Asymptotic analysis of genealogies induced by sequential Monte Carlo algorithms

Suzie Brown

October 4, 2019

1 Introduction

Monte Carlo methods are a class of algorithms which use random samples to approximate integrals. There are many different subclasses of which the most famous is probably Markov chain Monte Carlo (MCMC). These methods have proved enormously useful, particularly in higher-dimension problems where traditional numerical integration techniques do not perform well.

Another class of Monte Carlo methods, and the focus of this work, is sequential Monte Carlo (SMC). Since its introduction to the literature by Gordon et al. (1993), SMC has found a huge range of applications and has become an indispensable tool to practitioners from many fields. It is particularly useful in applications such as target tracking, where the model has a natural sequential structure. However, the sequential nature of SMC is helpful not only in inherently sequential settings, but in any setting where the quantity to be inferred has highly correlated components. Other application areas include image processing, molecular simulation and economics.

In these cases the more traditional Markov chain Monte Carlo methods are essentially useless because the strong dependence structure causes them to mix extremely slowly. Conversely, SMC exploits this dependence structure to create recursive Monte Carlo algorithms that can be efficient in this difficult setting. However, the method is not without its problems, and it is one of these problems that we will examine in this work.

Sections 2 and 3 explain the background material on SMC and coalescent theory respectively. In Section 4 we explain the contribution of this work in the context of the SMC literature. In Section 5 we present the first novel result of this work with an outline proof; the full proof is in the Appendix. Section 6 introduces the next focus of this work, for which the proofs are not yet complete. Future directions of the work, and its potential impact, are summarised in Section 7.

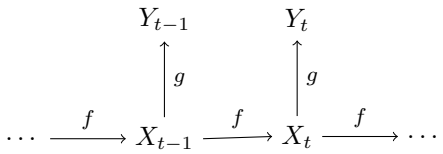
Throughout the document we will use the compact notation $X_{m:n}$ as shorthand for $X_m, X_{m+1}, \dots, X_n$, as well as $X_{-n} := X_0, \dots, X_{n-1}, X_{n+1}, \dots X_N$. We denote falling factorial powers $(x)_a := x(x-1)\dots(x-a+1)$, with the convention $(x)_0 = 1$.

1.1 State space models

For the purposes of presenting the algorithm, let us consider a model consisting of an unobservable time-homogeneous discrete-time Markov process $X_{0:T}$ and observables $Y_{0:T}$, satisfying the conditional independence structure

$$(X_{t+1:T} \perp\!\!\!\perp X_{0:t-1}) \mid X_t$$
$$(Y_t \perp\!\!\!\perp Y_{-t}, X_{-t}) \mid X_t$$

for all $t \in \{0, 1, \dots, T\}$, as represented by the graphical model below.



We assume for notational convenience that $x_0, \ldots, x_T$ take values in a common state space $\mathcal{X}$, and $y_0, \ldots, y_T$ in a common state space $\mathcal{Y}$, but these assumptions can be dropped. Such models are referred to as state space models or hidden Markov models; the terminology varies among authors. They provide a flexible framework for time series modelling, at the cost of being difficult to analyse.

In particular, consider the following model:

$$
\begin{aligned}
&X_0 \sim \mu(\cdot) \\
&X_{t+1} \mid (X_t = x_t) \sim f(\cdot|x_t) \qquad t = 0, \ldots, T-1 \\
&Y_t \mid (X_t = x_t) \sim g(\cdot|x_t) \qquad t = 0, \ldots, T
\end{aligned}
$$

We assume that the *transition* and *emission* kernels have densities with respect to a dominating measure dx which we denote by f and g respectively, but this is not necessary in general. All that is actually required is that we can sample from $\mu(\cdot)$ and $f(\cdot|x)$, and calculate *unnormalised* potentials $g(y|x)$, for all x, y.

As a concrete example, let us consider the application of target tracking. Suppose we are using radar to track the position of an aeroplane. The true trajectory of the aeroplane is unknown and is represented by $X_{0:T}$ (perhaps a sequence of positions in $\mathbb{R}^3$), with f encoding our model for how an aeroplane moves. What we observe is the output $Y_{0:T}$ of our radar equipment, which has some measurement uncertainty that is encoded in g.

1.2 Inference in state space models

Consider the Bayesian setting where μ is our prior distribution at time 0, observations $y_0, y_1, y_2 \ldots$, arrive sequentially, and we want to infer information about the hidden states $X_{0:T}$. The inference may be conducted on-line (sequentially as observations arrive) or off-line (retrospectively once all observations have been collected). The three main inference problems are:

Filtering (where is it now?) $p(x_t|y_{0:t})$

Prediction (where will it go next?) $p(x_{t+1}|y_{0:t})$

Smoothing (where has it been?) $p(x_{0:t}|y_{0:t})$

In the on-line setting, we take as our prior the posterior distribution from the previous time step $t-1$, and update it using the new observation y_t. The inference must be fast enough to keep up with the rate of arrival of observations, so in particular the complexity of the update must not increase with T. In the off-line setting, we take μ as the prior distribution, and infer the set of posteriors once all $T+1$ observations have arrived.

Prediction and filtering are essentially equivalent, because given a filtering distribution, the corresponding predictive distribution can be obtained by applying the transition kernel f. Smoothing is considered a harder task because it requires us to infer many more parameters from the same amount of information; the dimension of the problem increases linearly with T.

1.3 Deterministic solutions

In the case of linear Gaussian state space models, the posterior distributions of interest are also Gaussian, with mean and covariance available analytically by way of the Kalman filter (Kalman, 1960) and Rauch-Tung-Striebel (RTS) smoother recursions (Rauch et al., 1965). Recursions are also available for some other conjugate models: see for example Vidoni (1999). Another analytic case occurs if the state space $\mathcal{X}$ is finite, in which case any integrals become finite sums, and the forward-backward algorithm (Baum et al., 1970) yields the exact posteriors.

If the model is Gaussian but non-linear, the posterior filtering distributions can be estimated using the *extended Kalman filter* (see for example Jazwinski (2007)), which applies a first-order linearisation in order to make use of the Kalman filter. This method performs well on models that are "almost linear". The resulting predictor is only *optimal* when the model is actually linear, in which case the extended Kalman filter coincides with the Kalman filter.

For models that are highly non-linear or for which gradients are not readily available, a more suitable method is the *unscented Kalman filter* (Wan and Van Der Merwe, 2000). This involves taking a representative sample (which is chosen deterministically using the *unscented transformation*) to characterise the distribution at time t, and then propagating these points through the non-linear transition F to obtain a characterisation of the distribution at time $t+1$.

In more complex models such techniques are not feasible, and we are forced to resort to Monte Carlo methods. For state space models, Markov chain Monte Carlo methods are not very effective due to the high dimension of the parameter space. But we can exploit the sequential nature of the underlying dynamics to decompose the problem into a sequence of inferences of more manageable dimension. This is the motivation behind sequential Monte Carlo (SMC) methods.

2 Sequential Monte Carlo

References for this section are Doucet et al. (2001), Del Moral et al. (2006), and Doucet and Johansen (2011).

2.1 Particle approximation

The conditional independence structure in the model of Section 1.1 implies that the (joint) marginal distribution of the hidden states $X_{0:t}$ (i.e. the prior distribution at time t) is given by

$$p(x_{0:t}) = \mu(x_0) \prod_{i=1}^{t} f(x_i|x_{i-1})$$

and that the likelihood of the observations $y_{0:t}$ given the underlying states $x_{0:t}$ takes the form

$$p(y_{0:t}|x_{0:t}) = \prod_{i=0}^{t} g(y_i|x_i).$$

Using the conditional independence structure, we can write

$$p(x_{0:t}|y_{0:t}) \propto g(y_t|x_t) f(x_t|x_{t-1}) p(x_{0:t-1}|y_{0:t-1}) \tag{1}$$

$$\propto \mu(x_0) g(y_0|x_0) \prod_{i=1}^{t} f(x_i|x_{i-1}) g(y_i|x_i) \tag{2}$$

for $t = 0, \ldots, M$, where the one-step recursion (1) is obtained using Bayes rule, and (2) is obtained by applying (1) t times. The filtering distribution $p(x_t|y_{0:t})$ can be obtained from (1) by marginalising out $x_{0:t-1}$, which is straightforward if Monte Carlo samples are available. The predictive distributions can also be derived from the smoothing distributions using

$$p(x_{t+1}|y_{0:t}) \propto f(x_{t+1}|x_t) p(x_{0:t}|y_{0:t}).$$

SMC provides a particle method to approximate to (2), given a model specification and a sequence of observations. Like the underlying process, the algorithm proceeds sequentially, returning its approximation to the smoothing distribution at each time step. This approximation is the empirical distribution of the particles $X_{0:t}^{(1)}, \ldots, X_{0:t}^{(N)}$:

$$\hat{p}(x_{0:t}|y_{0:t}) = \frac{1}{N} \sum_{i=1}^{N} \delta_{X_{0:t}^{(i)}}(x_{0:t}) \tag{3}$$

The particle approximation is justified by various convergence results - see for example Del Moral (2013) for details.

A generic SMC algorithm is presented in Algorithm 1. For the state space model described above, we can take $K_{t+1}(x_t, \cdot) \equiv f(\cdot|x_t)$ and $g_{t+1}(x_t, x_{t+1}) \equiv g(y_{t+1}|x_{t+1})$. The form of the RESAMPLE function in Algorithm 1 is discussed in Section 6.

Figure 1 illustrates the particle approximation arising from such an algorithm on a linear Gaussian model, with the exact posterior for reference.

Algorithm 1 Standard SMC

Require: $N, T, \mu, \{K_t\}, \{g_t\}, y_{0:T}$

for $i \in \{1, \dots, N\}$ **do**
  Sample $X_0^{(i)} \sim \mu(\cdot)$ ▷ initialise
  $w_0^{(i)} \leftarrow g_0(X_0^{(i)})$
end for
$W \leftarrow \sum_{j=1}^{N} w_0^{(j)}$
for $i \in \{1, \dots, N\}$ **do**
  $w_0^{(i)} \leftarrow \frac{1}{W} w_0^{(i)}$ ▷ normalise weights
end for
for $t \in \{0, \dots, T-1\}$ **do**
  Sample $a_t^{(1:N)} \sim \text{RESAMPLE}(\{1, \dots, N\}, w_t^{(1:N)})$ ▷ resample particles
  for $i \in \{1, \dots, N\}$ **do**
    Sample $X_{t+1}^{(i)} \sim K_{t+1}(X_t^{(a_t^{(i)})}, \cdot)$ ▷ propagate particles
    $w_{t+1}^{(i)} \leftarrow g_{t+1}(X_t^{(a_t^{(i)})}, X_{t+1}^{(i)})$ ▷ calculate weights
  end for
  $W \leftarrow \sum_{j=1}^{N} w_{t+1}^{(j)}$
  for $i \in \{1, \dots, N\}$ **do**
    $w_{t+1}^{(i)} \leftarrow \frac{1}{W} w_{t+1}^{(i)}$ ▷ normalise weights
  end for
end for

If only the latest filtering distribution $p(x_t|y_{0:t})$ is required, we can marginalise out $x_{0:t-1}$ at each step by simply throwing away the particle histories and keeping only the particle approximation $X_t^{(1:N)}$ to the filtering distribution at the current time t. The algorithm progresses in a Markovian fashion, only ever referring to the particles at the immediately previous step, so filtering distributions can be approximated with minimal memory usage. This is vital if one wishes to carry out filtering in an on-line fashion, as it prevents the memory requirements accumulating more than necessary. In particular, the storage cost is $O(N)$, independently of T.

In the case of smoothing, the need to store information about all T time steps means that the memory requirements increase linearly with T. Storage considerations for smoothing are discussed further in Section 4.

SMC methods are justified by a central limit theorem and other convergence results, but these are beyond the scope of this report. See for example Del Moral (2013) for details.

3 Coalescent Theory

Definition 1 (Möhle (1998))**.** Let $\mathcal{E}_n$ denote the set of equivalence relations on $\{1, \dots, n\}$. A *discrete-time coalescent* is a stochastic process $(R_t)_{t \in \mathbb{N}}$ taking values in $\mathcal{E}_n$ such that $R_0 = \{(1,1), (2,2), \dots, (n,n)\}$ and $\mathbb{P}[R_{t+1} = \eta \mid R_{0:t-1}, R_t = \xi] > 0$ only if $\xi \subseteq \eta$.

That is, the initial state is the trivial relation where each index is in its own equivalence class, and the only possible forward-in-time transitions are staying the same or merging some equivalence classes together. An obvious consequence of this is that the state where all of the indices are in the same equivalence class is an absorbing state for the process.

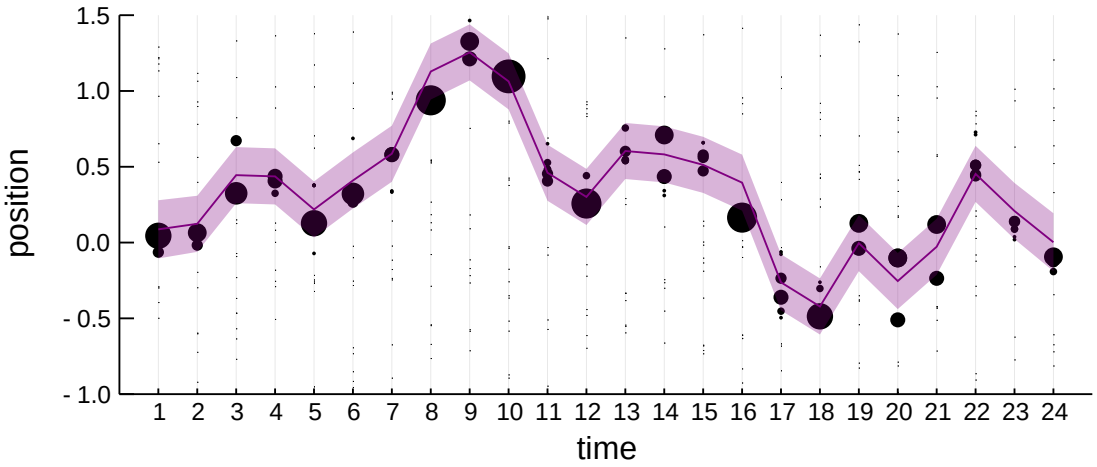


Figure 1: SMC particles before resampling for a linear Gaussian model. The purple ribbon shows the exact posterior mode and 95% highest posterior density interval, computed using the Kalman filter and RTS smoother. The black dots show the positions of the SMC particles, with size proportional to weight. After resampling all particles have equal weights but some are duplicated.

The genealogical interpretation of the equivalence relations is that $(i, j) \in R_t$ if and only if individuals i and j share a common ancestor in generation t. Note that the notion of a coalescent appears when we view the genealogy *backwards in time*.

3.1 Kingman's coalescent

Imagine we have a population with fixed size N over discrete generations, where each individual is descended from one randomly chosen individual of the previous generation. Then for each individual in the present generation, we can trace their *lineage* back through the generations. If we trace two lineages back in time, at some generation they may descend from the same individual, at which point we say they have *coalesced*. Once two lineages have coalesced they will stay together going backwards in time. The combined lineages of $n \leq N$ of the present individuals therefore forms a tree, or several non-overlapping trees, the entirety of which we refer to as the *ancestry* or *genealogy* of those n individuals.

Kingman's n-*coalescent* provides a model for such genealogies. Kingman showed in (Kingman, 1982*a*,*b*,*c*) that the n-coalescent is the limiting process for samples from a wide class of population models as $N \to \infty$.

The defining feature of the model is that each pair of lineages merges with unit rate. This means that, looking at the whole population, while there are more distinct lineages present, coalescences happen exponentially more frequently. In particular, the n-coalescent can be formulated as a Poisson process where pairs of lineages coalesce independently at rate 1, with the pair to coalesce being chosen uniformly at random (Wakeley, 2009, Section 3.2).

In the notation of Wakeley (2009), let T_i; $i = 2, \ldots, n$ be the i^{th} coalescence time, that is, the length of time for which there are exactly i branches in the sample genealogy. The n-coalescent is the process in which these times are distributed as independent Exponentials with rate $\binom{i}{2}$.

Möhle (1998) equivalently writes the process in terms of the infinitesimal generator Q of a Markov process on the set of equivalence relations on n elements, having entries

$$q_{\xi\eta} = \begin{cases} -\binom{b}{2} & \text{if } \xi = \eta \\ 1 & \text{if } \xi \prec \eta \\ 0 & \text{otherwise} \end{cases}$$

where b is the number of equivalence classes of ξ, and $\xi \prec \eta$ means that η is a state with exactly one more pair of lineages coalesced compared to ξ.

A realisation of the n-coalescent for $n = 50$ is shown in Figure 2.

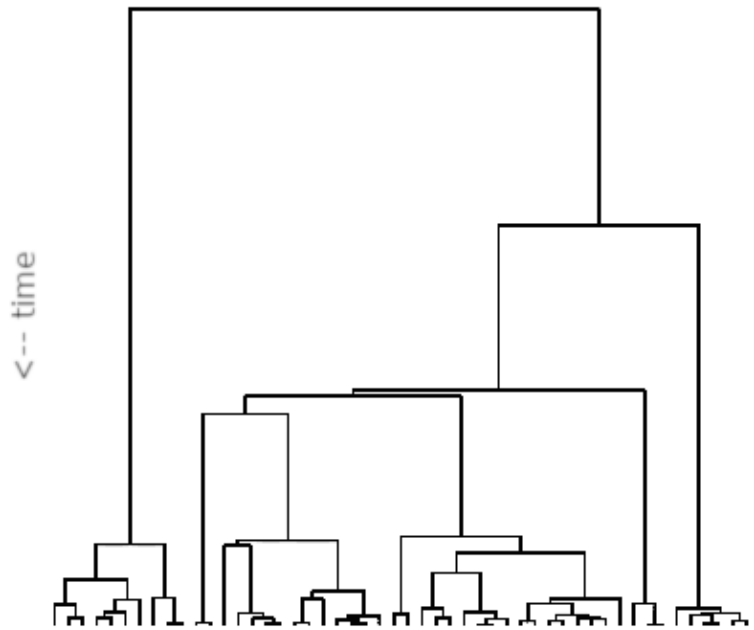


Figure 2: A realisation of Kingman's n-coalescent for a sample of size $n = 50$. At first there are many distinct lineages, and mergers happen rapidly. Once there are fewer distinct lineages left, they take longer to merge. The process spends about half of its time with just two or three distinct lineages. *(Source: Wikimedia Commons)*

The *Kingman coalescent* is a process on $\mathbb{N}$ such that the genealogy of any sample of size n individuals from the present generation is an n-coalescent. A precise formulation of this notion is given in (Wakeley, 2009).

3.2 Limiting genealogies for population models

The Kingman coalescent is the asymptotic process (in the large population limit) for the coalescents arising from a surprisingly wide range of population models. Some important examples are studied in this section.

3.2.1 Wright-Fisher model

The neutral Wright-Fisher model (Fisher, 1923, 1930; Wright, 1931) is one of the most studied models in population genetics.

Assume we have a population of constant size N which reproduces in discrete generations. At each time step, the existing generation dies, and N offspring are produced. The offspring descend from parents $(a_1, \ldots, a_N)$ that are independently selected according to

$$a_i \sim \text{Categorical}(\{1, \ldots, N\}, (1/N, \ldots, 1/N)).$$

The numbers of offspring of each parent are therefore jointly distributed

$$(v_1, \ldots, v_N) \sim \text{Multinomial}(N, (1/N, \ldots, 1/N)).$$

This model is "neutral" because all of the individuals are chosen to reproduce with the same probability. That is, no individual is considered "fitter" than any other, so there is no genetic selection. There are several variants of the Wright-Fisher model that include selection, but they are typically much less tractable than the neutral one.

Kingman showed in his original papers introducing the Kingman coalescent (Kingman, 1982*c*) that, under a certain time-scaling, genealogies of the neutral Wright-Fisher model converge to the Kingman coalescent as $N \to \infty$.

3.2.2 Cannings model

The neutral Cannings model (Cannings, 1974, 1975) is a more general construction which encompasses the neutral Wright-Fisher model as a special case.

Again we have a population of constant size N reproducing in discrete generations. Now, the particular selection mechanism is not specified; we only require that the offspring distribution is exchangeable, i.i.d. at each time step, and preserves the population size. That is, the probability of observing offspring counts $(v_1, \ldots, v_N)$ is invariant under permutations of this vector.

Invariance under permutations encodes the fact that all parents are treated equally, hence this is still a neutral model. It is clear that the neutral Wright-Fisher model falls under this framework, because the multinomial distribution with equal weights is indeed exchangeable.

Genealogies of the neutral Cannings model also converge to the Kingman coalescent, under some conditions and a suitable time-scaling, as $N \to \infty$ (Etheridge, 2011, Section 2.2).

3.2.3 Moran model

The neutral Moran model (Moran, 1958) is often preferred by mathematicians because of its tractability.

We still have a population of constant size N with discrete generations. At each time step, an ordered pair of individuals is selected uniformly at random. The first individual in this pair dies (i.e. leaves no offspring in the next generation), while the other reproduces (leaving two offspring). All of the other individuals leave exactly one offspring.

Usually the model is thought of as having "overlapping generations": the individuals having one offspring are considered to be not reproducing but rather surviving to appear again in the next generation. However, one can equally think of it as having non-overlapping generations and a low variance reproduction mechanism.

In fact, it is again a special case of the neutral Cannings model, where the offspring distribution is uniform over all permutations of $(0, 2, 1, 1, \ldots, 1)$. Therefore we know that under a suitable time-scaling, its genealogies converge to the Kingman coalescent.

The mathematical tractability of this model stems from the fact that, if we formulate it a little more carefully, its pre-limiting behaviour is exactly that of a Kingman coalescent. This requires us to work in continuous time, and now we say that the pairs are chosen to die/reproduce with exponential rate $\binom{N}{2}$. This gives the unit pairwise merger rate we need, and we can clearly see that the only possible mergers are between one pair of lineages at a time.

4 SMC genealogies as coalescents

In this section we reveal the link between sequential Monte Carlo and population genetics, and introduce the contribution of this work.

4.1 Ancestral degeneracy

The resampling step in Algorithm 1 induces a branching process which, viewed backwards in time, is a coalescent process.

During resampling, some particles have multiple offspring while others have none. This generates the genealogy, because particles that have no offspring in some generation "die out". An example of this is shown in Figure 3.

Tracing the lineages of the N time T particles backwards in time, they gradually coalesce until, at some time $s < T$, all of the time T particles descend from a single time s ancestor. If the number of particles N is big enough compared to the time horizon T, it could be that $s < 0$, in which case the genealogies do not fully coalesce within the sample. If N is not big enough, we end up with just one particle representing the marginal smoothing distributions at times $0, \ldots, s$. Typically, even at times quite close to T, there are no more than a few distinct particles in the sampling window.

In order to estimate filtering distributions $p(x_t|y_{0:t})$, we only require a sample of particles at the current time step, so if $\mathcal{X}$ is continuous we typically have N distinct positions given by the N particles. Then the

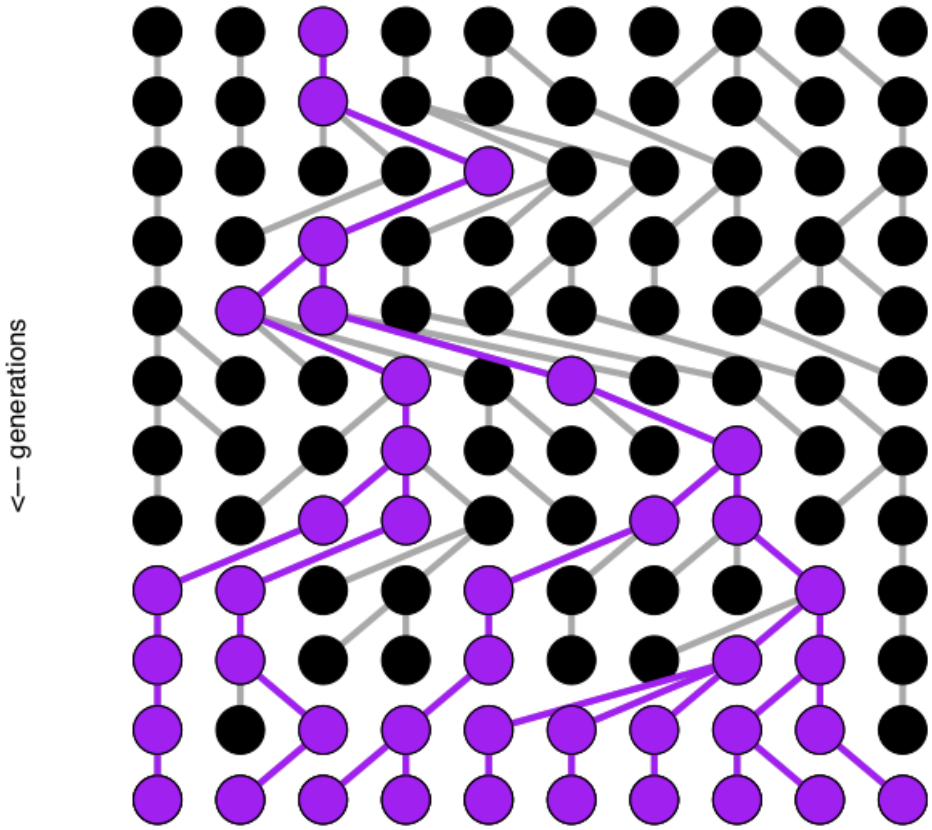


Figure 3: Genelaogical tree induced by resampling over 12 generations with $N = 10$ particles. At each resampling step, any particles with no offspring "die out"; they are not in the lineage of any time T particle. In this realisation, the N particles at time T all originate from the same time 0 ancestor. "Dead" particles/lineages are coloured black/grey, while the "live" tree is highlighted in purple.

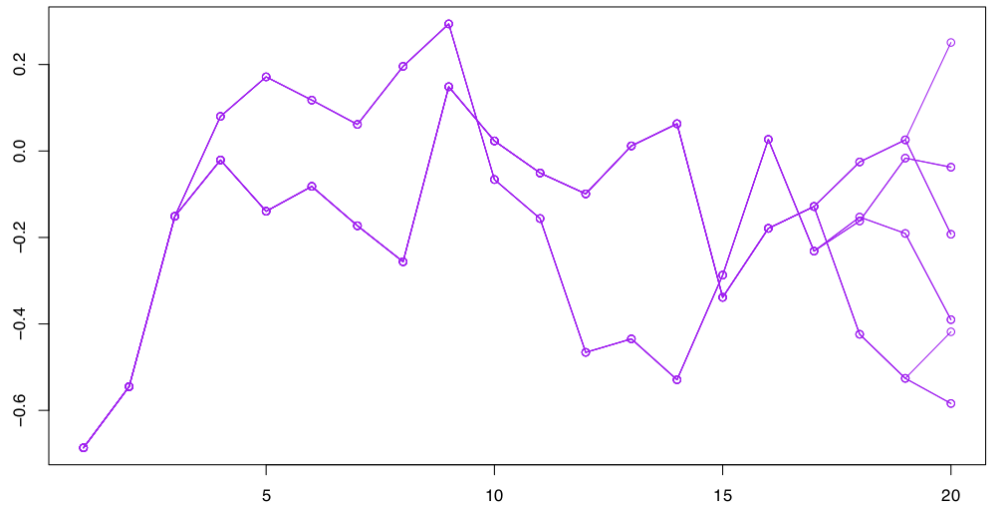


Figure 4: A sample of $N = 6$ trajectories, illustrating ancestral degeneracy. At the "present" time there are six distinct lineages, but just three steps back they have coalesced onto only two lineages, and it takes less than 20 steps back before only one lineage remains.

empirical measure

$$\hat{p}(dx_t|y_{0:t}) = \frac{1}{N}\sum_{i=1}^{N} \delta_{X_t^{(i)}}(dx_t)$$

has mass in N locations, and the Monte Carlo error for estimating expectations under $p(x_t|y_{0:t})$ scales like $N^{-1/2}$ (Doucet and Johansen, 2011).

However, in the case of estimating the smoothing distributions $p(x_{0:t}|y_{0:t})$, we require a sample of *trajectories* over times $0:t$ as opposed to a sample of particles at time t. The coalescence of lineages is an unavoidable effect of resampling, and it causes more and more of these trajectories to coincide the further into the past we look. The resulting empirical measure

$$\hat{p}(dx_{0:t}|y_{0:t}) = \frac{1}{N}\sum_{i=1}^{N} \delta_{X_{0:t}^{(i)}}(dx_{0:t})$$

typically consists of N distinct masses, but the marginals at early times may consist of only a single mass repeated N times. This phenomenon, known as *ancestral degeneracy*, is illustrated in Figure 4.

If we are really interested in the smoothed states a long way into the past, the estimation variance blows up. This problem was identified even in the early literature (Gordon et al., 1993), where some ad hoc methods were suggested to reduce it.

Since then many techniques have been proposed to mitigate ancestral degeneracy. When no resampling is done, the variance of the particle weights explodes, so that after a few iterations there is essentially just one particle contributing to the empirical measure. This is known as *weight degeneracy*, and it causes much the same problems as ancestral degeneracy, except that it worsens forwards in time, while ancestral degeneracy worsens backwards in time.

This means that there is a trade-off between the two, so achieving good performance is difficult. Various strategies have been proposed in the literature. These include adaptive resampling (resampling not at every step but only when the variance of the weights gets too high) and using lower-variance resampling schemes (this is discussed in detail in Section 6), as well as modified algorithms such as the auxiliary particle filter (Pitt and Shephard, 1999), resample-move (Gilks and Berzuini, 2001) and block sampling (Doucet et al., 2006).

4.2 Quantifying degeneracy

The problem of ancestral degeneracy can be mitigated by these various measures, but not entirely removed. Therefore it is natural to ask how bad we can expect it to be. While weight degeneracy is a reasonably well-quantified problem, there exists little in the way of tools for quantifying ancestral degeneracy. There have been some simulation studies attempting to cast light on the magnitude of this problem, but analytical findings remain elusive, since the complexity of the most popular particle methods makes it difficult to obtain any rigorous results.

If we could discover how quickly lineages will coalesce in a given algorithm, we could use this information to tune the number of particles, or even the time horizon, such that an acceptable number of distinct trajectories remain throughout the sampling window. For instance, in the *particle Gibbs* algorithm it is important to maintain at least two distinct trajectories with high probability, so the number of particles should be chosen large enough to ensure this (this is discussed further in Section 5).

In general, analysis of the genealogies can quantify the severity of ancestral degeneracy and thus the sampling variance to be expected with a given algorithm and number of particles. Variance estimation for SMC is an active research topic (Chan et al., 2013; Lee and Whiteley, 2018; Olsson et al., 2019), but the previously proposed estimators require the output from at least one run of the algorithm, whereas our analysis allows us to analyse aspects of an algorithm's performance without the need for any test runs.

Moreover, it could provide an analytical comparison of the efficacy of the various strategies proposed to reduce ancestral degeneracy, where previously recommendations have mostly been based on empirical performance. For instance, in Section 6 we discuss the relative performance of various resampling schemes.

There is also a connection with estimating the memory requirements of SMC algorithms. Storing the entire trajectory for each of N particles over time horizon T requires $O(NT)$ memory, but ancestral degeneracy means that much of this storage is redundant. This mean that more ancestral degeneracy is actually advantageous in terms of memory requirements. Jacob et al. (2015) construct an algorithm for efficiently storing the trajectories, and show that the expected cost is at most $O(N \log N)$, independent of T. Koskela et al. (2018) remark that this bound could perhaps be reduced to $O(N)$, based on empirical results.

4.3 Existing results

The first results showing convergence of population models to the Kingman coalescent appear in Kingman's original papers. This includes the neutral Wright-Fisher model and the neutral Moran model. Furthermore, it has been shown that the more general class of neutral Cannings models converge to the Kingman coalescent. Möhle (1998) proved convergence for a larger class still, encompassing some non-exchangeable models.

Koskela et al. (2018) presents the first application of this type of analysis to SMC genealogies. Their result relies heavily on the methods introduced by Möhle (1998). They prove convergence to the Kingman coalescent for genealogies induced by a first SMC algorithm, with multinomial resampling. In this work we attempt to extend their result to cover some other SMC algorithms.

It is interesting to note that many of the simplifying assumptions in population models actually hold in the case of SMC genealogies. The population size (number of particles) is a constant N; the generations are discrete and non-overlapping (all particles are resampled at once). The main difficulty in analysing SMC genealogies is the lack of neutrality: parents are selected with unequal weights, corresponding to "fitness" in population genetics.

5 Conditional SMC

5.1 Particle Gibbs

Consider a model like the one presented in Section 1.1, where the transition density of (X_t) depends on some parameter θ. That is,

$$
\begin{aligned}
&X_0 \sim \mu(\cdot) \\
&X_{t+1} \mid (X_t = x_t) \sim f_\theta(\cdot|x_t) \qquad t = 0, \dots, T-1 \\
&Y_t \mid (X_t = x_t) \sim g(\cdot|x_t) \qquad t = 0, \dots, T
\end{aligned}
$$

It is plausible to assume additionally the conditional independence statement $(\theta \perp\!\!\!\perp Y_t) \mid X_t$, although this may not be necessary.

In this scenario we might want to find the joint posterior distribution $p(\theta, x_{0:T}|y_{0:t})$. If the density f_θ is suitably tractable, then it should be possible to sample exactly from $p(\theta|x_{0:t}, y_{0:t}) = p(\theta|x_{0:t})$, given some value of $x_{0:t}$. This suggests employing a Gibbs sampler alternately sampling from $p(\theta|x_{0:t}, y_{0:t})$ and $p(x_{0:t}|\theta, y_{0:t})$.

The distribution $p(x_{0:t}|\theta, y_{0:t})$ plays the same role as $p(x_{0:t}|y_{0:t})$ in the model of Section 1.1, because although it is now parametrised by θ, we are conditioning on a specific value of the parameter. This is then exactly the distribution that we suggested targeting with SMC since it is too difficult to sample from exactly. So is it possible to use SMC within this Gibbs sampler?

This is precisely what was suggested in Andrieu et al. (2010) — although this is just one of many possibilities to which they apply their method. They point out that replacing an exact sample from $p(x_{0:t}|\theta, y_{0:t})$ with an SMC approximation does not yield a Gibbs sampler targeting the desired invariant distribution $p(\theta, x_{0:T}|y_{0:t})$. This can be corrected by using instead a *conditional SMC* update, which is described in Section 5.2.

Each iteration of the resulting particle Gibbs algorithm runs as follows: given the observations $y_{0:T}$ as well as positions $X_{0:T}^{(i-1)}$ and indices $a_{0:T}^{(i-1)}$ from the previous iteration,

1. Sample $\theta^{(i)} \sim p(\cdot|X_{0:T}^{(i-1)}, y_{0:T})$
2. Run a conditional SMC update conditional on $X_{0:T}^{(i-1)}, a_{0:T}^{(i-1)}$ to obtain the approximate distribution $\hat{p}(dx_{0:T}|\theta^{(i)}, y_{0:T})$
3. Sample $X_{0:T}^{(i)} \sim \hat{p}(dx_{0:T}|\theta^{(i)}, y_{0:T})$. This implicitly samples $a_{0:T}^{(i)}$, the particle indices of $X_{0:T}^{(i)}$.

These steps are repeated as many times as desired to produce the MCMC output.

5.2 The conditional SMC update

Conditional SMC differs from the standard algorithm in that one predetermined trajectory (that is, a sequence of particle positions and the corresponding ancestral indices) is conditioned to survive all of the propagation and resampling steps. We refer to this sequence of positions as the *immortal trajectory*, following the terminology used for conditioned Galton-Watson processes, and the *immortal particle* will refer to the particle in a particular generation that is part of the immortal trajectory.

When used as a component of the particle Gibbs algorithm, the immortal trajectory $x_{0:T}^*$ for each SMC run is sampled from the trajectories output from the previous run (Andrieu et al., 2010, Section 2.4.3). However, for our purposes we just consider a single conditional SMC run for which the immortal trajectory is fixed. A conditional SMC algorithm employing multinomial resampling is described in Algorithm 2. Our statement of the algorithm differs from the original formulation in that we do not include the immortal indices as an input; instead we sample them uniformly during the algorithm. For our purposes this is equivalent.

In the particle Gibbs sampler, it is crucial that the conditional SMC output maintains at least two distinct trajectories. The immortal trajectory will of course be among the surviving trajectories, but additionally, the new immortal trajectory (for the next SMC run) is chosen from among the surviving trajectories. Thus if all the trajectories coalesce onto the immortal trajectory, we are forced to choose the same immortal trajectory for the next run, at least for some early time steps $0 : s$. One can imagine that if there was a high probability of full coalescence on each run, we could easily end up with samples from $p(x_{0:T}|y_{0:T})$ that are identical in the coordinates $0 : s$, which would lead to poor results overall.

The problem can be avoided by using a sufficiently large number of particles for the fixed time window T of the conditional SMC runs. Corollary 1 (stated in Section 5.3) could possibly provide an a priori estimate of an appropraite N. If, say, we want to ensure that the probability of all N lineages coalescing is below a certain threshold, all of the relevant information is encoded in the distribution of the time to MRCA of the genealogical process. For the Kingman coalescent this distribution is known, and Corollary 1 states that as $N \to \infty$ the genealogy is a Kingman coalescent. The remaining question is whether the Kingman coalescent provides a reasonable approximation outside of the asymptotic regime - since in reality we simulate finitely many particles. We intend to investigate this question by way of a simulation study.

Algorithm 2 Conditional SMC with multinomial resampling

Require: $N, T, \mu, \{K_t\}, \{g_t\}, y_{0:T}, x^*_{0:T}$

for $i \in \{1, \dots, N\}$ **do**
  Sample $X_0^{(i)} \sim \mu(\cdot)$ ▷ initialise
end for
Sample $a_0^* \sim \text{Uniform}(\{1, \dots, N\})$
$X_0^{(a_0^*)} \leftarrow x_0^*$
for $i \in \{1, \dots, N\}$ **do**
  $w_0^{(i)} \leftarrow g_0(X_0^{(i)})$
end for
$W \leftarrow \sum_{j=1}^N w_0^{(j)}$
for $i \in \{1, \dots, N\}$ **do**
  $w_0^{(i)} \leftarrow \frac{1}{W} w_0^{(i)}$ ▷ normalise weights
end for
for $t \in \{0, \dots, T-1\}$ **do**
  Sample $a_t^{(1:N)} \sim \text{Categorical}(\{1, \dots, N\}, w_t^{(1:N)})$ ▷ resample particles
  Sample $a_{t+1}^* \sim \text{Uniform}(\{1, \dots, N\})$
  $a_t^{(a_{t+1}^*)} \leftarrow a_t^*$
  for $i \in \{1, \dots, N\}$ **do**
    Sample $X_{t+1}^{(i)} \sim K_{t+1}(X_t^{(a_t^{(i)})}, \cdot)$ ▷ propagate particles
  end for
  $X_{t+1}^{(a_{t+1}^*)} \leftarrow X_{t+1}^*$
  for $i \in \{1, \dots, N\}$ **do**
    $w_{t+1}^{(i)} \leftarrow g_{t+1}(X_t^{(a_t^{(i)})}, X_{t+1}^{(i)})$ ▷ calculate weights
  end for
  $W \leftarrow \sum_{j=1}^N w_{t+1}^{(j)}$
  for $i \in \{1, \dots, N\}$ **do**
    $w_{t+1}^{(i)} \leftarrow \frac{1}{W} w_{t+1}^{(i)}$ ▷ normalise weights
  end for
end for

5.3 Genealogies of conditional SMC algorithms

This section outlines the convergence proof for conditional SMC. We need to calculate various quantities related to the genealogical process induced by conditional SMC with multinomial resampling. By writing these in terms of the corresponding quantities for standard SMC with multinomial resampling, we are able to apply results from Koskela et al. (2018). In this way we will show that under a certain time-rescaling the genealogical process converges to the Kingman coalescent, in the sense of finite-dimensional distributions, as the number of particles $N \to \infty$.

Below is an overview of the proof. To prove convergence to the Kingman coalescent, we must control the rates of different types of mergers. In particular, we ensure that in the large population limit (under an appropriate time-scaling), pairwise mergers happen at the correct rate, and larger mergers never occur. The derivations of the expressions (4), (5), (6), along with details of the application of results from Koskela et al. (2018), are relegated to Appendix A.

A central quantity is

$$c_N(t) := \frac{1}{(N)_2} \sum_{i=1}^{N} (v_t^{(i)})_2.$$

This is the number of pairs merging in generation t. In the $N \to \infty$ limit, this will become the overall coalescence rate, so we must ensure that it matches asymptotically the coalescence rate of the Kingman coalescent. This is why we need to calculate its first and second moments below.

Another important quantity is

$$D_N(t) := \frac{1}{N(N)_2} \sum_{i=1}^{N} (v_t^{(i)})_2 \left(v_t^{(i)} + \frac{1}{N} \sum_{j \neq i} (v_t^{(j)})^2 \right)$$

which is asymptotically the rate of *super-binary mergers* — that is, more than two lineages merging simultaneously into one or more lineages. We must ensure that it vanishes as $N \to \infty$, since super-binary mergers are not allowed under the Kingman coalescent.

Throughout the following we use a tilde to indicate the conditional SMC versions of the quantities relating to standard SMC, always with multinomial resampling.

Firstly, we have the expected coalescence rate:

$$\mathbb{E}[\tilde{c}_N(t)|\mathcal{F}_{t-1}] = \frac{N-2}{N} \mathbb{E}[c_N(t)|\mathcal{F}_{t-1}] + \frac{2}{N} \mathbb{E}[w_t^{(1)}|\mathcal{F}_{t-1}] \tag{4}$$

Then the expected rate of super-binary mergers is bounded above by:

$$\begin{aligned} \mathbb{E}[\tilde{D}_N(t)|\mathcal{F}_{t-1}] \leq \mathbb{E}[D_N(t)|\mathcal{F}_{t-1}] &+ \frac{3}{N} \mathbb{E}[(w_t^{(1)})^2|\mathcal{F}_{t-1}] + \frac{4}{N^2} \mathbb{E}[w_t^{(1)}|\mathcal{F}_{t-1}] \\ &+ \frac{4}{N} \sum_{i=2}^{N} \mathbb{E}[w_t^{(1)} (w_t^{(i)})^2|\mathcal{F}_{t-1}] + \frac{2}{N^2} \sum_{i=2}^{N} \mathbb{E}[w_t^{(1)} w_t^{(i)}|\mathcal{F}_{t-1}] + \frac{1}{N^2} \sum_{i=2}^{N} \mathbb{E}[(w_t^{(i)})^2|\mathcal{F}_{t-1}] \end{aligned} \tag{5}$$

And lastly a bound on the expectation of the squared coalescence rate:

$$\begin{aligned} \mathbb{E}[\tilde{c}_N(t)^2|\mathcal{F}_{t-1}] \leq \mathbb{E}[c_N(t)^2|\mathcal{F}_{t-1}] &+ \frac{4}{N} \mathbb{E}[(w_t^{(1)})^3|\mathcal{F}_{t-1}] + \frac{12}{N^2} \mathbb{E}[(w_t^{(1)})^2|\mathcal{F}_{t-1}] + \frac{4}{N(N)_2} \mathbb{E}[w_t^{(1)}|\mathcal{F}_{t-1}] \\ &+ \frac{4}{N} \sum_{i=2}^{N} \mathbb{E}[w_t^{(1)} (w_t^{(i)})^2|\mathcal{F}_{t-1}] \end{aligned} \tag{6}$$

We then apply Lemma 3 of Koskela et al. (2018) to obtain the more tractable expressions

$$\frac{\varepsilon^4}{Na^4} + O(N^{-2}) \leq \mathbb{E}[\tilde{c}_N(t)|\mathcal{F}_{t-1}] \leq \frac{a^4}{N\varepsilon^4} + O(N^{-2})$$
$$\mathbb{E}[\tilde{D}_N(t)|\mathcal{F}_{t-1}] \leq \frac{C}{N}\mathbb{E}[\tilde{c}_N(t)|\mathcal{F}_{t-1}] + O(N^{-3})$$
$$\mathbb{E}[\tilde{c}_N(t)^2|\mathcal{F}_{t-1}] \leq \frac{C}{N}\mathbb{E}[\tilde{c}_N(t)|\mathcal{F}_{t-1}] + O(N^{-3})$$

and define the time-scaling

$$\tilde{\tau}_N(t) := \min\left\{ s \geq 1 : \sum_{r=1}^{s} \tilde{c}_N(r) \geq t \right\}. \tag{7}$$

Then, using Koskela et al. (2018, Lemma 2), which readily generalises to our modified quantities, we are able to verify the four conditions of Koskela et al. (2018, Theorem 1). Finally we are able to conclude the following.

Corollary 1. *Under the conditions of Koskela et al. (2018, Lemma 3), the genealogy of any n particles from a conditional SMC algorithm with multinomial resampling converges to Kingman's n-coalescent in the sense of finite-dimensional distributions, under the time-scaling defined in* (7)*.*

6 Alternative resampling schemes

There is a great deal of flexibility in the function referred to as RESAMPLE in Algorithm 1. Various possible schemes are presented in this section.

For a procedure to be considered a valid resampling scheme, we will require the following:

1. The total number of particles N remains fixed.
2. The particles after resampling are equally weighted.
3. The resampling scheme is unbiased; that is, the expected number of offspring of each particle i is equal to $Nw_t^{(i)}$.

Condition 3 ensures that the resulting SMC estimates are unbiased. Conditions 1 and 2 can be violated (for example Liu and Chen (1998) suggest sampling with the square roots of the weights and then correcting) but this is not generally advantageous.

6.1 Multinomial resampling

Multinomial resampling (Efron and Tibshirani, 1994) is one of the simplest resampling schemes. The parental indices are chosen independently from $\{1, \ldots, N\}$, each with probability given by the weight of the corresponding particle $w_t^{(i)}$. That is,

$$a_t^{(1:N)} \sim \text{Categorical}(\{1, \ldots, N\}, w_t^{(1:N)}).$$

This implies the joint distribution of the offspring counts is

$$v_t^{(1:N)} \overset{d}{=} \text{Multinomial}(N, w_t^{(1:N)}).$$

Note that in this case the parental indices are chosen independently, but the resulting offspring counts are negatively correlated.

A simple way to sample the parental indices is by inversion sampling: divide the unit interval into N disjoint subintervals each of which will correspond to a certain index i and has length equal to the weight

$w_t^{(i)}$; then draw N samples $U_i \sim \text{Uniform}(0,1)$ and classify them according to which of these subintervals they fall in. Explicitly, the parental index assigned to child i is the index a_i satisfying

$$\sum_{j=1}^{a_i-1} w_t^{(j)} \leq U_i \leq \sum_{j=1}^{a_i} w_t^{(j)} \tag{8}$$

This is illustrated in Figure 5a. Note that there exist more efficient methods to sample from a Multinomial distribution, so the inversion method may not be used in practice.

6.2 Residual resampling

Residual resampling is described in Liu and Chen (1998) and also in Whitley (1994) where it is called "remainder stochastic sampling".

Each particle $X_t^{(i)}$ is deterministically assigned $\lfloor N w_t^{(i)} \rfloor$ offspring, and the remaining $R := N - \sum_{i=1}^{N} \lfloor N w_t^{(i)} \rfloor$ offspring are assigned multinomially in proportion to the unaccounted-for weight. This yields a vector of offspring counts

$$v_t^{(1:N)} \stackrel{d}{=} \lfloor N w_t^{(1:N)} \rfloor + \text{Multinomial}(R, (N w_t^{(1:N)} - \lfloor N w_t^{(1:N)} \rfloor)/R).$$

The deterministic part ensures that every particle with weight $> 1/N$ is guaranteed to survive. This is a desirable property as it prevents the random loss of high-weighted particles.

6.3 Stratified resampling

Stratified resampling is introduced in (Kitagawa, 1996).

The scheme proceeds like Multinomial resampling, except that the Uniform samples that are fed in to do the Categorical sampling are produced in a different way. Instead of sampling N independent numbers from $\text{U}(0,1)$, one number is sampled uniformly from each subinterval of length $1/N$. That is,

$$U_i \sim \text{Uniform}\left(\frac{i-1}{N}, \frac{i}{N}\right).$$

The parents are then assigned as in (8). (Of course this means that the offspring distribution is no longer Multinomial, since parental indices are not chosen independently.) This scheme ensures that the samples are "well spread out", again reducing the probability of randomly losing high-weighted particles. The method is illustrated in Figure 5b.

6.4 Systematic resampling

Systematic resampling is described in Carpenter et al. (1999) and also in Whitley (1994) where it is called "stochastic universal sampling".

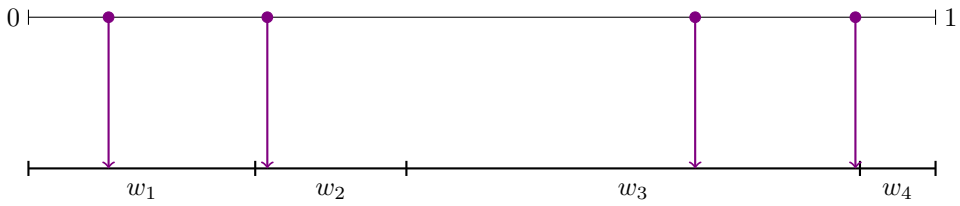
Like stratified resampling, it constitutes a change to the random number generator for sampling from the Categorical distriubtion. In this scheme, only one Uniform sample is drawn, $U \sim \text{U}(0, 1/N)$, and the other $N-1$ samples are generated deterministically by setting

$$U_i = U + \frac{i-1}{N}$$

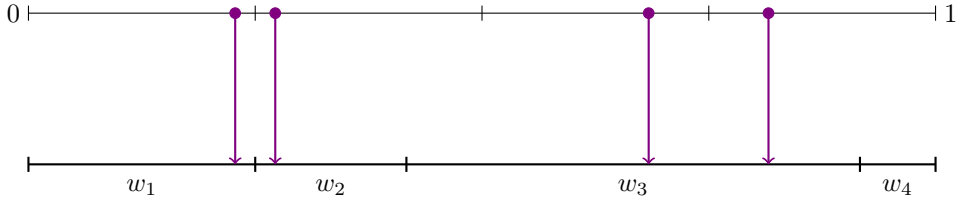
for each $i \in \{1, \ldots, N\}$. The parental indices are again selected according to (8). The method is illustrated in Figure 5c. This scheme again ensures the random numbers are "well spread out", even more so than with stratified resampling.

Systematic resampling is often preferred among practitioners because it is extremely easy to implement and also computationally efficient, requiring only one random number to be generated.

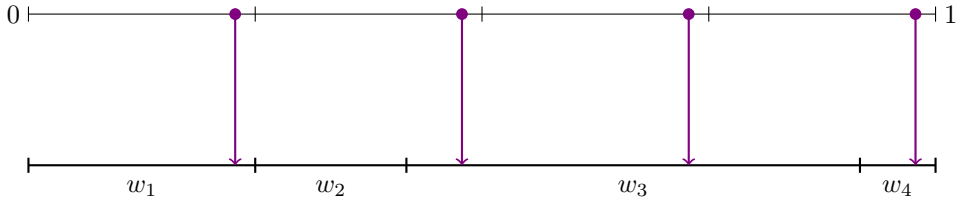
However, this scheme is known to exhibit pathological behaviour in some cases due to its dependence on the ordering of the subintervals (Douc et al., 2005). Such behaviour can be avoided by randomly permuting the intervals before sampling, and this is the recommended practice.



(a) Multinomial resampling



(b) Stratified resampling



(c) Systematic resampling

Figure 5: Illustration of the way parental indices are sampled in three different resampling schemes. For this example $N = 4$ and the weights are $w_t^{(1:4)} = \frac{1}{N}(1, \frac{2}{3}, 2, \frac{1}{3})$. In each case the indices are assigned by inversion sampling seeded with the same Uniform$(0, 1)$ samples.
(a) For Multinomial resampling, we sample N independent Uniform$(0, 1)$ random variables. In this example the sampled offspring counts are $(1, 1, 2, 0)$.
(b) For stratified resampling, the Uniform$(0, 1)$ samples are transformed to Uniform draws from the intervals (0,0.25), (0.25,0.5), (0.5, 0.75), (0.75,1). In this example the sampled offspring counts are $(1, 1, 2, 0)$.
(c) For systematic resampling, we use only the first draw and transform as in (b) to get a sample from Uniform$(0, 0.25)$. For the subsequent draws, we just add 0.25 each time to obtain a sample in each interval. In this example the sampled offspring counts are $(1, 0, 2, 1)$.

6.5 Remarks on performance

6.5.1 Variance

The most straightforward choice of resampling scheme, and also the easiest to analyse, is multinomial. However, multinomial resampling is well known to be sub-optimal in terms of the resulting Monte Carlo variance, and is rarely used in practice.

Douc et al. (2005) prove that both residual resampling and stratified resampling yield lower variance estimators. The variance we are referring to here is the variance of Monte Carlo estimators of an arbitrary test function f, conditional on the past:

$$\text{Var}\left[\frac{1}{N}\sum_{i=1}^{N} f(X_t^{(i)})\middle|\mathcal{F}_{t-1}\right]$$

The authors remark that, while the variance resulting from systematic resampling is not provably lower than that of multinomial resampling, empirical performance is comparable among residual, stratified and systematic resampling.

6.5.2 Support of offspring numbers

Let us consider the support of the marginal offspring distributions in each scheme, given the corresponding weight. Condition on the i^{th} weight lying in the interval $w_t^{(i)} \in [k/N, (k+1)/N]$, but leave the other weights unknown. By considering the best and worst cases for each scheme, we have:

Multinomial: $v_t^{(i)} \in \{0, \ldots, N\}$

Residual: $v_t^{(i)} \in \{k, \ldots, k+R\}$

Stratified: $v_t^{(i)} \in \{k-1, k, k+1, k+2\}$

Systematic: $v_t^{(i)} \in \{k, k+1\}$

We see that multinomial resampling allows the possibility of very good particles having 0 offspring, and of very bad particles having N offspring (although the probabilities associated to these events are low). Residual resampling ensures that good particles do not die out, but still allows bad particles to possibly have many offspring. Stratified resampling is more restrictive, although it allows the possibility of a particle with weight $> 1/N$ leaving no offspring. Systematic resampling is more restrictive still, allowing the number of offspring of each particle to vary from its expected value by no more than one.

6.5.3 Permutation invariance

A strange property of stratified and systematic resampling is that they are sensitive to the order in which the subintervals are placed. For example, in Figures 5b and 5c if the intervals w_2 and w_4 were swapped, the number of offspring assigned to particles 2 and 4 would be swapped in each case. We can also see that because w_1 has weight $\geq 1/N$ and is placed first, it is guaranteed at least one offspring.

This property can lead to pathological behaviour, but is easily avoided by applying a random permutation to the order of the subintervals. Gerber et al. (2017) also propose a variation on systematic resampling that avoids this property.

6.5.4 Degeneracy under equal weights

Suppose we somehow end up in the situation where all the weights are equal (i.e. $w_t^{(i)} = 1/N$ for all i). In this case, residual resampling will result in a deterministic assignment only: each particle will be assigned one offspring, and there will be no remainder left to assign randomly. This behaviour cannot be avoided, however the event that all weights are equal typically has zero measure.

Stratified and systematic resampling will have the same result: the intervals for sampling will correspond exactly to the weighted subintervals, so no matter which random numbers are sampled, exactly one will fall in each subinterval.

However, for stratified resampling, the formulation of Whitley (1994) avoids this behaviour. He imagines subdivisions of a circle rather than an interval, and then "spins the roulette wheel" around it, which shifts the sampling intervals by a random amount and thus prevents this degeneracy.

6.5.5 Exchangeability

We will call a resampling scheme exchangeable if the resulting distribution of parental indices is invariant under permutations of the children. To put it another way, each child chooses its parent from the same marginal distribution.

It is clear that multinomial resampling is exchangeable since in this case the parental indices are independent and identically distributed. However it is worth noting that some efficient implementations of multinomial sampling may not preserve exchangeability in practice.

Stratified and systematic resampling are clearly not exchangeable since, for instance, child 1 is more likely to choose parent 1 than child N is. However, this is merely a feature of the arbitrary ordering of the sampling steps: exchangeability can easily be reintroduced by applying a random permutation to the vector of parental indices after sampling. The same goes for residual resampling.

6.6 Optimal resampling

Crisan and Lyons (1999) introduce another resampling scheme based on a branching process, which they show to be optimal in some sense. However, their algorithm is not widely used in practice because it is much more complicated to implement than alternatives like systematic resampling which perform just as well empirically, and share some of its optimality properties (Bain and Crisan, 2008).

6.7 Genealogies of SMC algorithms with residual resampling

Similarly to Section 5.3, here we attempt to characterise the genealogies induced in the case of residual resampling. Convergence is not yet proved in this case, but the steps taken so far are summarised here, with full details in Appendix B.

The central quantity is again the coalescence rate, which we will now denote by $c_N^m(t)$ and $c_N^r(t)$ for multinomial and residual resampling respectively. The expected coalescence rate can be written:

$$\mathbb{E}[c_N^r(t)|\mathcal{F}_{t-1}] = \mathbb{E}[c_N^m(t)|\mathcal{F}_{t-1}]\left(1+\frac{1}{N-1}\right) - \frac{1}{(N)_2}\mathbb{E}\left[\frac{\sum_{i=1}^N\left(Nw_t^{(i)}-\lfloor Nw_t^{(i)}\rfloor\right)\left(Nw_t^{(i)}+\sum_{k\neq i}\lfloor Nw_t^{(k)}\rfloor\right)}{\sum_{j=1}^N\left(Nw_t^{(j)}-\lfloor Nw_t^{(j)}\rfloor\right)}\middle|\mathcal{F}_{t-1}\right]$$

Using that $\mathbb{E}[c_N^m(t)|\mathcal{F}_{t-1}] = O(N^{-1})$, since the subtracted term is non-negative, we have the upper bound

$$\mathbb{E}[c_N^r(t)|\mathcal{F}_{t-1}] \leq \mathbb{E}[c_N^m(t)|\mathcal{F}_{t-1}] + O(N^{-2}). \tag{9}$$

Before we can hope to verify the conditions of Koskela et al. (2018, Theorem 1) in this case, it remains to find similar bounds on $\mathbb{E}[D_N^r(t)|\mathcal{F}_{t-1}]$ and $\mathbb{E}[(c_N^r(t))^2|\mathcal{F}_{t-1}]$.

7 Discussion

We have successfully extended the result of Koskela et al. (2018) to show that conditional SMC genealogies also converge to the Kingman coalescent. The next step is to use numerical simulations to investigate whether this result provides a reasonable approximation in the finite-N regime that is of interest in practice.

Meanwhile, we are also looking into the influence of alternative resampling schemes on the asymptotic form of the coalescent. This is particularly interesting because these resampling schemes are intended to slow down coalescence, and more relevant than the multinomial case as these algorithms are widely used in practice.

The analyses here and in Koskela et al. (2018) are the first of their kind, and are a valuable contribution to the SMC literature because of the practical importance of ancestral degeneracy. We hope that this analysis may provide not only theoretically interesting results, but also directly benefit practitioners by providing a framework for optimising the design and parameters of SMC algorithms and providing a new view on the relative merits of existing methods.

A Proof of Corollary 1

In the derivation of (4) – (6) we will make extensive use of the formula for factorial moments of the multinomial distribution given in Mosimann (1962, p.67):

$$\mathbb{E}[(X_i)_a(X_j)_b] = (n)_{a+b}\, p_i^a p_j^b \tag{10}$$

where $(X_1, \dots X_k) \sim \text{Multinomial}(n, \mathbf{p})$. To apply this formula we need to write everything in terms of falling factorial powers. The required conversions are summarised in Table 1.

In standard SMC with multinomial resampling, the marginal offspring distributions, conditioned on the filtration $\mathcal{F}_{t-1}$ generated by the previous offspring counts, are

$$v_t^{(i)} \overset{d}{=} \text{Binomial}(N, w_t^{(i)}), \qquad i = 1, \dots, N$$

where $v_t^{(i)}$ is the number of offspring in generation $t+1$ of the ith particle in generation t, N is the number of particles and $w_t^{(i)}$ is the weight associated with the ith particle in generation t.

In conditional SMC we condition on the immortal trajectory surviving each resampling step. By exchangeability we can set without loss of generality that the immortal trajectory consists of particle 1 in each generation. At each resampling step, particle 1 must therefore choose particle 1 as its parent, while the remaining $N-1$ offspring are assigned multinomially to the N possible parents. The marginal offspring distributions are then

$$\begin{aligned}
\tilde{v}_t^{(1)} &\overset{d}{=} 1 + \text{Binomial}(N-1, w_t^{(1)}) \\
\tilde{v}_t^{(i)} &\overset{d}{=} \text{Binomial}(N-1, w_t^{(i)}), \qquad i = 2, \dots, N.
\end{aligned}$$

First let us consider the coalescence rate

$$c_N(t) := \frac{1}{(N)_2} \sum_{i=1}^{N} (v_t^{(i)})_2.$$

For standard SMC the expected value is, using the tower rule,

$$\mathbb{E}[c_N(t)|\mathcal{F}_{t-1}] = \frac{1}{(N)_2} \sum_{i=1}^{N} \mathbb{E}\left[\mathbb{E}[(v_t^{(i)})_2]|\mathcal{F}_{t-1}\right] = \frac{1}{(N)_2} \sum_{i=1}^{N} \mathbb{E}\left[(N)_2 (w_t^{(i)})^2 |\mathcal{F}_{t-1}\right] = \sum_{i=1}^{N} \mathbb{E}\left[(w_t^{(i)})^2 |\mathcal{F}_{t-1}\right]$$

as stated in Koskela et al. (2018, Remark 3). In the case of conditional SMC we separate the first term (corresponding to the immortal particle) from the sum to get

$$\begin{aligned}
\mathbb{E}[\tilde{c}_N(t)|\mathcal{F}_{t-1}] &= \frac{1}{(N)_2} \sum_{i=1}^{N} \mathbb{E}\left[(\tilde{v}_t^{(i)})_2 |\mathcal{F}_{t-1}\right] = \frac{1}{(N)_2} \mathbb{E}\left[(\tilde{v}_t^{(1)})_2 |\mathcal{F}_{t-1}\right] + \frac{1}{(N)_2} \sum_{i=2}^{N} \mathbb{E}\left[(\tilde{v}_t^{(i)})_2 |\mathcal{F}_{t-1}\right] \\
&= \frac{1}{(N)_2} \left\{ (N-1)_2 \mathbb{E}[(w_t^{(1)})^2 |\mathcal{F}_{t-1}] + 2(N-1)\mathbb{E}[w_t^{(1)}|\mathcal{F}_{t-1}] + \sum_{i=2}^{N} (N-1)_2 \mathbb{E}[(w_t^{(i)})^2 |\mathcal{F}_{t-1}] \right\} \\
&= \frac{(N-1)_2}{(N)_2} \sum_{i=1}^{N} \mathbb{E}[(w_t^{(i)})^2 |\mathcal{F}_{t-1}] + \frac{2(N-1)}{(N)_2} \mathbb{E}[w_t^{(1)}|\mathcal{F}_{t-1}] \\
&= \frac{N-2}{N} \mathbb{E}[c_N(t)|\mathcal{F}_{t-1}] + \frac{2}{N} \mathbb{E}[w_t^{(1)}|\mathcal{F}_{t-1}]
\end{aligned}$$

x	$=$	$(x)_1$
x^2	$=$	$(x)_2 + (x)_1$
x^3	$=$	$(x)_3 + 3(x)_2 + (x)_1$
x^4	$=$	$(x)_4 + 6(x)_3 + 7(x)_2 + (x)_1$
xy	$=$	$(x)_1(y)_1$
x^2y	$=$	$(x)_2(y)_1 + (x)_1(y)_1$
xy^2	$=$	$(x)_1(y)_2 + (x)_1(y)_1$
x^2y^2	$=$	$(x)_2(y)_2 + (x)_2(y)_1 + (x)_1(y)_2 + (x)_1(y)_1$
$(x+1)_2$	$=$	$(x)_2 + 2(x)_1$
$(x+1)^2$	$=$	$(x)_2 + 3(x)_1 + 1$
$(x+1)_2(x+1)$	$=$	$(x)_3 + 5(x)_2 + 4(x)_1$
$(x+1)_2^2$	$=$	$(x)_4 + 8(x)_3 + 14(x)_2 + 4(x)_1$

Table 1: Conversion of ordinary powers into falling factorial powers

which gives us (4).

An upper bound on the rate of super-binary mergers is given by

$$D_N(t) := \frac{1}{N(N)_2} \sum_{i=1}^{N} (v_t^{(i)})_2 \left(v_t^{(i)} + \frac{1}{N} \sum_{j \neq i} (v_t^{(j)})^2 \right).$$

In the standard case this quantity has expectation

$$\begin{aligned}\mathbb{E}[D_N(t)|\mathcal{F}_{t-1}] = \frac{1}{N(N)_2} \sum_{i=1}^{N} \Big\{ (N)_3 \mathbb{E}[(w_t^{(i)})^3|\mathcal{F}_{t-1}] + 2(N)_2 \mathbb{E}[(w_t^{(i)})^2|\mathcal{F}_{t-1}] \Big\} \\ + \frac{1}{N^2(N)_2} \sum_{i=1}^{N} \sum_{j \neq i} \Big\{ (N)_4 \mathbb{E}[(w_t^{(i)})^2 (w_t^{(j)})^2|\mathcal{F}_{t-1}] + (N)_3 \mathbb{E}[(w_t^{(i)})^2 w_t^{(j)}|\mathcal{F}_{t-1}] \Big\}\end{aligned}$$

while in the conditional case, again separating the terms involving particle 1,

$$\begin{aligned}\tilde{D}_N(t) &= \frac{1}{N(N)_2} (\tilde{v}_t^{(1)})_2 \left(\tilde{v}_t^{(1)} + \frac{1}{N} \sum_{j \neq 1} (\tilde{v}_t^{(j)})^2 \right) + \frac{1}{N(N)_2} \sum_{i \neq 1} (\tilde{v}_t^{(i)})_2 \left(\tilde{v}_t^{(i)} + \frac{1}{N} (\tilde{v}_t^{(1)})^2 + \frac{1}{N} \sum_{1 \neq j \neq i} (\tilde{v}_t^{(j)})^2 \right) \\ &= \frac{1}{N(N)_2} \left\{ (\tilde{v}_t^{(1)})_2 v_t^{(1)} + \frac{1}{N} \sum_{j \neq 1} (\tilde{v}_t^{(1)})_2 (\tilde{v}_t^{(j)})^2 + \frac{1}{N} \sum_{i \neq 1} (\tilde{v}_t^{(i)})_2 (\tilde{v}_t^{(1)})^2 \right\} \\ &\quad + \frac{1}{N(N)_2} \sum_{i \neq 1} \left\{ (\tilde{v}_t^{(i)})_2 \tilde{v}_t^{(i)} + \frac{1}{N} \sum_{1 \neq j \neq i} (\tilde{v}_t^{(i)})_2 (\tilde{v}_t^{(j)})^2 \right\} \\ &= \frac{1}{N(N)_2} \left\{ (\tilde{v}_t^{(1)})_2 \tilde{v}_t^{(1)} + \frac{1}{N} \sum_{i \neq 1} \left((\tilde{v}_t^{(1)})_2 (\tilde{v}_t^{(i)})^2 + (\tilde{v}_t^{(i)})_2 (\tilde{v}_t^{(1)})^2 \right) \right\} \\ &\quad + \frac{1}{N(N)_2} \sum_{i \neq 1} \left\{ (\tilde{v}_t^{(i)})_3 + 2(\tilde{v}_t^{(i)})_2 + \frac{1}{N} \sum_{1 \neq j \neq i} \left((\tilde{v}_t^{(i)})_2 (\tilde{v}_t^{(j)})_2 + (\tilde{v}_t^{(i)})_2 \tilde{v}_t^{(j)} \right) \right\}\end{aligned}$$

and so by applying the moments from (10) and Table 1 we find the expectation

$$
\begin{aligned}
&\mathbb{E}[\tilde{D}_N(t)|\mathcal{F}_{t-1}] = \\
&= \frac{1}{N(N)_2}\left\{(N-1)_3\mathbb{E}[(w_t^{(1)})^3|\mathcal{F}_{t-1}] + 5(N-1)_2\mathbb{E}[(w_t^{(1)})^2|\mathcal{F}_{t-1}] + 4(N-1)\mathbb{E}[w_t^{(1)}|\mathcal{F}_{t-1}]\right\} \\
&\quad + \frac{1}{N^2(N)_2}\sum_{i=2}^{N}\Big\{2(N-1)_4\mathbb{E}[(w_t^{(1)})^2(w_t^{(i)})^2|\mathcal{F}_{t-1}] + (N-1)_3\mathbb{E}[(w_t^{(1)})^2 w_t^{(i)}|\mathcal{F}_{t-1}] \\
&\qquad + 5(N-1)_3\mathbb{E}[w_t^{(1)}(w_t^{(i)})^2|\mathcal{F}_{t-1}] + 2(N-1)_2\mathbb{E}[w_t^{(1)}w_t^{(i)}|\mathcal{F}_{t-1}] + (N-1)_2\mathbb{E}[(w_t^{(i)})^2|\mathcal{F}_{t-1}]\Big\} \\
&\quad + \frac{1}{N(N)_2}\sum_{i=2}^{N}\left\{(N-1)_3\mathbb{E}[(w_t^{(i)})^3|\mathcal{F}_{t-1}] + 2(N-1)_2\mathbb{E}[(w_t^{(i)})^2|\mathcal{F}_{t-1}]\right\} \\
&\quad + \frac{1}{N^2(N)_2}\sum_{i=2}^{N}\sum_{1\neq j\neq i}\left\{(N-1)_4\mathbb{E}[(w_t^{(i)})^2(w_t^{(j)})^2|\mathcal{F}_{t-1}] + (N-1)_3\mathbb{E}[(w_t^{(i)})^2 w_t^{(j)}|\mathcal{F}_{t-1}]\right\} \\
&= \frac{1}{N(N)_2}\sum_{i=1}^{N}\left\{(N-1)_3\mathbb{E}[(w_t^{(i)})^3|\mathcal{F}_{t-1}] + 2(N-1)_2\mathbb{E}[(w_t^{(i)})^2|\mathcal{F}_{t-1}]\right\} \\
&\quad + \frac{1}{N^2(N)_2}\sum_{i=1}^{N}\sum_{j\neq i}\left\{(N-1)_4\mathbb{E}[(w_t^{(i)})^2(w_t^{(j)})^2|\mathcal{F}_{t-1}] + (N-1)_3\mathbb{E}[(w_t^{(i)})^2 w_t^{(j)}|\mathcal{F}_{t-1}]\right\} \\
&\quad + \frac{1}{N(N)_2}\left\{3(N-1)_2\mathbb{E}[(w_t^{(1)})^2|\mathcal{F}_{t-1}] + 4(N-1)\mathbb{E}[w_t^{(1)}|\mathcal{F}_{t-1}]\right\} \\
&\quad + \frac{1}{N^2(N)_2}\sum_{i=2}^{N}\left\{4(N-1)_3\mathbb{E}[w_t^{(1)}(w_t^{(i)})^2|\mathcal{F}_{t-1}] + 2(N-1)_2\mathbb{E}[w_t^{(1)}w_t^{(i)}|\mathcal{F}_{t-1}] + (N-1)_2\mathbb{E}[(w_t^{(i)})^2|\mathcal{F}_{t-1}]\right\} \\
&\leq \mathbb{E}[D_N(t)|\mathcal{F}_{t-1}] + \frac{1}{N(N)_2}\left\{3(N-1)_2\mathbb{E}[(w_t^{(1)})^2|\mathcal{F}_{t-1}] + 4(N-1)\mathbb{E}[w_t^{(1)}|\mathcal{F}_{t-1}]\right\} \\
&\quad + \frac{1}{N^2(N)_2}\sum_{i=2}^{N}\left\{4(N-1)_3\mathbb{E}[w_t^{(1)}(w_t^{(i)})^2|\mathcal{F}_{t-1}] + 2(N-1)_2\mathbb{E}[w_t^{(1)}w_t^{(i)}|\mathcal{F}_{t-1}] + (N-1)_2\mathbb{E}[(w_t^{(i)})^2|\mathcal{F}_{t-1}]\right\} \\
&\leq \mathbb{E}[D_N(t)|\mathcal{F}_{t-1}] + \frac{3}{N}\mathbb{E}[(w_t^{(1)})^2|\mathcal{F}_{t-1}] + \frac{4}{N^2}\mathbb{E}[w_t^{(1)}|\mathcal{F}_{t-1}] \\
&\quad + \frac{4}{N}\sum_{i=2}^{N}\mathbb{E}[w_t^{(1)}(w_t^{(i)})^2|\mathcal{F}_{t-1}] + \frac{2}{N^2}\sum_{i=2}^{N}\mathbb{E}[w_t^{(1)}w_t^{(i)}|\mathcal{F}_{t-1}] + \frac{1}{N^2}\sum_{i=2}^{N}\mathbb{E}[(w_t^{(i)})^2|\mathcal{F}_{t-1}]
\end{aligned}
$$

The second line of the first equality relies on multiplying the relevant terms in Table 1. For the second equality we recombine the terms in particle 1 into the sum. The inequalities follow by bounding e.g. $N-1$ by N, and identifying the first two lines with $\mathbb{E}[D_N(t)|\mathcal{F}_{t-1}]$. This gives us the inequality (5).

We also need control of the squared coalescence rate:

$$
c_N(t)^2 = \frac{1}{(N)_2^2}\left(\sum_{i=1}^{N}(v_t^{(i)})_2\right)^2 = \frac{1}{(N)_2^2}\left\{\sum_{i=1}^{N}\mathbb{E}[(v_t^{(i)})_2^2] + \sum_{i=1}^{N}\sum_{j\neq i}\mathbb{E}[(v_t^{(i)})_2(v_t^{(j)})_2]\right\}
$$

A bound on its expected value is proved in Koskela et al. (2018), but here we will use a different, more explicit

bound to allow direct comparison between the standard and conditional cases. For standard SMC we have:

$$\begin{aligned}\mathbb{E}[c_N(t)^2|\mathcal{F}_{t-1}] &= \frac{1}{(N)_2^2}\left\{(N)_4\sum_{i=1}^{N}\mathbb{E}[(w_t^{(i)})^4|\mathcal{F}_{t-1}] + 4(N)_3\sum_{i=1}^{N}\mathbb{E}[(w_t^{(i)})^3|\mathcal{F}_{t-1}] + 2(N)_2\sum_{i=1}^{N}\mathbb{E}[(w_t^{(i)})^2|\mathcal{F}_{t-1}]\right\} \\ &\quad + \frac{1}{(N)_2^2}(N)_4\sum_{i=1}^{N}\sum_{j\neq i}\mathbb{E}[(w_t^{(i)})^2(w_t^{(j)})^2|\mathcal{F}_{t-1}] \\ &= \frac{1}{(N)_2}\left\{(N-2)_2\sum_{i=1}^{N}\mathbb{E}[(w_t^{(i)})^4|\mathcal{F}_{t-1}] + 4(N-2)\sum_{i=1}^{N}\mathbb{E}[(w_t^{(i)})^3|\mathcal{F}_{t-1}] + 2\sum_{i=1}^{N}\mathbb{E}[(w_t^{(i)})^2|\mathcal{F}_{t-1}]\right. \\ &\qquad \left. + (N-2)_2\sum_{i=1}^{N}\sum_{j\neq i}\mathbb{E}[(w_t^{(i)})^2(w_t^{(j)})^2|\mathcal{F}_{t-1}]\right\}\end{aligned}$$

For conditional SMC, we again separate the terms involving particle 1:

$$\begin{aligned}\tilde{c}_N(t)^2 &= \frac{1}{(N)_2^2}\left\{\sum_{i=1}^{N}\mathbb{E}[(\tilde{v}_t^{(i)})_2^2] + \sum_{i=1}^{N}\sum_{j\neq i}\mathbb{E}[(\tilde{v}_t^{(i)})_2(\tilde{v}_t^{(j)})_2]\right\} \\ &= \frac{1}{(N)_2^2}\left\{\sum_{i=2}^{N}\mathbb{E}[(\tilde{v}_t^{(i)})_2^2] + \mathbb{E}[(\tilde{v}_t^{(1)})_2^2] + \sum_{i=2}^{N}\sum_{1\neq j\neq i}\mathbb{E}[(\tilde{v}_t^{(i)})_2(\tilde{v}_t^{(j)})_2] + 2\sum_{i=2}^{N}\mathbb{E}[(\tilde{v}_t^{(1)})_2(\tilde{v}_t^{(i)})_2]\right\}\end{aligned}$$

and then use the same techniques as for $\tilde{D}_N(t)$ to calculate the expectation:

$$
\begin{aligned}
&\mathbb{E}[\tilde{c}_N(t)^2|\mathcal{F}_{t-1}] = \\
&= \frac{1}{(N)_2^2}\left\{(N-1)_4\sum_{i=2}^{N}\mathbb{E}[(w_t^{(i)})^4|\mathcal{F}_{t-1}] + 4(N-1)_3\sum_{i=2}^{N}\mathbb{E}[(w_t^{(i)})^3|\mathcal{F}_{t-1}] + 2(N-1)_2\sum_{i=2}^{N}\mathbb{E}[(w_t^{(i)})^2|\mathcal{F}_{t-1}]\right\} \\
&\quad + \frac{1}{(N)_2^2}\left\{(N-1)_4\mathbb{E}[(w_t^{(1)})^4|\mathcal{F}_{t-1}] + 8(N-1)_3\mathbb{E}[(w_t^{(1)})^3|\mathcal{F}_{t-1}] + 14(N-1)_2\mathbb{E}[(w_t^{(1)})^2|\mathcal{F}_{t-1}]\right\} \\
&\quad + \frac{1}{(N)_2^2}4(N-1)\mathbb{E}[w_t^{(1)}|\mathcal{F}_{t-1}] + \frac{1}{(N)_2^2}(N-1)_4\sum_{i=2}^{N}\sum_{1\neq j\neq i}\mathbb{E}[(w_t^{(i)})^2(w_t^{(j)})^2|\mathcal{F}_{t-1}] \\
&\quad + \frac{2}{(N)_2^2}\sum_{i=2}^{N}\left((N-1)_4\mathbb{E}[(w_t^{(1)})^2(w_t^{(i)})^2|\mathcal{F}_{t-1}] + 2(N-1)_3\mathbb{E}[w_t^{(1)}(w_t^{(i)})^2|\mathcal{F}_{t-1}]\right) \\
&= \frac{1}{(N)_2^2}\left\{(N-1)_4\sum_{i=1}^{N}\mathbb{E}[(w_t^{(i)})^4|\mathcal{F}_{t-1}] + 4(N-1)_3\sum_{i=1}^{N}\mathbb{E}[(w_t^{(i)})^3|\mathcal{F}_{t-1}] + 2(N-1)_2\sum_{i=1}^{N}\mathbb{E}[(w_t^{(i)})^2|\mathcal{F}_{t-1}]\right\} \\
&\quad + \frac{(N-1)_4}{(N)_2^2}\sum_{i=1}^{N}\sum_{j\neq i}\mathbb{E}[(w_t^{(i)})^2(w_t^{(j)})^2|\mathcal{F}_{t-1}] \\
&\quad + \frac{1}{(N)_2^2}\left\{4(N-1)_3\mathbb{E}[(w_t^{(1)})^3|\mathcal{F}_{t-1}] + 12(N-1)_2\mathbb{E}[(w_t^{(1)})^2|\mathcal{F}_{t-1}] + 4(N-1)\mathbb{E}[w_t^{(1)}|\mathcal{F}_{t-1}]\right\} \\
&\quad + \frac{1}{(N)_2^2}\left\{4(N-1)_3\sum_{i=2}^{N}\mathbb{E}[w_t^{(1)}(w_t^{(i)})^2|\mathcal{F}_{t-1}]\right\} \\
&\leq \mathbb{E}[c_N(t)^2|\mathcal{F}_{t-1}] + \frac{1}{(N)_2^2}\left\{4(N-1)_3\mathbb{E}[(w_t^{(1)})^3|\mathcal{F}_{t-1}] + 12(N-1)_2\mathbb{E}[(w_t^{(1)})^2|\mathcal{F}_{t-1}]\right\} \\
&\quad + \frac{1}{(N)_2^2}\left\{4(N-1)\mathbb{E}[w_t^{(1)}|\mathcal{F}_{t-1}] + 4(N-1)_3\sum_{i=2}^{N}\mathbb{E}[w_t^{(1)}(w_t^{(i)})^2|\mathcal{F}_{t-1}]\right\} \\
&\leq \mathbb{E}[c_N(t)^2|\mathcal{F}_{t-1}] + \frac{4}{N}\mathbb{E}[(w_t^{(1)})^3|\mathcal{F}_{t-1}] + \frac{12}{N^2}\mathbb{E}[(w_t^{(1)})^2|\mathcal{F}_{t-1}] + \frac{4}{N(N)_2}\mathbb{E}[w_t^{(1)}|\mathcal{F}_{t-1}] \\
&\quad + \frac{4}{N}\sum_{i=2}^{N}\mathbb{E}[w_t^{(1)}(w_t^{(i)})^2|\mathcal{F}_{t-1}]
\end{aligned}
$$

The conditions (18) and (19) of Koskela et al. (2018, Lemma 3) give us control over the weights so that we have $w_t^{(i)} = O(1)$ for all i. Under these conditions, in the limit as $N \to \infty$, the three modified expectations derived above simplify to:

$$
\begin{aligned}
&\mathbb{E}[\tilde{c}_N(t)|\mathcal{F}_{t-1}] \leq \mathbb{E}[c_N(t)|\mathcal{F}_{t-1}] + O(N^{-2}) \\
&\mathbb{E}[\tilde{D}_N(t)|\mathcal{F}_{t-1}] \leq \mathbb{E}[D_N(t)|\mathcal{F}_{t-1}] + O(N^{-3}) \\
&\mathbb{E}[\tilde{c}_N(t)^2|\mathcal{F}_{t-1}] \leq \mathbb{E}[c_N(t)^2|\mathcal{F}_{t-1}] + O(N^{-3})
\end{aligned}
$$

This shows that each of these quantities for conditional SMC is bounded above by the corresponding standard SMC quantity, plus some vanishing error term. This will allow us to apply Koskela et al. (2018, Theorem 1), as we will show in the following.

Next we apply the result of Koskela et al. (2018, Lemma 3), so that for our modified quantity $\tilde{c}_N(t)$ we

have the upper bound:

$$\begin{aligned}\mathbb{E}[\tilde{c}_N(t)|\mathcal{F}_{t-1}] &= \frac{N-2}{N}\mathbb{E}[c_N(t)|\mathcal{F}_{t-1}] + \frac{2}{N}\mathbb{E}[w_t^{(1)}|\mathcal{F}_{t-1}] \\ &\leq \mathbb{E}[c_N(t)|\mathcal{F}_{t-1}] + \frac{2}{N}\mathbb{E}[w_t^{(1)}|\mathcal{F}_{t-1}] \\ &\leq \frac{a^4}{N\varepsilon^4} + \frac{2}{N}\mathbb{E}[w_t^{(1)}|\mathcal{F}_{t-1}] \\ &= \frac{a^4}{N\varepsilon^4} + O(N^{-2}) \end{aligned} \tag{11}$$

and lower bound:

$$\begin{aligned}\mathbb{E}[\tilde{c}_N(t)|\mathcal{F}_{t-1}] &= \frac{N-2}{N}\mathbb{E}[c_N(t)|\mathcal{F}_{t-1}] + \frac{2}{N}\mathbb{E}[w_t^{(1)}|\mathcal{F}_{t-1}] \\ &\geq \frac{N-2}{N}\frac{\varepsilon^4}{Na^4} + O(N^{-2}) \\ &= \frac{\varepsilon^4}{Na^4} - \frac{2\varepsilon^4}{N^2a^4} + O(N^{-2}) \\ &= \frac{\varepsilon^4}{Na^4} + O(N^{-2}) \end{aligned} \tag{12}$$

corresponding to (22) in Koskela et al. (2018), except for the addition of a vanishing error term. Furthermore, we obtain for the other quantities (where the constant C may change from one line to the next):

$$\begin{aligned}\mathbb{E}[\tilde{D}_N(t)|\mathcal{F}_{t-1}] &\leq \mathbb{E}[D_N(t)|\mathcal{F}_{t-1}] + O(N^{-3}) \\ &\leq \frac{C}{N}\mathbb{E}[c_N(t)|\mathcal{F}_{t-1}] + O(N^{-3}) \\ &= \frac{C}{N}\mathbb{E}[\tilde{c}_N(t)|\mathcal{F}_{t-1}] + O(N^{-3}) \end{aligned} \tag{13}$$

and

$$\begin{aligned}\mathbb{E}[\tilde{c}_N(t)^2|\mathcal{F}_{t-1}] &\leq \mathbb{E}[c_N(t)^2|\mathcal{F}_{t-1}] + O(N^{-3}) \\ &\leq \frac{C}{N}\mathbb{E}[c_N(t)|\mathcal{F}_{t-1}] + O(N^{-3}) \\ &= \frac{C}{N}\mathbb{E}[\tilde{c}_N(t)|\mathcal{F}_{t-1}] + O(N^{-3}) \end{aligned} \tag{14}$$

according to equations (20) and (21) in Koskela et al. (2018), again with additional error terms.

Now let us define the time-scaling:

$$\tilde{\tau}_N(t) := \min\left\{ s \geq 1 : \sum_{r=1}^{s} \tilde{c}_N(r) \geq t \right\}$$

which is a generalised inverse of $\tilde{c}_N(t)$ and thus satisfies the property:

$$t - s - 1 \leq \sum_{r=\tilde{\tau}_N(s)+1}^{\tilde{\tau}_N(t)} \tilde{c}_N(r) \leq t - s + 1. \tag{15}$$

We are finally ready to verify the conditions of Koskela et al. (2018, Theorem 1). The conditions are the following.

(Standing Assumption) The conditional distribution of parental indices $a_t^{(1:N)}$ given offspring counts $v_t^{(1:N)}$ is uniform over all valid assignments.

(A) $\lim_{N\to\infty} \mathbb{E}\left[\sum_{r=\tilde{\tau}_N(s)+1}^{\tilde{\tau}_N(t)} \tilde{D}_N(r)\right] = 0$

(B) $\lim_{N\to\infty} \mathbb{E}[\tilde{c}_N(t)] = 0$

(C) $\lim_{N\to\infty} \mathbb{E}\left[\sum_{r=\tilde{\tau}_N(s)+1}^{\tilde{\tau}_N(t)} \tilde{c}_N(r)^2\right] = 0$

(D) $\mathbb{E}[\tilde{\tau}_N(t) - \tilde{\tau}_N(s)] \leq C_{t,s}N$

These five conditions are verified below.

(Standing Assumption) This holds by the exchangeability of offspring assignments arising from Algorithm 2.

(B) Using (11) and applying the tower rule, we find

$$\mathbb{E}[\tilde{c}_N(t)] = \mathbb{E}[\mathbb{E}[\tilde{c}_N(t)|\mathcal{F}_{t-1}]] \leq \frac{a^4}{N\varepsilon^4} + O(N^{-2}) \overset{N\to\infty}{\longrightarrow} 0$$

(C) Using Koskela et al. (2018, Lemma 2) along with (14) and the upper bound in (15),

$$\begin{aligned}
\mathbb{E}\left[\sum_{r=\tilde{\tau}_N(s)+1}^{\tilde{\tau}_N(t)} \tilde{c}_N(r)^2\right] &= \mathbb{E}\left[\sum_{r=\tilde{\tau}_N(s)+1}^{\tilde{\tau}_N(t)} \mathbb{E}[\tilde{c}_N(r)^2|\mathcal{F}_{t-1}]\right] \leq \mathbb{E}\left[\sum_{r=\tilde{\tau}_N(s)+1}^{\tilde{\tau}_N(t)} \left(\frac{C}{N}\mathbb{E}[\tilde{c}_N(t)|\mathcal{F}_{t-1}] + O(N^{-3})\right)\right] \\
&= \frac{C}{N}\mathbb{E}\left[\sum_{r=\tilde{\tau}_N(s)+1}^{\tilde{\tau}_N(t)} \mathbb{E}[\tilde{c}_N(t)|\mathcal{F}_{t-1}]\right] + O(N^{-2}) = \frac{C}{N}\mathbb{E}\left[\sum_{r=\tilde{\tau}_N(s)+1}^{\tilde{\tau}_N(t)} \tilde{c}_N(r)\right] + O(N^{-2}) \\
&\leq \frac{C}{N}(t-s+1) + O(N^{-2}) \overset{N\to\infty}{\longrightarrow} 0
\end{aligned}$$

(A) The above calculation replacing (14) with (13) yields

$$\begin{aligned}
\mathbb{E}\left[\sum_{r=\tilde{\tau}_N(s)+1}^{\tilde{\tau}_N(t)} \tilde{D}_N(r)\right] &= \mathbb{E}\left[\sum_{r=\tilde{\tau}_N(s)+1}^{\tilde{\tau}_N(t)} \mathbb{E}[\tilde{D}_N(r)|\mathcal{F}_{t-1}]\right] \\
&\leq \mathbb{E}\left[\sum_{r=\tilde{\tau}_N(s)+1}^{\tilde{\tau}_N(t)} \left(\frac{C}{N}\mathbb{E}[\tilde{c}_N(t)|\mathcal{F}_{t-1}] + O(N^{-3})\right)\right] \overset{N\to\infty}{\longrightarrow} 0
\end{aligned}$$

(D) Using (12), the upper bound in (15) and Koskela et al. (2018, Lemma 2),

$$\begin{aligned}
\mathbb{E}[\tilde{\tau}_N(t) - \tilde{\tau}_N(s)] &= \mathbb{E}\left[\sum_{r=\tilde{\tau}_N(s)+1}^{\tilde{\tau}_N(t)} 1\right] = \mathbb{E}\left[\sum_{r=\tilde{\tau}_N(s)+1}^{\tilde{\tau}_N(t)} \frac{\mathbb{E}[\tilde{c}_N(r)|\mathcal{F}_{t-1}]}{\mathbb{E}[\tilde{c}_N(r)|\mathcal{F}_{t-1}]}\right] \leq \mathbb{E}\left[\sum_{r=\tilde{\tau}_N(s)+1}^{\tilde{\tau}_N(t)} \frac{\mathbb{E}[\tilde{c}_N(r)|\mathcal{F}_{t-1}]}{\frac{\varepsilon^4}{Na^4} + O(N^{-2})}\right] \\
&= \frac{1}{\frac{\varepsilon^4}{Na^4} + O(N^{-2})}\mathbb{E}\left[\sum_{r=\tilde{\tau}_N(s)+1}^{\tilde{\tau}_N(t)} \mathbb{E}[\tilde{c}_N(r)|\mathcal{F}_{t-1}]\right] = \frac{1}{\frac{\varepsilon^4}{Na^4} + O(N^{-2})}\mathbb{E}\left[\sum_{r=\tilde{\tau}_N(s)+1}^{\tilde{\tau}_N(t)} \tilde{c}_N(r)\right] \\
&\leq \frac{t-s+1}{\frac{\varepsilon^4}{Na^4} + O(N^{-2})} = \frac{(t-s+1)a^4N}{\varepsilon^4 + O(N^{-1})} = (t-s+1)\frac{a^4}{\varepsilon^4}N + O(1)
\end{aligned}$$

where the last equality follows by a Taylor expansion of $(\frac{\varepsilon^4}{Na^4} + O(N^{-2}))^{-1}$.
Similarly we derive a lower bound using (11), the lower bound in (15) and Koskela et al. (2018, Lemma

2):

$$\mathbb{E}[\tilde{\tau}_N(t) - \tilde{\tau}_N(s)] = \mathbb{E}\left[\sum_{r=\tilde{\tau}_N(s)+1}^{\tilde{\tau}_N(t)} 1\right] = \mathbb{E}\left[\sum_{r=\tilde{\tau}_N(s)+1}^{\tilde{\tau}_N(t)} \frac{\mathbb{E}[\tilde{c}_N(r)|\mathcal{F}_{t-1}]}{\mathbb{E}[\tilde{c}_N(r)|\mathcal{F}_{t-1}]}\right] \geq \mathbb{E}\left[\sum_{r=\tilde{\tau}_N(s)+1}^{\tilde{\tau}_N(t)} \frac{\mathbb{E}[\tilde{c}_N(r)|\mathcal{F}_{t-1}]}{\frac{a^4}{N\varepsilon^4} + O(N^{-2})}\right]$$
$$= \frac{1}{\frac{a^4}{N\varepsilon^4} + O(N^{-2})}\mathbb{E}\left[\sum_{r=\tilde{\tau}_N(s)+1}^{\tilde{\tau}_N(t)} \mathbb{E}[\tilde{c}_N(r)|\mathcal{F}_{t-1}]\right] = \frac{1}{\frac{a^4}{N\varepsilon^4} + O(N^{-2})}\mathbb{E}\left[\sum_{r=\tilde{\tau}_N(s)+1}^{\tilde{\tau}_N(t)} \tilde{c}_N(r)\right]$$
$$\geq \frac{t-s-1}{\frac{a^4}{N\varepsilon^4} + O(N^{-2})} = \frac{(t-s-1)\varepsilon^4 N}{a^4 + O(N^{-1})} = (t-s-1)\frac{\varepsilon^4}{a^4}N + O(1)$$

Therefore we have as required

$$\mathbb{E}[\tilde{\tau}_N(t) - \tilde{\tau}_N(s)] \sim C_{t,s}N$$

as $N \to \infty$.

This concludes the proof of Corollary 1. □

B Derivation of (9)

In residual resampling offspring counts are sampled according to:

$$v_t^{(i)} = \lfloor N w_t^{(i)} \rfloor + X_i$$
$$X_i \sim \text{Multinomial}(N-k, (\bar{w}_t^{(1)}, \ldots, \bar{w}_t^{(N)}))$$

where $k := \sum_{i=1}^N \lfloor N w_t^{(i)} \rfloor$ is the number of offspring assigned deterministically, and $\bar{w}_t^{(i)} := \frac{N w_t^{(i)} - \lfloor N w_t^{(i)} \rfloor}{N-k}$ are the residual weights. Let us also define the residuals $r_i := N w_t^{(i)} - \lfloor N w_t^{(i)} \rfloor$. This also implies $\sum_{i=1}^N r_i = N-k$.

The coalescence rate is defined as

$$c_N(t) := \frac{1}{(N)_2}\sum_{i=1}^N (v_t^{(i)})_2.$$

We use $c_N^m(t)$ and $c_N^r(t)$ to denote the coalescence rates with multinomial and residual resampling respectively. The expectation then comes out as

$$\mathbb{E}[(v_t^{(i)})_2|\mathcal{F}_{t-1}] = \mathbb{E}[(v_t^{(i)})^2|\mathcal{F}_{t-1}] - \mathbb{E}[v_t^{(i)}|\mathcal{F}_{t-1}]$$
$$= \mathbb{E}[\lfloor N w_t^{(i)} \rfloor^2|\mathcal{F}_{t-1}] + 2\mathbb{E}[\lfloor N w_t^{(i)} \rfloor r_i|\mathcal{F}_{t-1}] + \mathbb{E}\left[r_i\left(1 - \frac{r_i}{N-k} + r_i\right)|\mathcal{F}_{t-1}\right] - \mathbb{E}[N w_t^{(i)}|\mathcal{F}_{t-1}]$$
$$= \mathbb{E}[\lfloor N w_t^{(i)} \rfloor^2|\mathcal{F}_{t-1}] - \mathbb{E}[\lfloor N w_t^{(i)} \rfloor|\mathcal{F}_{t-1}] + 2\mathbb{E}[\lfloor N w_t^{(i)} \rfloor r_i|\mathcal{F}_{t-1}] + \mathbb{E}\left[r_i^2\left(1 - \frac{1}{N-k}\right)|\mathcal{F}_{t-1}\right]$$
$$= \mathbb{E}[(N w_t^{(i)})^2|\mathcal{F}_{t-1}] - \mathbb{E}[\lfloor N w_t^{(i)} \rfloor|\mathcal{F}_{t-1}] - \mathbb{E}\left[\frac{r_i^2}{N-k}|\mathcal{F}_{t-1}\right]$$

so we get

$$\begin{aligned}
\mathbb{E}[c_N^r(t)|\mathcal{F}_{t-1}] &= \frac{1}{(N)_2}\sum_{i=1}^{N}\mathbb{E}[(v_t^{(i)})_2|\mathcal{F}_{t-1}] \\
&= \frac{N}{N-1}\sum_{i=1}^{N}\mathbb{E}[(w_t^{(i)})^2|\mathcal{F}_{t-1}] - \frac{1}{(N)_2}\sum_{i=1}^{N}\mathbb{E}\left[\frac{r_i^2}{N-k}|\mathcal{F}_{t-1}\right] - \frac{1}{(N)_2}\mathbb{E}[k|\mathcal{F}_{t-1}] \\
&= \mathbb{E}[c_N^m(t)|\mathcal{F}_{t-1}]\left(1+\frac{1}{N-1}\right) - \frac{1}{(N)_2}\mathbb{E}\left[\frac{\sum_{i=1}^{N}(Nw_t^{(i)} - \lfloor Nw_t^{(i)}\rfloor)^2}{\sum_{j=1}^{N}(Nw_t^{(j)} - \lfloor Nw_t^{(j)}\rfloor)}\middle|\mathcal{F}_{t-1}\right] \\
&\quad - \frac{1}{(N)_2}\mathbb{E}\left[\sum_{i=1}^{N}\lfloor Nw_t^{(i)}\rfloor\middle|\mathcal{F}_{t-1}\right] \qquad (16)
\end{aligned}$$

We can write this in a different form by combining the second and third terms of (16):

$$\begin{aligned}
&-\frac{1}{(N)_2}\mathbb{E}\left[\frac{\sum_{i=1}^{N}(Nw_t^{(i)} - \lfloor Nw_t^{(i)}\rfloor)^2}{\sum_{j=1}^{N}(Nw_t^{(j)} - \lfloor Nw_t^{(j)}\rfloor)}|\mathcal{F}_{t-1}\right] - \frac{1}{(N)_2}\mathbb{E}\left[\sum_{k=1}^{N}\lfloor Nw_t^{(k)}\rfloor\middle|\mathcal{F}_{t-1}\right] \\
&=: -\frac{1}{(N)_2}\mathbb{E}[A|\mathcal{F}_{t-1}]
\end{aligned}$$

Then

$$\begin{aligned}
A &= \frac{\sum_{i=1}^{N}(Nw_t^{(i)} - \lfloor Nw_t^{(i)}\rfloor)^2}{\sum_{j=1}^{N}(Nw_t^{(j)} - \lfloor Nw_t^{(j)}\rfloor)} + \sum_{k=1}^{N}\lfloor Nw_t^{(k)}\rfloor \\
&= \frac{\sum_{i=1}^{N}(Nw_t^{(i)} - \lfloor Nw_t^{(i)}\rfloor)^2 + \sum_{i=1}^{N}\sum_{k=1}^{N}\lfloor Nw_t^{(k)}\rfloor(Nw_t^{(i)} - \lfloor Nw_t^{(i)}\rfloor)}{\sum_{j=1}^{N}(Nw_t^{(j)} - \lfloor Nw_t^{(j)}\rfloor)} \\
&=: \frac{A'}{\sum_{j=1}^{N}(Nw_t^{(j)} - \lfloor Nw_t^{(j)}\rfloor)}
\end{aligned}$$

Then

$$\begin{aligned}
A' &= \sum_{i=1}^{N}(Nw_t^{(i)} - \lfloor Nw_t^{(i)}\rfloor)^2 + \sum_{i=1}^{N}\sum_{k=1}^{N}\lfloor Nw_t^{(k)}\rfloor(Nw_t^{(i)} - \lfloor Nw_t^{(i)}\rfloor) \\
&= \sum_{i=1}^{N}\left\{\left(Nw_t^{(i)} - \lfloor Nw_t^{(i)}\rfloor\right)^2 + \lfloor Nw_t^{(i)}\rfloor\left(Nw_t^{(i)} - \lfloor Nw_t^{(i)}\rfloor\right) + \sum_{k\neq i}\lfloor Nw_t^{(k)}\rfloor\left(Nw_t^{(i)} - \lfloor Nw_t^{(i)}\rfloor\right)\right\} \\
&= \sum_{i=1}^{N}\left\{(Nw_t^{(i)})^2 - Nw_t^{(i)}\lfloor Nw_t^{(i)}\rfloor + \sum_{k\neq i}\lfloor Nw_t^{(k)}\rfloor\left(Nw_t^{(i)} - \lfloor Nw_t^{(i)}\rfloor\right)\right\} \\
&= \sum_{i=1}^{N}\left\{\left(Nw_t^{(i)} - \lfloor Nw_t^{(i)}\rfloor\right)\left(Nw_t^{(i)} + \sum_{k\neq i}\lfloor Nw_t^{(k)}\rfloor\right)\right\}
\end{aligned}$$

So we have

$$\begin{aligned}
\mathbb{E}[c_N^r(t)|\mathcal{F}_{t-1}] &= \mathbb{E}[c_N^m(t)|\mathcal{F}_{t-1}]\left(1+\frac{1}{N-1}\right) \\
&\quad - \frac{1}{(N)_2}\mathbb{E}\left[\frac{\sum_{i=1}^{N}\left(Nw_t^{(i)} - \lfloor Nw_t^{(i)}\rfloor\right)\left(Nw_t^{(i)} + \sum_{k\neq i}\lfloor Nw_t^{(k)}\rfloor\right)}{\sum_{j=1}^{N}\left(Nw_t^{(j)} - \lfloor Nw_t^{(j)}\rfloor\right)}\middle|\mathcal{F}_{t-1}\right]
\end{aligned}$$

Under the assumptions of Koskela et al. (2018, Lemma 3) we have that $\mathbb{E}[c_N^m(t)|\mathcal{F}_{t-1}] = O(N^{-1})$. Combining this with the observation that the subtracted term above is non-negative, we obtain the bound

$$\mathbb{E}[c_N^r(t)|\mathcal{F}_{t-1}] \leq \mathbb{E}[c_N^m(t)|\mathcal{F}_{t-1}] + O(N^{-2})$$

which was stated in (9). □

References

Andrieu, C., Doucet, A. and Holenstein, R. (2010), 'Particle Markov chain Monte Carlo methods', *Journal of the Royal Statistical Society: Series B (Statistical Methodology)* **72**(3), 269–342.

Bain, A. and Crisan, D. (2008), *Fundamentals of Stochastic Filtering*, Springer.

Baum, L. E., Petrie, T., Soules, G. and Weiss, N. (1970), 'A maximization technique occurring in the statistical analysis of probabilistic functions of Markov chains', *The Annals of Mathematical Statistics* **41**, 164–171.

Cannings, C. (1974), 'The latent roots of certain markov chains arising in genetics: a new approach, i. haploid models', *Advances in Applied Probability* **6**(2), 260–290.

Cannings, C. (1975), 'The latent roots of certain markov chains arising in genetics: a new approach, ii. further haploid models', *Advances in Applied Probability* **7**(2), 264–282.

Carpenter, J., Clifford, P. and Fearnhead, P. (1999), 'Improved particle filter for nonlinear problems', *IEE Proceedings - Radar, Sonar and Navigation* **146**(1), 2–7.

Chan, H. P., Lai, T. L. et al. (2013), 'A general theory of particle filters in hidden Markov models and some applications', *The Annals of Statistics* **41**(6), 2877–2904.

Crisan, D. and Lyons, T. (1999), 'A particle approximation of the solution of the Kushner–Stratonovitch equation', *Probability Theory and Related Fields* **115**(4), 549–578.

Del Moral, P. (2013), *Mean Field Simulation for Monte Carlo Integration*, Chapman and Hall/CRC.

Del Moral, P., Doucet, A. and Jasra, A. (2006), 'Sequential Monte Carlo samplers', *Journal of the Royal Statistical Society: Series B (Statistical Methodology)* **68**(3), 411–436.

Douc, R., Cappé, O. and Moulines, E. (2005), Comparison of resampling schemes for particle filtering, *in* 'Image and Signal Processing and Analysis, 2005. ISPA 2005. Proceedings of the 4th International Symposium on', IEEE, pp. 64–69.

Doucet, A., Briers, M. and Sénécal, S. (2006), 'Efficient block sampling strategies for sequential Monte Carlo methods', *Journal of Computational and Graphical Statistics* **15**(3), 693–711.

Doucet, A., De Freitas, N. and Gordon, N. (2001), An introduction to sequential Monte Carlo methods, *in* 'Sequential Monte Carlo methods in Practice', Springer, pp. 3–14.

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

Efron, B. and Tibshirani, R. J. (1994), *An Introduction to the Bootstrap*, CRC press.

Etheridge, A. (2011), *Some Mathematical Models from Population Genetics: École D'Été de Probabilités de Saint-Flour XXXIX-2009*, Springer.

Fisher, R. A. (1923), 'On the dominance ratio', *Proceedings of the Royal Society of Edinburgh* **42**, 321–341.

Fisher, R. A. (1930), 'The distribution of gene ratios for rare mutations', *Proceedings of the Royal Society of Edinburgh* **50**, 205–220.

Gerber, M., Chopin, N. and Whiteley, N. (2017), 'Negative association, ordering and convergence of resampling methods', *arXiv preprint arXiv:1707.01845* .

Gilks, W. R. and Berzuini, C. (2001), 'Following a moving target — Monte Carlo inference for dynamic Bayesian models', *Journal of the Royal Statistical Society: Series B (Statistical Methodology)* **63**(1), 127–146.

Gordon, N. J., Salmond, D. J. and Smith, A. F. (1993), Novel approach to nonlinear/non-Gaussian Bayesian state estimation, *in* 'IEE Proceedings F (Radar and Signal Processing)', Vol. 140, IET, pp. 107–113.

Jacob, P. E., Murray, L. M. and Rubenthaler, S. (2015), 'Path storage in the particle filter', *Statistics and Computing* **25**(2), 487–496.

Jazwinski, A. H. (2007), *Stochastic Processes and Filtering Theory*, Courier Corporation.

Kalman, R. E. (1960), 'A new approach to linear filtering and prediction problems', *Journal of Basic Engineering* **82**(1), 35–45.

Kingman, J. (1982*a*), 'The coalescent', *Stochastic Processes and Their Applications* **13**(3), 235–248.

Kingman, J. (1982*b*), Exchangeability and the evolution of large populations, *in* 'Proceedings of the International Conference on Exchangeability in Probability and Statistics, Rome, 6th-9th April, 1981, in Honour of Professor Bruno de Finetti', North-Holland, Amsterdam.

Kingman, J. (1982*c*), 'On the genealogy of large populations', *Journal of Applied Probability* **19**(A), 27–43.

Kitagawa, G. (1996), 'Monte Carlo filter and smoother for non-Gaussian nonlinear state space models', *Journal of Computational and Graphical Statistics* **5**(1), 1–25.

Koskela, J., Jenkins, P. A., Johansen, A. M. and Spanò, D. (2018), 'Asymptotic genealogies of interacting particle systems with an application to sequential Monte Carlo', *arXiv preprint arXiv:1804.01811* .

Lee, A. and Whiteley, N. (2018), 'Variance estimation in the particle filter', *Biometrika* **105**(3), 609–625.

Liu, J. S. and Chen, R. (1998), 'Sequential Monte Carlo methods for dynamic systems', *Journal of the American Statistical Association* **93**(443), 1032–1044.

Möhle, M. (1998), 'Robustness results for the coalescent', *Journal of Applied Probability* **35**(2), 438–447.

Moran, P. A. P. (1958), Random processes in genetics, *in* 'Mathematical Proceedings of the Cambridge Philosophical Society', Vol. 54, Cambridge University Press, pp. 60–71.

Mosimann, J. E. (1962), 'On the compound multinomial distribution, the multivariate β-distribution, and correlations among proportions', *Biometrika* **49**(1/2), 65–82.

Olsson, J., Douc, R. et al. (2019), 'Numerically stable online estimation of variance in particle filters', *Bernoulli* **25**(2), 1504–1535.

Pitt, M. K. and Shephard, N. (1999), 'Filtering via simulation: Auxiliary particle filters', *Journal of the American Statistical Association* **94**(446), 590–599.

Rauch, H. E., Striebel, C. and Tung, F. (1965), 'Maximum likelihood estimates of linear dynamic systems', *AIAA Journal* **3**(8), 1445–1450.

Vidoni, P. (1999), 'Exponential family state space models based on a conjugate latent process', *Journal of the Royal Statistical Society: Series B (Statistical Methodology)* **61**(1), 213–221.

Wakeley, J. (2009), *Coalescent Theory: An Introduction*, Roberts & Co. Publishers.

Wan, E. A. and Van Der Merwe, R. (2000), The unscented Kalman filter for nonlinear estimation, *in* 'Proceedings of the IEEE 2000 Adaptive Systems for Signal Processing, Communications, and Control Symposium', IEEE, pp. 153–158.

Whitley, D. (1994), 'A genetic algorithm tutorial', *Statistics and Computing* **4**(2), 65–85.

Wright, S. (1931), 'Evolution in mendelian populations', *Genetics* **16**(2), 97–159.