Demographic inference using a particle filter for continuous Markov Jump processes

Donna Henderson*, Sha (Joe) Zhu*, Chris Cole and Gerton Lunter†

Wellcome Centre for Human Genetics
Roosevelt Drive
Oxford OX3 7BN, UK
e-mail: donnah@well.ox.ac.uk

Big Data Institute
Old Road
Oxford OX3 7FZ, UK
e-mail: joezhu@well.ox.ac.uk

MRC Weatherall Institute of Molecular Medicine
John Radcliffe Hospital
Headington
Oxford Oxford OX3 9DS, UK
e-mail: ccole@well.ox.ac.uk *e-mail:* gerton.lunter@well.ox.ac.uk

Abstract: Demographic events shape a population's genetic diversity, a process described by the coalescent-with-recombination (CwR) model that relates demography and genetics by an unobserved sequence of genealogies. The space of genealogies over genomes is large and complex, making inference under this model challenging.

We approximate the CwR with a continuous-time and -space Markov jump process. We develop a particle filter for such processes, using waypoints to reduce the problem to the discrete-time case, and generalising the Auxiliary Particle Filter for discrete-time models. We use Variational Bayes for parameter inference 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

MSC 2010 subject classifications: Primary 60G55, 62M05, 62M20, 62F15; secondary 92D25.
Keywords and phrases: Continuous-time continuous-space Markov jump process, Particle filter, Sequential Monte Carlo, Variational Bayes, Demographic Inference.

*Equal contributions.

†We thank Arnaud Doucet for helpful discussions, and Paul Staab for implementing the SCRM library. This work was supported by Wellcome Trust grants 090532/Z/09/Z and 102423/Z/13/Z.

1. 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 shape the pattern and frequency of observed genetic polymorphisms 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 polymorphisms. 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 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 52]. A common approximation is to treat recombination events as known and assume unlinked loci, either by treating each mutation as independent [2, 48, 3, 26, 25, 16], or by first identifying tracts of genetic material unbroken by recombination [15, 6, 46, 28, 29]. To account for recombination while retaining power to infer earlier demographic events, it is necessary to model the genealogy directly. ARGWeaver [49] implements the full coalescent with recombination (CwR) 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 mutations are only weakly informative about genealogies this leaves the inferred trees biased towards the prior model and less directly suitable for inferring demography. Restricting to single diploid genomes, the Pairwise Sequentially Markovian Coalescent (PSMC) model [35] 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 [53, 51, 57, 60], 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 a particle filter. Particle filters have many desireable properties [23, 12, 1, 14], and applications to a range of problems in computational biology have started to appear [59, 55, 21, 61]. 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.

Originally, particle filters were introduced for models with discrete time evolution and with either discrete or continuous state variables [50, 23]. Here, 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 every 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 Markov jump process (or purely discontinuous Markov process, [18]), 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 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 [10, 22, 17], as well as to Markov jump processes on discrete state spaces [44, 42], and hybrids of the two [13, 54], as well as to piecewise deterministic processes [63]; for a general treatment see [11, 9]. Here we focus on the case of Markov jump processes that are continuous in both time and space. The algorithm we propose relies on Radon-Nikodym derivatives [see e.g. 17], 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 depletion remains under control.

Although the algorithm generally works well, we found that for the CwR model we obtain biased inferences for some parameters whose associated latent variables have long forgetting times, requiring fixed-lag smoothing with long lags resulting in particle degeneracy [45]. For discrete-time models the Auxiliary Particle Filter [47] addresses a related problem by "guiding" the sampler 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. Using simulations we show that this substantially reduces the bias.

The particle filter generates samples from the posterior distribution of the latent variable, and we infer the model parameters from this sample. One strategy is to use stochastic expectation-maximization [SEM; 43]. An issue with this approach is that the joint posterior distribution over parameters is represented by a point-mass, ignoring uncertainty in parameter estimates. 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 a zero event rate, which is a fixed point of an SEM procedure. In principle this could 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 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.

2. Methods

2.1. Particle filters

Particle filters methods, also known as Sequential Monte Carlo (SMC) [14], 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 $\{(x^{(i)}, w^{(i)})\}_{i=1,\dots,N}$, approximate a distribution with density $p(x)$, such that

$$E_p[f(X)] = \int f(x)p(x)\mathrm{d}x \approx \frac{1}{\sum_{i=1}^N w^{(i)}} \sum_{i=1}^N w^{(i)} f(x^{(i)}) \tag{2.1}$$

for any bounded continuous function f, where $X \sim p(x)\mathrm{d}x$. We use "approximate" and $\approx$ to mean that $X_n \sim \sum_i w^{(i)}\delta_{x^{(i)}}(x)$ converges in distribution to $X \sim p(x)\mathrm{d}x$, so that equality holds in (2.1) as $N \to \infty$. Under some conditions, we can use IS to obtain particles approximating another distribution $q(x)\mathrm{d}x$:

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

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 $q(x)\mathrm{d}x$. 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 (both taken to be discrete in this section) have the form

$$P(X = x_{1\cdot\cdot s}) = p(x_1)p(x_2|x_1)\cdots p(x_s|x_{s-1}) \tag{2.2}$$

$$P(Y = y_{1\cdot\cdot s}|X = x_{1\cdot\cdot s}) = g(y_1|x_1)\cdots g(y_s|x_s) \tag{2.3}$$

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 are vectors. Let $\{(x^{(i)}, w^{(i)})\}$ be particles approximating the posterior $P(X_{1\cdot\cdot s} = x_{1\cdot\cdot s}|Y_{1\cdot\cdot s} = y_{1\cdot\cdot s})$, which for brevity we write as $P(x_{1\cdot\cdot s}|y_{1\cdot\cdot s})$. If $\widetilde{x}^{(i)}$ is the vector obtained by extending $x^{(i)}$ with a sample from $P(x_{s+1}|{x^{(i)}}_s)$, then from (2.2) and (2.3) it follows that $\{(\widetilde{x}^{(i)}, w^{(i)})\}$ 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

$$\widetilde{w}^{(i)} = w^{(i)}g(y_{s+1}|\widetilde{x}^{(i)}_{s+1}) \tag{2.4}$$

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

$$\widetilde{w}^{(i)} = w^{(i)}g(y_{s+1}|\widetilde{x}^{(i)}_{s+1})\frac{p(\widetilde{x}^{(i)}_{s+1}|{x^{(i)}}_s)}{q(\widetilde{x}^{(i)}_{s+1}|{x^{(i)}}_s)}. \tag{2.5}$$

For now we will choose q to be independent of y. Because samples from q do not follow the desired posterior $p(x|y)$, the fraction of particles close to the posterior's mode diminishes exponentially at each iteration until (2.1) fails altogether. To address this, samples are drawn from the approximating distribution itself, assigning each resampled particle weight $1/N$ – interestingly, this is the same process that is used in the Wright-Fisher model with selection to describe how fitness differences shape an evolving population [30]. This procedure tends to remove particles that have drifted from the mode of the posterior and have low weight, and duplicates particles with large weights, while preserving the validity of (2.1). Although resampling substantially decreases the future variance of (2.1), it increases the variance at the present iteration. To avoid increasing this variance unnecessarily, resampling is performed only when the estimated sample size $ESS = (\sum w^{(i)})^2/\sum(w^{(i)})^2$ drops below a threshold, often $N/2$. In addition, we use systematic resampling to minimize the variance introduced when resampling is performed [7]. This leads to Algorithm 1 [23].

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 [14] in calling it a particle filter algorithm instead of a smoothing algorithm (although our use of fixed-lag distributions for parameter estimation, Section 3.4, is a partial smoothing operation). The marginal likelihood can be estimated (although with high variance, see [4]) by setting the weights to $N^{-1}\sum_i {w^{(i)}}_s$ rather than N^{-1} when particles are resampled. This makes the weights asymptotically normalized, so that (2.1) becomes $E_{p(X,Y=y)}[f] \approx \sum_i w^{(i)}f(x^{(i)})$, and $P(Y = y) = \int p(x, y)\mathrm{d}x = E_{p(X,Y=y)}[1] \approx \sum_i {w^{(i)}}_L$.

Algorithm 1 Particle filter

Input: $y_{1..L}$
Output: Particles $\{({x^{(i)}}_{1..L}, {w^{(i)}}_L)\}$ approximating $p(x_{1..L}|y_{1..L})$
  ${w^{(i)}}_0 \leftarrow 1/N$, ${x^{(i)}}_0 \leftarrow \emptyset$ $(i = 1, \ldots, N)$
  For s from 0 to $L-1$:
    Loop invariant: $\{({x^{(i)}}_{1..s}, {w^{(i)}}_s)\} \sim p(x_{1..s}|y_{1..s})$
    If $ESS < N/2$:
      Resample, with replacement, $\{{x^{(i)}}_{1..s}\}$ proportional to $\{{w^{(i)}}_s\}$
      ${w^{(i)}}_s \leftarrow N^{-1}$ $(i = 1, \ldots, N)$
    For i from 1 to N:
      Sample ${x^{(i)}}_{s+1} \sim q(x_{s+1}|{x^{(i)}}_s)$
      ${w^{(i)}}_{s+1} \leftarrow {w^{(i)}}_s \frac{p({x^{(i)}}_{s+1}|{x^{(i)}}_s)}{q({x^{(i)}}_{s+1}|{x^{(i)}}_s)} g(y_{s+1}|{x^{(i)}}_{s+1})$

2.2. Continuous-time and -space Markov jump processes

We now consider Markov jump processes that have as realisations piecewise constant functions $x : [1, L) \mapsto \mathbb{T}$ where $\mathbb{T}$ is the state space of the Markov process. We consider the case that $\mathbb{T}$ is uncountable, e.g. $\mathbb{T} = \mathbb{R}^n$ (our main interest is in the case $\mathbb{T} = \mathbb{R}^{n-1} \times T_n$ of genealogical trees over n sequences; T_n is the set of rooted tree topologies over n tips). Let $(\mathcal{X}, \mathcal{F}_x, \pi_x)$ be a probability space with $\mathcal{X} = \mathbb{T}^{[1,L)}$ the space of possible realisations of the stochastic process $X = \{X_s\}_{s\in[1,L)}$, $\mathcal{F}_x \subset \mathcal{P}(\mathcal{X})$ the σ-algebra of events, and $\pi_x(X)$ the probability measure on $\mathcal{X}$ induced by the stochastic process X. The definition of a Markov model for uncountable phase space $\mathbb{T}$ involves the concept of conditional distributions; see the Appendix for details. For the hidden Markov model, assume X is not observed, and that observations Y are generated by a point process whose intensity at time s depends on X_s [a Cox process, see e.g. 34]. We define the space of observations $\mathcal{Y} = \mathcal{P}_{<\mathbb{N}}([1, L) \times M)$ of finite subsets of $[1, L) \times M$, where M is a discrete set of potential events that may occur at some $s \in [1, L)$. For an element $y = ((\tilde{s}_1, m_1), \ldots, (\tilde{s}_k, m_k)) \in \mathcal{Y}$, let $|y| := k$ be the number of events in the full observation, and let $\lambda(y)$ be the Lebesgue measure $(\mathrm{d}s)^{|y|}$, so that the "emission" distribution $\pi(Y|X = x)$ has a density $h(y|x)$ relative to λ; for Cox processes this density factorizes as $h(y|x) = \prod_{i=1}^{|y|} h(\tilde{s}_i, m_i|x_{\tilde{s}_i})$. The probability space for the joint process is $(\mathcal{X} \times \mathcal{Y}, \mathcal{F}, \pi)$, and the posterior distribution of interest is π conditioned on an observation $y \in \mathcal{Y}$, written as $\pi(X|Y = y)$.

To describe the Markov jump process version of algorithm 1 we introduce some more notation. As above π_x denotes the prior distribution of the latent variable X, 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 $[a, b) \subseteq [1, L)$; similarly $y_{a:b} := y \cap ([a, b) \times M)$ and $X_{a:b} := \{X_s\}_{s\in[a,b)}$. 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 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 $1 = s_0 < s_1 < \ldots < s_K = L$.
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 $K - 1$:
  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 $\{w^{(i)}{}_{s_j}\}$
    $w^{(i)}{}_{s_j} \leftarrow N^{-1}$ $(i = 1, \ldots, N)$
  For i from 1 to N:
    Sample $x^{(i)}{}_{s_j:s_{j+1}} \sim \xi_x(X_{s_j:s_{j+1}}|X_{s_j} = x^{(i)}{}_{s_j})$
    $w^{(i)}{}_{s_{j+1}} \leftarrow w^{(i)}{}_{s_j} \dfrac{\mathrm{d}\pi_x}{\mathrm{d}\xi_x}(x^{(i)}{}_{s_j:s_{j+1}}|X_{s_j} = x^{(i)}{}_{s_j}) \dfrac{\mathrm{d}\pi}{\mathrm{d}\lambda}(y_{s_j:s_{j+1}}|X_{s_j:s_{j+1}} = x^{(i)}{}_{s_j:s_{j+1}})$

The choice of waypoints $s_1, \ldots, s_K$ is discussed in section 2.3 below; in particular they need not coincide with the event times $\tilde{s}_1, \ldots, \tilde{s}_{|y|}$ in the observation y. 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 loop invariant is trivially true 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 $N^{-1}\sum_i w^{(i)}{}_{s_j}$; then the likelihood density w.r.t. $\lambda(\mathrm{d}y) = (\mathrm{d}s)^{|y|}$ is approximated by $\sum_i w^{(i)}{}_L$.

Note that by the nature of Markov jump processes, particles that start from identical states 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.

The absence of events along a time interval is also an "observation" and is assigned a likelihood density. In practice some intervals may not be observable for technical reasons. We assume that the observability process is independent of the Markov jump process, so that the likelihood density $\mathrm{d}\pi/\mathrm{d}\lambda$ can be set to 1 across these intervals.

2.3. 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 affect where the algorithms consider whether or not to resample. A high density of waypoints is therefore always acceptable, but a low density may result in particle degeneration. 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. Here we derive a criterion on the waypoints that limits this effect.

Let $R(s) = f(X_s)$ be the stochastic variable that measures the instantaneous rate of occurrence of 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 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 $\langle V(X)\rangle := \int V(X)\mathrm{d}\pi(X)$ for the expectation of V over $\pi(X)$. Writing $R(s) = \mu + r(s)$ where $\mu = \langle R(s)\rangle$ is the mean event rate (which is independent of s by assumption), then

$$\langle W(L)\rangle = \langle W_0 e^{-\int_0^L \mu + r(s)\mathrm{d}s}\rangle = \langle W_0\rangle e^{-\mu L}\langle \prod_{i=1}^{k}(1 - r(s_i)\Delta s)\rangle \tag{2.6}$$

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

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

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 $r(t)$, 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 $\sigma^2 := K(s,s)$ 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

$$\langle W\rangle = \langle W_0\rangle e^{-\mu L + \frac{1}{2}C} \ge \langle W_0\rangle e^{-\mu L}, \tag{2.7}$$

$$\langle W^2\rangle = \langle W_0^2\rangle e^{-2\mu L + C} \le \langle W_0^2\rangle e^{-2\mu L + \sigma^2 L^2}, \tag{2.8}$$

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

$$ESS = \frac{\left(\sum_{i=1}^{N} w^{(i)}\right)^2}{\sum_{i=1}^{N}(w^{(i)})^2} \approx \frac{N\langle W\rangle^2}{N^2\langle W^2\rangle} \ge \frac{\langle W_0\rangle^2 e^{-2\mu L}}{N\langle W_0^2\rangle e^{-2\mu L + \sigma^2 L^2}} = ESS_0 e^{-\sigma^2 L^2} \tag{2.9}$$

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

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. Therefore the minimum waypoint density derived here serves as a guide; breakdown of the assumptions made here can be monitored by tracking the ESS, increasing the density of waypoints if necessary.

2.4. 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 $\pi(X_{1:s_j}|y_{1:s_j})$ is independent of data beyond s_j. However, not all particles are equally important for approximating subsequent posteriors. Emphasising particles that will be relevant in future at the expense of those relevant only to $\pi(X_{1:s_j}|y_{1:s_j})$ would improve 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 [47] implements this intuition for discrete-time models $p(X_{1:n}, Y_{1:n})$ by targeting a special resampling distribution [33]. 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 [14], 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 not used to define a distribution over y, but to quantify the plausibility of a current state ${x^{(i)}}_s$. 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 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 $\widetilde{\pi}^s(x_{1:s}, y_{1:L}) = \pi^{1:s}(x_{1:s}, y_{1:s})h^s(y_{s:L}|x_s)\lambda^{s:L}(y_{s:L})$ on $\mathcal{X}_{1:s} \times \mathcal{Y}$. The 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 and one set of samples, together forming 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.

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 $\{(x^{(i)}{}_{1:L}, w^{(i)}{}_L)\}$ approximating $\pi(X|Y_{1:L} = y_{1:L})$

$w^{(i)}{}_1 \leftarrow 1/N$, $v_1^{(i)} \leftarrow 1/N$, $x^{(i)}{}_{1:1} \leftarrow \emptyset$ $(i = 1, \ldots, N)$
For j from 0 to $K - 1$:
  Loop invariant: $\{(x^{(i)}{}_{1:s_j}, w^{(i)}{}_{s_j})\} \approx \pi^{1:s_j}(X_{1:s_j}|Y_{1:s_j} = y_{1:s_j})$
  Loop invariant: $\{(x^{(i)}{}_{1:s_j}, v_{s_j}^{(i)})\} \approx \widetilde{\pi}^{s_j}(X_{1:s_j}|Y = y_{1:L})$
  If $ESS(\{v_{s_j}^{(i)}\}) < N/2$:
    Resample $\{x^{(i)}{}_{1:s_j}\}$ with probabilities proportional to $\{v_{s_j}^{(i)}\}$
    $w^{(i)}{}_{s_j} \leftarrow N^{-1} w^{(i)}{}_{s_j}/v_{s_j}^{(i)}$ $(i = 1, \ldots, N)$
    $v_{s_j}^{(i)} \leftarrow N^{-1}$ $(i = 1, \ldots, N)$
  For i from 1 to N:
    Sample $x^{(i)}{}_{s_j:s_{j+1}} \sim \xi_x(X_{s_j:s_{j+1}}|X_{s_j} = x^{(i)}{}_{s_j})$
    $w^{(i)}{}_{s_{j+1}} \leftarrow w^{(i)}{}_{s_j} \dfrac{\mathrm{d}\pi_x}{\mathrm{d}\xi_x}(x^{(i)}{}_{s_j:s_{j+1}}|X_{s_j} = x^{(i)}{}_{s_j}) \dfrac{\mathrm{d}\pi}{\mathrm{d}\lambda}(y_{s_j:s_{j+1}}|X_{s_j:s_{j+1}} = x^{(i)}{}_{s_j:s_{j+1}})$
    $v_{s_{j+1}}^{(i)} \leftarrow v_{s_j}^{(i)} \dfrac{\mathrm{d}\pi_x}{\mathrm{d}\xi_x}(x^{(i)}{}_{s_j:s_{j+1}}|X_{s_j} = x^{(i)}{}_{s_j}) \dfrac{\mathrm{d}\widetilde{\pi}^{s_{j+1}}}{\mathrm{d}\widetilde{\pi}^{s_j}}(y_{s_j:L}|X_{s_j:s_{j+1}} = x^{(i)}{}_{s_j:s_{j+1}})$

2.5. Parameter inference

Parameters can be inferred by stochastic expectation maximization (SEM), which involves maximizing the expected log likelihood over the posterior distribution of latent variables. Note that the expected log likelihood for a Poisson process with rate θ is $c \log \theta - q\theta$ where c and q are the expected event count and opportunity, which is maximized for $\theta = c/q$. We consider Markov jump processes X_s parameterized by a vector of parameters θ, for which the distribution $\pi_x(\mathrm{d}x|\theta)$ can be written as

$$\pi_x(x|\theta)\mathrm{d}x = \prod_i \exp\{-\theta_i Q_i(x)\}\theta_i^{|x|_i}\mathrm{d}x, \tag{2.10}$$

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

$$\theta'_{i,\mathsf{EM}} = \frac{\mathbb{E}_{\pi(x|y,\theta)}[\ |x|_i\ ]}{\mathbb{E}_{\pi(x|y,\theta)}[\ Q_i(x)\ ]} \tag{2.11}$$

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

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 θ. 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 θ, and an appropriate prior over θ 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

$$\theta'_{i,\mathsf{VB}} \sim \Gamma(\alpha_0 + \mathbb{E}[\ |x|_i\ ], \beta_0 + \mathbb{E}[\ Q_i(x)\ ]) \tag{2.12}$$

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 θ. It would appear that obtaining samples from this distribution is intractable. However, if $\pi(\theta)$ is a Gamma distribution, θ 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; here ψ is the digamma function, see the Appendix for details. These scaling 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 generate samples from this distribution at no additional computational cost.

3. Results

3.1. The sequential coalescent with recombination model

The coalescent-with-recombination (CwR) process, and the graph structure that results from it, was first described by [32], and was given an elegant mathematical description by [24], 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 [64], 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) developed by [56] 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 [40, 38]. 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 [56].

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 realisation x can be written as $\pi_x(x|\rho, C)\mathrm{d}x =$

$$\exp\left\{-\int B(x_s)\rho(s)\mathrm{d}s\right\}\left[\prod_{j=1}^{|x|}\exp\left\{-\int_{\nu_j}^{\tau_j} b_u(x_{s_j})C(u)\mathrm{d}u\right\}\rho(s_j)C(\tau_j)\right](\mathrm{d}s)^{|x|}(\mathrm{d}u)^{2|x|}. \tag{3.1}$$

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, so that $\rho(s)B(x_s)$ is the exit rate of the Markov process in state x_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. See Section 5.3 for more details.

Mutations follow a Poisson process whose rate at s depends on the latent variable 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 $A_j \in \{0, 1\}^S$ represents the distribution of wildtype (0) and alternative (1) alleles observed in the S samples. The conditional probability measure of the observations y given a realisation x is

$$\pi(y|X = x)\mathrm{d}y = \exp\left\{-\int B(x_s)\mu(s)\mathrm{d}s\right\}\left[\prod_{j=1}^{|y|} P(A_j|x_{s_j}, \mu(s_j))\right](\mathrm{d}s)^{|y|} \tag{3.2}$$

where $P(A|x_s, \mu)$ is the probability of observing the allelic pattern A given a genealogy x_s and a mutation rate μ per nucleotide and per generation; this probably is calculated using Felsenstein's peeling algorithm [19].

3.2. Choice of waypoints

To compute the minimum distance for waypoints, we need to compute the variance of the mutation rate before conditioning on data. For a complex model this is most straightforwardly estimated by simulation. Alternatively, assuming 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 $(4N_e)^2 \sum_{i=1}^{n-1} i^{-2}$ [58]. The variance of mutation rates σ^2 is obtained by multiplying this by μ^2, since the rate of mutations on the coalescent tree is μ 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 $1/\sqrt{2\sigma^2} = \sqrt{3}/\pi\theta \approx 1/2\theta$.

3.3. The 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 digest consists of the distance s_i to the nearest future singleton for each haploid sequence, and the $\leq n/2$ mutually consistent cherries (i.e. two-tipped nodes that can coexist in the same tree) $c_k = (a_k, b_k)$, together with positions $s'_k \leq s''_k$ where their first and last supporting doubleton were observed. Under some simplifying assumptions we derive an approximation of the likelihood of the current genealogy given these data; see Section 5.5 for details.

3.4. Parameter inference

From (3.1) and Section 2.5, the parameters $\theta' = (\rho_{\mathsf{EM}}, C_{\mathsf{EM}})$ that maximize the expected log of $\pi_x(x|\theta')$, where the expectation is taken over $x \sim \pi(x|y,\theta)\mathrm{d}x$ as approximated by algorithm 3, is

$$\rho'_{\mathsf{EM}} = \frac{\mathbb{E}[\ |x|\ ]}{\mathbb{E}[\int B(x_s)\mathrm{d}s]} \quad \text{and} \quad C'_{\mathsf{EM}} = \frac{\mathbb{E}[\ |x|\ ]}{\mathbb{E}[\sum_{j=1}^{|x|} \int_{\nu_j}^{\tau_j} b_u(x_{s_j})]}, \tag{3.3}$$

where $\theta = (\rho, C)$ is the vector of current parameter estimates. Note that C'_{EM} in (3.3) is constant in time. In practice we maximize (3.1) with respect to piecewise constant functions $C'_{\mathsf{EM}}(t)$, which yields

$$C'_{\mathsf{EM}}(t) = \frac{\mathbb{E}[\ |x|_{\nu,\tau}\ ]}{\mathbb{E}[\sum_{j=1}^{|x|} \int_{u\in[\nu_j,\tau_j)\cap[\nu,\tau)} b_u(x_{s_j})\mathrm{d}u]} \tag{3.4}$$

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

uses

$$\rho'_{\mathsf{VB}} \sim \Gamma(\alpha_\rho + \mathbb{E}[\ |x|\ ], \beta_\rho + \mathbb{E}[\textstyle\int B(x_s)\mathrm{d}s]) \tag{3.5}$$

$$C'_{\mathsf{VB}} \sim \Gamma(\alpha_C + \mathbb{E}[\ |x|\ ], \beta_C + \mathbb{E}[\textstyle\sum_{j=1}^{|x|} \int_{\nu_j}^{\tau_j} b_u(x_{s_j})]) \tag{3.6}$$

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 (3.5-3.6) of the previous iteration, and α_ρ, β_ρ, α_C, β_C parameterize the prior distributions $\rho \sim \Gamma(\alpha_\rho, \beta_\rho)$ and $C \sim \Gamma(\alpha_C, \beta_C)$.

To estimate the expectations appearing in (3.3-3.6) 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 [5] are difficult to apply here since finite-time transition probabilities are intractable. For discrete-time models fixed-lag smoothing is often used [45]. 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 $\alpha = 1$.

3.5. Simulation study

We implemented the model and algorithm above in a 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 ($N_e = 10,000$, $\mu = 2.5 \times 10^{-8}$ / year, $\rho = 10^{-8}$ / year, generation time $g = 30$ years). We inferred population sizes $N_e = 1/(2C)$ through time as a piecewise constant function across 9 epochs (with boundaries at 400, 800, 1200, $2k$, $4k$, $8k$, $20k$, $40k$ and $60k$ generations) using particle filters Alg. 2 and Alg. 3, as well as a recombination rate, which was taken to be constant across time. Recombination rates were inferred accurately (data not shown) and throughout we focus on the accuracy of N_e through 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. 1a).

On phased data (Fig. 1**a**, 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. 1**a**, bottom rows), the default particle filter continues to work reasonably well; in fact the bias appears somewhat reduced compared to phased data analyses,

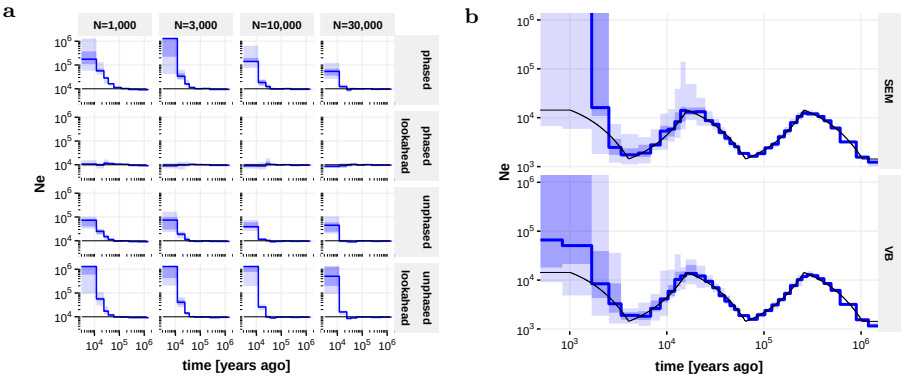

FIG 1. *True population sizes (black), and median inferred population sizes across 15 independent runs (blue); shaded areas denote quartiles and full extent.* **a** *Impact of lookahead, phasing and number of particles on the bias in population size estimates for recent epochs, for data simulated under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

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 unphased data, making the lookahead procedure less effective, and even counterproductive for early epochs. Recombination rate inferences (assumed constant in time) were accurate in all experiments (data not shown).

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 [51] and Terhorst, Kamm and Song [60], and inferred N_e across 37 approximately exponentially spaced epochs; see Section 5.6 for 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 (data not shown; but see Fig. 2b). Inferences in the most recent epochs remain biased, which however is mitigated considerably by the Variational Bayes approach compared to SEM (Fig. 1b).

3.6. Inference on human subpopulations

We applied `SMCSMC` 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` [51] on the same data, and on subsets of 2 and 1 diploid samples. Inferences show good agreement where

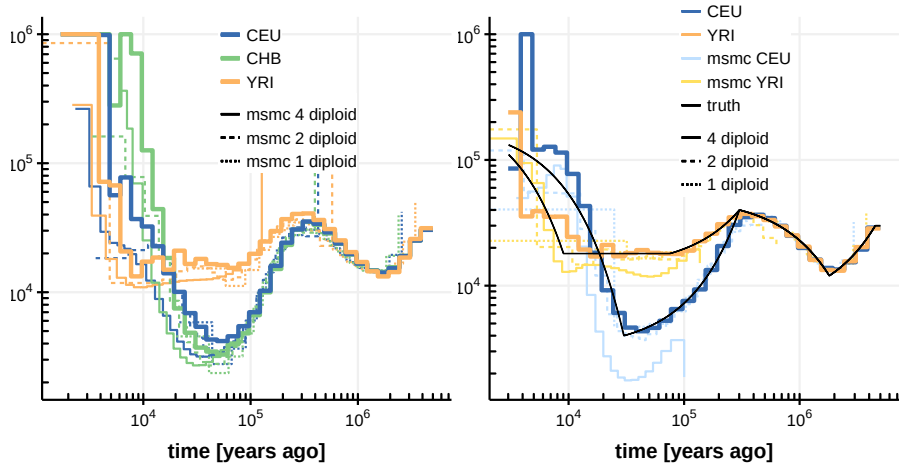

FIG 2. *Population size inferences by* SMCSMC *on four diploid samples. Left, three human populations (CEU, CHB, YRI), together with inferences from* msmc *using 1, 2 and 4 diploid samples. Right, simulated populations resembling CEU and YRI population histories. All inferences (*SMCSMC *,* msmc*) were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

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 5.6), and 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.

4. Discussion

Motivated by the problem of recovering a population's demographic history from the genomes of a sample of its individuals [52], we have introduced a continuous-sequence approximation of the CwR model, and developed a particle filter algorithm for continuous-time Markov jump processes with a continuous phase space, by restricting the doubly-continuous process to 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 [47] 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 polymorphism 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. 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 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 [35, 53, 51, 57, 60], 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., 29] 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, Kamm and Song [60] 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 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 [56], 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 [27, 62], and background selection profoundly shapes the heterozygosity in the human genome [41]. 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 data 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. A further advantage of a particle filter is that it provides a sample from the posterior distribution of ancestral recombination graphs

(ARGs). This allows estimating 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 [49], 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 [35], 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.

References

[1] Arulampalam, M. S., Maskell, S., Gordon, N. and Clapp, T. (2002). A Tutorial on Particle Filters for Online Nonlinear/Non-Gaussian Bayesian Tracking. *IEEE Trans. Signal Processing* **50** 174-188.

[2] Beaumont, M. (1999). Detecting Population Expansion and Decline Using Microsatellites. *Genetics* **153** 2013-2029.

[3] Beaumont, M., Zhang, W. and Balding, D. J. (2002). Approximate Bayesian Computation in Population Genetics. *Genetics* **162** 2025-2035.

[4] Bérard, J., Del Moral, P. and Doucet, A. (2014). A lognormal central limit theorem for particle approximations of normalizing constants. *Electron. J. Probab.* **19** 1-28.

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

[6] 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.

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

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

[9] Del Moral, P. (2016). *Mean Field Simulation for Monte Carlo Integration*. Chapman and Hall/CRC.

[10] 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.

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

[12] Doucet, A., Godsill, S. and Andrieu, C. (2000). On sequential Monte Carlo sampling methods for Bayesian filtering. *Statistics and computing* **10** 197–208.

[13] Doucet, A., Gordon, N. J. and Krishnamurthy, V. (2001). Particle Filters for State Estimation of Jump Markov Linear Systems. *IEEE Transactions on Signal Processing* **49** 613–624.

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

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

[16] 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.

[17] Fearnhead, P., Papaspiliopoulos, O. and Roberts, G. O. (2008). Particle Filters for Partially Observed Diffusions. *Journal of the Royal Statistical Society: Series B* **70** 755–777.

[18] Feller, W. (1940). On the Integro-Differential Equations of Purely Discontinuous Markoff Processes. *Transactions of the American Mathematical Society* **48** 488-515.

[19] Felsenstein, J. (1981). Evolutionary trees from DNA sequences: A maximum likelihood approach. *Journal of Molecular Evolution* **17** 368-376.

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

[21] Fourment, M., Claywell, B. C., McCoy, C., Matsen Iv, F. A. and Darling, A. E. (2018). Effective Online Bayesian Phylogenetics via Sequential Monte Carlo with Guided Proposals. *Systematic Biology*.

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

[23] Gordon, N. J., Salmond, D. J. and Smith, A. F. M. (1993). Novel approach to nonlinear/non-Gaussian Bayesian state estimation. *IEEE Proceedings F, Radar and Signal Processing* **140** 107-113.

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

[25] 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.

[26] 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.

[27] 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.

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

[29] 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.

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

[31] 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.

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

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

[34] 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* **54** 496-506.

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

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

[37] Mackay, D. (1997). Ensemble learning for hidden Markov models.

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

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

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

[41] McVicker, G., Gordon, D. and Davis, P. Colleen Green (2009). Widespread Genomic Signatures of Natural Selection in Hominid Evolution. *PLoS Genet* **5** e1000471.

[42] Ng, B., Pfeffer, A. and Dearden, R. (2005). Continuous Time Particle Filtering. In *Proceedings of the IJCAI* 1360-1365.

[43] Nielsen, S. F. (2000). The stochastic EM algorithm: estimation and asymptotic results. *Bernoulli* **6** 457-489.

[44] Nodelman, U., Shelton, C. R. and Koller, D. (2002). Continuous Time Bayesian Networks. In *Proceedings of the UAI*.

[45] Olsson, J., Capp'e, O., Douc, R. and Moulines, É. (2008). Sequential Monte Carlo smoothing with application to parameter estimation in nonlinear state space models. *Bernoulli* **14** 155-179.

[46] 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.

[47] Pitt, M. K. and Shephard, N. (1999). Filtering via Simulation: Auxiliary Particle Filters. *Journal of the American Statistical Association* **94** 590-599.

[48] Pritchard, J. K., Stephens, M. and Donnelly, P. (2000). Inference of Population Structure Using Multilocus Genotype Data. *Genetics* **155** 945–959.

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

[50] Rosenbluth, M. N. and Rosenbluth, A. W. (1955). Monte Carlo Calculation of the Average Extension of Molecular Chains. *J. Chem. Phys.* **23** 356-359.

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

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

[53] 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.

[54] Sherlock, C., Golightly, A. and Gillespie, C. S. (2014). Bayesian Inference for Hybrid Discrete-Continuous Systems Biology Models. *Inverse Problems* **30** 114005.

[55] Smith, R. A., Ionides, E. L. and King, A. A. (2017). Infectious Disease Dynamics Inferred from Genetic Data via Sequential Monte Carlo. *Molecular Biology and Evolution*.

[56] 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.

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

[58] Tavaré, S. (1984). Line-of-descent and genealogical processes, and their application in population genetic models. *Theoretical Population Biology* **26** 119-164.

[59] Taylor, S., Ridall, G., Sherlock, C. and Fearnhead, P. (2014). Particle learning approach to Bayesian model selecion: An application from neurology. *Springer Proceedings in Mathematics and Statistics* **63** 165-167.

[60] 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.
[61] Wang, L., Wang, S. and Bouchard-Côté, A. (2019). An Annealed Sequential Monte Carlo Method for Bayesian Phylogenetics. *Systematic Biology.*
[62] Whelan, S. and Goldman, N. (2004). Estimating the Frequency of Events That Cause Multiple-Nucleotide Changes. *Genetics* **167** 2027-2043.
[63] 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.
[64] Wiuf, C. and Hein, J. (1999). Recombination as a Point Process along Sequences. *Theoretical Population Biology* **55** 248-259.

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 [8], 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:

$$E_\pi[f(x_t)|x_{s_1} = \tau_1, \ldots, x_{s_k} = \tau_k] = E_\pi[f(x_t)|x_{s_k} = \tau_k] \tag{5.1}$$

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 loop invariant is trivially true, 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 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 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 $\{w_{s_{j+1}}^{(i)}, X_{1:s_{j+1}}^{(i)}\}$ 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 $(v_{s_{j+1}}^{(i)}, X_{1:s_{j+1}}^{(i)}) \approx \widetilde{\pi}^{s_j}(x_{1:s_{j+1}}|Y = y_{1:L})$ is analogous. This proves the loop invariant for $j+1$, and the algorithm.

5.3. The sequential coalescent with recombination process

To make formula (3.1) 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 (3.1) 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 (3.1) 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 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.

5.4. 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 $\mathrm{d}x\mathrm{d}y$ admits a probability density of the form

$$\pi_{xy}(x, y|\theta)\mathrm{d}x\mathrm{d}y = \pi_y(y|x) \prod_i \exp\left\{-\theta_i B_i(x)\right\} \theta_i^{|x|_i} \mathrm{d}x\mathrm{d}y. \tag{5.2}$$

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

$$B_U^R(x) = \int_s \int_{u \in U} b_u(x_s)\mathrm{d}u\mathrm{d}s; \qquad B_U^C(x) = \sum_{j=1}^{|x|} \int_{u \in [\nu_j, \tau_j] \cap U} b_u(x_{s_j})\mathrm{d}u, \tag{5.3}$$

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 (here the constant of proportionality implied by "$\propto$" hides a constant density $\lambda(y)$). Following Hinton and van Camp [31] and Mackay [37], 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 [20],

$$F(\phi) = -\int_x \int_\theta \phi(x, \theta) \log\left[\frac{\pi(x, \theta|y)}{\phi(x, \theta)}\right] \mathrm{d}\theta\mathrm{d}x. \tag{5.4}$$

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

$$F(\phi) = -\int \phi_x(x)\phi_\theta(\theta) \left\{\sum_i |x|_i \log\theta_i - B_i(x)\theta_i + \log\pi_\theta(\theta) - \log\phi_\theta(\theta)\right\} \mathrm{d}\theta\mathrm{d}x + const \tag{5.5}$$

$$= \int \phi_\theta(\theta) \log \frac{\phi_\theta(\theta)}{\pi_\theta(\theta) \prod_i \theta_i^{\mathbb{E}_{\phi_x}[|x|_i]} \exp\{-\mathbb{E}_{\phi_x}[B_i(x)]\theta_i\}} \mathrm{d}\theta + const \tag{5.6}$$

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

$$\phi_\theta(\theta) = \prod_i \Gamma(\alpha_i', \beta_i') \tag{5.7}$$

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,

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

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, $F(\phi)$ is minimized if the numerator and denominator are equal, which happens for

$$\phi_x(x) \propto \pi_y(y|x)\prod_i \exp\{-\bar{\theta}_i B_i(x)\}(\theta_i^*)^{|x|_i} \propto \pi(x|y,\bar{\theta})\prod_i\left(\frac{\theta_i^*}{\bar{\theta}_i}\right)^{|x|_i} = \pi(x|y,\bar{\theta})\prod_i \delta_i^{|x|_i} \tag{5.8}$$

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

5.5. Approximate lookahead likelihood

Let s_i denote the distance to the nearest future singleton in each sequence, and let $c_k = (a_k, b_k)$ be $\leq n/2$ mutually consistent cherries with positions $s_k' \leq s_k''$ of their first and last supporting doubleton. To simplify notation we assume that the current position is 0 (Fig. 3a).

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. 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 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

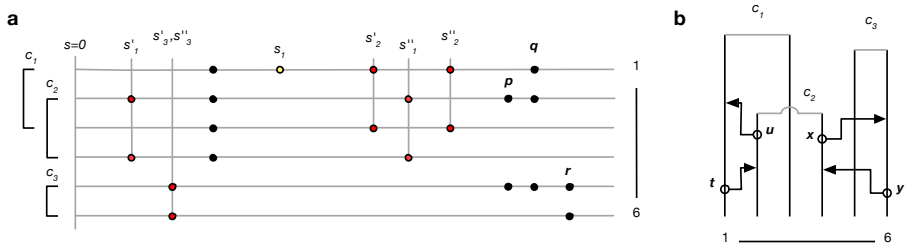

FIG 3. **a.** *Example of data digest. Lines represent genomes of* 6 *lineages, circles observed genetic variants. Of the data shown, one singleton (yellow) and five doubletons (red) contribute to the digest. Cherry* c_3 *is supported by a single doubleton;* r *does not contribute because the mutation 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 3 (x,y).*

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

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

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 μ_i and μ_i' in the likelihood, so that

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

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 (Fig. 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 $n \geq 3$ [39], 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

$$p(a_k, b_k, s_k', s_k'' | (a_k, b_k; l) \in \tau) = e^{-\rho_C s_k''} + \frac{2}{3(n-1)}(1 - e^{-\rho_C s_k''}), \tag{5.9}$$

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

$$p(a_k, b_k, s'_k, s''_k | \bar{l}, (a_k, b_k) \notin \tau) = \frac{2}{3(n-1)}(1 - e^{-\rho'_C s'_k}), \tag{5.10}$$

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 (5.9-5.10) are likelihoods for τ supporting c_k at the given position, 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 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.

5.6. 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 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 relatively slow evaluations of the exponential function, often for small exponents. We use the continued-fraction

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

approximation $e^x \approx 1 + 2x/(2 - x + \frac{1}{6}x^2)$ for $|x| < 0.03$, with relative error bounded by 10^{-10} [36].

Table 1 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 [35]). 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`).