can be expressed in terms of conditional expectations:

$$E_\pi[f(x_t)|x_{s_1} = \tau_1, \ldots, x_{s_k} = \tau_k] = E_\pi[f(x_t)|x_{s_k} = \tau_k] \tag{13}$$

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

5.2 | Proof of algorithm 3

The algorithm is proved by induction on j. For $j = 0$ the invariant is trivially true, while for $j = K$ the invariant implies the output condition. Suppose the 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 kth new particle, $\hat{v}^{(k)} = N^{-1}$ and $\hat{w}^{(k)} = N^{-1}w^{(i_k)}/v^{(i_k)}$ be its weights, and write $\pi^j = \pi(X_{1:s_j}|Y_{1:s_j} = y_{1:s_j})$, $\widetilde{\pi}^j = \widetilde{\pi}^{s_j}(X_{1:s_j}|Y_{1:s_j} = y_{1:s_j})$, then

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

so that the 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 the two distributions gives

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

and since $\pi(x_{1:s_{j+1}}, y_{s_j:s_{j+1}}|Y_{1:s_j} = y_{1:s_j}) \propto \pi(x_{1:s_{j+1}}|Y_{1:s_{j+1}} = y_{1:s_{j+1}})\lambda_{s_j:s_{j+1}}(y_{s_j:s_{j+1}})$ so that the weights $w^{(i)}_{s_{j+1}}$ together with the samples $X^{(i)}_{1:s_{j+1}}$ form particles approximating $\pi(x_{1:s_{j+1}}|Y_{1:s_{j+1}} = y_{1:s_{j+1}})$. The argument showing that $(v^{(i)}_{s_{j+1}}, X^{(i)}_{1:s_{j+1}}) \approx \widetilde{\pi}^{s_j}(x_{1:s_{j+1}}|Y = y_{1:L})$ is analogous. This proves the invariant for $j+1$, and the algorithm.

5.3 | 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 $\mathrm{d}x\mathrm{d}y$ admit a probability density of the form

$$
\pi_{xy}(x, y|\theta)\mathrm{d}x\mathrm{d}y = \pi_y(y|x)\prod_i \exp\{-\theta_i B_i(x)\}\,\theta_i^{|x|_i}\,\mathrm{d}x\mathrm{d}y. \tag{14}
$$

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

$$
B_U^R(x) = \int_s \int_{u\in U} b_u(x_s)\mathrm{d}u\mathrm{d}s; \qquad B_U^C(x) = \sum_{j=1}^{|x|}\int_{u\in[\nu_j,\tau_j]\cap U} b_u(x_{s_j})\mathrm{d}u, \tag{15}
$$

and $|x|_U^R = \#\{j : \nu_j \in U\}$, $|x|_U^C = \#\{j : \tau_j \in U\}$, 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 π_θ is a prior on the parameters, with a probability density $\phi(x, \theta)$ that is easier to work with.[5] Following Hinton and van Camp (1993) and Mackay (1997), we choose to constrain ϕ 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 (Feynman, 1972),

$$
F(\phi) = -\int_x \int_\theta \phi(x, \theta)\log\left[\frac{\pi(x, \theta|y)}{\phi(x, \theta)}\right]\mathrm{d}\theta\mathrm{d}x. \tag{16}
$$

[5] The constant of proportionality implied by "$\propto$" hides a density $\lambda(y)$.

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

$$F(\phi) = -\int \phi_x(x)\phi_\theta(\theta)\left\{\sum_i |x|_i \log\theta_i - B_i(x)\theta_i + \log\pi_\theta(\theta) - \log\phi_\theta(\theta)\right\}\mathrm{d}\theta\mathrm{d}x + const \tag{17}$$

$$= \int \phi_\theta(\theta)\log\frac{\phi_\theta(\theta)}{\pi_\theta(\theta)\prod_i \theta_i^{\mathbb{E}_{\phi_x}[|x|_i]}\exp\{-\mathbb{E}_{\phi_x}[B_i(x)]\theta_i\}}\mathrm{d}\theta + const \tag{18}$$

This is minimized by setting $\log\phi_\theta(\theta)$ equal to 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

$$\phi_\theta(\theta) = \prod_i \Gamma(\alpha_i',\beta_i') \tag{19}$$

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 ϕ_x with ϕ_θ fixed,

$$F(\phi) = -\int \phi_x(x)\phi_\theta(\theta)\left\{\sum_i |x|_i \log\theta_i - B_i(x)\theta_i + \log\pi_y(y|x) - \log\phi_x(x)\right\}\mathrm{d}\theta\mathrm{d}x + const \tag{20}$$

$$= \int \phi_x(x)\log\frac{\phi_x(x)}{\pi_y(y|x)\prod_i \exp\{|x|_i\mathbb{E}_{\phi_\theta}[\log\theta_i] - B_i(x)\mathbb{E}_{\phi_\theta}[\theta_i]\}}\mathrm{d}x + const \tag{21}$$

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 ψ is the digamma function. Again, (21) is minimized if the numerator and denominator are equal, which happens for

$$\phi_x(x) \propto \pi_y(y|x)\prod_i \exp\left\{-\bar\theta_i B_i(x)\right\}(\theta_i^*)^{|x|_i} \propto \pi(x|y,\bar\theta)\prod_i\left(\frac{\theta_i^*}{\bar\theta_i}\right)^{|x|_i} = \pi(x|y,\bar\theta)\prod_i \delta_i^{|x|_i} \tag{22}$$

where $\delta_i = \theta_i^*/\bar\theta_i = \exp\{\psi(\alpha_i')\}/\alpha_i'$. As given, the SMC 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 δ_i in a particle's weight for every event of type i that occurs.

5.4 | Approximate lookahead likelihood

The lookahead particle filter requires a tractable approximate likelihood of future data given a current genealogy. To achieve this we make a number of simplifications. First, most data are ignored, except for a digest of singletons and doubletons that are informative of the topology and branch lengths near the tips of the genealogy. This consists of the distance s_i to the nearest future singleton in each sequence, and the $\leq n/2$ mutually consistent cherries $c_k = (a_k, b_k)$ (i.e. having disjunct doubletons) with positions $s_k' \leq s_k''$ where their first and last supporting doubleton were observed (see Figure 3a). To simplify notation we assume that the current position is 0.

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 (Figure 3b). To compute the likelihood that the first singleton in lineage i occurs at position 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

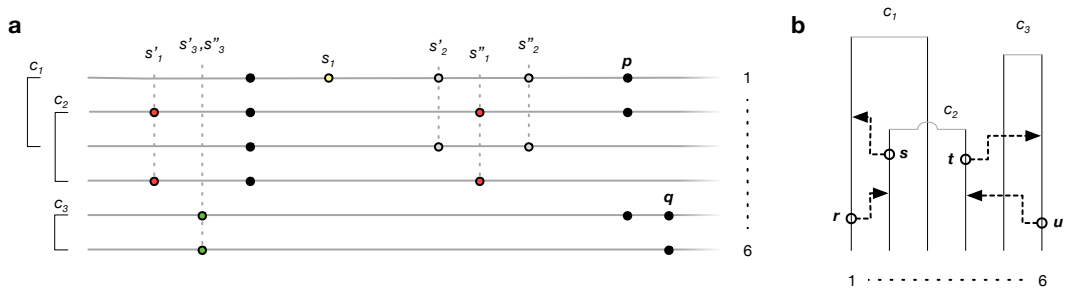

FIGURE 3 **a.** Example of data digest. Lines represent genomes of 6 lineages, circles observed genetic variants. Of the data shown, one singleton in lineage 1 and five doubletons contribute to the digest. Cherry c_3 is supported by a single doubleton; q does not contribute because the mutation pattern p is incompatible with c_3. **b.** Partial genealogy (unbroken lines) over 6 lineages. Open circles and arrows represent examples of recombination and coalescence events that would change the terminal branch length for lineage 1 (r, s), and remove cherry 3 (t, u).

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 $\mu_i := \mu l_i$ 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

$$p(\text{first singleton in } i \text{ at } s_i|l_i, l_i') = (\rho_i + \mu_i - \mu_i')^{-1}\left(\rho_i\mu_i' e^{-\mu_i' s_i} + (\mu_i - \mu_i')(\rho_i + \mu_i)e^{-(\rho_i+\mu_i)s_i}\right)\mathrm{d}t. \quad (23)$$

In the case where no singleton was observed up until s_i but data was missing thereafter, the same probability densities apply except for the factors μ_i and μ_i' in the likelihood, resulting in

$$p(\text{no singleton in } i \text{ until } s_i|l_i, l_i') = (\rho_i + \mu_i - \mu_i')^{-1}\left(\rho_i e^{-\mu_i' s_i} + (\mu_i - \mu_i')e^{-(\rho_i+\mu_i)s_i}\right)\mathrm{d}t. \quad (24)$$

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 cherry $c = (a, b)$ 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 (Figure 3b). 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 $n \geq 3$ (McKenzie and Steel, 2000), 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 position s_k''

conditional on the tree currently containing c_k is

$$p(a_k, b_k, s'_k, s''_k | (a_k, b_k; l) \in \tau) = e^{-\rho_C s''_k} + \frac{2}{3(n-1)}(1 - e^{-\rho_C s''_k}), \tag{25}$$

where $(a_k, b_k; l) \in \tau$ expresses that τ 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 τ. 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 position s'_k is

$$p(a_k, b_k, s'_k, s''_k | \bar{l}, (a_k, b_k) \notin \tau) = \frac{2}{3(n-1)}(1 - e^{-\rho'_C s'_k}), \tag{26}$$

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 (25-26) are likelihoods for τ supporting c_k at the given position, rather than for a doubleton mutation actually occurring.

These likelihood 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 μ 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.

5.5 | Implementation details

We use systematic resampling to minimize the variance introduced through resampling (Carpenter et al., 1999).

The waypoints s_j are chosen to coincide with mutation loci, with additional ones added to ensure they are never more than 10kb apart.

By the nature of Markov jump processes, particles that start from identical states have a positive probability of remaining identical after a finite amount of evolution. Combined with resampling, this causes a considerable number of particles to have identical states throughout the filtering process. For efficiency we represent such particles once, keeping track of their multiplicity k. When evolving a particle with multiplicity $k > 1$, recombination events occur as normal, but at a k-fold increased rate. When a recombination occurs one particle is spawned off, while $k - 1$ continue unchanged.

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 potentially many event records when particles are duplicated at resampling, they 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

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`

TABLE 1 Commands to generate simulation data

particles fail to be resampled, ensuring that memory usage is bounded.

The likelihood calculations involve many relatively slow evaluations of the exponential function, often for small exponents. We use the continued-fraction approximation $e^x \approx 1 + 2x/(2 - x + \frac{1}{6}x^2)$ for $|x| < 0.03$, with relative error bounded by 10^{-10} (Lorentzen and Waadeland, 2008).

Table 1 shows the commands to generate the data for the zigzag, CEU and YRI simulation experiments.

ACKNOWLEDGEMENTS

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

REFERENCES

Arulampalam, M. S., Maskell, S., Gordon, N. and Clapp, T. (2002) A tutorial on particle filters for online nonlinear/non-gaussian bayesian tracking. *IEEE Transactions on Signal Processing*, **50**, 174–188.

Beaumont, M. (1999) Detecting population expansion and decline using microsatellites. *Genetics*, **153**, 2013–2029.

Beaumont, M., Zhang, W. and Balding, D. J. (2002) Approximate bayesian computation in population genetics. *Genetics*, **162**, 2025–2035.

Briers, M., Doucet, A. and Maskell, S. (2009) Smoothing algorithms for state-space models. *Annals of the Institute of Statistical Mathematics*, **62**, 61–89.

Browning, S. R. and Browning, B. L. (2011) A fast, powerful method for detecting identity by descent. *American Journal of Human Genetics*, **88**, 173–182.

Carpenter, J., Clifford, P. and Fearnhead, P. (1999) Improved particle filter for nonlinear problems. *IEE Proceedings - Radar, Sonar and Navigation*, **146**, 2–7.

Chang, J. T. and Pollard, D. (1997) Conditioning as disintegration. *Statistica Neerlandica*, **51**, 287–317.

Del Moral, P., Jacob, J. and Protter, P. (2002) The monte carlo method for filtering with discrete-time observations. *Probability Theory and Related Fields*, **120**, 346–368.

Doucet, A., Godsill, S. and Andrieu, C. (2000) On sequential monte carlo sampling methods for bayesian filtering. *Statistics and computing*, **10**, 197–208.

Doucet, A., Gordon, N. and Krishnamurthy, V. (2001) Particle filters for state estimation of jump markov linear systems. *IEEE Transactions on Signal Processing*, **49**, 613–624. URL: `http://dx.doi.org/10.1109/78.905890`.

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

Doucet, A. and Sénécal, S. (2004) Fixed-lag sequential monte carlo. In *Signal Processing Conference, 2004 12th European*, 861–864. IEEE.

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

Excoffier, L., Dupanloup, I., Huerta-Sánchez, E., Sousa, V. C. and Foll, M. (2013) Robust demographic inference from genomic and snp data. *PLoS Genet*, **9**, e1003905.

Feller, W. (1940) On the integro-differential equations of purely discontinuous markoff processes. *Transactions of the American Mathematical Society*, **48**, 488–515.

Felsenstein, J. (1981) Evolutionary trees from dna sequences: A maximum likelihood approach. *Journal of Molecular Evolution*, **17**, 368–376.

Feynman, R. P. (1972) *Statistical Mechanics: A Set Of Lectures*. Reading, Mass: W. A. Benjamin.

Golightly, A. and Wilkinson, D. J. (2006) Bayesian sequential inference for nonlinear multivariate diffusions. *Statistics and Computing*, **16**, 323–338.

Gordon, N., Salmond, D. and Smith, A. (1993) Novel approach to nonlinear/non-gaussian bayesian state estimation. *IEEE Proceedings F, Radar and Signal Processing*, **140**, 107–113.

Griffiths, R. C. and Marjoram, P. (1997) An ancestral recombination graph. In *Progress in Population Genetics and Human Evolution* (eds. P. Donnelly and S. Tavaré). Springer-Verlag.

Gronau, I., Hubisz, M. J., Gulko, B., Danko, C. G. and Siepel, A. (2011) Bayesian inference of ancient human demography from individual genome sequences. *Nature genetics*, **43**, 1031–1034.

Gutenkunst, R. N., Hernandez, R. D., Williamson, S. H. and Bustamante, C. D. (2009) Inferring the joint demographic history of multiple populations from multidimensional snp frequency data. *PLoS Genet*, **5**, e1000695.

Harpak, A., Lan, X., Gao, Z. and Pritchard, J. K. (2017) Frequent nonallelic gene conversion on the human lineage and its effect on the divergence of gene duplicates. *Proc. Nat. Acad. Sci.*, **114**, 12779–12784.

Harris, K. and Nielsen, R. (2013) Inferring demographic history from a spectrum of shared haplotype lengths. *PLoS Genet*, **9**, e1003521.

Hellenthal, G., Busby, G. B., Band, G., Wilson, J. F., Capelli, C., Falush, D. and Myers, S. (2014) A genetic atlas of human admixture history. *Science*, **343**, 747–751.

Heyde, C. C. (1977) The effect of selection on genetic balance when the population size is varying. *Th Pop Biol*, **11**, 249–251.

Hinton, G. and van Camp, D. (1993) Keeping neural networks simple by minimizing the description length of their weights. In *Proceedings of the COLT'93*, 5–13.

Hudson, R. R. (1983) Properties of a neutral allele model with intragenic recombination. *Theoretical Population Biology*, **23**, 183–201.

Johansen, A. M. and Doucet, A. (2008) A note on auxiliary particle filters. *Statistics and Probability Letters*, **78**, 1498–1504.

Kou, S. C., Xie, X. S. and Liu, J. S. (2005) Bayesian analysis of single-molecule experimental data. *Journal of the Royal Statistical Society Series C*, 496–506.

Li, H. and Durbin, R. (2011) Inference of human population history from individual whole-genome sequences. *Nature*, **475**, 493–496.

Lorentzen, L. and Waadeland, H. (2008) Continued fractions. In *Atlantis Studies in Mathematics for Engineering and Science 1* (ed. C. K. Chui). Atlantis Press / World Scientific.

Mackay, D. (1997) Ensemble learning for hidden markov models. URL: `www.inference.org.uk/mackay/ensemblePaper.pdf`.

Marjoram, P. and Wall, J. D. (2006) Fast "coalescent" simulation. *BMC Genetics*, **7**.

McKenzie, A. and Steel, M. (2000) Distributions of cherries for two models of trees. *Math Biosci*, **164**, 81–92.

McVean, G. A. and Cardin, N. J. (2005) Approximating the coalescent with recombination. *Philos Trans R Soc Lond B Biol Sci*, **360**, 1387–1393.

McVicker, G., Gordon, D. and Davis, Colleen Green, P. (2009) Widespread genomic signatures of natural selection in hominid evolution. *PLoS Genet*, **5**, e1000471.

Ng, B., Pfeffer, A. and Dearden, R. (2005) Continuous time particle filtering. In *Proceedings of the IJCAI*, 1360–1365.

Nielsen, S. F. (2000) The stochastic em algorithm: estimation and asymptotic results. *Bernoulli*, **6**, 457–489.

Nodelman, U., Shelton, C. R. and Koller, D. (2002) Continuous time bayesian networks. In *Proceedings of the UAI*.

Palamara, P. F., Lencz, T., Darvasi, A. and Peér, I. (2012) Length distributions of identity by descent reveal fine-scale demographic history. *Am. J. Hum. Genet.*, **91**, 809–22.

Pitt, M. K. and Shephard, N. (1999) Filtering via simulation: Auxiliary particle filters. *Journal of the American Statistical Association*, **94**, 590–599.

Pritchard, J. K., Stephens, M. and Donnelly, P. (2000) Inference of population structure using multilocus genotype data. *Genetics*, **155**, 945–959.

Rasmussen, M. D., Hubisz, M. J., Gronau, I. and Siepel, A. (2014) Genome-wide inference of ancestral recombination graphs. *PLoS Genet*, **10**, e1004342.

Schiffels, S. and Durbin, R. (2014) Inferring human population size and separation history from multiple genome sequences. *Nature genetics*, **46**, 919–925.

Schraiber, J. G. and Akey, J. M. (2015) Methods and models for unravelling human evolutionary history. *Nature Reviews Genetics*.

Sheehan, S., Harris, K. and Song, Y. S. (2013) Estimating variable effective population sizes from multiple genomes: a sequentially markov conditional sampling distribution approach. *Genetics*, **194**, 647–662.

Sherlock, C., Golightly, A. and Gillespie, C. S. (2014) Bayesian inference for hybrid discrete-continuous systems biology models. *Inverse Problems*, **30**, 114005.

Staab, P. R., Zhu, S., Metzler, D. and Lunter, G. (2015) scrm: efficiently simulating long sequences using the approximated coalescent with recombination. *Bioinformatics*, **31**, 1680–1682.

Steinrücken, M., Kamm, J. A. and Song, Y. S. (2015) Inference of complex population histories using whole-genome sequences from multiple populations. *bioRxiv*, 026591.

Terhorst, J., Kamm, J. A. and Song, Y. S. (2017) Robust and scalable inference of population history from hundreds of unphased whole genomes. *Nature Genetics*, **49**, 303–309.

Whelan, S. and Goldman, N. (2004) Estimating the frequency of events that cause multiple-nucleotide changes. *Genetics*, **167**, 2027–2043.

Witeley, N., Johansen, A. M. and Godsill, S. (2011) Monte carlo filtering of piecewise deterministic processes. *Journal of Computational and Graphical Statistics*, **20**, 119–139.

Wiuf, C. and Hein, J. (1999) Recombination as a point process along sequences. *Theoretical Population Biology*, **55**, 248–259.