Biometrika (2026), **112**, 1, *p*. 1
Printed in Great Britain Advance Access publication on

Phylodynamic Inference with Bounded Coalescent: A Point Process Perspective

By Bingjing Tang

Department of Epidemiology and Biostatistics, University of California San Franscisco
550 16th Street 2nd floor, San Francisco, CA 94158 USA
bingjing.tang@gmail.com

Shuangping Li

Department of Statistics and Data Science, Yale University
Kline Tower, 219 Prospect Street, New Haven, CT 06511 USA
shuangping.li@yale.edu

Julia A. Palacios

Department of Statistics, Stanford University,
Sequoia Hall, 390 Serra Mall, Stanford University, Stanford, CA 94305 USA
juliapr@stanford.edu

Summary

The coalescent is a central framework in population genetics for modeling the ancestral relationships of a sample of individuals via a genealogy, represented as a rooted and ranked binary tree. In this model, lineages coalesce at a rate inversely proportional to the effective population size, a time-varying quantity of primary interest. The bounded coalescent conditions the genealogy on the time to the most recent common ancestor being upper-bounded by some fixed time. This model is useful in various contexts, such as multi-species modeling and single-cell lineage tracing in synthetic barcoding experiments. To our knowledge, no existing tools provide a unified framework for likelihood derivation, coalescent simulation, and inference of effective population size trajectories under the bounded coalescent with a time-varying effective population size. We view likelihood derivation under the bounded coalescent as equivalent to computing an inhomogeneous phase-type distribution. We provide an algorithm for coalescent simulation under the bounded coalescent using point process methodologies, retaining the exactness of naive rejection sampling while substantially reducing computational cost and avoiding the approximation error inherent in inverse transformation methods. We then develop an exact MCMC procedure for posterior inference of effective population size trajectories, validate it on simulated data by attaining smaller sum of squared error than approaches based on the standard coalescent likelihood, and demonstrate its practical utility on genetic sequence data from single-cell lineage tracing.

Some key words: Bounded coalescent; Singel-cell lineage tracing; Effective population size; Gaussian processes; Phase-type.

1. INTRODUCTION

The coalescent with variable population size (Kingman, 1982; Tavaré, 2004) is a commonly used prior on the latent genealogy of a sample of individuals in Bayesian inference from molecular sequences. In this model, sampled lineages coalesce back in time with rate inversely proportional to a quantity of interest called effective population size. The effective population size trajectory, denoted by $N_e(t)$, is proportional to the rate at which genetic diversity is lost or gained in a randomly mating population and so it provides valuable information about the past population history. The bounded coalescent model is a variant of the coalescent in which the genealogy is conditioned on the time to the most recent common ancestor (TMRCA) being smaller than some known fixed time τ (Carson et al., 2022). This setting arises in multi-species settings when jointly modeling species phylogeny and gene trees. Specifically, when a gene duplication occurs on a branch of the species tree at time τ, all descendant gene lineages must originate from that duplication event, and thus must share a common ancestor before τ (Pamilo & Nei, 1988; Rasmussen & Kellis, 2012; Li et al., 2021). Similarly, in the context of infectious diseases, when modeling within-host pathogen transmission (Didelot et al., 2017), coalescence within host must occur before coalescence across different hosts. Another example in studying infectious diseases occurs when the introduction event time is known and the coalescent genealogy must coalesce before that time. In this work we will show that the bounded coalescent also emerges in the context of single cell lineage tracing data in which the experimenter knows all lineages need to have coalesced before the start of the experiment (tracing lineages from the end of the experiment to the start).

In Carson et al. (2022), the authors studied the bounded coalescent and showed that ignoring the bound introduces systematic bias, leading to negatively biased estimates of effective population sizes. Although the bounded coalescent model has been used across various applications, its use has been restricted to the case of constant effective population sizes. To the best of our knowledge, no prior work has extended the bounded coalescent model to the more general setting of a time-varying effective population size $N_e(t)$. In this work, we develop a method for Bayesian nonparametric inference of variable effective population size trajectories under the bounded coalescent. We first show that the distribution of the TMRCA corresponds to an inhomogeneous phase-type distribution (Albrecher & Bladt, 2019; Horváth & Telek, 2024) that can be explicitly computed without matrix exponential. Recently, phase-type distributions have emerged as a powerful framework in mathematical population genetics. In Hobolth et al. (2024), the authors focus on homogeneous standard coalescent models and derive distributions for key population-genetic quantities, including TMRCA, by representing them as phase-type distributions. We extend this work on TMRCA for the inhomogeneous case. A brief overview of phase-type distributions is given in section 7.

For Bayesian inference, we place a transformed Gaussian process prior on $N_e(t)$ and develop an MCMC algorithm for posterior exploration. Gaussian process-based priors have been used extensively in the standard coalescent. In Minin et al. (2008); Palacios & Minin (2012); Gill et al. (2013), the authors place a log-Gaussian Markov random field prior on $N_e(t)$, however the exact likelihood becomes intractable and it is therefore approximated via discretization. In Palacios & Minin (2013), this discretization is avoided by introducing latent variables and targeting an augmented posterior over coalescent times and auxiliary event times, yielding a tractable augmented likelihood. This strategy, however, relies on point-process thinning and does not extend directly to the bounded coalescent. To overcome this limitation, we adapted a recently proposed

method for Cox Processes that avoids likelihood approximation (Tang & Palacios, 2024). The key insight is to jointly a priori model $1/N_e(t)$ and its associated integrated quantities of the form $\int_*^* dt/N_e(t)$ under a truncated Gaussian prior, thereby effectively avoiding discretization in the calculation of these integrals in the likelihood. We tested our method in simulations and a single-lineage tracing dataset. In addition to providing an inference method of $N_e(t)$ from bounded genealogies, we develop a novel and efficient method for simulating genealogies under this model.

In Carson et al. (2022), the authors proposed a simulation algorithm for the bounded coalescent with constant population size. The algorithm first samples the number of extant lineages at discretized time points (bins) and iteratively refines the bins until the number of lineages across consecutive bins changes by one. The algorithm then samples coalescent times by inverse transformation. They demonstrated that this direct sampling method significantly outperforms naive rejection sampling in terms of computational efficiency. In this work, we propose a novel and efficient method for simulating genealogies under the bounded coalescent with variable population size, including constant. Our approach combines time transformation and thinning. In particular, we provide a time-varying upper bound on the coalescent intensity rate and use it to simulate proposals of coalescent events via a time transformation method. We then accept a subset of those events via thinning. Thinning is an accept/reject algorithm that was first proposed for inhomogeneous Poisson processes (Lewis & Shedler, 1979), and later extended to a more general class of point processes (Ogata, 1981). Our synthetic experiments show that this method significantly outperforms both naive rejection sampling and the direct sampling algorithm proposed by Carson et al. (2022) in terms of computational efficiency.

2. Bounded coalescent

We assume that a genealogy with n tips sampled at present time (assumed to be 0), such as the one depicted in Figure 1, is available to us. The coalescent time T_k denotes the time when two of the k extant lineages merge into a single ancestral lineage. Under the standard coalescent, the density of T_k given there are k lineages after time t_{k+1} is

$$f(t_k \mid t_{k+1}, N_e(t)) := \frac{\binom{k}{2}}{N_e(t_k)} \exp\left\{ - \int_{t_{k+1}}^{t_k} \frac{\binom{k}{2}}{N_e(s)} \, d\, s \right\},$$

where $N_e(t)$ denotes the effective population size at time t (Slatkin & Maddison, 1989; Tavaré, 2004). Consequently, a full realization of the coalescent process is the ordered vector $\mathbf{t} = (t_n, \ldots, t_2)$ satisfying $0 < t_n < t_{n-1} < \cdots < t_2$, with likelihood:

$$f(\mathbf{t} \mid N_e(t)) = \prod_{k=n}^{2} f(t_k \mid t_{k+1}, N_e(t)).$$

Under the bounded coalescent (Carson et al., 2022), we further assume that T_2, the time to the most recent common ancestor, is bounded by a known quantity denoted by τ. Hence, the corresponding coalescent likelihood is:

$$f(\mathbf{t} \mid N_e(t), T_2 \leq \tau) = \prod_{k=n}^{2} f(t_k \mid t_{k+1}, N_e(t)) \frac{\mathbb{1}(t_2 \leq \tau)}{\mathrm{pr}(T_2 \leq \tau \mid N_e(t))}. \tag{1}$$

Many genealogy statistics, including the TMRCA T_2, are known to follow phase-type distributions (Hobolth et al., 2024). When the effective population size varies through time, the

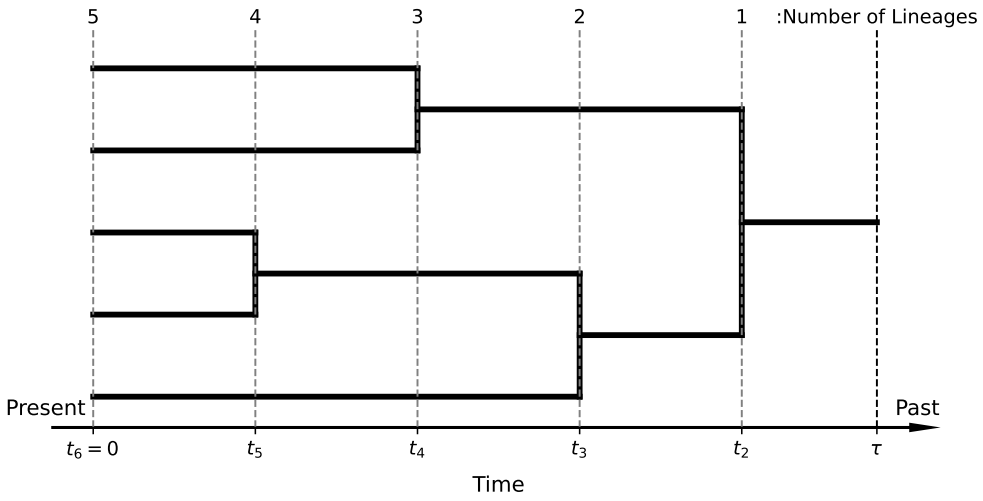

Figure 1: An example genealogy with $n = 5$ tips sampled at the present time $t_6 = 0$. Coalescent times are denoted by $t_5, \ldots, t_2$, where t_k denotes the time at which two of the k extant lineages merge into a single lineage.

distribution becomes inhomogeneous phase-type. Recall that a phase-type distribution is the law of the waiting time until absorption in a finite state space continuous-time Markov chain (Horváth & Telek, 2024). In the next proposition, we show that we can avoid matrix exponential computations and calculate the bound probability $\mathrm{pr}(T_2 \le \tau \mid N_e(t))$ as a linear combination of simpler terms. This is not only useful for reducing computational burden for inference, but also for designing an efficient simulation algorithm under this model.

PROPOSITION 1. *Let T_2 denote the TMRCA of a standard coalescent tree with n tips evolving under the population effective size trajectory $N_e(t)$, then*

$$pr(T_2 \le \tau \mid N_e(t)) = \sum_{j=1}^{n} r_{j,n}\, e^{-\binom{j}{2}\Lambda(\tau)},$$

where the coefficients are defined as

$$r_{j,n} := (-1)^{j-1}(2j-1)\frac{(n)_j}{(n-1+j)_j} \quad (j = 1, \ldots, n),$$

$\Lambda(\tau) := \int_0^\tau \frac{1}{N_e(u)} du$, *and* $(x)_j := x(x-1)\cdots(x-j+1) = \frac{x!}{(x-j)!}$.

The result in Proposition 1 is obtained by finding the spectral expansion of the generator. The coefficients in the sum are the same as in the coalescent with constant population size (Tavaré, 1984). Full proofs of all propositions in this paper are provided in Section 8. Throughout, we adopt the convention that

$$\binom{1}{2} = 0.$$

3. The bounded coalescent as a point process

In this section, we derive some results that are needed for devising a novel simulation algorithm for the bounded coalescent with a time-varying effective population size (Section 3.1).

In order to define an algorithm that sequentially generates coalescent times under the model, we re-express the likelihood function in Equation (1) as a product of Markov bridge kernels conditioned on $T_2 \leq \tau$ as follows:

$$\begin{aligned} f(\mathbf{t} \mid T_2 \leq \tau) &= \prod_{k=n}^{2} f(t_k \mid t_{k+1}, T_2 \leq \tau) \\ &= \prod_{k=n}^{2} f(t_k \mid t_{k+1}) \frac{\text{pr}\left(T_2 \leq \tau \mid T_k = t_k\right)}{\text{pr}\left(T_2 \leq \tau \mid T_{k+1} = t_{k+1}\right)} \\ &:= \prod_{k=n}^{2} f^B(t_k \mid t_{k+1}), \end{aligned}$$

where we have suppressed $N_e(t)$ in our notation, though we are still conditioning on a given effective population trajectory $N_e(t)$. The general form of these Markov bridge kernels is:

$$f^B(t_k \mid t_{k+1}) = f(t_k \mid t_{k+1}) g_k(t_k, t_{k+1}, \tau) \quad (k = 2, 3, \ldots, n),$$

where

$$g_k(t_k, t_{k+1}, \tau) := \frac{\text{pr}\left(T_2 \leq \tau \mid T_k = t_k\right)}{\text{pr}\left(T_2 \leq \tau \mid T_{k+1} = t_{k+1}\right)} = \frac{\sum_{j=1}^{k-1} r_{j,k-1} \exp\left[\binom{j}{2}\{\Lambda(t_k) - \Lambda(\tau)\}\right]}{\sum_{j=1}^{k} r_{j,k} \exp\left[\binom{j}{2}\{\Lambda(t_{k+1}) - \Lambda(\tau)\}\right]}$$

is obtained from the following Corollary 1, a consequence of Proposition 1.

Corollary 1. *For all integers $k = 2, \ldots, n$ and $r_{j,k}$ coefficients defined in Proposition 1 ,we have*

$$\begin{aligned} pr(T_2 \leq \tau \mid T_{k+1} = u, N_e(t)) &= pr(T_2 \leq \tau \mid T_k > u, N_e(t)), \\ &= \sum_{j=1}^{k} r_{j,k} \exp\left[\binom{j}{2}\{\Lambda(u) - \Lambda(\tau)\}\right]. \end{aligned}$$

We now derive the conditional intensity function for the bounded coalescent point process based on its likelihood function in Equation (1). The conditional intensity function of a general point process is defined as

$$\lambda(t) = \frac{f(t \mid \mathcal{H}_t)}{1 - \int_{t_0}^{t} f(s \mid \mathcal{H}_t)\, ds},$$

where $\mathcal{H}_t$ denotes the filtration, that is, the history of the process up to time t (not including t), t_0 denotes the time of the last observed event before time t, and $f(t \mid \mathcal{H}_t)$ denotes the conditional density on all past events (Rasmussen, 2018). Therefore, the conditional intensity function of the kth coalescence is

$$\lambda_k^B(t) = \frac{f^B(t \mid t_{k+1})}{1 - \int_{t_{k+1}}^{t} f^B(s \mid t_{k+1}) ds}, \quad t \in (t_{k+1}, t_k],$$

$$= \frac{\binom{k}{2}}{N_e(t)} g_k(t, t, \tau), \quad t \in (t_{k+1}, t_k]. \tag{2}$$

It is also noticeable that

$$\lim_{t \to \tau} \lambda_k^B(t) = \infty. \tag{3}$$

All derivations can be found in Section 10.

3.1. *Sampling from a bounded coalescent point process*

Simulation of genealogies under the bounded coalescent is useful both for inference and for understanding how different effective population size trajectories and bounds influence the branch length distribution of genealogies. A naive approach is to perform rejection sampling (Tavaré, 2004; Didelot et al., 2014); although conceptually straightforward, this method becomes inefficient when the rejection probability is high. Carson et al. (2022) proposed a sampling algorithm for the bounded coalescent under a constant effective population size trajectory; however, this approach does not extend to time-varying effective population size trajectories. Two common approaches for simulating point processes are the inverse transformation method (Slatkin & Hudson, 1991) and the thinning algorithm (Lewis & Shedler, 1979). Both approaches have been successfully adapted to simulation of the standard coalescent point process (Hein et al., 2004; Palacios & Minin, 2013). We briefly describe these two methods for the standard coalescent point process. The inverse transformation method exploits the fact that

$$\Lambda_k(T_k) - \Lambda_k(T_{k+1}) \sim \text{Exp}(1),$$

where

$$\Lambda_k(t) := \binom{k}{2} \int_0^t \frac{1}{N_e(u)} du = \binom{k}{2} \Lambda(t).$$

Consequently, given T_{k+1} and a standard exponential random variable W, the next event time is obtained as

$$T_k = \Lambda_k^{-1}(\Lambda_k(T_{k+1}) + W).$$

Similar in spirit to rejection sampling, the thinning algorithm requires a constant upper bound $\bar{\lambda}_k$ such that

$$\bar{\lambda}_k \geq \lambda_k(t) := \binom{k}{2} / N_e(t).$$

Given T_{k+1} and $\bar{\lambda}_k$, we initially assign $W = T_{k+1}$ and sequentially update $W = W + E_i$, with $E_i \sim \text{Exp}(\bar{\lambda}_k)$, until $T_k = W$ is accepted with probability $\lambda_k(W)/\bar{\lambda}_k$.

Applying the inverse-transform method to the bounded coalescent requires tractable expressions for both $\Lambda_k^B(t) = \int_0^t \lambda_k^B(u) du$ and its inverse $(\Lambda_k^B)^{-1}(t)$. However, even under the assumption that $N_e(t)$, $\Lambda(t)$, and $\Lambda^{-1}(t)$ are all tractable, neither $\Lambda_k^B(t)$ nor its inverse $(\Lambda_k^B)^{-1}(t)$ generally admits a closed-form expression. This is because $\Lambda_k^B(t)$ involves an intractable integral of a ratio of polynomial functions of $\exp[\Lambda(t) - \Lambda(\tau)]$. Although numerical approximation of $(\Lambda_k^B)^{-1}(t)$ is in principle feasible, it introduces several practical difficulties beyond approximation error. Specifically, according to Equation (4), simulating t_k from the bounded coalescent conditional on t_{k+1} requires evaluating $(\Lambda_k^B)^{-1}$ at $\Lambda_k^B(t_{k+1}) + W$. This evaluation is typically carried out via a root-finding procedure, which in turn requires repeated numerical approximation of Λ_k^B at successive iterates of the root-finding algorithm. Consequently, evaluating $(\Lambda_k^B)^{-1}$ at a single

point entails a nested numerical computation. Moreover, because Λ_k^B depends explicitly on k, numerical computations cannot be reused across coalescent events. Finally, numerical inversion near the boundary $t \to \tau$ may suffer from instability and loss of precision, since the bounded coalescent intensity $\lambda_k^B(t)$ diverges as $t \to \tau$ for all k. Indeed notice that

$$\begin{aligned}\Lambda_k^B(t) &= \int_0^t \frac{\binom{k}{2}}{N_e(u)}\, g_k(u,u,\tau)\, du, \\ &= \binom{k}{2} \int_0^t \frac{\sum_{j=1}^{k-1} r_{j,k-1} \exp\left[\binom{j}{2}\{\Lambda(u)-\Lambda(\tau)\}\right]}{\sum_{j=1}^{k} r_{j,k} \exp\left[\binom{j}{2}\{\Lambda(u)-\Lambda(\tau)\}\right]} d\Lambda(u) \end{aligned} \tag{4}$$

Due to the unbounded nature (see Equation (3)) of the bounded coalescent point process, the thinning algorithm cannot be applied directly using a constant upper bound. Instead, we employ a thinning algorithm with a time-varying upper bound $\lambda_k^U(t)$. The main challenge is to construct such a $\lambda_k^U(t)$ so that the corresponding dominating point process can be simulated easily and the acceptance ratio $\lambda_k^B(t)/\lambda_k^U(t)$ is tractable. The following Lemma and Proposition 2 allow us to express the bound probabilities in a polynomial form that is useful for defining $\lambda_k^U(t)$.

Lemma 1. *For all integers $k = 2, \ldots, n$, and coefficients $r_{j,k}$ defined in Proposition 1, the polynomial on $x \in \mathbb{R}$ given by*

$$\sum_{j=1}^{k} r_{j,k}\, x^{\binom{j}{2}}$$

admits the factorization

$$\sum_{j=1}^{k} r_{j,k}\, x^{\binom{j}{2}} = (1-x)^{k-1} \left(\sum_{i=0}^{M_k} a_{i,k}\, x^i \right),$$

where

$$M_k = \binom{k-1}{2},$$

and the coefficients $\{a_{i,k}\}$ are defined recursively as

$$a_{i,k} = \begin{cases} \displaystyle\sum_{s=1}^{k-1} (-1)^{s+1} \binom{k-1}{s} a_{i-s,k} \;+\; \sum_{j=1}^{k} r_{j,k}\, \mathbb{1}\left\{\binom{j}{2} = i\right\} & (i = 0, \ldots, M_k), \\ 0 & \text{otherwise}. \end{cases}$$

Remark 1 (*Connection to Touchard–Riordan numbers*). The coefficients $r_{j,k}$ appearing in Lemma 1 are closely related to the Touchard–Riordan numbers $t_{k,j}$ (OEIS Foundation Inc., 2026; Riordan, 1975). Our proof of Lemma 1 provides a self-contained algebraic derivation that avoids the combinatorial framework of chord crossings used by Riordan (1975). A precise correspondence is discussed in Section 9.

Proposition 2.

$$\frac{\sum_{j=1}^{k-1} r_{j,k-1} x^{\binom{j}{2}}}{\sum_{j=1}^{k} r_{j,k} x^{\binom{j}{2}}} \le \frac{1}{1-x} \quad (x \in [0,1];\ k = 3, \ldots, n).$$

The proof of Proposition 2 is provided in Section 8. Substituting $x = \exp\{\Lambda(t) - \Lambda(\tau)\}$ into the inequality yields $g_k(t,t,\tau) \le 1/(1-\exp\{\Lambda(t)-\Lambda(\tau)\})$, implying that $\lambda_k^B(t) \le \lambda_k^U(t)$, where $\lambda_k^U(t)$ is an upper-bound intensity constructed as follows:

$$\lambda_k^U(t) \;:=\; \frac{\binom{k}{2}}{N_e(t)} \frac{1}{1-\exp\{\Lambda(t)-\Lambda(\tau)\}}. \tag{5}$$

As mentioned above, the construction of $\lambda_k^U(t)$ brings both practical simulation and tractable acceptance ratio. We can then use the time-transformation method to simulate time event realizations with intensity $\lambda_k^U(t)$, and then use thinning until acceptance. Specifically, to simulate t_k from the bounded coalescent conditioned on t_{k+1}, we initialize $t_{\text{old}} = t_{k+1}$ and repeat the following two steps until acceptance:

Step 1 (Inverse transformation). According to Equation (5),

$$\Lambda_k^U(t) = \int_0^t \lambda_k^U(s)\,ds = \binom{k}{2} \log\left(\frac{e^{\Lambda(\tau)}-1}{e^{\Lambda(\tau)-\Lambda(t)}-1}\right).$$

Given t_{old}, we draw $W \sim \text{Exp}(1)$ and solve

$$W = \Lambda_k^U(t_{\text{new}}) - \Lambda_k^U(t_{\text{old}})$$

for t_{new}, to obtain

$$t_{\text{new}} = \Lambda^{-1}\left[\Lambda(\tau) - \log\left\{1 + e^{-W/\binom{k}{2}}\left(e^{\Lambda(\tau)-\Lambda(t_{\text{old}})}-1\right)\right\}\right].$$

Step 2 (Thinning). We accept t_{new} as t_k with probability

$$\frac{\lambda_k^B(t_{\text{new}})}{\lambda_k^U(t_{\text{new}})} = g_k(t_{\text{new}}, t_{\text{new}}, \tau)\left(1 - e^{\Lambda(t_{\text{new}})-\Lambda(\tau)}\right).$$

If the proposal is rejected, we set $t_{\text{old}} = t_{\text{new}}$ and repeat Steps 1–2.

Algorithm 1 provides a procedural implementation of this scheme.

Algorithm 1. Simulation of coalescent times by thinning with tractable $N_e(t)$, $\Lambda(t)$, and $\Lambda^{-1}(t)$.

Input: $k = n$, $t = 0$, $N_e(t)$, $\Lambda(t)$, $\Lambda^{-1}(t)$.
Output: $\{t_k\}_{k=n}^2$.
repeat
  Transformation:
  Sample $E \sim \text{Exp}\left(\binom{k}{2}\right)$ and $U \sim \text{Unif}(0,1)$;

$$t = \Lambda^{-1}\left[\Lambda(\tau) + E - \log\left\{e^{\Lambda(\tau)-\Lambda(t)} + e^E - 1\right\}\right]$$

  Thinning:
  if $U \le \lambda_k^B(t)/\lambda_k^U(t)$ **then**
    $t_k \leftarrow t, \quad k \leftarrow k-1$;
until $k < 2$;

We can extend Algorithm 1 to a more general setting in which only $N_e(t)$ and the cumulative intensity function $\Lambda(t)$ are numerically evaluable, while the inverse $\Lambda^{-1}(t)$ is not available.

Specifically, we assume

$$L \leq \frac{1}{N_e(t)} \leq M \quad \text{for all } t.$$

Details of this extension are provided in Algorithm 2.

4. Inference method

We are interested in estimating the posterior distribution of $N_e(t)$. Following the Bayesian nonparametric inference in Palacios & Minin (2013) for standard coalescent models, in this study we also a priori model $N_e(t)$ as a transformed Gaussian process, and naturally the problem is reduced to posterior inference over Gaussian processes. In a standard coalescent model, since its likelihood includes intractable integral terms $\int_{t_{k+1}}^{t_k} ds/N_e(s)$ for $k = 2, \ldots, n$, the posterior distribution over $N_e(t)$ is *doubly-intractable* (Murray et al., 2012). Similar in a bounded coalescent model , with one more extra integral term $\int_{t_2}^{\tau} ds/N_e(s)$. To perform exact MCMC inference without discretization, Palacios & Minin (2013) proposed a data augmentation scheme by introducing rejected proposals in a virtual thinning generative process. This method requires an upper bound intensity function whose point process likelihood is tractable and a tractable intensity function ratio. However, in a bounded coalescent model, Equation (2) shows that it is difficult to find an upper bound intensity function with both tractable likelihood and intensity ratio over $\lambda^B(t)$ (typically the cumulative function of the nonparametric estimate of $N_e(t)$ is intractable). To solve this, we follow the random integral (RI) method proposed in Tang & Palacios (2024), where more technical details about this section could be found.

At a high level, in this study the adopted RI approach is to treat the integrals as latent random variables and jointly a priori model the function values of $1/N_e(t)$ at finite locations with its integrals. We aim to define a joint positive prior constructed from Gaussian processes on

$$\boldsymbol{\lambda} := \left[\frac{1}{N_e(x_1)}, \ldots, \frac{1}{N_e(x_m)}, \int_0^{t_n} \frac{1}{N_e(s)} ds, \int_{t_n}^{t_{n-1}} \frac{1}{N_e(s)} ds, \ldots, \int_{t_3}^{t_2} \frac{1}{N_e(s)} ds, \int_{t_2}^{\tau} \frac{1}{N_e(s)} ds\right]^{\mathrm{T}}$$

where $\{x_i\}_{i=1}^m$ are locations of interest, including both observed points $\{t_k\}_{k=2}^{n+1}$, and prediction (test) locations $\{s_l\}_{l=1}^{m-n}$, that is, $\{x_i\}_{i=1}^m := \{t_k\}_{k=2}^{n+1} \cup \{s_l\}_{l=1}^{m-n}$.

Theorem 1. *Suppose the Gaussian process $f(\cdot)$ on the compact space $\mathcal{X}$ satisfies the assumption that its mean function $\mu(\cdot)$ and covariance kernel $k(\cdot,\cdot)$ are integrable, i.e., $\int_{\mathcal{X}} \mu(s)ds, \int_{\mathcal{X}} k(s,t)dt$ and $\int \int_{\mathcal{X}\times\mathcal{X}} k(s,t)dsdt$ exist. For every finite set of vectors $s_1, \ldots, s_p \in \mathcal{X}$ and subsets $\{\mathcal{X}_i\}_{i=1}^q$ where $\mathcal{X}_i \subset \mathcal{X}$, the vector $\mathbf{f} := \left[f(s_1), \ldots, f(s_p), \int_{\mathcal{X}_1} f(s)ds, \ldots, \int_{\mathcal{X}_q} f(s)ds\right]^{\mathrm{T}}$ follows a Gaussian distribution and*

$$\mathbf{f} \sim \mathcal{N}\left(\boldsymbol{\mu}, \begin{pmatrix} \boldsymbol{V}_{SS} & \boldsymbol{V}_{SI} \\ \boldsymbol{V}_{SI}^{\mathrm{T}} & \boldsymbol{V}_{II} \end{pmatrix}\right),$$

where $\boldsymbol{\mu} := [\mu(s_1), \ldots, \mu(s_p), \int_{\mathcal{X}_1} \mu(s)ds, \int_{\mathcal{X}_q} \mu(s)ds]^{\mathrm{T}}$, $\boldsymbol{V}_{SI}$ is a $p \times q$ matrix formed by covariance terms between function values $\{f(s_i)\}_{i=1}^p$ and integral values $\{\int_{\mathcal{X}_j} f(s)ds\}_{j=1}^q$ with ij-th term being $\int_{\mathcal{X}_j} k(s_i,t)dt$, and $\boldsymbol{V}_{II}$ is a $q \times q$ matrix containing covariance terms for all pairs of integral values $\{\int_{\mathcal{X}_j} f(s)ds\}_{j=1}^q$ with ij-th term being $\int \int_{\mathcal{X}_i\times\mathcal{X}_j} k(s,t)dsdt$.

Following theorem 1, which is the extension of Theorem 2.1 in Tang & Palacios (2024) from one integral term to multiple integrals, we place a truncated positive Gaussian distribution on $\boldsymbol{\lambda}$, imposing an additional positive constraint to it. Precisely, $\boldsymbol{\lambda} \sim \mathcal{TN}(\boldsymbol{\mu}, \boldsymbol{V})$, with density:

$$p(\boldsymbol{\lambda}) = \frac{\exp\left\{-\frac{1}{2}(\boldsymbol{\lambda}-\boldsymbol{\mu})^{\mathrm{T}}\boldsymbol{V}^{-1}(\boldsymbol{\lambda}-\boldsymbol{\mu})\right\}}{\int_{\mathcal{S}} \exp\left\{-\frac{1}{2}(\boldsymbol{\lambda}-\boldsymbol{\mu})^{\mathrm{T}}\boldsymbol{V}^{-1}(\boldsymbol{\lambda}-\boldsymbol{\mu})\right\} d\boldsymbol{\lambda}} \cdot \mathbb{1}(\boldsymbol{\lambda} > \mathbf{0})$$

where $\mathcal{S} = [0, +\infty]^{n+m}$ and $\mathbb{1}(\boldsymbol{\lambda} > \mathbf{0})$ is an indicator function that takes 1 if all elements of $\boldsymbol{\lambda}$ are positive. Both mean vector $\boldsymbol{\mu}$ and covariance matrix V are defined according to Theorem 1. The presence of multiple integrals in $\boldsymbol{\lambda}$ induces a more complex covariance structure for $\boldsymbol{V}$ than that in Tang & Palacios (2024), which may lead to numerical difficulties in computing its inverse and Cholesky decomposition during posterior inference; see section 5.2 for details. In this work, we set mean $\boldsymbol{\mu}$ to be zero and consider the following two covariance kernels for $\boldsymbol{V}$: the squared exponential kernel with hyperparameters $\theta = (\theta_0, \theta_1)$, i.e., $k_{SE}(x, x') = \exp\left(-\theta_1 \|x - x'\|^2/2\right)/\theta_0$, and the Brownian motion covariance kernel, i.e., $k_{BM}(x, x') = \min(x, x')/\theta$, where θ is the hyperparameter, denoting the precision parameter. Let C/θ denote the Brownian motion covariance obtained from Theorem 1 using the Brownian motion kernel $k_{\mathrm{BM}}(x, x')$, where

$$C = \begin{pmatrix} x_1 & \dots & \min(x_1, x_M) & \int_0^{t_n} \min(x_1, t)dt & \dots & \int_{t_2}^{\tau} \min(x_1, t)dt \\ \vdots & \ddots & \vdots & \vdots & \vdots & \\ \min(x_M, x_1) & \dots & x_M & \int_0^{t_n} \min(x_M, t)dt & \dots & \int_{t_2}^{\tau} \min(x_M, t)dt \\ \int_0^{t_n} \min(x_1, t)dt & \dots & \int_0^{t_n} \min(x_M, t)dt & \int_0^{t_n}\int_0^{t_n} \min(s, t)dsdt & \dots & \int_0^{t_n}\int_{t_2}^{\tau} \min(s, t)dsdt \\ \vdots & \ddots & \vdots & \vdots & \vdots & \\ \int_{t_2}^{\tau} \min(x_1, t)dt & \dots & \int_{t_2}^{\tau} \min(x_M, t)dt & \int_{t_2}^{\tau}\int_0^{t_n} \min(s, t)dsdt & \dots & \int_{t_2}^{\tau}\int_{t_2}^{\tau} \min(s, t)dsdt \end{pmatrix}.$$

However, as discussed in Tang & Palacios (2024), a limitation of the mean-zero Brownian motion prior is that it assigns very small prior variance to function values at locations near the origin, which reduces model flexibility and typically leads to low posterior values in these regions. To address this issue, we adopt a boundary-corrected Brownian motion prior (Rue & Held, 2005; Tang & Palacios, 2024), with covariance $\tilde{C}/\theta$, where

$$\tilde{C} := \left(C^{-1} - \frac{C^{-1}\boldsymbol{l}\boldsymbol{l}^{\mathrm{T}}C^{-1}}{\boldsymbol{l}^{\mathrm{T}}C^{-1}\boldsymbol{l}} + \epsilon\mathbf{I}\right)^{-1}, \tag{6}$$

Here ϵ denotes a jitter term (i.e., a small positive constant, e.g., 10^{-16}) added for numerical stability and

$$\boldsymbol{l} := (\underbrace{1, \dots, 1}_{m},\ t_n,\ t_{n-1} - t_n,\ \dots,\ t_2 - t_3,\ \tau - t_2)^{\mathrm{T}}.$$

The matrix $C^{-1} - C^{-1}\boldsymbol{l}\boldsymbol{l}^{\mathrm{T}}C^{-1}/(\boldsymbol{l}^{\mathrm{T}}C^{-1}\boldsymbol{l})$ is rank deficient, therefore we add the jitter term ϵ to its diagonal to ensure invertibility. Note that the vector $\boldsymbol{l}$ is defined differently from that in Tang & Palacios (2024), reflecting the structure induced by the multiple integral terms considered here. In general, this boundary-corrected Brownian motion prior is equivalent to placing a noninformative prior on the Brownian motion initial value and subsequently marginalizing it out, thereby alleviating the degeneracy of the mean-zero Brownian motion prior near the origin.

After defining the prior, the posterior distribution is

$$p(\boldsymbol{\lambda}, \theta | \{x_i\}_{i=1}^m) \propto\ p_\theta(\theta)\mathcal{TN}(\boldsymbol{\lambda}; \mathbf{0}, V_\theta)$$

$$\prod_{k=n}^{2} \frac{\binom{k}{2}}{N_e(t_k)} \exp\left\{-\binom{k}{2}\int_{t_{k+1}}^{t_k} \frac{1}{N_e(s)} ds\right\} \frac{\mathbb{1}(t_2 \le \tau)}{r_{1,n} + \sum_{j=2}^{n} r_{j,n} \exp\left\{-\binom{j}{2}\int_0^{\tau} \frac{1}{N_e(s)} ds\right\}}$$

where the covariance V_θ is constructed from the kernel function $k_\theta(\cdot,\cdot)$ as described in Theorem 1, the mean of the GP prior is assumed to be zero, and $p_\theta(\theta)$ denotes the prior distribution on the kernel hyperparameter θ. we estimate the posterior distribution via Metropolis-within-Gibbs sampling in two steps, alternating between $\boldsymbol{\lambda}$ and θ.

Sample $\boldsymbol{\lambda} \mid \theta, \{x_i\}_{i=1}^m$:

$$\begin{aligned} p(\boldsymbol{\lambda} \mid \theta, \{x_i\}_{i=1}^M) \propto \mathcal{N}(\boldsymbol{\lambda}; \mathbf{0}, V_\theta) & \prod_{i=1}^{m} \mathbb{1}\left(\frac{1}{N_e(x_i)} > 0\right) \\ & \prod_{k=n}^{2} \mathbb{1}\left(\int_{t_{k+1}}^{t_k} \frac{1}{N_e(s)}\, ds > 0\right) \mathbb{1}\left(\int_{t_2}^{\tau} \frac{1}{N_e(s)}\, ds > 0\right) \\ & \prod_{k=n}^{2} \frac{\binom{k}{2}}{N_e(t_k)} \exp\left\{-\binom{k}{2}\int_{t_{k+1}}^{t_k} \frac{1}{N_e(s)}\, ds\right\} \\ & \frac{\mathbb{1}(t_2 \le \tau)}{r_{1,n} + \sum_{j=2}^{n} r_{j,n} \exp\left\{-\binom{j}{2}\int_0^{\tau} \frac{1}{N_e(s)}\, ds\right\}}. \end{aligned}$$

The conditional no longer involves any intractable terms. Due to the positivity constraint, a Metropolis-Hastings algorithm with Gaussian proposal would lead to rare acceptance, while a Metropolis-Hastings algorithm with truncated Gaussian proposals would lead to both inefficiency in sampling from truncated multivariate normal distribution (especially for high-dimensional) and inaccuracy in acceptance rate estimation. Fortunately, we found that a routine elliptical slice sampler (ESS)(Murray et al., 2010) is particularly effective in this context due to its adaptive bracket size and the fact that it transforms the original high-dimensional sampling problem into a one-dimensional sampling problem. Additionally, since it is designed for target distributions of the form $p(f) = \mathcal{N}(f; 0, \Sigma)L(f)/Z$, we can incorporate the indicator terms inherited from the truncated normal prior into $L(f)$, thereby avoiding inefficiencies associated with truncated normal distributions.

Sample $\theta \mid \boldsymbol{\lambda}, \{x_i\}_{i=1}^m$:

$$p(\theta|\boldsymbol{\lambda}, \{x_i\}_{i=1}^m) \propto\ p_\theta(\theta) \frac{\exp\left\{-\frac{1}{2}\boldsymbol{\lambda}^{\mathrm{T}} V_\theta^{-1} \boldsymbol{\lambda}\right\}}{\int_{\mathcal{F}} \exp\left\{-\frac{1}{2}\boldsymbol{\lambda}^{\mathrm{T}} V_\theta^{-1} \boldsymbol{\lambda}\right\} d\boldsymbol{\lambda}},$$

where $\mathcal{F} = [0, +\infty)^{m+n}$. We note that, for a general kernel, the normalizing constant $\int_{\mathcal{F}} \exp\left\{-\boldsymbol{\lambda}^{\mathrm{T}} V_\theta^{-1} \boldsymbol{\lambda}/2\right\} d\boldsymbol{\lambda}$ arising from the truncated multivariate normal density is analytically intractable and can only be approximated numerically, typically via evaluation of a multivariate normal CDF over a hyperrectangle. We elaborate on this point using the squared exponential kernel. In this case, the covariance matrix can be written as

$V_\theta = K_{\theta_1}/\theta_0$, where

$$K_{\theta_1} = \begin{pmatrix} 1 & \dots & \exp\left(-\frac{\theta_1\|x_1 - x_M\|^2}{2}\right) & \dots & \int_{t_2}^{\tau} \exp\left(-\frac{\theta_1\|x_1 - t\|^2}{2}\right) dt \\ \vdots & \ddots & \vdots & \ddots & \vdots \\ \exp\left(-\frac{\theta_1\|x_M - x_1\|^2}{2}\right) & \dots & 1 & \dots & \int_{t_2}^{\tau} \exp\left(-\frac{\theta_1\|x_M - t\|^2}{2}\right) dt \\ \vdots & \ddots & \vdots & \ddots & \vdots \\ \int_{t_2}^{\tau} \exp\left(-\frac{\theta_1\|x_1 - t\|^2}{2}\right) dt & \dots & \int_{t_2}^{\tau} \exp\left(-\frac{\theta_1\|x_M - t\|^2}{2}\right) dt & \dots & \int_{t_2}^{\tau}\int_{t_2}^{\tau} \exp\left(-\frac{\theta_1\|s - t\|^2}{2}\right) ds\, dt \end{pmatrix}$$

The posterior distribution over θ_0 and θ_1 is given by

$$\begin{aligned} p(\theta_0, \theta_1 | \boldsymbol{\lambda}, \{x_i\}_{i=1}^m) &\propto p_{\theta_0}(\theta_0) p_{\theta_1}(\theta_1) \frac{\exp\left\{-\frac{\theta_0}{2}\boldsymbol{\lambda}^{\mathrm{T}} K_{\theta_1}^{-1}\boldsymbol{\lambda}\right\}}{\int_{\mathcal{F}} \exp\left\{-\frac{\theta_0}{2}\boldsymbol{\lambda}^{\mathrm{T}} K_{\theta_1}^{-1}\boldsymbol{\lambda}\right\} d\boldsymbol{\lambda}} \\ &\propto p_{\theta_0}(\theta_0) p_{\theta_1}(\theta_1) \frac{\sqrt{\theta_0}^{m+n} \exp\left\{-\frac{\theta_0}{2}\boldsymbol{\lambda}^{\mathrm{T}} K_{\theta_1}^{-1}\boldsymbol{\lambda}\right\}}{\int_{\mathcal{F}} \exp\left\{-\frac{1}{2}\sqrt{\theta_0}\boldsymbol{\lambda}^{\mathrm{T}} K_{\theta}^{-1}\sqrt{\theta_0}\boldsymbol{\lambda}\right\} d\sqrt{\theta_0}\boldsymbol{\lambda}} \\ &\propto p_{\theta_0}(\theta_0) p_{\theta_1}(\theta_1) \frac{\sqrt{\theta_0}^{m+n} \exp\left\{-\frac{\theta_0}{2}\boldsymbol{\lambda}^{\mathrm{T}} K_{\theta_1}^{-1}\boldsymbol{\lambda}\right\}}{\int_{\mathcal{F}} \exp\left\{-\frac{1}{2}\boldsymbol{z}^{\mathrm{T}} K_{\theta_1}^{-1}\boldsymbol{z}\right\} d\boldsymbol{z}} \end{aligned}$$

where $\boldsymbol{z} = \theta_0^{1/2}\boldsymbol{\lambda}$. We update θ_0 and θ_1 using a Gibbs sampling scheme. Conditional on θ_1, if we set $p_\theta(\theta) = \Gamma(\alpha, \beta)$, $p(\theta_0 \,|\, \theta_1, \boldsymbol{\lambda}, \{x_i\}_{i=1}^m)$ is conjugate and follows a Gamma distribution with parameters $\tilde{\alpha} = \alpha + (m+n)/2$ and $\tilde{\beta} = \beta + \boldsymbol{\lambda}^{\mathrm{T}} K_{\theta_1}^{-1}\boldsymbol{\lambda}/2$. Conditional on θ_0, we update θ_1 using a Metropolis-Hastings algorithm. The acceptance ratio involves computing $\int_{\mathcal{F}} \exp\left\{-\boldsymbol{z}^{\mathrm{T}} K_{\theta_1}^{-1}\boldsymbol{z}/2\right\} d\boldsymbol{z}$, which is approximated numerically using a Genz-type algorithm for multivariate normal probabilities (Genz & Bretz, 2009; Genz & Trinh, 2016), as implemented in the mvstdnormcdf function of the statsmodels Python package (Seabold et al., 2010). In constrast, for the boundary-corrected Brownian motion covariance $V_\theta = \tilde{C}/\theta$, the conditional posterior simplifies considerably. In this case, we only need to sample from

$$p(\theta | \boldsymbol{\lambda}, \{x_i\}_{i=1}^m) \propto p_\theta(\theta) \sqrt{\theta}^{m+n} \exp\left\{-\frac{\theta}{2}\boldsymbol{\lambda}^{\mathrm{T}} \tilde{C}^{-1}\boldsymbol{\lambda}\right\}.$$

By assign a Gamma prior $p_\theta(\theta) = \Gamma(\alpha, \beta)$, the posterior distribution of θ remain conjugate and is a Gamma distribution with parameters $\tilde{\alpha} = \alpha + (m+n)/2$ and $\tilde{\beta} = \beta + \boldsymbol{\lambda}^{\mathrm{T}} C^{-1}\boldsymbol{\lambda}/2$, which can be sampled exactly without requiring any numerical approximation.

5. Experiment Results

5.1. *Simulation results*

We first validate our implementation of Algorithm 1 by comparing simulations from Algorithm 1 with those obtained via rejection sampling. We simulated 1000 replicates of genealogies of 10 tips with both algorithms, for two scenarios: (1) $N_e = 1$ and $\tau = 0.5$, and (2) $N_e(t) = 25e^{-5t}$ and $\tau = 0.9$. Figure 2 shows histograms of T_3, the coalescent time when there are

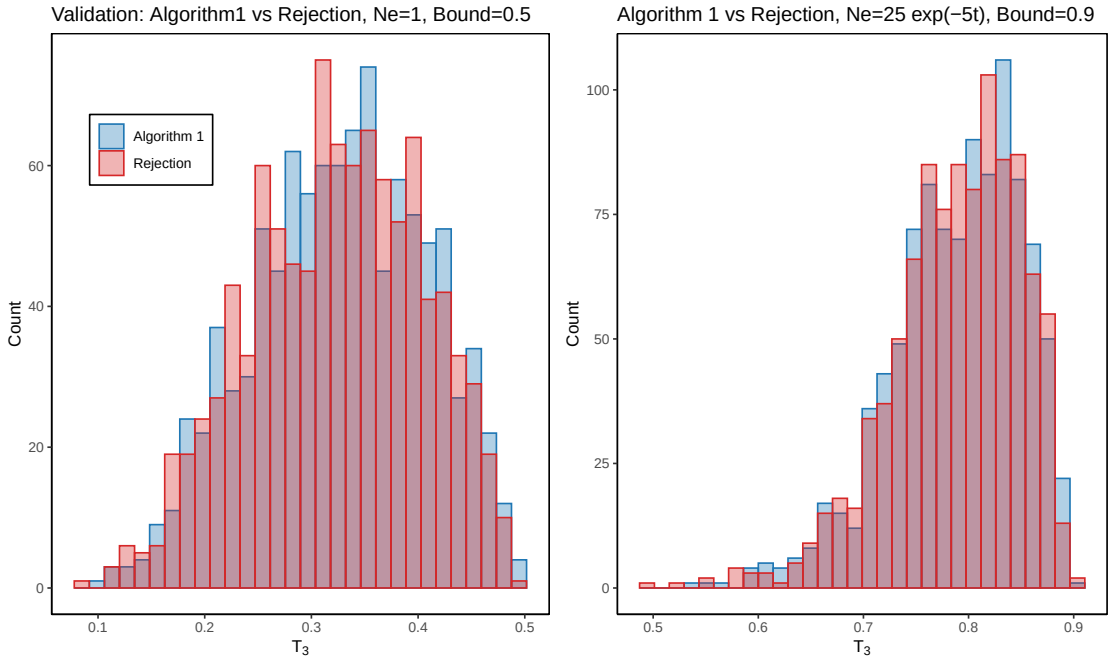

Figure 2: Comparison of Algorithm 1 and rejection sampling. Histograms of T_3 are based on 1000 realizations of genealogies with 10 tips under the bounded coalescent. The first panel corresponds to simulations with $N_e(t) = 1$, and $\tau = 0.5$, and the second panel corresponds to simulations with $N_e(t) = 25e^{-5t}$, and $\tau = 0.9$.

3 lineages. The total variation distances between the distributions produced by the two methods are 0.0310 and 0.028 in the two scenarios, respectively, while two independent distributions simulated with Algorithm 1 have an average total variation distance of 0.0370 and 0.0385 in each scenario across 100 simulations. Code and documentation for all methods are available at `https://github.com/bingjingle/boundedcoal`.

To evaluate the computational performance of our proposed simulation algorithm, we generated 3,000 replicates under $N_e(t) = 1$ with upper bounds $\tau = 0.50$ and $\tau = 1.00$, and under $N_e(t) = 25e^{-5t}$ with upper bounds $\tau = 0.55$ and $\tau = 0.71$, each for genealogies with 50 and 100 tips. We compared Algorithm 1 against rejection sampling and the simulation method of Carson et al. (Carson et al., 2022). For each method, we report the mean wall-clock time per genealogy. Because the method of Carson et al. is designed for constant effective population sizes, we apply it only to simulations with $N_e(t) = 1$. Results in Table 1 demonstrate the efficiency and superiority of Algorithm 1 across all settings.

The left panel of Fig. 3 shows the empirical distributions of the coalescent times of genealogies with 10 tips under a constant effective population size, comparing the bounded coalescent with $\tau = 0.5$ and the standard coalescent. Although the distributions of the early coalescent times (e.g. t_{10}, t_9) are nearly identical in the two models, the distributions of coalescent times close to the TMRCA are very distinct. These discrepancies are more pronounced in the right panel of Fig. 3, which displays the difference between the two distributions when $N_e(t) = 25e^{-5t}$ and $\tau = 0.5$. This observation is consistent with Carson et al.'s finding that ignoring the bound in the likelihood can bias $N_e(t)$ estimation.

5.2. *Synthetic datasets from standard coalescent models*

We simulated 30 datasets under the standard coalescent model for each of three effective population size trajectories: (1) $N_{e_1}(t) = 1$; (2) $N_{e_2}(t) = 3\exp\{-t\}$; (3) $N_{e_3}(t) = 25\exp\{-5t\}$,

Effective Population Size	Bound	Ntips	Methods	Running Time
$N_e(t) = 1$	0.50	50	Algorithm 1	0.002 s
			Naive rejection	0.161 s
			Carson et al.	0.002 s
		100	Algorithm 1	0.003 s
			Naive rejection	0.354 s
			Carson et al.	0.017 s
	1.00	50	Algorithm 1	0.001 s
			Naive rejection	0.008 s
			Carson et al.	0.003 s
		100	Algorithm 1	0.002 s
			Naive rejection	0.010 s
			Carson et al.	0.015 s
$N_e(t) = 25e^{-5t}$	0.55	50	Algorithm 1	0.004 s
			Naive rejection	> 201.600 s
		100	Algorithm 1	0.013 s
			Naive rejection	> 201.600 s
	0.71	50	Algorithm 1	0.003 s
			Naive rejection	4.548 s
		100	Algorithm 1	0.008 s
			Naive rejection	31.075 s

Table 1: Comparison of average running times for simulations under $N_e(t) = 1$ and $N_e(t) = 25e^{-5t}$ with two different upper bounds τ. Rows correspond to two numbers of tips ($n = 50$ and $n = 100$) and three simulation methods (Algorithm 1, naive rejection, and Carson et al. sampling).

with both 50 and 100 tips, where $N_{e_3}(t)$ exhibits a much steeper decline than $N_{e_2}(t)$. For each simulated dataset, we compared our proposed random integral (RI) method using both a Brownian motion kernel and a squard exponential kernel with the existing INLA approach, which empploys a Brownian motion kernel by default. Comparative results are summarized in Table 2. We evaluated the performance of these methods according to sum of squared errors (SSE), coverage, credible intervals width and average running time across the 30 simulated datasets (see their definitions and details in both Table 2 and Section 13).

In posterior inference, we require both the covariance matrix inverse (for hyperparameter updates) and its Cholesky decomposition (for updating $\boldsymbol{\lambda}$ via elliptical slice sampling). When the covariance matrix is ill-conditioned, both operations become unreliable. A common practical approach to stabilizing Gaussian process covariance matrices is to add a small jitter term to the diagonal. However, jitter selection involves a trade-off: jitter values that are too small fail to sufficiently reduce the condition number, whereas overly large jitter distorts the posterior geometry, both leading to poor mixing in Gaussian process posterior inference. A standard diagnostic strategy is therefore to choose the smallest jitter value that yields well-mixing trace plots.

Compared to Tang & Palacios (2024), the covariance matrix $\boldsymbol{V}$ in this study has a more complex structure and is therefore more prone to ill-conditioning, with smallest eigenvalues approaching zero. As a result, the simple jitter strategy described above is not always sufficient. In particular, for the random integral method with the boundary-corrected Brownian motion covariance, we require accurate computation of both the inverse and the Cholesky decomposition of $\tilde{C}$ (as defined in Equation (6)). Since the original Brownian motion covariance matrix C is typically severely ill-conditioned, a natural approach is to add diagonal jitter to C and replace C^{-1} in Equation (6) with

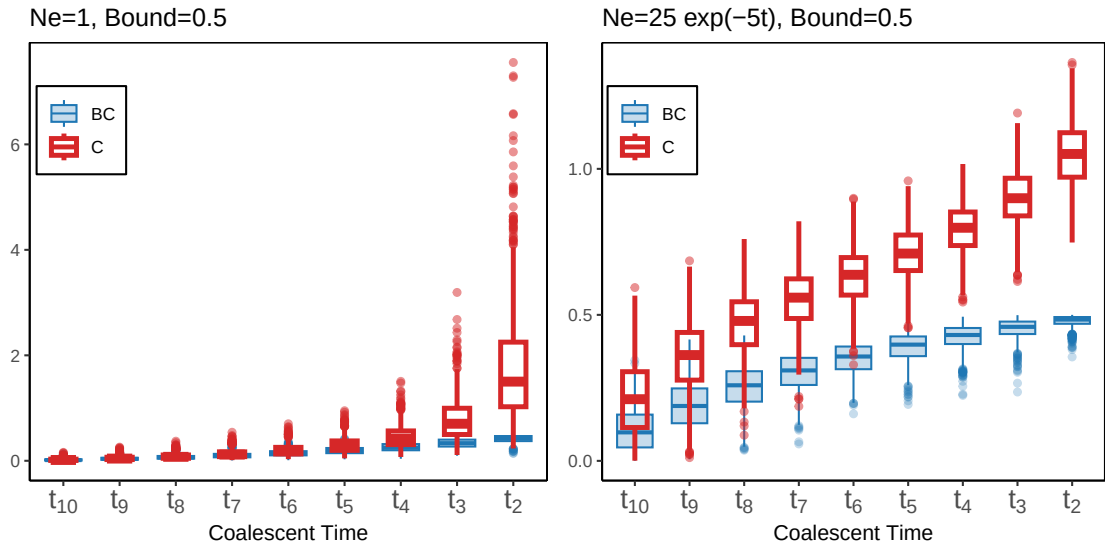

Figure 3: Boxplots of 1000 simulated coalescent times of genealogies with 10 tips under the bounded coalescent (BC, blue) with $\tau = 0.5$ and the standard coalescent (C, red) in two scenarios: (left) $N_e = 1$; (right) $N_e(t) = 25e^{-5t}$.

$C_0^{-1} := (C + \epsilon_0)^{-1}$. However, even when ϵ_0 is chosen to be the smallest value that guarantees invertibility, the resulting distortion of the covariance geometry can be substantial, leading to inaccurate estimates of both $\tilde{C}^{-1}$ and chol$(\tilde{C})$. To mitigate this issue, we employ the technique described in Section 12 to obtain more accurate estimates. Nevertheless, even this improved approach can fail in certain challenging cases; we illustrate such scenarios in Table 2 and Fig. 4.

Effective Population Size	Ntips	Methods	SSE at 100 Grids	Coverage at 100 Grids	Credible Interval Width	Running Time
$N_{e_1}(t)$	50	SC-INLA-BM	4.28 (3.24, 11.53)	100% (100%, 100%)	3.12 (1.63, 4.15)	0.70 ± 0.09 s
		SC-RI-BM	4.60 (1.91, 10.73)	100% (100%, 100%)	1.86 (1.41, 2.68)	1.58 ± 0.09 s
		SC-RI-SE	**2.58 (1.06, 8.07)**	100% (100%, 100%)	1.25 (0.83, 4.90)	749.21 ± 223.25 s
	100	SC-INLA-BM	2.48 (0.84, 7.61)	100% (100%, 100%)	1.88 (1.42, 2.80)	0.61 ± 0.08 s
		SC-RI-BM	2.86 (1.80, 7.55)	100% (100%, 100%)	1.71 (1.08, 2.08)	3.47 ± 1.45 s
		SC-RI-SE	**1.29 (0.30, 4.74)**	100% (100%, 100%)	1.09, 0.57, 4.78)	1388.51 ± 317.80 s
$N_{e_2}(t)$	50	SC-INLA-BM	20.69 (13.33, 33.99)	100% (100%, 100%)	3.35 (3.22, 3.86)	0.65 ± 0.05 s
		SC-RI-BM	**8.35 (6.71, 18.53)**	100% (100%, 100%)	3.49 (3.09, 4.02)	1.47 ± 0.05 s
		SC-RI-SE	17.73, (9.63, 26.82)	100% (98%, 100%)	3.47 (2.71, 5.24)	592.05 ± 172.02 s
	100	SC-INLA-BM	36.60 (8.97, 53.45)	100% (100%, 100%)	2.99 (2.73, 3.36)	0.73 ± 0.35 s
		SC-RI-BM	**10.82 (7.66, 19.33)**	100% (96%, 100%)	3.07 (2.84, 3.41)	2.25 ± 0.67 s
		SC-RI-SE	25.17 (12.07, 43.07)	97% (77%, 100%)	2.98 (2.17, 4.70)	1665.90 ± 302.85 s
$N_{e_3}(t)$	50	SC-INLA-BM	381.84 (240.66, 752.09)	100% (100%, 100%)	10.54 (9.47, 12.89)	0.68 ± 0.09 s
		SC-RI-BM	492.94 (336.67, 628.67)	62% (59%, 69%)	24.46 (19.44, 27.72)	2.43 ± 0.94 s
		SC-RI-SE	**369.07 (204.61, 957.61)**	72% (64%, 82%)	7.54, (6.49, 9.31)	713.03± 144.59 s
	100	SC-INLA-BM	233.81 (134.82, 434.61)	100% (100%, 100%)	6.63 (6.04, 7.41)	0.63 ± 0.07 s
		SC-RI-BM	500.64 (414.03, 842.48)	58% (55%, 62%)	18.83 (15.51, 23.22)	2.78 ± 1.03 s
		SC-RI-SE	**203.02 (141.38, 374.92)**	62% (58%, 70%)	3.75 (3.43, 4.88)	1459.47 ± 424.09 s

Table 2: Performance is evaluated over 100 grids (test points) across 30 simulations generated under effective population size trajectories $N_{e_1}(t)$, $N_{e_2}(t)$ and $N_{e_3}(t)$. For each dataset, inference for SC-RI-BM is based on 1,000,000 MCMC iterations following a burn-in of 1,000,000 iterations, whereas inference for SC-RI-SE is based on 100,000 MCMC iterations following a burn-in of 100,000 iterations. Columns 4–6 report results in the format: 0.50 quantile (0.25 quantile, 0.75 quantile). Boldface indicates the best-performing method among those compared. In the last column, we report the average running time and its standard deviation across the 30 simulated datasets, measured per 10,000 MCMC iterations for all MCMC-based methods; for INLA, we report the total running time. Additional details on the definitions and computation of the evaluation metrics are provided in Section 13.

Results for one simulated dataset with effective population size trajectories $N_{e_1}(t)$, $N_{e_2}(t)$ and $N_{e_3}(t)$ are depicted in Fig. 4. Although all Bayesian methods have comparable performance for

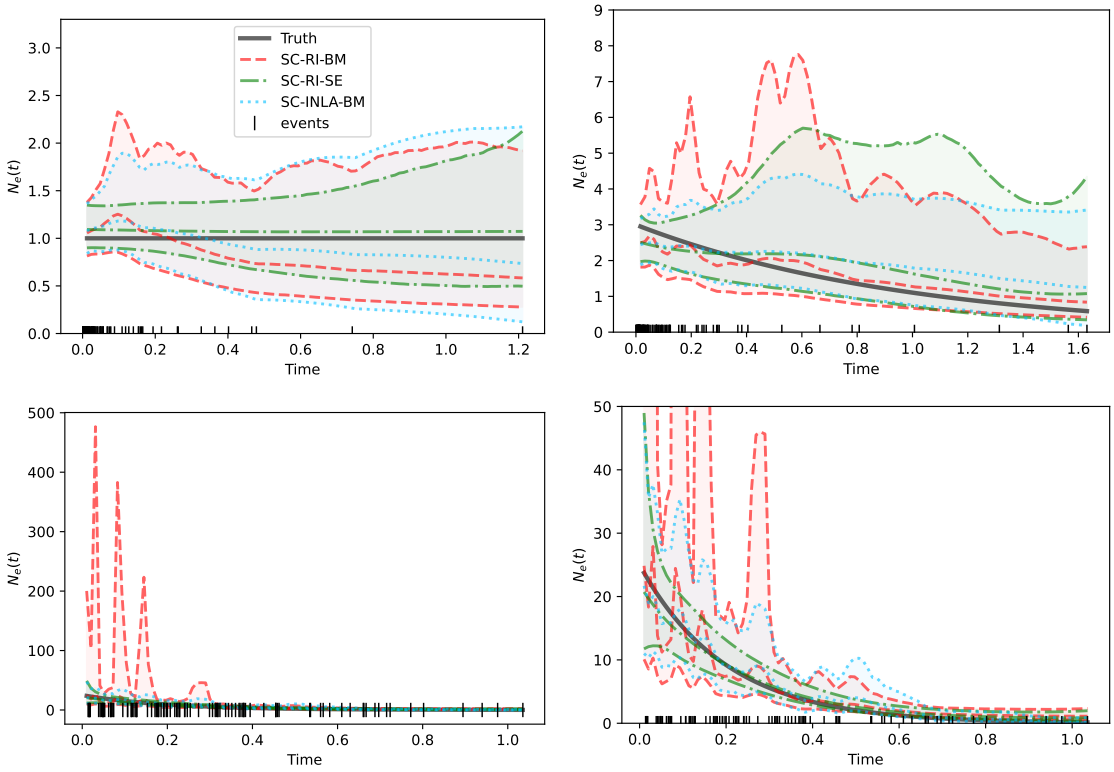

Figure 4: SC-X-BM refers to method X using a Brownian motion kernel under standard coalescent likelihood, while SC-X-SE denotes method X using a squared exponential kernel under standard coalescent likelihood. The top-left panel shows posterior inference of the effective population size trajectory $N_{e_1}(t)$ from one simulated dataset with 100 tips, while the top-right panel shows posterior inference of $N_{e_2}(t)$ from one simulated dataset with 100 tips.The two bottom panels both correspond to posterior inference of $N_{e_3}(t)$ from one simulated dataset with 100 tips, displayed with different y-axis ranges to highlight features at different scales. The true trajectories are depicted by solid black curves, posterior medians by dashed curves, and 95% credible intervals by shaded regions. The times of simulated events are indicated by tick marks along the bottom of each plot. Different methods are distinguished by colors as indicated in the legend, which is shared across all panels.

the first two effective population size trajectories, SC-RI-BM (red shaded region) shows much higher uncertainty and inaccuracy than any other method for $N_{e_3}(t)$. This is consistent with the quantitative results across 30 simulations in Table 2. The worst performance of SC-RI-BM could be due to several reasons: (1) as mentioned in the previous paragraph and discussed in Section 12, even the improved approach for computing matrices associated with $\tilde{C}$ performs poorly for datasets simulated under $N_{e_3}(t)$; (2) Compared to the squared exponential kernel, which has two hyperparameters—θ_0 controlling the scale and θ_1 controlling the smoothness (length scale) of the Gaussian process—the Brownian motion kernel is less flexible, as it involves only a single hyperparameter θ that simultaneously governs both scale and smoothness. For the steep trajectory $1/N_{e_3}(t)$, this coupling forces θ into an unfavorable compromise, resulting in poor estimation of both the overall scale and the local smoothness. As a consequence, SC-RI-BM exhibits excessive variability at early times and an insufficient scale to capture the tail behavior at later times, leading to reduced accuracy relative to SC-RI-SE. This effect is evident in the bottom-right panel of Fig. 4 as well as in the coverage results reported in Table 2; (3) Compared to SC-INLA-BM, which models $-\log N_{e_3}(t)$ using a Gaussian process with a Brownian motion kernel, SC-RI-BM instead places a Gaussian process with a Brownian motion kernel directly

on $1/N_{e_3}(t)$. Moreover, SC-RI-BM requires a more involved boundary-correction procedure due to the presence of multiple integral terms, which further exacerbates numerical instability in this setting. Performance statistics based on 30 simulations shown in Table 2 indicate that, in terms of SSE, SC-RI-SE always outperforms SC-INLA-BM and is the best performing method among those three methods for most cases. Yet SC-INLA-BM is fastest among those three in terms of running time. For simulated datasets from $N_{e_3}(t)$, both SC-RI-BM and SC-RI-SE fails to capture the tail of the ground-truth effective population size trajectory, therefore resulting in lower coverage than SC-INLA-BM.

5.3. *Synthetic datasets from bounded coalescent models*

We use the naive rejection sampling scheme to simulate 30 datasets under the bounded coalescent model for each of the same three effective population size trajectories, now incorporating an upper bound τ, with both 50 and 100 tips: (1) $N_{e_1}(t) = 1$, $\tau_1 = 1$; (2) $N_{e_2}(t) = 3\exp\{-t\}$, $\tau_2 = 0.7$; (3) $N_{e_3}(t) = 25\exp\{-5t\}$, $\tau_3 = 0.71$. For each simulated dataset, we compared the performance of our proposed random integral (RI) method—using both a Brownian motion kernel and a squared exponential kernel—under the bounded coalescent likelihood against its performance under the standard coalescent likelihood.

Results for one simulated dataset with effective population size trajectories $N_{e_1}(t)$, $N_{e_2}(t)$, and $N_{e_3}(t)$ under upper constraints τ_1, τ_2, and τ_3 are shown in Fig. 5. In most cases, the random integral method under the bounded coalescent likelihood outperforms its counterpart under the standard coalescent likelihood in terms of SSE and coverage, which is consistent with the quantitative results across 30 simulations reported in Table 3. For $N_{e_3}(t)$ with 100 tips, however, the random integral method with both a Brownian motion kernel and a squared exponential kernel under the standard coalescent likelihood exhibits smaller SSE values in Table 3. As illustrated in the bottom panels of Fig. 5, although the random integral method under the bounded coalescent likelihood behaves better overall, it fails to accurately capture the tail behavior, a limitation already observed in Fig. 4. This apparent advantage of the standard coalescent likelihood should not be interpreted as superior accuracy. First, because coalescent events are much sparser in the tail, inaccuracies in tail estimation are generally less consequential than inaccuracies in the earlier and intermediate portions of the trajectory. Second, under the standard coalescent likelihood, the inverse of the truncated Gaussian process actually estimates $N_e(t)/g_k(t,t,\tau)$ rather than $N_e(t)$. Since $N_e(t)/g_k(t,t,\tau)$ is always smaller than $N_e(t)$, the inverse of the truncated Gaussian process systematically underestimates the true effective population size, as evident across all panels in Fig. 5. Nevertheless, as shown in the bottom-right panel of Fig. 4, the inverse of the truncated Gaussian process under the standard coalescent likelihood tends to overestimate $N_e(t)/g_k(t,t,\tau)$ in the tail. Coincidentally, this overestimation produces tail behavior that is numerically closer to the true $N_e(t)$ than that obtained under the bounded coalescent likelihood, leading to the accidental improvement in SSE.

5.4. *Real example: DNA Typewriter lineage tracing data*

We analyze DNA Typewriter lineage-tracing data from a 25-day monoclonal expansion of HEK293T cells (Choi et al., 2022; Seidel et al., 2024). We focus on a random subset of 100 cells constructed by Seidel et al. (2024), and infer the lineage tree using the unweighted pair group method with arithmetic mean (UPGMA). Conditioning on this estimated tree and the 25-day upper bound on the time to the most recent common ancestor, we apply the random integral method under the bounded coalescent likelihood to perform posterior inference on the effective population size $N_e(t)$, using both a squared exponential kernel and a Brownian motion kernel. Time is plotted in reverse order, with day 25 corresponding to day 0 in real time; black

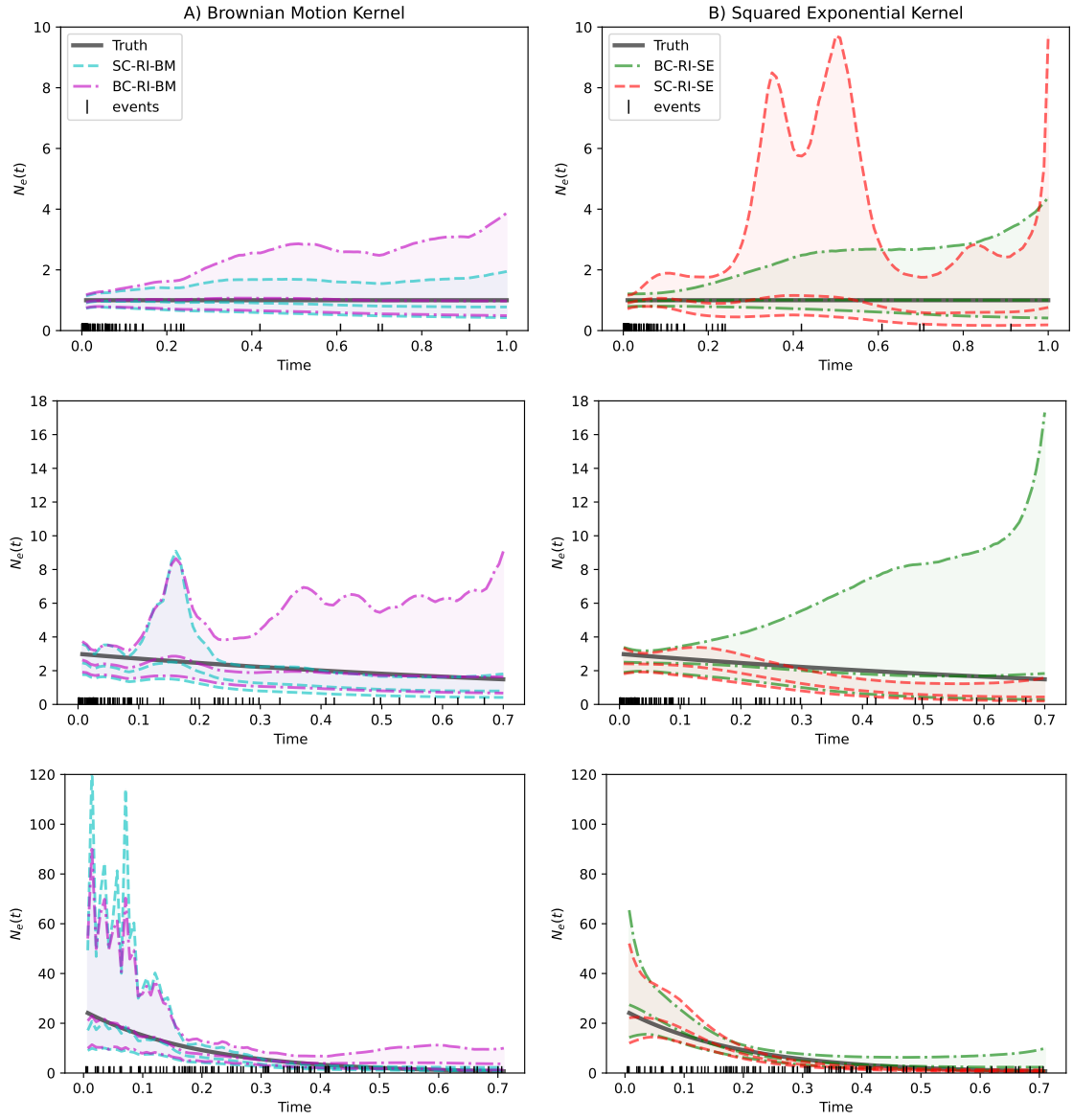

Figure 5: SC-RI-X refers to the random integral method using kernel X under the standard coalescent likelihood, while BC-RI-X denotes the random integral method using kernel X under the bounded coalescent likelihood. Rows from top to bottom show posterior inference of the effective population size trajectories $N_{e,1}(t)$, $N_{e,2}(t)$, and $N_{e,3}(t)$ from a single simulated dataset with 100 tips. The true trajectories are depicted by solid black curves, posterior medians by dashed curves, and 95% credible intervals by shaded regions. The times of simulated events are indicated by tick marks along the bottom of each plot. Different methods are distinguished by colors as described in the legend boxes, and plots within the same column share a common legend. Methods shown in panel A assume Brownian motion covariance kernels, whereas those in panel B assume squared exponential covariance kernels.

ticks therefore mark early cell expansion events near day 25, with no events observed before approximately day 13 (axis time). This highly uneven event structure leads to a severely ill-conditioned kernel covariance matrix. To address this issue, we perform posterior inference only on the observed events and subsequently make predictions on regular grids using the truncated conditional normal distribution. Equivalently, this corresponds to adopting a modified prior that

Effective Population Size	Ntips	Methods	SSE at 100 Grids	Coverage at 100 Grids	Credible Interval Width	Running Time
$N_{e_1}(t)$	50	BC-RI-BM	**4.46** (**2.06**, **7.46**)	100% (100%, 100%)	2.17 (1.93, 2.42)	1.81 ± 0.10 s
		SC-RI-BM	11.60 (8.04, 16.17)	100% (100%, 100%)	1.25 (1.12, 1.44)	3.15 ± 0.25 s
		BC-RI-SE	**2.16** (**0.70**, **3.94**)	100% (100%, 100%)	2.46 (1.09, 3.83)	860.61 ± 193.03 s
		SC-RI-SE	3.45 (0.46, 8.39)	100% (100%, 100%)	0.88 (0.63, 1.32)	807.72 ± 165.13 s
	100	BC-RI-BM	**2.53** (**1.19**, **4.95**)	100% (100%, 100%)	1.88 (1.76, 2.11)	2.64 ± 0.62 s
		SC-RI-BM	8.59 (5.00, 11.55)	100% (100%, 100%)	1.09 (1.01, 1.16)	4.40 ± 0.29 s
		BC-RI-SE	**0.32** (**0.14**, **1.32**)	100% (100%, 100%)	1.51 (0.91, 2.38)	1706.63 ± 584.14 s
		SC-RI-SE	1.76 (0.60, 4.83)	100% (100%, 100%)	0.94 (0.64, 1.40)	983.70 ± 133.78 s
$N_{e_2}(t)$	50	BC-RI-BM	**13.37** (**9.11**, **22.36**)	100% (100%, 100%)	5.15 (4.57, 5.49)	1.83 ± 0.04 s
		SC-RI-BM	60.87 (36.35, 71.99)	90% (76%, 100%)	1.68 (1.53, 1.91)	2.98 ± 0.12 s
		BC-RI-SE	**18.04** (**13.26**, **40.73**)	100% (100%, 100%)	7.42 (6.84, 8.31)	1255.52 ± 324.23 s
		SC-RI-SE	87.44 (76.03, 112.62)	50% (42%, 55%)	1.57 (1.39, 1.79)	1176.8 ± 245.64 s
	100	BC-RI-BM	**21.93** (**10.93**, **41.43**)	100% (100%, 100%)	5.01 (4.71, 5.75)	2.47 ± 0.08 s
		SC-RI-BM	46.76 (34.54, 64.46)	85% (60%, 100%)	1.82 (1.62, 2.21)	3.93 ± 0.73 s
		BC-RI-SE	**25.29** (**17.98**, **40.82**)	100% (100%, 100%)	6.97 (6.38, 7.99)	1421.92 ± 249.37 s
		SC-RI-SE	82.71 (65.31, 93.72)	44% (39%, 53%)	1.35 (1.26, 1.51)	1561.98 ± 415.60 s
$N_{e_3}(t)$	50	BC-RI-BM	**792.68** (617.27, 1474.27)	70% (61%, 65%)	32.56 (25.57, 37.72)	2.86 ± 0.84 s
		SC-RI-BM	824.54 (**557.52**, **1266.75**)	100% (98%, 100%)	27.68 (21.83, 34.76)	2.68 ± 0.96 s
		BC-RI-SE	**773.37** (604.65, 1875.43)	83% (68%, 90%)	15.24 (12.84, 21.88)	765.46 ± 153.22 s
		SC-RI-SE	1114.62 (**517.6**, **1636.88**)	70% (59%, 86%)	12.05 (8.83, 15.29)	644.93 ± 219.83 s
	100	BC-RI-BM	720.98 (603.5, 1023.02)	60% (56%, 65%)	14.18 (12.86, 16.98)	4.14 ± 1.41 s
		SC-RI-BM	**604.57** (**450.75**, **992.29**)	93% (88%, 95%)	15.98 (13.43, 19.42)	2.76 ± 1.15 s
		BC-RI-SE	436.98 (271.18, 902.31)	74% (67%, 83%)	8.28 (7.65, 9.28)	1135.68 ± 162.94 s
		SC-RI-SE	**375.39** (**151.81**, **540.59**)	74% (55%, 88%)	5.86 (5.29, 6.58)	1144.29 ± 83.55 s

Table 3: Performance is evaluated over 100 grid points across 30 simulated datasets generated under the effective population size trajectories $N_{e_1}(t)$, $N_{e_2}(t)$, and $N_{e_3}(t)$, with corresponding upper constraints τ_1, τ_2, and τ_3. For each dataset, inference for X-RI-BM is based on 1,000,000 MCMC iterations following a burn-in of 1,000,000 iterations, whereas inference for X-RI-SE is based on 100,000 MCMC iterations following a burn-in of 100,000 iterations. Columns 4–6 report results in the format 0.50 quantile (0.25 quantile, 0.75 quantile). Boldface indicates the best-performing method between the standard coalescent and bounded coalescent likelihoods when using the same kernel. In the last column, we report the average running time and its standard deviation across the 30 simulated datasets, measured per 10,000 MCMC iterations.

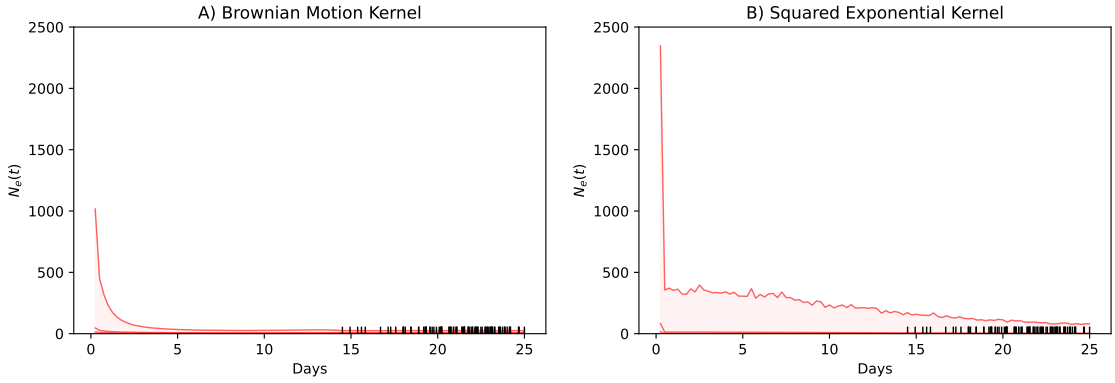

Figure 6: Both panels show the posterior median (red curve) and 95% credible intervals (shaded regions). Black ticks represent cell expansion events. The left panel corresponds to the Brownian motion kernel, while the right panel corresponds to the squared exponential kernel.

factorizes into a truncated normal distribution over the observed events and a truncated conditional normal distribution over grid locations given the observed events. The results are shown in Fig. 6. We find that both kernels yield estimates of $N_e(t)$ with similar overall shape, although the squared exponential kernel produces slightly higher estimates over most of the time interval.

6. DISCUSSION

In this work, we derive the first bounded coalescent model for time-varying effective population size trajectories and develop corresponding algorithms for simulation and posterior inference. We apply the proposed methods to DNA Typewriter lineage-tracing data from a 25-day monoclonal expansion of HEK293T cells. Several open issues remain for future investigation. First, in posterior inference using the random integral method with squared exponential kernels, we approximate the updates of kernel hyperparameters using multivariate normal cumulative distribution functions. The accuracy of this approximation deteriorates as the dimension increases, making it of interest to develop exact MCMC methods for hyperparameter inference. Due to the presence of multiple integrals in the bounded coalescent likelihood, existing approaches in empirical Bayes for kernel hyperparameter estimation such as Tang & Palacios (2024) and simple cross-validation schemes do not extend in a straightforward manner. Second, the current random integral method with both squared exponential and Brownian motion kernels does not adequately capture the tail behavior of steep effective population size trajectories. Improving tail performance is therefore an important direction for future work. Finally, it is of considerable interest to extend the random integral framework to scalable inference, for example by leveraging faster matrix inversion and Cholesky decomposition techniques.

DECLARATION OF THE USE OF GENERATIVE AI AND AI-ASSISTED TECHNOLOGIES

During the preparation of this work the author(s) used ChatGPT in order to assist with sentence polishing. After using this tool/service the author(s) reviewed and edited the content as necessary and take(s) full responsibility for the content of the publication.

ACKNOWLEDGEMENT

We thank Michael Howes for drawing our attention to the Touchard–Riordan numbers in the On-Line Encyclopedia of Integer Sequences (OEIS).

SUPPLEMENTARY MATERIAL

7. PHASE-TYPE DISTRIBUTIONS

Phase-type distributions are defined as the distribution of the time until absorption in a finite-state continuous-time Markov chain (CTMC). Specifically, assume $X(t)$ is a homogeneous CTMC with k transient states $1, \ldots, k$ and one absorbing state $k+1$, and $\boldsymbol{\pi} = (\boldsymbol{\alpha}, 0) \in \mathbb{R}^{k+1}$ is the initial distribution of the CTMC. The generator matrix of such a CTMC has the structure:

$$\mathbf{Q} = \begin{bmatrix} \mathbf{A} & \mathbf{a} \\ \mathbf{0} & \mathbf{0} \end{bmatrix}.$$

where $\boldsymbol{A}$ is named subintensity matrix, which is the generator associated to transient states. We use T to denote the time to absorption, that is, $T = \min\{t : X(t) = k+1, t \geq 0\}$, then $T \sim PH(\boldsymbol{\alpha}, \mathbf{A})$ (Horváth & Telek, 2024), and

$$\begin{aligned} \text{pr}(T < t) = \text{pr}(X(t) = k+1) &= \boldsymbol{\pi} e^{\mathbf{Q}t} \mathbf{e}_{k+1}^T \\ &= 1 - \boldsymbol{\alpha} e^{\mathbf{A}t} \mathbb{1} \end{aligned} \tag{7}$$

where $\mathbf{e}_{k+1}$ is the unit vector with all values 0 in the first k entries and a single 1 in the $k+1$ entry.

An inhomogeneous phase-type distribution with initial distribution $\boldsymbol{\pi} = (\boldsymbol{\alpha}, \mathbf{0})$ and time-varying generator $\mathbf{Q}(t)$ is denoted by $IPH(\boldsymbol{\alpha}, \mathbf{A}(t))$. The key difference between homogeneous and inhomogeneous phase-type distributions is that the generator matrix is time-varying:

$$\mathbf{Q}(t) = \begin{pmatrix} \mathbf{A}(t) & \mathbf{a}(t) \\ \mathbf{0} & \mathbf{0} \end{pmatrix}.$$

Albrecher & Bladt (2019) showed that for inhomogeneous phase-type distributions with generator matrix $\mathbf{Q}(t)$ and transient states generator $\mathbf{A}(t)$, the corresponding cumulative distribution of T can be expressed in terms of the product integral. In particular their Corollary 2.4 states that if $\mathbf{A}(t_1)$ and $\mathbf{A}(t_2)$ commute for all $t_1, t_2 \geq 0$, then

$$\text{pr}(T < t) = 1 - \boldsymbol{\alpha} e^{\int_0^t \mathbf{A}(u)du} \mathbb{1}. \tag{8}$$

Fortunately, this is the case for the coalescent with variable population size.

The coalescent with constant population size is a pure death process with $n-1$ transient states, corresponding to $n, \ldots, 2$ lineages, and a single absorbing state corresponding to one lineage, representing the most recent common ancestor. The initial distribution is $\boldsymbol{\pi} = (1, \underbrace{0, \ldots, 0}_{n-1})$, and the generator matrix is

$$\mathbf{Q} = \begin{pmatrix} -\binom{n}{2} & \binom{n}{2} & 0 & 0 & \cdots & 0 & 0 \\ 0 & -\binom{n-1}{2} & \binom{n-1}{2} & 0 & \cdots & 0 & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots & 0 & 0 \\ 0 & 0 & 0 & \cdots & -\binom{3}{2} & \binom{3}{2} & 0 \\ 0 & 0 & 0 & \cdots & 0 & -\binom{2}{2} & \binom{2}{2} \\ 0 & 0 & 0 & \cdots & 0 & 0 & 0 \end{pmatrix}. \tag{9}$$

The absorption time T is T_2, the TMRCA, whose cumulative distribution function is given in Equation (7). The coalescent with time-varying effective population size $N_e(t)$ is an inhomogeneous continuous time Markov chain with an initial distribution $\boldsymbol{\pi}$ and generator

$$\boldsymbol{Q}(t) = \frac{1}{N_e(t)} \boldsymbol{Q}.$$

We have now all background needed to prove the following proposition.

8. PROOFS

PROPOSITION 1. *Let T_2 denote the TMRCA of a standard coalescent tree with n tips evolving under the population effective size trajectory $N_e(t)$, then*

$$pr(T_2 \leq \tau \mid N_e(t)) = \sum_{j=1}^{n} r_{j,n}\, e^{-\binom{j}{2}\Lambda(\tau)},$$

where the coefficients are defined as

$$r_{j,n} := (-1)^{j-1}(2j-1)\frac{(n)_j}{(n-1+j)_j} \quad (j = 1, \ldots, n),$$

$\Lambda(\tau) \coloneqq \int_0^\tau \frac{1}{N_e(u)} du$, *and* $(x)_j \coloneqq x(x-1)\cdots(x-j+1) = \frac{x!}{(x-j)!}$.

Proof. The cumulative distribution function of the TMRCA T_2 can be obtained from Equation (8) and after some algebraic computation as

$$\text{pr}(T_2 < \tau \mid N_e(t)) = 1 - \boldsymbol{\alpha} \sum_{k=0}^{\infty} \frac{\Lambda(\tau)^k}{k!} \boldsymbol{A}^k \mathbb{1} = \boldsymbol{\pi} \exp\{\Lambda(\tau)\boldsymbol{Q}\}\ \mathbf{e}_n^T \tag{10}$$

where $\mathbf{e}_n$ denotes the unit vector whose first $n-1$ entries are zero and whose last entry equals one, $\Lambda(t) = \int_0^t du/N_e(u)$ and $\mathbf{A}$ is the submatrix obtained from $\mathbf{Q}$ in Equation (9) by eliminating the last column and the last row.

Through algebraic computation, the matrix $\boldsymbol{Q}$ admits the left-right eigenvector decomposition

$$\boldsymbol{Q} = \sum_{i=1}^{n} \lambda_i \boldsymbol{u}_i \boldsymbol{v}_i,$$

where the eigenvalues are

$$\lambda_i = -\binom{n-i+1}{2} \quad (i = 1, \ldots, n),$$

and the corresponding left and right eigenvectors satisfy $\boldsymbol{v}_i \boldsymbol{u}_j = \mathbb{1}(i = j)$. The i-th left eigenvector is a column vector:

$$\mathbf{u}_{k,i} = \begin{cases} 0 & k > i \\ 1 & k = i \\ \prod_{j=k}^{i-1} \frac{(n-j+1)(n-j)}{(i-j)(2n-j-i+1)} & k < i \end{cases}$$

and i-th right eigenvector is a row vector:

$$\mathbf{v}_{i,k} = \begin{cases} 0 & k < i \\ 1 & k = i \\ \prod_{j=i+1}^{k} -\frac{(n-j+2)(n-j+1)}{(j-i)(2n-j-i+1)} & k > i \end{cases}$$

Consequently,

$$\begin{aligned} \exp\{\Lambda(\tau)\boldsymbol{Q}\} &= \sum_{k=0}^{\infty} \frac{\Lambda(\tau)^k}{k!} \boldsymbol{Q}^k = \sum_{k=0}^{\infty} \frac{\Lambda(\tau)^k}{k!} \sum_{i=1}^{n} \lambda_i^k \mathbf{u_i} \mathbf{v_i}, \\ &= \sum_{i=1}^{n} \exp\{\Lambda(\tau)\lambda_i\}\, \mathbf{u_i} \mathbf{v_i}, \end{aligned}$$

Substituting this expression into Equation (10) yields

$$\begin{aligned} \text{pr}(T < \tau \mid N_e(t)) &= \boldsymbol{\pi} \left(\sum_{i=1}^{n} \exp\{\Lambda(\tau)\lambda_i\} \mathbf{u_i} \mathbf{v_i} \right) \mathbf{e}_n^T \\ &= \sum_{i=1}^{n} \exp\{\Lambda(\tau)\lambda_i\}\, \boldsymbol{\pi} \mathbf{u_i} \mathbf{v_i}\, \mathbf{e}_n^T \\ &= \sum_{i=1}^{n} \exp\left\{ -\binom{n-i+1}{2} \Lambda(\tau) \right\} \mathbf{u}_{1i} \mathbf{v}_{in}. \end{aligned}$$

A direct calculation (assuming $0! = 1$) gives

$$r^*_{i,n} = \mathbf{u}_{1i}\mathbf{v}_{in} = \frac{(-1)^{n-i} n!(n-1)!(2n-2i+1)}{(i-1)!(2n-i)!}$$

Re-indexing with $j = n - i + 1$, we obtain

$$\begin{aligned} r_{j,n} = r^*_{n-i+1,n} &= \frac{(-1)^{j-1} n!(n-1)!(2j-1)}{(n-j)!(n+j-1)!} \\ &= (-1)^{j-1}(2j-1)\frac{(n)_j}{(n-1+j)_j}. \end{aligned}$$

which completes the proof. □

Corollary 1. *For all integers $k = 2, \ldots, n$ and $r_{j,k}$ coefficients defined in Proposition 1 ,we have*

$$\begin{aligned} pr(T_2 \leq \tau \mid T_{k+1} = u, N_e(t)) &= pr(T_2 \leq \tau \mid T_k > u, N_e(t)), \\ &= \sum_{j=1}^{k} r_{j,k} \exp\left[\binom{j}{2}\{\Lambda(u) - \Lambda(\tau)\}\right]. \end{aligned}$$

Proof. These two quantities $\mathrm{pr}(T_2 \leq \tau \mid T_{k+1} = u, N_e(t))$ and $\mathrm{pr}(T_2 \leq \tau \mid T_k > u, N_e(t))$ are equal since they can both be interpreted as the cumulative distribution function of the TMRCA of a genealogy that starts at time u with k tips. Then, from Equation (8), we obtain

$$\begin{aligned} \mathrm{pr}(T_2 \leq \tau \mid T_{k+1} = u,\ N_e(t)) &= \mathrm{pr}(u \leq T_2 \leq \tau \mid T_{k+1} = u,\ N_e(t)) \\ &= 1 - \boldsymbol{\alpha}_k e^{\int_u^\tau \mathbf{A}(t)dt}\mathbb{1}, \end{aligned}$$

with initial probability $\boldsymbol{\alpha}_k = (1, 0, \ldots, 0)$ a vector of size k. The result then follows from Proposition 1, by replacing $\Lambda(\tau) = \int_0^\tau dt/N_e(t)$ with $\Lambda(\tau) - \Lambda(u) = \int_u^\tau dt/N_e(t)$. □

Lemma 1. *For all integers $k = 2, \ldots, n$, and coefficients $r_{j,k}$ defined in Proposition 1, the polynomial on $x \in \mathbb{R}$ given by*

$$\sum_{j=1}^{k} r_{j,k}\, x^{\binom{j}{2}}$$

admits the factorization

$$\sum_{j=1}^{k} r_{j,k}\, x^{\binom{j}{2}} = (1-x)^{k-1}\left(\sum_{i=0}^{M_k} a_{i,k}\, x^i\right),$$

where

$$M_k = \binom{k-1}{2},$$

and the coefficients $\{a_{i,k}\}$ *are defined recursively as*

$$a_{i,k} = \begin{cases} \displaystyle\sum_{s=1}^{k-1} (-1)^{s+1} \binom{k-1}{s} a_{i-s,k} \;+\; \sum_{j=1}^{k} r_{j,k}\, \mathbb{1}\left\{\binom{j}{2} = i\right\} & (i = 0, \ldots, M_k), \\ 0 & \textit{otherwise}. \end{cases}$$

Proof. We first use induction to prove that the polynomial $f_k(x) \coloneqq \sum_{j=1}^{k} r_{j,k} x^{\binom{j}{2}}$ is divisible by $(1-x)^{k-1}$ and then obtain expressions for $a_{i,k}$ by applying the polynomial long-division algorithm.

We begin with an induction proof of the statement that

$$S_{j,k} \coloneqq \sum_{q=1}^{j} r_{q,k} = (-1)^{j-1} j \prod_{m=1}^{j-1} \frac{k-1-m}{k+m} \quad (j \in \{2, \ldots, k\};\ k \in \{2, \ldots, n\}) \tag{11}$$

Base Case. Let $j = 2$. Then

$$S_{2,k} = r_{1,k} + r_{2,k} = 1 - 3\frac{k-1}{k+1} = -2\,\frac{k-2}{k+1}.$$

This agrees with Equation (11) for $j = 2$, completing the base case.

Inductive hypothesis: Assume the statement holds for $j = i$, namely

$$S_{i,k} = (-1)^{i-1} i \prod_{m=1}^{i-1} \frac{k-1-m}{k+m}.$$

Inductive step: We aim to show that the statement holds for $j = i+1$, that is,

$$S_{i+1,k} = (-1)^{i} (i+1) \prod_{m=1}^{i} \frac{k-1-m}{k+m}.$$

Recall that

$$r_{i+1,k} = (-1)^{i+1-1}(2i+2-1)\frac{(k)_{i+1}}{(k-1+i+1)_{i+1}} = (-1)^i (2i+1) \prod_{m=1}^{i} \frac{k-m}{k+m}.$$

Then

$$\begin{aligned} S_{i+1,k} &= S_{i,k} + r_{i+1,k} \\ &= (-1)^{i-1} i \prod_{m=1}^{i-1} \frac{k-1-m}{k+m} + (-1)^i (2i+1) \prod_{m=1}^{j} \frac{k-m}{k+m} \\ &= \frac{(-1)^i}{\prod_{m=1}^{i} k+m} \left[-i(k+i) \left(\prod_{m=1}^{i-1} k-1-m \right) + (2i+1) \left(\prod_{m=1}^{i} k-m \right) \right] \\ &= \frac{(-1)^i}{\prod_{m=1}^{i} k+m} (i+1) \left(\prod_{m=1}^{i} k-1-m \right) \\ &= (-1)^i (i+1) \prod_{m=1}^{i} \frac{k-1-m}{k+m} \end{aligned}$$

This completes the inductive step.

In particular, taking $j = k$ in Equation (11) yields

$$\sum_{q=1}^{k} r_{q,k} = 0 \quad (k = 2, \ldots, n). \tag{12}$$

Next, we establish the central step of the proof, showing that all derivatives of order up to $k-2$ of the function f_k vanish at 1, namely,

$$f_k^{(s)}(1) = 0, \qquad \forall\, s = 0, 1, \ldots, k-2. \tag{13}$$

In general, for $s = 0, \ldots, k-2$, we have

$$\begin{aligned} f_k^{(s)}(x) &= \sum_{j=1}^{k} r_{j,k} \prod_{m=0}^{s-1} \left\{ \binom{j}{2} - m \right\} x^{\binom{j}{2}-s} \mathbb{1}\left\{ \binom{j}{2} \geq s \right\} \\ &= \sum_{\substack{1 \leq j \leq k \\ \binom{j}{2} \geq s}} r_{j,k} \prod_{m=0}^{s-1} \left\{ \binom{j}{2} - m \right\} x^{\binom{j}{2}-s}. \end{aligned}$$

Here we adopt the convention that

$$\prod_{m=0}^{-1} \left\{ \binom{j}{2} - m \right\} = 1,$$

which corresponds to the case $s = 0$. By Equation (12), it follows that $f_k^{(0)}(1) = 0$ for all $k = 2, \ldots, n$. We now proceed with the induction step.

Base Case: $f_2^{(0)}(1) = 0$ and $f_3^{(0)}(1) = 0$.

Inductive hypothesis: Assume that $f_k^{(s)}(1) = 0$ and $f_{k+1}^{(s)}(1) = 0$, that is,

$$\sum_{\substack{1 \leq j \leq k \\ \binom{j}{2} \geq s}} r_{j,k} \prod_{m=0}^{s-1} \left\{ \binom{j}{2} - m \right\} = 0, \tag{14}$$

$$\sum_{\substack{1 \leq j \leq k+1 \\ \binom{j}{2} \geq s}} r_{j,k+1} \prod_{m=0}^{s-1} \left\{ \binom{j}{2} - m \right\} = 0. \tag{15}$$

Inductive step: We aim to show that $f_{k+1}^{(s+1)}(1) = 0$, that is,

$$\sum_{\substack{1 \leq j \leq k+1 \\ \binom{j}{2} \geq s+1}} r_{j,k+1} \prod_{m=0}^{s} \left\{ \binom{j}{2} - m \right\} = 0.$$

Denote

$$j^* := \min \left\{ j \in \mathbb{Z} : \binom{j}{2} \geq s \right\}.$$

If $\binom{j^*}{2} > s$, then

$$f_{k+1}^{(s+1)}(1) = \sum_{j=j^*}^{k+1} r_{j,k+1} \prod_{m=0}^{s} \left\{ \binom{j}{2} - m \right\}.$$

If $\binom{j^*}{2} = s$, then

$$\begin{aligned} f_{k+1}^{(s+1)}(1) &= \sum_{j=j^*+1}^{k+1} r_{j,k+1} \prod_{m=0}^{s} \left\{ \binom{j}{2} - m \right\} \\ &= r_{j^*,k+1} \prod_{m=0}^{s} \left\{ \binom{j^*}{2} - m \right\} + \sum_{j=j^*+1}^{k+1} r_{j,k+1} \prod_{m=0}^{s} \left\{ \binom{j}{2} - m \right\} \\ &= \sum_{j=j^*}^{k+1} r_{j,k+1} \prod_{m=0}^{s} \left\{ \binom{j}{2} - m \right\}, \end{aligned}$$

where the first term vanishes since $\binom{j^*}{2} - s = 0$. In summary, it suffices to show that

$$\sum_{j=j^*}^{k+1} r_{j,k+1} \prod_{m=0}^{s} \left\{ \binom{j}{2} - m \right\} = 0.$$

$$\begin{aligned} \sum_{j=j^*}^{k+1} r_{j,k+1} \left\{ \prod_{m=0}^{s} \binom{j}{2} - m \right\} &= \sum_{j=j^*}^{k} r_{j,k+1} \left\{ \prod_{m=0}^{s} \binom{j}{2} - m \right\} \; + \; r_{k+1,k+1} \left\{ \prod_{m=0}^{s} \binom{j}{2} - m \right\} \\ &= \sum_{j=j^*}^{k} r_{j,k+1} \left\{ \prod_{m=0}^{s-1} \binom{j}{2} - m \right\} \left\{ \binom{j}{2} - s \right\} \; + \; r_{k+1,k+1} \left\{ \prod_{m=0}^{s-1} \binom{k+1}{2} - m \right\} \left\{ \binom{k+1}{2} - s \right\} \\ &= \sum_{j=j^*}^{k} r_{j,k+1} \left\{ \prod_{m=0}^{s-1} \binom{j}{2} - m \right\} \left\{ \binom{j}{2} - s \right\} - \sum_{j=j^*}^{k} r_{j,k+1} \left\{ \prod_{m=0}^{s-1} \binom{j}{2} - m \right\} \left\{ \binom{k+1}{2} - s \right\} \\ &= \sum_{j=j^*}^{k} r_{j,k+1} \left\{ \prod_{m=0}^{s-1} \binom{j}{2} - m \right\} \left\{ \binom{j}{2} - \binom{k+1}{2} \right\} \\ &= -\binom{k+1}{2} \sum_{j=j^*}^{k} r_{j,k} \left\{ \prod_{m=0}^{s-1} \binom{j}{2} - m \right\} \\ &= -\binom{k+1}{2} \sum_{j=j^*}^{k} r_{j,k} \left\{ \prod_{m=0}^{s-1} \binom{j}{2} - m \right\} \\ &= 0. \end{aligned}$$

From the second to the third line, we apply Equation (15). From the fourth to the fifth line, we use the identity

$$r_{j,k+1} = r_{j,k} \frac{\binom{k+1}{2}}{\binom{k+1}{2} - \binom{j}{2}},$$

which follows directly from the definition of $r_{j,k}$ in Proposition 1. The final equality follows from Equation (14). This completes the induction and thereby establishes the statement in Equation (13).

By Taylor expansion about $x = 1$, we have

$$f_k(x) = \sum_{s=0}^{\binom{k}{2}} \frac{f_k^{(s)}(1)}{s!} (x-1)^s.$$

According to Equation (13), all derivatives of order up to $k-2$ vanish at $x = 1$, that is, $f_k^{(s)}(1) = 0$ for $s = 0, 1, \ldots, k-2$. Therefore, the Taylor expansion of $f_k(x)$ about $x = 1$ starts at order $(x-1)^{k-1}$, and hence $f_k(x)$ is divisible by $(1-x)^{k-1}$.

Finally, by applying the polynomial long-division algorithm in ascending degree order, we obtain explicit expressions for the coefficients $a_{i,k}$.

Proposition 2.

$$\frac{\sum_{j=1}^{k-1} r_{j,k-1} x^{\binom{j}{2}}}{\sum_{j=1}^{k} r_{j,k} x^{\binom{j}{2}}} \le \frac{1}{1-x} \quad (x \in [0,1];\ k = 3, \ldots, n).$$

Proof. We denote the two polynomials in Lemma 1 as

$$f_k(x) \coloneqq \sum_{j=1}^{k} r_{j,k}\, x^{\binom{j}{2}}, \quad g_k(x) \coloneqq \sum_{i=0}^{\binom{k-1}{2}} a_{i,k}\, x^i.$$

Our goal is to show that

$$\frac{f_{k-1}(x)}{f_k(x)} \le \frac{1}{1-x}.$$

By applying Lemma 1, it suffices to prove that $g_{k-1}(x) \le g_k(x)$. We establish this result in four main steps by analyzing the relationship between $a_{i,k}$ and $a_{i,k-1}$.

(i) We first state an intermediate conclusion about $f_k(x)$:

$$f_k(x) - f_{k-1}(x) = \frac{x f_k'(x)}{\binom{k}{2}} \quad (k = 3, 4, \ldots, n) \tag{16}$$

Recall that

$$\begin{aligned} r_{j,k} &\coloneqq (-1)^{j-1}(2j-1)\frac{(k)_j}{(k-1+j)_j} \\ &= (-1)^{j-1}(2j-1)\prod_{m=1}^{j-1}\frac{k-m}{k+m}. \end{aligned}$$

In this section, we use the above simplified expression for $r_{j,k}$. As before, we adopt the convention that for the case $j = 1$,

$$\prod_{m=1}^{0} \frac{k-m}{k+m} = 1.$$

Below we provide the proof for Equation (16).

$$\begin{aligned}
\text{LHS} &= f_k(x) - f_{k-1}(x) \\
&= \sum_{j=1}^{k} r_{j,k}\, x^{\binom{j}{2}} - \sum_{j=1}^{k-1} r_{j,k-1}\, x^{\binom{j}{2}} \\
&= \sum_{j=2}^{k-1} \left(r_{j,k} - r_{j,k-1}\right) x^{\binom{j}{2}} + r_{k,k}\, x^{\binom{k}{2}} \\
&= \sum_{j=2}^{k-1} (-1)^{j-1}(2j-1)\, x^{\binom{j}{2}} \left(\prod_{m=1}^{j-1} \frac{k-m}{k+m} - \prod_{m=1}^{j-1} \frac{k-1-m}{k-1+m} \right) + (-1)^{k-1}(2k-1)\, x^{\binom{k}{2}} \prod_{m=1}^{k-1} \frac{k-m}{k+m} \\
&= \sum_{j=2}^{k-1} (-1)^{j-1}(2j-1)\, x^{\binom{j}{2}} \left(1 - \frac{(k-j)(k+j-1)}{k(k-1)} \right) \prod_{m=1}^{j-1} \frac{k-m}{k+m} + (-1)^{k-1}(2k-1)\, x^{\binom{k}{2}} \prod_{m=1}^{k-1} \frac{k-m}{k+m} \\
&= \sum_{j=2}^{k-1} (-1)^{j-1}(2j-1)\, x^{\binom{j}{2}} \frac{j(j-1)}{k(k-1)} \prod_{m=1}^{j-1} \frac{k-m}{k+m} + (-1)^{k-1}(2k-1)\, x^{\binom{k}{2}} \prod_{m=1}^{k-1} \frac{k-m}{k+m} \\
&= \frac{1}{\binom{k}{2}} \sum_{j=2}^{k} (-1)^{j-1}(2j-1) \binom{j}{2} x^{\binom{j}{2}} \prod_{m=1}^{j-1} \frac{k-m}{k+m}.
\end{aligned}$$

$$\begin{aligned}
\text{RHS} &= \frac{x f_k'(x)}{\binom{k}{2}} \\
&= \frac{x}{\binom{k}{2}} \left[\sum_{j=1}^{k} (-1)^{j-1}(2j-1)\, x^{\binom{j}{2}} \prod_{m=1}^{j-1} \frac{k-m}{k+m} \right]' \\
&= \frac{1}{\binom{k}{2}} \sum_{j=1}^{k} (-1)^{j-1}(2j-1) \binom{j}{2} x^{\binom{j}{2}} \prod_{m=1}^{j-1} \frac{k-m}{k+m}.
\end{aligned}$$

Hence Equation (16) is proved.

(ii) Applying Lemma 1, this intermediate result can be expressed in terms of $a_{i,k}$, $a_{i-1,k}$, and $a_{i,k-1}$.

$$(1-x)^{k-1} g_k(x) - (1-x)^{k-2} g_{k-1}(x) = \frac{x}{\binom{k}{2}} \left[(1-x)^{k-1} g_k(x) \right]' \quad \forall x \in \mathbb{R}$$

$$(1-x)^{k-1} g_k(x) - (1-x)^{k-2} g_{k-1}(x) = \frac{x}{\binom{k}{2}} \left[(1-x)^{k-1} g_k'(x) - (k-1)(1-x)^{k-2} g_k(x) \right] \quad \forall x \in \mathbb{R}$$

$$\binom{k}{2}(1-x) g_k(x) - \binom{k}{2} g_{k-1}(x) = x(1-x) g_k'(x) - (k-1) x g_k(x) \quad \forall x \neq 1$$

$$\left[\binom{k}{2}(1-x)+(k-1)x\right]g_k(x)-\binom{k}{2}g_{k-1}(x)=x(1-x)g_k'(x)\quad\forall x\neq 1$$

$$\left[\binom{k}{2}(1-x)+(k-1)x\right]\sum_{i=0}^{\binom{k-1}{2}}a_k^i x^i-\binom{k}{2}\sum_{i=0}^{\binom{k-2}{2}}a_{k-1}^i x^i=x(1-x)\sum_{i=1}^{\binom{k-1}{2}}ia_k^i x^{i-1}\quad\forall x\neq 1$$

$$\binom{k}{2}\sum_{i=0}^{\binom{k-1}{2}}a_k^i x^i(1-x)+(k-1)\sum_{i=0}^{\binom{k-1}{2}}a_k^i x^{i+1}-\binom{k}{2}\sum_{i=0}^{\binom{k-2}{2}}a_{k-1}^i x^i=\sum_{i=1}^{\binom{k-1}{2}}ia_k^i x^i(1-x)\quad\forall x\neq 1$$

$$\sum_{i=0}^{\binom{k-1}{2}}\left[\binom{k}{2}-i\right]a_k^i x^i(1-x)+(k-1)\sum_{i=0}^{\binom{k-1}{2}}a_k^i x^{i+1}-\binom{k}{2}\sum_{i=0}^{\binom{k-2}{2}}a_{k-1}^i x^i=0\quad\forall x\neq 1$$

$$\sum_{i=0}^{\binom{k-1}{2}}\left[\binom{k}{2}-i\right]a_k^i x^i-\sum_{i=0}^{\binom{k-1}{2}}\left[\binom{k}{2}-i\right]a_k^i x^{i+1}+(k-1)\sum_{i=0}^{\binom{k-1}{2}}a_k^i x^{i+1}-\binom{k}{2}\sum_{i=0}^{\binom{k-2}{2}}a_{k-1}^i x^i=0\quad\forall x\neq 1$$

$$\sum_{i=0}^{\binom{k-1}{2}}\left[\binom{k}{2}-i\right]a_k^i x^i+\sum_{i=0}^{\binom{k-1}{2}}\left[k-1-\binom{k}{2}+i\right]a_k^i x^{i+1}-\binom{k}{2}\sum_{i=0}^{\binom{k-2}{2}}a_{k-1}^i x^i=0\quad\forall x\neq 1$$

$$\sum_{i=0}^{\binom{k-1}{2}}\left[\binom{k}{2}-i\right]a_k^i x^i+\sum_{i=1}^{\binom{k-1}{2}+1}\left[k-1-\binom{k}{2}+i-1\right]a_k^{i-1}x^i-\binom{k}{2}\sum_{i=0}^{\binom{k-2}{2}}a_{k-1}^i x^i=0\quad\forall x\neq 1$$

$$\sum_{i=0}^{\binom{k-1}{2}}\left[\binom{k}{2}-i\right]a_k^i x^i+\sum_{i=1}^{\binom{k-1}{2}+1}\left[-\binom{k-1}{2}+i-1\right]a_k^{i-1}x^i-\binom{k}{2}\sum_{i=0}^{\binom{k-2}{2}}a_{k-1}^i x^i=0\quad\forall x\neq 1$$

$$\sum_{i=0}^{\binom{k-1}{2}}\left[\binom{k}{2}-i\right]a_k^i x^i+\sum_{i=1}^{\binom{k-1}{2}}\left[-\binom{k-1}{2}+i-1\right]a_k^{i-1}x^i-\binom{k}{2}\sum_{i=0}^{\binom{k-2}{2}}a_{k-1}^i x^i=0\quad\forall x\neq 1$$

$$\binom{k}{2}\left(a_k^0-a_{k-1}^0\right)+\sum_{i=1}^{\binom{k-2}{2}}\left\{\left[\binom{k}{2}-i\right]a_k^i+\left[-\binom{k-1}{2}+i-1\right]a_k^{i-1}-\binom{k}{2}a_{k-1}^i\right\}x^i$$

$$+\sum_{i=\binom{k-2}{2}+1}^{\binom{k-1}{2}}\left\{\left[\binom{k}{2}-i\right]a_k^i+\left[-\binom{k-1}{2}+i-1\right]a_k^{i-1}\right\}x^i=0\quad\forall x\neq 1$$

$$(1-x)^{k-1}g_k(x)-(1-x)^{k-2}g_{k-1}(x)=\frac{x}{\binom{k}{2}}\left[(1-x)^{k-1}g_k(x)\right]'\quad\forall x\in\mathbb{R}$$

$$(1-x)^{k-1}g_k(x)-(1-x)^{k-2}g_{k-1}(x)=\frac{x}{\binom{k}{2}}\left[(1-x)^{k-1}g_k'(x)-(k-1)(1-x)^{k-2}g_k(x)\right]\quad\forall x\in\mathbb{R}$$

$$\binom{k}{2}(1-x)g_k(x)-\binom{k}{2}g_{k-1}(x)=x(1-x)g_k'(x)-(k-1)xg_k(x)\quad\forall x\neq 1$$

$$\left[\binom{k}{2}(1-x)+(k-1)x\right]g_k(x)-\binom{k}{2}g_{k-1}(x)=x(1-x)g_k'(x)\quad\forall x\neq 1$$

$$\left[\binom{k}{2}(1-x)+(k-1)x\right]\sum_{i=0}^{\binom{k-1}{2}}a_{i,k}x^i-\binom{k}{2}\sum_{i=0}^{\binom{k-2}{2}}a_{i,k-1}x^i=x(1-x)\sum_{i=1}^{\binom{k-1}{2}}ia_{i,k}x^{i-1}\quad\forall x\neq 1$$

$$\binom{k}{2}\sum_{i=0}^{\binom{k-1}{2}}a_{i,k}x^i(1-x)+(k-1)\sum_{i=0}^{\binom{k-1}{2}}a_{i,k}x^{i+1}-\binom{k}{2}\sum_{i=0}^{\binom{k-2}{2}}a_{i,k-1}x^i=\sum_{i=1}^{\binom{k-1}{2}}ia_{i,k}x^i(1-x)\quad\forall x\neq 1$$

$$\sum_{i=0}^{\binom{k-1}{2}}\left[\binom{k}{2}-i\right]a_{i,k}x^i(1-x)+(k-1)\sum_{i=0}^{\binom{k-1}{2}}a_{i,k}x^{i+1}-\binom{k}{2}\sum_{i=0}^{\binom{k-2}{2}}a_{i,k-1}x^i=0\quad\forall x\neq 1$$

$$\sum_{i=0}^{\binom{k-1}{2}}\left[\binom{k}{2}-i\right]a_{i,k}x^i-\sum_{i=0}^{\binom{k-1}{2}}\left[\binom{k}{2}-i\right]a_{i,k}x^{i+1}+(k-1)\sum_{i=0}^{\binom{k-1}{2}}a_{i,k}x^{i+1}-\binom{k}{2}\sum_{i=0}^{\binom{k-2}{2}}a_{i,k-1}x^i=0\quad\forall x\neq 1$$

$$\sum_{i=0}^{\binom{k-1}{2}}\left[\binom{k}{2}-i\right]a_{i,k}x^i+\sum_{i=0}^{\binom{k-1}{2}}\left[k-1-\binom{k}{2}+i\right]a_{i,k}x^{i+1}-\binom{k}{2}\sum_{i=0}^{\binom{k-2}{2}}a_{i,k-1}x^i=0\quad\forall x\neq 1$$

$$\sum_{i=0}^{\binom{k-1}{2}}\left[\binom{k}{2}-i\right]a_{i,k}x^i+\sum_{i=1}^{\binom{k-1}{2}+1}\left[k-1-\binom{k}{2}+i-1\right]a_{i-1,k}x^i-\binom{k}{2}\sum_{i=0}^{\binom{k-2}{2}}a_{i,k-1}x^i=0\quad\forall x\neq 1$$

$$\sum_{i=0}^{\binom{k-1}{2}}\left[\binom{k}{2}-i\right]a_{i,k}x^i+\sum_{i=1}^{\binom{k-1}{2}}\left[-\binom{k-1}{2}+i-1\right]a_{i-1,k}x^i-\binom{k}{2}\sum_{i=0}^{\binom{k-2}{2}}a_{i,k-1}x^i=0\quad\forall x\neq 1$$

$$\binom{k}{2}(a_{0,k}-a_{0,k-1})+\sum_{i=1}^{\binom{k-2}{2}}\left\{\left[\binom{k}{2}-i\right]a_{i,k}+\left[-\binom{k-1}{2}+i-1\right]a_{i-1,k}-\binom{k}{2}a_{i,k-1}\right\}x^i$$

$$+\sum_{i=\binom{k-2}{2}+1}^{\binom{k-1}{2}}\left\{\left[\binom{k}{2}-i\right]a_{i,k}+\left[-\binom{k-1}{2}+i-1\right]a_{i-1,k}\right\}x^i=0\quad\forall x\neq 1$$

By setting all coefficients of x^i to be zero, we obtain the following general equation for the $a_{i,k}$'s:

$$a_{i,k}=\frac{\left[\binom{k-1}{2}-i+1\right]a_{i-1,k}+\binom{k}{2}a_{i,k-1}}{\binom{k}{2}-i},\quad\left(k=3,\ldots,n;\ i=0,\ldots,\binom{k-1}{2}\right).\tag{17}$$

Notice that, according to the definition of $a_{i,k}$ in Lemma 1, we obtain

$$a_{-1,k}=0,\quad a_{i,k-1}=0\quad\left(i=\binom{k-2}{2}+1,\ldots,\binom{k-1}{2}\right).$$

(iii) Next, we perform induction to prove the following mathematical statement, denoted by $H(k,i)$:

$$a_{i,k}\geq 0,\quad\left(k=2,\ldots,n;\ i=0,\ldots,\binom{k-1}{2}\right).$$

Base Case: Based on the definition of $a_{i,k}$ in Lemma 1, we have $a_{0,2}=1$. Thus, $H(2,0)$ holds. Furthermore, by definition, $H(3,-1)$ holds.

Inductive hypothesis: Assume that $H(q, r-1)$ and $H(q-1, r)$ are correct for

$$q = 3, \ldots, n, \quad r = 0, \ldots, \binom{q-1}{2}.$$

That is,

$$a_{r-1,q} \geq 0, \qquad a_{r,q-1} \geq 0 \quad \left(q = 3, \ldots, n;\ r = 0, \ldots, \binom{q-1}{2}\right).$$

Inductive step: Applying Equation (17), we obtain

$$a_{r,q} = \frac{\left[\binom{q-1}{2} - r + 1\right] a_{r-1,q} + \binom{q}{2} a_{r,q-1}}{\binom{q}{2} - r}.$$

Since

$$\binom{q-1}{2} - r + 1 > 0 \quad \text{and} \quad \binom{q}{2} - r > 0,$$

it follows that $a_{r,q} \geq 0$.
Hence, by mathematical induction, $H(k, i)$ is correct for

$$k = 2, \ldots, n, \quad i = 0, \ldots, \binom{k-1}{2}.$$

(iv) We observe that

$$\binom{k-1}{2} - i + 1 > 0, \qquad \binom{k}{2} > \binom{k}{2} - i \quad \left(k = 3, \ldots, n;\ i = 0, \ldots, \binom{k-2}{2}\right).$$

Following the statement $H(k, i)$, we have that

$$a_{i-1,k} \geq 0, \qquad a_{i,k-1} \geq 0.$$

Therefore, the first term in Equation (17) is nonnegative and the second term is no less than $a_{i,k-1}$, so that

$$a_{i,k} \geq a_{i,k-1}.$$

Since now we have proved that,

$$a_{i,k} \geq a_{i,k-1} \quad \left(k = 3, \ldots, n;\ i = 0, \ldots, \binom{k-2}{2}\right),$$

$$a_{i,k} \geq 0 \quad \left(k = 3, \ldots, n;\ i = \binom{k-2}{2} + 1, \ldots, \binom{k-1}{2}\right).$$

Thus, we conclude that

$$\frac{g_{k-1}(x)}{g_k(x)} = \frac{\sum_{i=0}^{\binom{k-2}{2}} a_{i,k-1} x^i}{\sum_{i=0}^{\binom{k-1}{2}} a_{i,k} x^i} \leq 1, \quad x \in [0, \infty),$$

$$\frac{(1-x)^{k-2}}{(1-x)^{k-1}}\frac{g_{k-1}(x)}{g_k(x)} \le \frac{1}{1-x}, \quad x \in [0,1],$$
$$\frac{f_{k-1}(x)}{f_k(x)} \le \frac{1}{1-x}, \quad x \in [0,1].$$

Now Proposition 2 is proved.

9. CONNECTION BETWEEN THE COEFFICIENTS $r_{j,n}$ AND THE TOUCHARD–RIORDAN NUMBERS $t_{n,j}$

This appendix discusses the relationship between the coefficients $r_{j,n}$ and the Touchard–Riordan numbers $t_{n,j}$; see Riordan (1975) and OEIS sequence A067311 (OEIS Foundation Inc., 2026).

Equation (1) in Riordan (1975) states an identity similar to Lemma 1:

$$(1-x)^k \sum_{i=0}^{\binom{k}{2}} T_{k,i} x^i = \sum_{j=0}^{k} (-1)^j t_{k,j} x^{\binom{j+1}{2}}.$$

Here,

$$t_{k,j} = \binom{2k}{k-j} - \binom{2k}{k-j-1},$$

and

$$T_{k,i} = \sum_{j=1}^{j^*} (-1)^{j-1} \binom{i - \binom{j}{2} + k - 1}{i - \binom{j}{2}} t_{k,j-1}, \qquad j^* = \operatorname*{argmax}_{j:\, \binom{j}{2} \le i} j.$$

By careful comparison, we identify the correspondence between the Touchard–Riordan numbers $t_{k,j}$, $T_{k,i}$ and the coefficients $r_{j,k}$, $a_{i,k}$ in Lemma 1 as

$$r_{j,k+1} = (-1)^{j-1} \frac{t_{k,j-1}}{C_k}, \quad a_{i,k+1} = \frac{T_{k,i}}{C_k},$$

where the Catalan number is given by

$$C_k = \binom{2k}{k} - \binom{2k}{k+1}.$$

While Riordan (1975) derived this identity via the distribution of chord crossings in a circle, our proof of Lemma 1 provides an alternative algebraic derivation of the same identity.

10. CONDITIONAL INTENSITY DERIVATION

According to the definition of conditional intensity (Rasmussen, 2018), for $t \in (t_{k+1}, t_k]$ and $k = 2, \ldots, n$, we obtain

$$\lambda_k^B(t) = \frac{f^B(t \mid t_{k+1})}{1 - \int_{t_{k+1}}^{t} f^B(u \mid t_{k+1})\, du}$$
$$= \frac{f^B(t \mid t_{k+1})}{\mathrm{pr}(T_k > t \mid T_{k+1} = t_{k+1}, T_2 \le \tau)}$$

$$
\begin{aligned}
&= \frac{f(t \mid t_{k+1})\, g_k(t, t_{k+1}, \tau)\, \mathrm{pr}(T_{k+1} = t_{k+1}, T_2 \le \tau)}{\mathrm{pr}(T_k > t, T_{k+1} = t_{k+1}, T_2 \le \tau)} \\
&= \frac{f(t \mid t_{k+1})\, g_k(t, t_{k+1}, \tau)\, \mathrm{pr}(T_2 \le \tau \mid T_{k+1} = t_{k+1})\, \mathrm{pr}(T_{k+1} = t_{k+1})}{\mathrm{pr}(T_2 \le \tau \mid T_k > t)\, \mathrm{pr}(T_k > t \mid T_{k+1} = t_{k+1}) \mathrm{pr}(T_{k+1} = t_{k+1})} \\
&= \frac{f(t \mid t_{k+1})\, \mathrm{pr}(T_2 \le \tau \mid T_k = t)}{\mathrm{pr}(T_2 \le \tau \mid T_k > t)\, \mathrm{pr}(T_k > t \mid T_{k+1} = t_{k+1})} \\
&= \frac{f(t \mid t_{k+1})\, \mathrm{pr}(T_2 \le \tau \mid T_k = t)}{\mathrm{pr}(T_2 \le \tau \mid T_{k+1} = t) \int_t^\infty f(t_k \mid t_{k+1})\, dt_k} \\
&= \frac{\binom{k}{2}}{N_e(t)} \exp\left\{ \binom{k}{2} \big(\Lambda(t_{k+1}) - \Lambda(t)\big) \right\} \frac{\mathrm{pr}(T_2 \le \tau \mid T_k = t)}{\exp\left\{ \binom{k}{2} \big(\Lambda(t_{k+1}) - \Lambda(t)\big) \right\} \mathrm{pr}(T_2 \le \tau \mid T_{k+1} = t)} \\
&= \frac{\binom{k}{2}}{N_e(t)} \cdot \frac{\mathrm{pr}(T_2 \le \tau \mid T_k = t)}{\mathrm{pr}(T_2 \le \tau \mid T_{k+1} = t)} \\
&= \frac{\binom{k}{2}}{N_e(t)}\, g_k(t, t, \tau).
\end{aligned}
$$

A minor notational correction: when $k = 2$ we have $t \in (t_3, \tau]$.

Next, we show that

$$\lim_{t \to \tau} \lambda_k^B(t) = \infty.$$

Again, denote $x = e^{\Lambda(t) - \Lambda(\tau)}$; then $\lim_{t \to \tau} x = 1$. We have

$$
\begin{aligned}
\lim_{t \to \tau} \lambda_k^B(t) &= \lim_{t \to \tau} \frac{\binom{k}{2}}{N_e(t)} \frac{\sum_{j=1}^{k-1} r_{j,k-1} x^{\binom{j}{2}}}{\sum_{j=1}^{k} r_{j,k} x^{\binom{j}{2}}} \\
&= \lim_{t \to \tau} \frac{\binom{k}{2}}{N_e(t)} \frac{1}{1 - x} \frac{\sum_{i=0}^{\binom{k-2}{2}} a_{i,k-1} x^i}{\sum_{i=0}^{\binom{k-1}{2}} a_{i,k} x^i} \\
&= \infty,
\end{aligned}
$$

where the second-to-last line is obtained according to Lemma 1.

11. ALGORITHM 2

Assume $L \le \frac{1}{N_e(t)} \le M \quad \forall t \in [0, \tau]$, then

$$\frac{\binom{k}{2}}{N_e(t)} \cdot \frac{1}{1 - \exp\{\Lambda(t) - \Lambda(\tau)\}} \le \frac{\binom{k}{2} M}{1 - \exp(-L(\tau - t))}.$$

We construct a new upper bound for $\lambda_k^B(t)$, denoted by $\lambda_k^U(t)$, as follows:

$$\lambda_k^U(t) := \frac{\binom{k}{2} M}{1 - \exp\{-L(\tau - t)\}}.$$

We need to make some modifications to both steps and describe the two steps as follow.

Step 1 (Inverse transformation). We derive $\Lambda_k^U(t) = \int_0^t \lambda_k^U(s) ds = \frac{\binom{k}{2} M}{L} \log\left(\frac{\exp(L\tau) - 1}{\exp(L(\tau - t)) - 1} \right)$.

Given t_{old}, we simulate a standard exponential distribution random variable W and solve

the following equation

$$W = \Lambda_k^U(t_{\text{new}}) - \Lambda_k^U(t_{\text{old}}).$$

We obtain that

$$t_{\text{new}} = \tau - \frac{1}{L}\log\left(\exp\{L(\tau - t_{\text{old}})\} + \exp\left\{\frac{WL}{\binom{k}{2}M}\right\} - 1\right) + \frac{W}{\binom{k}{2}M}.$$

Step 2 (Thinning). We accept t_{new} as t_k with probability

$$\frac{g_k(t_{\text{new}}, t_{\text{new}}, \tau)\,(1 - \exp\{L(t_{\text{new}} - \tau)\})}{MN_e(t_{\text{new}})}.$$

If the proposal is rejected, we set $t_{\text{old}} = t_{\text{new}}$ and repeat Steps 1–2.

Algorithm 2. Simulation of coalescent times by thinning with evaluable $N_e(t)$ and $\Lambda(t)$.

Input: $k = n$, $t = 0$, $N_e(t)$, $\Lambda(t)$.
Output: $\{t_k\}_{k=n}^2$.
repeat
  Transformation:
  Sample $E \sim \text{Exp}\left(\binom{k}{2}\right)$ and $U \sim \text{Unif}(0,1)$;

$$t = \tau + \frac{E}{M} - \frac{1}{L}\log\left(\exp\left(\frac{LE}{M}\right) + \exp\big(L(\tau - t)\big) - 1\right)$$

  Thinning:
  if $U \le \lambda_k^B(t)/\lambda_k^U(t)$ **then**
    $t_k \leftarrow t, \quad k \leftarrow k - 1;$
until $k < 2$;

12. TECHNIQUES FOR COVARIANCE MATRIX

In this work, we apply the following technique to compute matrices related to $\tilde{C}$, thereby sidestepping the explicit computation of C^{-1}. Define $A := C^{-1} + \epsilon\mathbf{I}$, $\boldsymbol{w} := C^{-1}\boldsymbol{l}$, and $d := \boldsymbol{l}^{\mathrm{T}}\boldsymbol{w}$. Starting from the definition of $\tilde{C}$ in Equation (6), and noting that the jitter term ϵ has already been set therein (typically $\epsilon = 10^{-16}$), we obtain

$$\begin{aligned}
\tilde{C} &= \left(C^{-1} - \frac{C^{-1}\boldsymbol{l}\boldsymbol{l}^{\mathrm{T}}C^{-1}}{\boldsymbol{l}^{\mathrm{T}}C^{-1}\boldsymbol{l}} + \epsilon\mathbf{I}\right)^{-1} \\
&= \left(A - \frac{\boldsymbol{w}\boldsymbol{w}^{\mathrm{T}}}{d}\right)^{-1} \\
&= A^{-1} + \frac{A^{-1}\boldsymbol{w}\boldsymbol{w}^{\mathrm{T}}A^{-1}}{d - \boldsymbol{w}^{\mathrm{T}}A^{-1}\boldsymbol{w}}, \qquad \text{applying Woodbury matrix identity.}
\end{aligned}$$

The matrix A^{-1} is computed exactly as $C(\boldsymbol{I} + \epsilon C)^{-1}$, which avoids forming C^{-1} explicitly. The vector $\boldsymbol{w}$ is approximately computed as $(LL^{\mathrm{T}})^{-1}\boldsymbol{l}$, where $L := \text{chol}(C + \epsilon_0\boldsymbol{I})$, using a Cholesky-based solver rather than an explicit inverse. The corresponding implementation is shown in the

following Python code block. Compared to directly using C_0^{-1}, this approach has two key advantages: (i) the discrepancy between C and $LL^\top$ is substantially smaller than that between CC_0^{-1} and $\mathbf{I}$; (ii) Cholesky-based solvers (`cho_solve`) are more numerically stable and accurate than forming explicit matrix inverses. However, for datasets simulated under $N_{e,3}(t)$, the discrepancy between $\boldsymbol{Cw}$ and $\boldsymbol{l}$ is larger than in the other cases, which contributes to the poorest performance of SC-RI-BM among the three methods. After constructing $\tilde{C}$, we compute its Cholesky decomposition $\tilde{L} \coloneqq \mathrm{chol}(\tilde{C} + \epsilon_0 \boldsymbol{I})$. Rather than forming $\tilde{C}^{-1}$ explicitly, we compute products of the form $\tilde{C}^{-1}\boldsymbol{\lambda}$ (required by **??**) via `cho_solve`$(\tilde{L}, \boldsymbol{\lambda})$.

The second jitter parameter ϵ_0 is selected using the diagnostic procedure described in Section 5.2. In addition, we offer a quantitative guideline for its selection based on the condition number of C_{jittered} and the Frobenius norm of the perturbation $\|C_{\text{jittered}} - C\|_F$ by ensuring that the condition number is less than 10^{10} and that the Frobenius norm is less than 10^{-7}. In practice, ϵ_0 typically lies between 10^{-9} and 10^{-6}.

```
jitter = 1e-7  #\epsilon_0
C_jittered = C + np.eye(C.shape[0]) * jitter
L = np.linalg.cholesky(C_jittered)
w = cho_solve((L, True), l)
```

13. EVALUATION METRICS

Sum of square errors at grid points : is computed by summing up squared differences between the median of predicted effective population size trajectory values and ground-truth values.

Coverage at grid points : is the proportion of points at which the true effective population size trajectory lies within the corresponding 95% credible intervals.

Credible interval width : is computed as the average width of the 95% posterior credible intervals across all grid points.

These three statistics are evaluated on a regular grid of 100 points per dataset. We report the 25th, 50th (median), and 75th percentiles of the resulting values across 30 simulated datasets.

BIBLIOGRAPHY

ALBRECHER, H. & BLADT, M. (2019). Inhomogeneous phase-type distributions and heavy tails. *Journal of Applied Probability* **56**, 1044–1064.

CARSON, J., LEDDA, A., FERRETTI, L., KEELING, M. & DIDELOT, X. (2022). The bounded coalescent model: conditioning a genealogy on a minimum root date. *Journal of Theoretical Biology* **548**, 111186.

CHOI, J., CHEN, W., MINKINA, A., CHARDON, F. M., SUITER, C. C., REGALADO, S. G., DOMCKE, S., HAMAZAKI, N., LEE, C., MARTIN, B. et al. (2022). A time-resolved, multi-symbol molecular recorder via sequential genome editing. *Nature* **608**, 98–107.

DIDELOT, X., FRASER, C., GARDY, J. & COLIJN, C. (2017). Genomic infectious disease epidemiology in partially sampled and ongoing outbreaks. *Molecular biology and evolution* **34**, 997–1007.

DIDELOT, X., GARDY, J. & COLIJN, C. (2014). Bayesian inference of infectious disease transmission from whole-genome sequence data. *Molecular biology and evolution* **31**, 1869–1879.

GENZ, A. & BRETZ, F. (2009). *Computation of multivariate normal and t probabilities*, vol. 195. Springer Science & Business Media.

GENZ, A. & TRINH, G. (2016). Numerical computation of multivariate normal probabilities using bivariate conditioning. In *Monte Carlo and Quasi-Monte Carlo Methods: MCQMC, Leuven, Belgium, April 2014*. Springer, pp. 289–302.

GILL, M. S., LEMEY, P., FARIA, N. R., RAMBAUT, A., SHAPIRO, B. & SUCHARD, M. A. (2013). Improving Bayesian population dynamics inference: a coalescent-based model for multiple loci. *Molecular Biology and Evolution* **30**, 713–724.

HEIN, J., SCHIERUP, M. & WIUF, C. (2004). *Gene genealogies, variation and evolution: a primer in coalescent theory*. Oxford University Press, USA.

HOBOLTH, A., RIVAS-GONZÁLEZ, I., BLADT, M. & FUTSCHIK, A. (2024). Phase-type distributions in mathematical population genetics: An emerging framework. *Theoretical Population Biology* .

HORVÁTH, A. & TELEK, M. (2024). *Phase Type Distributions: Theory and Application*. John Wiley & Sons.

KINGMAN, J. F. C. (1982). The coalescent. *Stochastic processes and their applications* **13**, 235–248.

LEWIS, P. W. & SHEDLER, G. S. (1979). Simulation of nonhomogeneous Poisson processes by thinning. *Naval research logistics quarterly* **26**, 403–413.

LI, Q., SCORNAVACCA, C., GALTIER, N. & CHAN, Y.-B. (2021). The multilocus multispecies coalescent: a flexible new model of gene family evolution. *Systematic Biology* **70**, 822–837.

MININ, V. N., BLOOMQUIST, E. W. & SUCHARD, M. A. (2008). Smooth Skyride through a Rough Skyline: Bayesian coalescent-based Inference of Population Dynamics. *Molecular Biology and Evolution* **25**, 1459–1471.

MURRAY, I., ADAMS, R. & MACKAY, D. (2010). Elliptical slice sampling. In *Proceedings of the thirteenth International Conference on Artificial Intelligence and Statistics*. JMLR Workshop and Conference Proceedings.

MURRAY, I., GHAHRAMANI, Z. & MACKAY, D. (2012). MCMC for doubly-intractable distributions. *arXiv preprint arXiv:1206.6848* .

OEIS FOUNDATION INC. (2026). Entry A067311 in The On-Line Encyclopedia of Integer Sequences. https://oeis.org/A067311. The On-Line Encyclopedia of Integer Sequences.

OGATA, Y. (1981). On Lewis' simulation method for point processes. *IEEE transactions on information theory* **27**, 23–31.

PALACIOS, J. A. & MININ, V. N. (2012). Integrated nested Laplace approximation for Bayesian nonparametric phylodynamics. In *Proceedings of the Twenty-Eighth Conference on Uncertainty in Artificial Intelligence*.

PALACIOS, J. A. & MININ, V. N. (2013). Gaussian process-based Bayesian nonparametric inference of population size trajectories from gene genealogies. *Biometrics* **69**, 8–18.

PAMILO, P. & NEI, M. (1988). Relationships between gene trees and species trees. *Molecular biology and evolution* **5**, 568–583.

RASMUSSEN, J. G. (2018). Lecture notes: Temporal point processes and the conditional intensity function. *arXiv preprint arXiv:1806.00221* .

RASMUSSEN, M. D. & KELLIS, M. (2012). Unified modeling of gene duplication, loss, and coalescence using a locus tree. *Genome research* **22**, 755–765.

RIORDAN, J. (1975). The distribution of crossings of chords joining pairs of 2 points on a circle. *Mathematics of Computation* **29**, 215–222.

RUE, H. & HELD, L. (2005). *Gaussian Markov random fields: theory and applications*. CRC press.

SEABOLD, S., PERKTOLD, J. et al. (2010). Statsmodels: econometric and statistical modeling with python. *SciPy* **7**, 92–96.

SEIDEL, S., ZWAANS, A., REGALADO, S., CHOI, J., SHENDURE, J. & STADLER, T. (2024). Sciphy: A Bayesian phylogenetic framework using sequential genetic lineage tracing data. *bioRxiv* , 2024–10.

SLATKIN, M. & HUDSON, R. R. (1991). Pairwise comparisons of mitochondrial DNA sequences in stable and exponentially growing populations. *Genetics* **129**, 555–562.

SLATKIN, M. & MADDISON, W. P. (1989). A cladistic measure of gene flow inferred from the phylogenies of alleles. *Genetics* **123**, 603–613.

Tang, B. & Palacios, J. (2024). Exact Bayesian Gaussian Cox Processes Using Random Integral. *arXiv preprint arXiv:2406.19722* .

Tavaré, S. (1984). Line-of-descent and genealogical processes, and their applications in population genetics models. *Theoretical population biology* **26**, 119–164.

Tavaré, S. (2004). *Ancestral inference in population genetics*. Lectures on probability theory and statistics: Ecole d'Eté de Probabilités de Saint-Flour XXXI-2001. Springer.

[*Received on* 2 *January* 2026. *Editorial decision on* 2 *January* 2026]