Phylodynamic Inference with Bounded Coalescent: A Point Process Perspective

Bingjing Tang, Shuangping Li and Julia A. Palacios

January 20, 2026

Abstract

Coalescent models are powerful tools in evolutionary biology, representing a stochastic process that describes genealogies of randomly sampled individuals from a population, where the inverse of effective population size trajectories serves as its intensity function. The bounded coalescent model is obtained by conditioning the coalescent tree so that the time to the most recent common ancestor is 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. Existing research restricts the effective population size to constant; in contrast, we extend the framework to Gaussian process–based Bayesian nonparametric approaches. We first develop a probabilistic generative model to simulate from the bounded coalescent, surpassing naive rejection sampling. Posterior inference of effective population size involves multiple intractable integrals over it in the likelihood, leading to a doubly intractable posterior distribution. We propose to jointly model the intensity and the integrals as a transformed Gaussian process, allowing us to directly bypass the need of approximating those integrals. We propose an exact MCMC sampler for posterior inference and evaluate its performance on simulated data. We demonstrate the utility of our method using sequential genetic lineage tracing data.

1 Introduction

Coalescent models are powerful tools in evolutionary biology, representing a stochastic process that describes genealogies of randomly sampled individuals from a population (Kingman, 1982). This probablistic model describes the relationship between a gene genealogy of a random sample of molecular sequences and effective population size trajectory, denoted $N_e(t)$, which is a function that captures how genetic drift behaves over time in a population. Coalescent-based inference methods allow us to estimate $N_e(t)$ from gene genealogies. Coalescent times could be viewed as realizations from a point process with an intensity function related to the inverse of $N_e(t)$.

The bounded coalescent model is a variant of coalescent models in which the genealogical tree is conditioned so that the time to the most recent common ancestor (TMRCA) is upper-bounded by some known fixed time τ (Carson et al., 2022). This modeling idea was first introduced as a natural outcome of the DL-Coal model (Rasmussen and Kellis, 2012), which unifies gene duplication, gene loss, and coalescence within a single probabilistic framework. Specifically, when a gene duplication occurs on a branch of the species tree at time τ, it gives rise to a daughter locus. All descendant gene lineages in this locus must originate from that duplication event, and thus must share a common ancestor more recent than τ. In backward-time coalescent models, we trace lineages from the present back in time until they coalesce; therefore, the TMRCA must be less than τ. Consequently, the bounded coalescent model has been widely adopted in both multi-species modeling and within-host pathogen transmission analyses. In the former, coalescence of organisms of the same species is enforced before speciation time τ (Li et al., 2021), while in the latter, coalescence of virons is enforced within hosts before transmission at time τ (Didelot et al., 2017). More recently, bounded coalescent models have also emerged in the context of single cell lineage tracing using synthetic barcoding experiments — an application that motivates our current interest.

Although the bounded coalescent model has been increasingly used across various biological research domains, over the past decade it has appeared solely as a subcomponent of larger models, such as the DLCoal model. The bounded coalescent model was first formally defined and studied as an independent model in Carson et al. (2022). Similar to Rasmussen and Kellis (2012), Carson et al. (2022) focused on the case of constant effective population size. The authors proposed a forward algorithm to compute the bound probability—that is, the probability that the TMRCA is less than τ. Crucially, they observed that this probability is equivalent to the probability that the number of extant lineages at τ is exactly one. This insight enabled efficient computation via dynamic programming, by modeling the number of extant lineages across discretized time points as a hidden Markov model (HMM). In addition, they introduced a direct simulation algorithm for generating coalescent times by repeatedly sampling the number of extant lineages at discretized time points until each time interval contains at most one coalescent event, followed by inverse transform sampling to draw each coalescence time. They demonstrated that this direct sampling method significantly outperforms naive rejection sampling in terms of computational efficiency. Furthermore, they performed posterior inference on the effective population size using methods originally designed for unbounded coalescent models (Palacios and Minin, 2012b; Karcher et al., 2017); however, their results demonstrated that ignoring the bound introduces systematic bias, leading to negatively biased estimates of the effective population size trajectory.

Previous studies on the bounded coalescent model assumed a constant effective population size. However, this assumption is often unrealistic in many biological contexts where population sizes vary over time. To the best of our knowledge, no prior work has extended the bounded coalescent model to the more general and realistic setting of a time-varying effective population size $N_e(t)$. Moreover, the existing methods for computing the bound probability and simulating coalescent times without rejection, for example, the forward algorithm and the direct sampling algorithm proposed in Carson et al. (2022), critically rely on the constant effective population size assumption and cannot be directly applied to time-varing $N_e(t)$. In this work, we address this gap by developing the first bounded coalescent framework for time-varying $N_e(t)$. We derive the likelihood, develop simulation procedures, and perform posterior inference under this generalized model. We introduce a novel two-step simulation algorithm that combines time transformation with thinning. 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.

Recently, phase-type distributions (Horváth and Telek, 2024) have emerged as a powerful framework in mathematical population genetics. The time to reach the absorbing state of a continuous-time Markov chain is phase-type distributed. In Hobolth et al. (2024), the authors focus on homogeneous standard coalescent models and derive distributions for key population-genetic quantities, such as tree height, by representing them as phase-type distributions. Earlier approaches to computing these distributions relied on analytically cumbersome formulations and were largely restricted to homogeneous coalescent models. By contrast, Hobolth et al. (2024) shows that these quantities can be computed in a systematic and tractable way by formulating the coalescent process as an absorbing Markov jump process and leveraging the well-defined matrix-based formulas available within the phase-type distribution framework. We extend this approach to the computation of the tree height under an inhomogeneous standard coalescent model, using the theory of inhomogeneous phase-type distributions (Albrecher and Bladt, 2019). A brief overview of phase-type distributions is given in Section 7.

Even for the case of constant effective population size, posterior inference under the bounded coalescent model remains an open problem. To the best of our knowledge, we are the first to propose a posterior inference framework for time-varying effective population size $N_e(t)$ under the bounded coalescent model, with the constant effective population size setting as a special instance. In the standard (unbounded) coalescent literature, Bayesian inference of effective population size trajectories $N_e(t)$ could be performed using methods such as Integrated Nested Laplace Approximation under piecewise linear assumptions(Palacios and Minin, 2012a), and large data augmentation relying on thinning with Gaussian process (GP) priors (Palacios and Minin, 2013b). However, these methods do not extend to the bounded coalescent setting. For example, for the method of data augmentation relying on thinning, the coalescent point process intensity allows an upper bounding point process whose likelihood is tractable, and the ratio of the two intensities is also tractable.

In the bounded coalescent model, the intensity function lacks this feature, making direct application of this method infeasible. To overcome this limitation, we develop a nonparametric Bayesian inference method for $N_e(t)$ using Gaussian process priors, relying on a recently proposed framework for exact MCMC inference in inhomogeneous Poisson processes (Tang and Palacios, 2024).

2 Bounded coalescent

The coalescent model with variable effective population (Slatkin and Maddison, 1989; Tavaré, 2004) is an inhomogenous Markov death process that keeps track of the number of ancestral lineages and ancestors of a sample of individuals with labels $1, \ldots, n$. The chain starts at time 0 with n lineages, and proceeds back in time until there are 2 lineages at the time to the most recent common ancestor (TMRCA) T_2 when a single node (the root), is the ancestor of all n samples. Given there are k lineages, the chain transitions to $k-1$ by choosing two of the k lineages to coalesce into one. This transition occurs with cumulative intensity $\int_{t_k}^{t} \binom{k}{2} \frac{du}{N_e(u)}$, where $N_e(t)$ denotes the effective population size at time t, a relative measure of genetic diversity over time. A full realization of the coalescent process is a gene genealogy with n tips and $n-1$ coalescent times $0 < T_n < T_{n-1} < \cdots < T_2$ (assume $T_{n+1} = t_{n+1} = 0$ for notation convenience). The coalescent time T_i indicates the time when two of i lineages coalesce into a single lineage in the genealogy. Under the standard coalescent model, the sequence of coalescent times forms a Markov process backward in time. Consequently, the genealogical density can be expressed as a product of Markov transition kernels:

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

where

$$\mathbf{t} \coloneqq (t_n, t_{n-1}, \ldots, t_3, t_2)$$
$$f(t_k \mid t_{k+1}, N_e(t)) \coloneqq \frac{\binom{k}{2}}{N_e(t_k)} \exp\left\{ - \int_{t_{k+1}}^{t_k} \frac{\binom{k}{2}}{N_e(s)} d\, s \right\}.$$

The bounded coalescent (Carson et al., 2022) arises when the TMRCA is upper bounded by some fixed time τ. In this setting, the genealogical density is given by:

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

To evaluate the likelihood under the bounded coalescent model, the key challenge lies in computing the denominator in Equation (1), namely $\mathrm{P}(T_2 \leq \tau \mid N_e(t))$. This quantity is precisely the cumulative distribution function of the tree height under the standard coalescent model. We state the main result in Proposition 1. Although an inductive proof is feasible, it relies on cumbersome analytical derivations. Instead, we derive the result using the theory of inhomogeneous phase-type distributions (Albrecher and Bladt, 2019), which leads to a more elegant and convenient matrix-based formulation. Full proofs of all propositions in this paper are provided in Section 8. Throughout, we adopt the convention that $\binom{1}{2} = 0$.

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

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

where the coefficients are defined as

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

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

Proposition 1 can be extended to obtain the generalized distributions $\mathrm{P}(T_2 \leq \tau \mid T_{k+1} = u, N_e(t))$ and $\mathrm{P}(T_2 \leq \tau \mid T_k > u, N_e(t))$ for $k = 2, 3, \ldots, n$, as stated in Proposition 2. Together with Lemma 3, this result is used for simulation in Section 3.

Proposition 2. *For all integers $k = 2, \ldots, n$, we have*

$$\begin{aligned} P(T_2 \leq \tau \mid T_{k+1} = u, N_e(t)) &= P(T_2 \leq \tau \mid T_k > u, N_e(t)), \\ &= \sum_{j=1}^{k} r_{j,k}\, x^{\binom{j}{2}}, \end{aligned}$$

where

$$x = e^{\Lambda(u)-\Lambda(\tau)}.$$

Lemma 3. *For all integers $k = 2, \ldots, n$, 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

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

Remark 1 (Connection to Touchard–Riordan numbers (Riordan, 1975))**.** *The coefficients $r_{j,k}$ appearing in Lemma 3 are closely related to the Touchard–Riordan numbers $t_{k,j}$. Our proof of Lemma 3 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.*

3 Simulation

Simulation of genealogies under the bounded coalescent is useful both for inference and for understanding how different effective population size trajectories influence the 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 forward filtering backward 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. In this study, we propose a novel algorithm for sampling from a bounded coalescent with a time-varying effective population size. We compare our proposed algorithm with both naive rejection sampling and Carson's forward filtering backward sampling algorithm in terms of efficiency and effectiveness in Section 5.1. In this section we suppress $N_e(t)$ in our notation, though we are still conditioning on a given effective population trajectory $N_e(t)$.

3.1 The bounded coalescent as a point process

Naturally we want to re-express the likelihood function under the bounded coalescent model in Equation (1) as a product of Markov bridge kernels conditioned on $T_2 \leq \tau$, denoted as $f^B(t_k \mid t_{k+1})$.

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

The general form of these Markov bridge kernels is:

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

where

$$g_k(t_k, t_{k+1}, \tau) \coloneqq \frac{\mathrm{P}\left(T_2 \leq \tau \mid T_k = t_k\right)}{\mathrm{P}\left(T_2 \leq \tau \mid T_{k+1} = t_{k+1}\right)} = \frac{\sum_{j=1}^{k-1} r_{j,k-1} \cdot \exp\left\{\binom{j}{2} \cdot (\Lambda(t_k) - \Lambda(\tau))\right\}}{\sum_{j=1}^{k} r_{j,k} \cdot \exp\left\{\binom{j}{2} \cdot (\Lambda(t_{k+1}) - \Lambda(\tau))\right\}}, \quad k = 2, 3, \ldots, n.$$

is obtained from Proposition 2.

It is well known that the standard coalescent model could be viewed as a point process with the conditional intensity function $\lambda^S(t) \coloneqq \frac{\binom{k}{2}}{N_e(t)}$. We derive the conditional intensity function for the bounded coalescent point process based on its likelihood function in Equation (1). Since the conditional intensity function (Rasmussen, 2018) 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 history of the process up to time t (not including t), t_0 denotes the last observed event up to time t, and $f(t \mid \mathcal{H}_t)$ denotes the conditional density on all past events. In the bounded coalescent case, the conditional intensity function of the kth coalescent event is written as

$$\begin{aligned} \lambda^B(t) &= \frac{f^B(t \mid t_{k+1})}{1 - \int_{t_{k+1}}^{t} f^B(s \mid t_{k+1}) ds}, \quad t \in (t_{k+1}, t_k] \\ &= \frac{\binom{k}{2}}{N_e(t)} g_k(t, t, \tau), \quad t \in (t_{k+1}, t_k] \end{aligned} \tag{3}$$

It is also noticeable that

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

All derivations can be found in Section 10.

3.2 Sampling from a bounded coalescent point process

Two common approaches for simulating point processes are the inverse transformation method (Slatkin and Hudson, 1991) and the thinning algorithm (Lewis and Shedler, 1979). Both approaches have been successfully applied to simulation of the standard coalescent point process (Hein et al., 2004; Palacios and Minin, 2013a). We briefly describe these two methods for the standard coalescent point process. The inverse transformation method exploits the fact that

$$\Lambda(T_k) - \Lambda(T_{k+1}) \sim \text{Exp}(1).$$

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

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

Similar in spirit to rejection sampling, the thinning algorithm requires a constant upper bound $\bar{\lambda}$ such that $\bar{\lambda} \geq \lambda^S(t)$. Given T_{k+1} and $\bar{\lambda}$, candidate waiting times $W \sim \text{Exp}(\bar{\lambda})$ are repeatedly proposed and accepted with probability

$$\frac{\lambda^S(T_{k+1} + W)}{\bar{\lambda}}.$$

Applying the inverse-transform method to the bounded coalescent requires tractable expressions for both the cumulative intensity $\Lambda^B(t)$ and its inverse $(\Lambda^B)^{-1}(t)$; however, neither is available in closed form. Although numerical approximation of $(\Lambda^B)^{-1}(t)$ is theoretically feasible, it introduces several practical difficulties beyond approximation error. According to Equation (3), for each coalescent event,

$$\Lambda^B(t) = \int_{t_{k+1}}^{t} \frac{\binom{k}{2}}{N_e(u)}\, g_k(u, t_{k+1}, \tau)\, du,$$

which requires numerical integration over (t_{k+1}, t). Consequently, evaluating $(\Lambda^B)^{-1}(t)$ entails a nested computation, in which the numerical integration defining $\Lambda^B(t)$ is repeatedly performed within a root-finding procedure. Moreover, since $\Lambda^B(t)$ depends on k, a different cumulative intensity must be evaluated for each coalescent event, preventing reuse of numerical computations across events. Finally, because $\lambda^B(t)$ diverges as $t \to \tau$, numerical inversion of Λ^B near the boundary may suffer from instability and loss of precision.

Due to the unbounded nature of the bounded coalescent point process (see **??**), the thinning algorithm cannot be applied directly using a constant upper bound. Instead, we employ a thinning algorithm with a time-varying upper bound $\lambda^U(t)$. The main challenge is to construct such a $\lambda^U(t)$ so that the corresponding dominating point process can be simulated easily and the acceptance ratio $\lambda^B(t)/\lambda^U(t)$ is tractable. According to Proposition 4, under the assumption that $N_e(t)$, $\Lambda(t)$, and $\Lambda^{-1}(t)$ are all tractable, we construct the corresponding upper-bound intensity as follows:

$$\lambda^U(t) \;\coloneqq\; \frac{\binom{k}{2}}{N_e(t)} \cdot \frac{1}{1 - \exp\{\Lambda(t) - \Lambda(\tau)\}}. \tag{4}$$

Proposition 4.

$$\frac{\sum_{j=1}^{k-1} r_{j,k-1} x^{\binom{j}{2}}}{\sum_{j=1}^{k} r_{j,k} x^{\binom{j}{2}}} \leq \frac{1}{1-x} \quad \forall x \in [0,1] \quad \forall k = 3, 4, \ldots, n$$

The proof of Proposition 4 is provided in Section 8. Substituting $x = \exp\{\Lambda(t) - \Lambda(\tau)\}$ into the inequality yields $\Lambda^B(t) \leq \Lambda^U(t)$ establishing that $\lambda^U(t)$ is a valid dominating intensity for the bounded coalescent point process.

As mentioned above, the construction of $\lambda^U(t)$ brings both easy simulation and tractable acceptance ratio. To simulate from a point process with $\lambda^U(t)$, we could apply the inverse transformation method by

mapping Poisson arrival times to event times of a point process with $\lambda^U(t)$. For each of candidate proposals, we accept each with probability

$$g_k(t,t,\tau)\left(1-\exp\left\{\Lambda(t)-\Lambda(\tau)\right\}\right).$$

For example, we simulate a k-th event t_k from a bounded coalescent point process with $\lambda^B(t)$ via two recursive steps until an acceptance:

Step 1 Inverse transformation.

According to Equation (4), we derive $\Lambda^U(t)=\int_0^t\lambda^U(s)ds=\binom{k}{2}\log\left(\frac{\exp(\Lambda(\tau))-1}{\exp(\Lambda(\tau)-\Lambda(t))-1}\right)$. Given the last sample $\tilde{t}_i$ from the dominating point process, we simulate a standard exponential distribution random variable W and solve the following equation

$$W=\Lambda^U(\tilde{t}_{i+1})-\Lambda^U(\tilde{t}_i).$$

We obtain that

$$\tilde{t}_{i+1}=\Lambda^{-1}\left(\Lambda(\tau)+\frac{W}{\binom{k}{2}}-\log\left(\exp(\Lambda(\tau)-\Lambda(\tilde{t}_i))+\exp\left(\frac{W}{\binom{k}{2}}\right)-1\right)\right).$$

Step 2 Thinning.

We accept $\tilde{t}_{i+1}$ as a sample from the bounded coalescent point process with probability

$$g_k(\tilde{t}_{i+1},\tilde{t}_{i+1},\tau)\left(1-\exp\left\{\Lambda(\tilde{t}_{i+1})-\Lambda(\tau)\right\}\right).$$

Details could be found in Algorithm 1. We can extend Algorithm 1 to a more general setting in which only

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

Input: $k=n$, $t_{n+1}=0$, $t=0$, $N_e(t)$, $\Lambda(t)$, $\Lambda^{-1}(t)$.
Output: $\{t_k\}_{k=n}^2$.
repeat
  Sample $E\sim Exponential\left(\binom{k}{2}\right)$ and $U\sim U(0,1)$;
  $t=\Lambda^{-1}\left(\Lambda(\tau)+E-\log\left(\exp(\Lambda(\tau)-\Lambda(t))+\exp(E)-1\right)\right)$;
  if $U\leq\lambda^B(t)/\lambda^U(t)$ **then**
    $t_k\leftarrow t,\quad k\leftarrow k-1$
  end
until $k<2$;

$N_e(t)$ and the cumulative intensity function $\Lambda(t)$ are numerically evaluable, while the inverse $\Lambda^{-1}(t)$ is not available. For example, when $N_e(t)$ is modeled via a transformed Gaussian process. Specifically, we assume that the intensity satisfies

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

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 and Minin (2013b) 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 $\left\{\int_{t_{k+1}}^{t_k} \frac{1}{N_e(s)} ds\right\}_{k=n}^{2}$, 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 $\int_{t_2}^{\tau} \frac{1}{N_e(s)ds}$. To perform exact MCMC inference without discretization, Palacios and Minin (2013b) 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 (3) 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 cumularive function of the nonparametric estimate of $N_e(t)$ is intractable). To solve this, we follow the random integral (RI) method proposed in Tang and 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 $\frac{1}{N_e(t)}$ at finite locations with its integrals. We aim to define a joint positive prior constructed from Gaussian processes on

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

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, $\{x_i\}_{i=1}^m \coloneqq \{t_k\}_{k=2}^{n+1} \cup \{s_l\}_{l=1}^{m-n}$.

Theorem 5. *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 $\mathbf{f} \coloneqq \left[f(s_1), \ldots, f(s_p), \int_{\mathcal{X}_1} f(s)ds, \ldots, \int_{\mathcal{X}_q} f(s)ds\right]'$ follows a Gaussian distribution and*

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

where $\boldsymbol{\mu} \coloneqq [\mu(s_1), \ldots, \mu(s_p), \int_{\mathcal{X}_1} \mu(s)ds, \int_{\mathcal{X}_q} \mu(s)ds]'$, $\boldsymbol{V}_{SI}$ 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 5, which is the extension of Theorem 2.1 in Tang and 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:

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

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 5. The presence of multiple integrals in $\boldsymbol{\lambda}$ induces a more complex covariance structure for $\boldsymbol{V}$ than that in Tang and 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., $k_{SE}(x,x') = \frac{1}{\theta_0} \exp\left(-\frac{\theta_1 \|x-x'\|^2}{2}\right)$, and the Brownian motion covariance kernel, i.e., $k_{BM}(x,x') = \frac{1}{\theta} \min(x,x')$, where θ is the hyperparameter, denoting the precision parameter. Let $\frac{1}{\theta}C$ denote the Brownian motion covariance

obtained from Theorem 5 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 and 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 and Held, 2005; Tang and Palacios, 2024), with covariance $\frac{1}{\theta}\tilde{C}$, where

$$\tilde{C} \coloneqq \left(C^{-1} - \frac{C^{-1}\boldsymbol{l}\boldsymbol{l}'C^{-1}}{\boldsymbol{l}'C^{-1}\boldsymbol{l}} + \epsilon \mathbf{I} \right)^{-1}, \tag{5}$$

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

$$\boldsymbol{l} \coloneqq (\underbrace{1, \dots, 1}_{m},\ t_n,\ t_{n-1} - t_n,\ \dots,\ t_2 - t_3,\ \tau - t_2)'.$$

The matrix $C^{-1} - \frac{C^{-1}\boldsymbol{l}\boldsymbol{l}'C^{-1}}{\boldsymbol{l}'C^{-1}\boldsymbol{l}}$ 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 and 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 given by

$$p(\boldsymbol{\lambda}, \theta | \{x_i\}_{i=1}^m) \propto\ p_\theta(\theta)\mathcal{TN}(\boldsymbol{\lambda}; \mathbf{0}, V_\theta) \cdot \prod_{k=n}^{2} \frac{\binom{k}{2}}{N_e(t_k)} \exp\left\{ \binom{k}{2} \left(-\int_{t_{k+1}}^{t_k} \frac{1}{N_e(s)} ds \right) \right\} \frac{\mathbb{1}(t_2 \le \tau)}{r_n^1 + \sum_{j=2}^n r_n^j \exp\left\{ -\binom{j}{2} \int_0^\tau \frac{1}{N_e(s)} ds \right)\}}$$

where the covariance V_θ is constructed from the kernel function $k_\theta(\cdot, \cdot)$ as described in Theorem 5, 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 $\boldsymbol{\lambda} \,|\, \theta, \{x_i\}_{i=1}^m$:

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

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 $p(f) = \frac{1}{Z}\mathcal{N}(f; 0, \Sigma)L(f)$, we can incorporate the indicator terms inherited from the truncated normal prior into $L(f)$, thereby avoiding inefficiencies associated with truncated normal distributions.

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

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

where $\mathcal{F} = [0, +\infty)^{m+n}$. We note that, for a general kernel, the normalizing constant $\int_{\mathcal{F}} \exp\left\{-\frac{1}{2}\boldsymbol{\lambda}' V_\theta^{-1}\boldsymbol{\lambda}\right\} d\boldsymbol{\lambda}$ 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 $V_\theta = \dfrac{1}{\theta_0} K_{\theta_1}$, 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

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

where $\boldsymbol{z} = \sqrt{\theta_0}\boldsymbol{\lambda}$. 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 $\tilde{\alpha} = \alpha + \frac{m+n}{2}$ and $\tilde{\beta} = \beta + \frac{1}{2}\boldsymbol{\lambda}' K_{\theta_1}^{-1}\boldsymbol{\lambda}$. Conditional on θ_0, we update θ_1 using a Metropolis-Hastings algorithm. The acceptance ratio involves computing $\int_{\mathcal{F}} \exp\left\{-\frac{1}{2}\boldsymbol{z}' K_{\theta_1}^{-1}\boldsymbol{z}\right\} d\boldsymbol{z}$, which is approximated numerically using a Genz-type algorithm for multivariate normal probabilities (Genz and Bretz, 2009; Genz and 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 $V_\theta = \frac{1}{\theta}\tilde{C}$, the conditional posterior simplifies considerably. In this case, we only need to sample from

$$p(\theta|\boldsymbol{\lambda}, \{x_i\}_{i=1}^m) \propto\ p_\theta(\theta) \cdot \sqrt{\theta}^{m+n} \exp\left\{-\frac{\theta}{2}\boldsymbol{\lambda}' \tilde{C}^{-1}\boldsymbol{\lambda}\right\}. \tag{6}$$

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 $\tilde{\alpha} = \alpha + \frac{m+n}{2}$ and $\tilde{\beta} = \beta + \frac{1}{2}\boldsymbol{\lambda}' C^{-1}\boldsymbol{\lambda}$, which can be sampled exactly without requiring any numerical approximation.

5 Experiment Results

In this section, we demonstrate the efficiency and effectiveness of both Algorithm 1 for simulation and the random integral method for posterior inference. Specifically, in Section 5.1, we compare our proposed thinning algorithm with naive rejection sampling and Carson's forward–filtering backward–sampling algorithm (Carson et al., 2022). In Section 5.2, we compare the proposed random integral method with the existing integrated nested Laplace approximation (INLA) approach using datasets simulated from standard coalescent models. In Section 5.3, we apply the random integral method to datasets simulated from both bounded and standard coalescent models to demonstrate the advantages of the bounded coalescent framework. All simulations in Sections 5.1 to 5.3 are based on three effective population trajectories: (1) $N_{e_1}(t) = 1$; (2) $N_{e_2}(t) = 3\exp\{-t\}$; (3) $N_{e_3}(t) = 25\exp\{-5t\}$. Finally, in Section 5.4, we apply our method to DNA Typewriter lineage-tracing data. We implement our algorithm in Python and run it on a shared computing cluster consisting of 24 CPU cores and 191 GB of memory and set a time limit of 7 days. Code and documentation for all methods are available at https://github.com/bingjingle/boundedcoal.

5.1 Simulation results

We simulated 3,000 datasets under $N_{e_1}(t)$ with upper bounds $\tau = 0.50$ and $\tau = 1.00$, and under $N_{e_3}(t)$ with upper bounds $\tau = 0.55$ and $\tau = 0.71$, each with 50 and 100 tips, using our proposed thinning algorithm, naive rejection sampling, and Carson's forward–filtering backward–sampling algorithm. For each method, we report the average running time per sample. Since Carson's approach is designed for constant effective population sizes, we apply it only to simulations from $N_{e_1}(t)$. Results in Table 1 demonstrate the efficiency and superiority of our thinning algorithm across all settings. In addition, Figure 1 and Table 2 show that datasets simulated from $N_{e_1}(t)$ with $\tau = 0.50$ using thinning and naive rejection sampling are statistically similar.

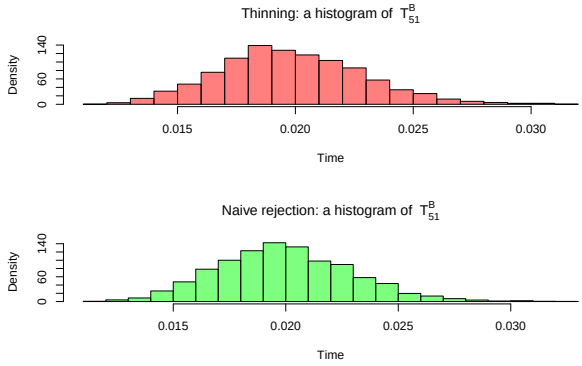

Figure 1: Histograms of T^B_{51} based on 3,000 simulations from $N_{e_1}(t)$ with 100 leaves and upper bound $\tau = 0.5$, generated using naive rejection sampling and our thinning algorithm.

Methods	**Mean ± Std**
Thinning	0.0198 ± 0.003
Naive rejection	0.0199 ± 0.003

Table 2: Mean and standard deviation of T^B_{51} based on 3,000 simulations from $N_{e_1}(t)$ with 100 leaves and upper bound $\tau = 0.5$, generated using naive rejection sampling and our thinning algorithm.

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 $N_{e_1}(t)$, $N_{e_2}(t)$ and $N_{e_3}(t)$, 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,

Effective Population Size	Bound	Ntips	Methods	Running Time
$N_{e_1}(t)$	0.50	50	Thinning	0.002 s
			Naive rejection	0.161 s
			Carson	0.002 s
		100	Thinning	0.003 s
			Naive rejection	0.354 s
			Carson	0.017 s
	1.00	50	Thinning	0.001 s
			Naive rejection	0.008 s
			Carson	0.003 s
		100	Thinning	0.002 s
			Naive rejection	0.010 s
			Carson	0.015 s
$N_{e_3}(t)$	0.55	50	Thinning	0.004 s
			Naive rejection	$>$ 201.600 s
		100	Thinning	0.013 s
			Naive rejection	$>$ 201.600 s
	0.71	50	Thinning	0.003 s
			Naive rejection	4.548 s
		100	Thinning	0.008 s
			Naive rejection	31.075 s

Table 1: Average running time comparison for $N_{e_1}(t)$ and $N_{e_3}(t)$ under two different upper bounds τ. Rows correspond to two numbers of tips ($n = 50$ and $n = 100$) and three simulation methods (thinning, naive rejection, and Carson's forward–filtering backward–sampling).

which empploys a Brownian motion kernel by default. Comparative results are summarized in Table 3. 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 3 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 and 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 (5)). 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 (5) with $C_0^{-1} := (C + \epsilon_0)^{-1}$. 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 $\text{chol}(\tilde{C})$. 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 3 and Figure 2.

Results for one simulated dataset with effective population size trajectories $N_{e_1}(t)$, $N_{e_2}(t)$ and $N_{e_3}(t)$

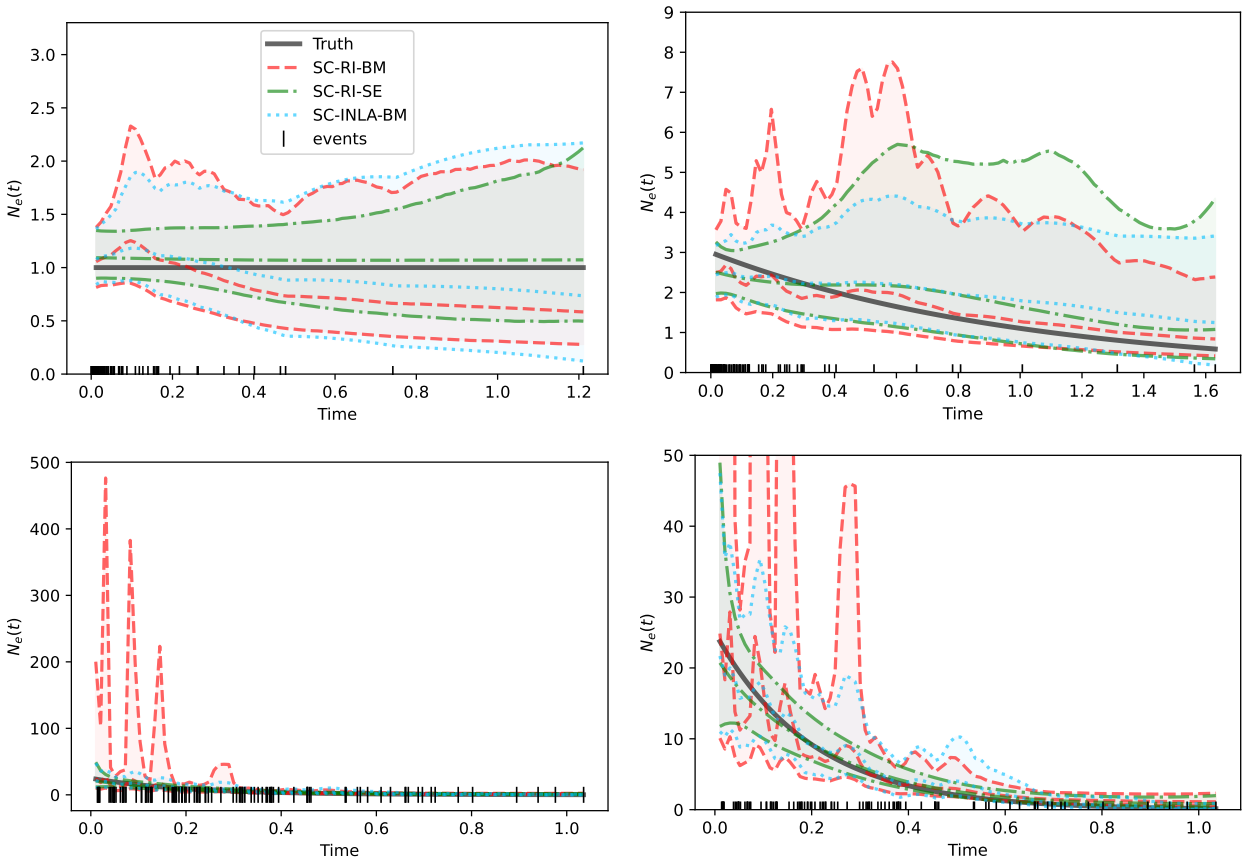

Figure 2: SC-X-BM refers to method X using a Brownian motion kernel under standard coalescent likelihood, while SC-X-SE denotes method X using a squared exponential kernel under standard coalescent likelihood. The top-left panel shows posterior inference of the effective population size trajectory $N_{e_1}(t)$ 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.

are depicted in Figure 2. Although all Bayesian methods have comparable performance for 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 3. 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 Figure 2 as well as in the coverage results reported in Table 3; (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

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 ± 317.8 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)	1665.9 ± 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 3: 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.

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 3 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 Figure 3. 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 4. 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 4. As illustrated in the bottom panels of Figure 3, 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 Figure 2. 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 $\binom{k}{2}/\lambda^B(t)$ rather than $N_e(t)$. Since $\binom{k}{2}/\lambda^B(t)$ is always

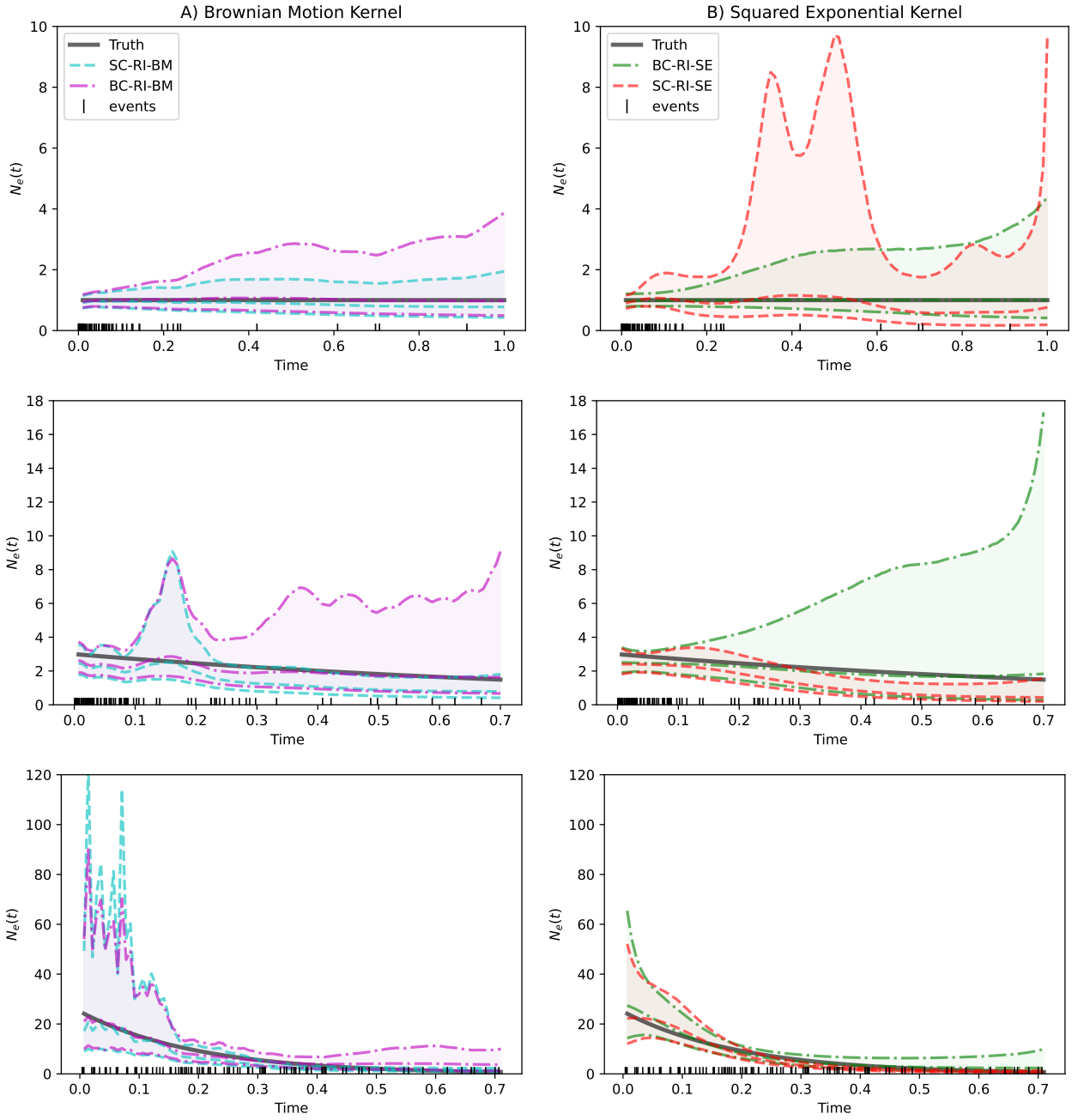

Figure 3: SC-RI-X refers to the random integral method using kernel X under the standard coalescent likelihood, while BC-RI-X denotes the random integral method using kernel X under the bounded coalescent likelihood. Rows from top to bottom show posterior inference of the effective population size trajectories $N_{e,1}(t)$, $N_{e,2}(t)$, 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.

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 Figure 3. Nevertheless, as shown in the bottom-right panel of Figure 2, the inverse of the truncated Gaussian process under the standard coalescent likelihood tends to overestimate $\binom{k}{2}/\lambda^B(t)$ 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.

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 4: 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.

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 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 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 Figure 4. 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.

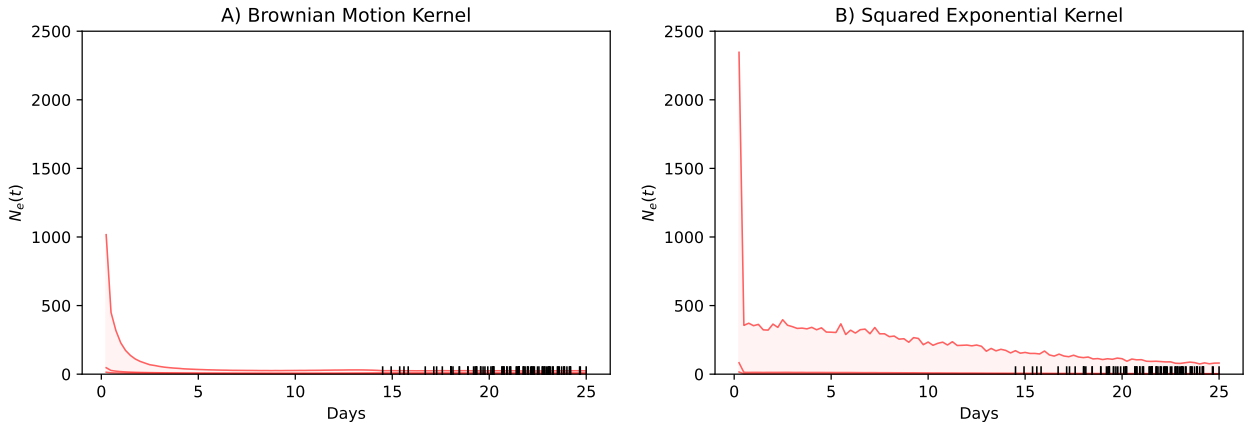

Figure 4: Both panels show the posterior median (red curve) and 95% credible intervals (shaded regions). Black ticks represent cell expansion events. The left panel corresponds to the Brownian motion kernel, while the right panel corresponds to the squared exponential kernel.

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 for empirical Bayes estimation (Tang and Palacios, 2024) or 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.

Appendix

7 Appendix A: 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 $T \sim PH(\boldsymbol{\alpha}, \mathbf{A})$. As shown in Equation 2.1 of Horváth and Telek (2024), its cumulative distribution could be in written in a matrix

formula fo two forms:

$$\mathrm{P}(T < t) = \mathrm{P}(X(t) = k+1) = \boldsymbol{\pi} e^{\mathbf{Q}t} \mathbf{e}_{k+1}^T \tag{7}$$

$$= 1 - \boldsymbol{\alpha} e^{\mathbf{A}t} \mathbb{1}, \tag{8}$$

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.

Hobolth et al. (2024) formulate the homogeneous standard coalescent with n tips as a continuous-time Markov jump process $\{X(t)\}_{t\geq 0}$, where $X(t)$ denotes the number of lineages at time t. The process has $n-1$ transient states, corresponding to $n, n-1, \ldots, 2$ lineages, and a single absorbing state corresponding to one lineage, representing the most recent common ancestor. In the standard coalescent case, the initial distribution is

$$\boldsymbol{\pi} = (1, \underbrace{0, \ldots, 0}_{n-1}), \tag{9}$$

and the generator matrix is

$$\mathbf{Q} = \begin{pmatrix} -\binom{n}{2} & \binom{n}{2} & 0 & 0 & \cdots & 0 & 0 \\ 0 & -\binom{n-1}{2} & \binom{n-1}{2} & 0 & \cdots & 0 & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots & 0 & 0 \\ 0 & 0 & 0 & \cdots & -\binom{3}{2} & \binom{3}{2} & 0 \\ 0 & 0 & 0 & \cdots & 0 & -\binom{2}{2} & \binom{2}{2} \\ 0 & 0 & 0 & \cdots & 0 & 0 & 0 \end{pmatrix}. \tag{10}$$

The absorption time T is exactly the tree height, whose cumulative distribution function is given in Equation (7).

In Proposition 1, we extend this distribution to the inhomogeneous standard coalescent using inhomogeneous phase-type distributions. 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:

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

We finish this section with a key intermediate result about inhomogeneous phase-type distribution from Albrecher and Bladt (2019) that we will use to prove Proposition 1.

Lemma 6. *[Corollary 2.6 in Albrecher and Bladt (2019)] If $\boldsymbol{A}(t) = \lambda(t)\boldsymbol{A}$, where $\lambda(t)$ is a known nonnegative real function and A is a subintensity matrix, and we write $T \sim \mathrm{IPH}(\boldsymbol{\alpha}, \lambda(t)\mathbf{A})$, then the density function f and the distribution function F of the absorption time T are given by*

$$F(t) = 1 - \boldsymbol{\alpha} \exp\left(\int_0^t \lambda(u)\, du\, \mathbf{A} \right) \mathbb{1}.$$

8 Appendix B: proposition proofs

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

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

where the coefficients are defined as

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

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

Proof: An inhomogeneous standard coalescent with time-varying effective population size $N_e(t)$ can be viewed as an inhomogeneous continuous-time Markov jump process with an initial distribution $\boldsymbol{\pi}$ and generator

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

where $\boldsymbol{\pi}$ and $\boldsymbol{Q}$ is defined as Equations (9) and (10).

By Lemma 6, the cumulative distribution function of the tree height T_2 satisfies

$$\begin{aligned}
\mathrm{P}(T_2 < t \mid N_e(t)) &= 1 - \boldsymbol{\alpha} \exp\{\Lambda(t)\boldsymbol{A}\}\, \mathbb{1} \\
&= 1 - \boldsymbol{\alpha} \sum_{k=0}^{\infty} \frac{\Lambda(t)^k}{k!} \boldsymbol{A}^k \mathbb{1} \\
&= 1 - (\boldsymbol{\alpha}, 0) \sum_{k=0}^{\infty} \frac{\Lambda(t)^k}{k!} \begin{pmatrix} \boldsymbol{A}^k & \cdot \\ 0 & 0 \end{pmatrix} \begin{pmatrix} \mathbb{1} \\ 0 \end{pmatrix} \\
&= 1 - \boldsymbol{\pi} \sum_{k=0}^{\infty} \frac{\Lambda(t)^k}{k!} \boldsymbol{Q}^k \begin{pmatrix} \mathbb{1} \\ 0 \end{pmatrix} \\
&= 1 - \boldsymbol{\pi} \exp\{\Lambda(t)\boldsymbol{Q}\} \begin{pmatrix} \mathbb{1} \\ 0 \end{pmatrix} \\
&= \boldsymbol{\pi} \exp\{\Lambda(t)\boldsymbol{Q}\}\, \mathbf{e}_n^T
\end{aligned} \tag{11}$$

where $\mathbf{e}_n$ denotes the unit vector whose first $n-1$ entries are zero and whose last entry equals one.

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

$$\lambda_i = -\binom{n-i+1}{2}, \qquad i = 1, \ldots, n,$$

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,

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

Substituting this expression into Equation (11) yields

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

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. □

Proposition 2. *For all integers $k = 2, \ldots, n$, we have*

$$\begin{aligned}P(T_2 \leq \tau \mid T_{k+1} = u, N_e(t)) &= P(T_2 \leq \tau \mid T_k > u, N_e(t)), \\ &= \sum_{j=1}^{k} r_{j,k} \, x^{\binom{j}{2}},\end{aligned}$$

where

$$x = e^{\Lambda(u) - \Lambda(\tau)}.$$

Proof: These two quantities $\mathrm{P}(T_2 \leq \tau \mid T_{k+1} = u, N_e(t))$ and $\mathrm{P}(T_2 \leq \tau \mid T_k > u, N_e(t))$ are equal since they can both be interpreted as the cumulative distribution function of the tree height of a standard coalescent tree that starts at time 0 with k tips and evolves under the effective population size trajectory $N_e(u + t)$.

Specifically, we write

$$\begin{aligned}
\mathrm{P}(T_2 \le \tau \mid T_{k+1} = u,\, N_e(t)) &= \mathrm{P}(T_2 \le \tau \mid T_k > u,\, N_e(t)) \\
&= \mathrm{P}(T_2 \le \tau - u \mid N_e(t+u)) \\
&= \sum_{j=1}^{k} r_{j,k} \exp\left\{ -\binom{j}{2} \int_0^{\tau-u} \frac{1}{N_e(u+t)}\, dt \right\} \\
&= \sum_{j=1}^{k} r_{j,k} \exp\left\{ -\binom{j}{2} \int_u^{\tau} \frac{1}{N_e(z)}\, dz \right\} \\
&= \sum_{j=1}^{k} r_{j,k} \exp\left\{ \binom{j}{2} \left(\Lambda(u) - \Lambda(\tau)\right) \right\}.
\end{aligned}$$

□

Lemma 3. *For all integers $k = 2, \ldots, n$, 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

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

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

$$S_{j,k} \coloneqq \sum_{q=1}^{j} r_{q,k} = (-1)^{j-1} j \prod_{m=1}^{j-1} \frac{k-1-m}{k+m}, \quad \forall j \in \{2, \ldots, k\},\ \forall k \in \{2, \ldots, n\}. \tag{12}$$

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 (12) 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 (12) yields

$$\sum_{j=1}^{k} r_{j,k} = 0, \qquad \forall\, k = 2,\ldots,n. \tag{13}$$

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,

$$f_k^{(s)}(1) = 0, \qquad \forall\, s = 0,1,\ldots,k-2. \tag{14}$$

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 (13), 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,

$$\sum_{\substack{1\le j\le k\\ \binom{j}{2}\ge s}} r_{j,k} \prod_{m=0}^{s-1} \left(\binom{j}{2} - m \right) = 0, \tag{15}$$

$$\sum_{\substack{1\le j\le k+1\\ \binom{j}{2}\ge s}} r_{j,k+1} \prod_{m=0}^{s-1} \left(\binom{j}{2} - m \right) = 0. \tag{16}$$

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

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

Denote

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

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 (16). 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 (15). This completes the induction and thereby establishes the statement in Equation (14).

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 (14), 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 4.

$$
\frac{\sum_{j=1}^{k-1} r_{j,k-1}x^{\binom{j}{2}}}{\sum_{j=1}^{k} r_{j,k}x^{\binom{j}{2}}} \le \frac{1}{1-x} \quad \forall x \in [0,1] \quad \forall k = 3, 4, \ldots, n
$$

Proof: We denote the two polynomials in Lemma 3 as

$$
f_k(x) \coloneqq \sum_{j=1}^{k} r_{j,k}\, x^{\binom{j}{2}}, \tag{17}
$$

$$
g_k(x) \coloneqq \sum_{i=0}^{\binom{k-1}{2}} a_{i,k}\, x^i. \tag{18}
$$

Our goal is to show that

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

By applying Lemma 3, it suffices to prove that $g_{k-1}(x) \leq g_k(x)$. 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 $f_k(x)$, $\forall k \in \{3, 4, \ldots, n\}$:

$$f_k(x) - f_{k-1}(x) = \frac{x f_k'(x)}{\binom{k}{2}} \tag{19}$$

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 (19).

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

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

Hence Equation (19) is proved.

(ii) Applying Lemma 3, this intermediate result can be expressed in terms of $a_{i,k}$, $a_{i-1,k}$, and $a_{i,k-1}$.

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

$$a_{i,k}=\frac{\left[\binom{k-1}{2}-i+1\right]a_{i-1,k}+\binom{k}{2}a_{i,k-1}}{\binom{k}{2}-i}, \quad \forall k\in\{3,4,\ldots,n\},\ \forall i\in\{0,1,2,\ldots,\binom{k-1}{2}\}. \tag{20}$$

Notice that, according to the definition of $a_{i,k}$ in Lemma 3,

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

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

$$a_{i,k} \geq 0, \quad \forall k \in \{2,3,\ldots,n\},\ \forall i \in \{0,1,2,\ldots,\tbinom{k-1}{2}\}.$$

Base Case: Based on the definition of $a_{i,k}$ in Lemma 3, we have

$$a_{0,2} = 1.$$

Thus, $H(2,0)$ holds. Furthermore, by definition, $H(3,-1)$ holds.

Inductive hypothesis: Let $q \in \{3,4,\ldots,n\}$ and $r \in \{0,1,2,\ldots,\tbinom{q-1}{2}\}$ be arbitrary integers. Assume that $H(q,r-1)$ and $H(q-1,r)$ are correct. That is,

$$a_{r-1,q} \geq 0, \qquad a_{r,q-1} \geq 0.$$

Inductive step: Applying Equation (20), we obtain

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

Since $\tbinom{q-1}{2} - r + 1 > 0$ and $\tbinom{q}{2} - r > 0$, it follows that $a_{r,q} \geq 0$.
Hence, by mathematical induction, $H(k,i)$ is correct for all $k \in \{2,3,\ldots,n\}$ and $i \in \{0,1,2,\ldots,\tbinom{k-1}{2}\}$.

(iv) Again, $\forall k \in \{3,4,\ldots,n\}$ and $\forall i \in \{0,1,2,\ldots,\tbinom{k-2}{2}\}$, we observe that

$$\binom{k-1}{2} - i + 1 > 0, \qquad \binom{k}{2} > \binom{k}{2} - i.$$

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 (20) 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, $\forall k \in \{3,4,\ldots,n\}$,

$$a_{i,k} \geq a_{i,k-1}, \quad \forall i \in \left\{0,1,\ldots,\binom{k-2}{2}\right\},$$
$$a_{i,k} \geq 0, \quad \forall i \in \left\{\binom{k-2}{2}+1, \binom{k-2}{2}+2, \ldots, \binom{k-1}{2}\right\}.$$

Thus, we conclude that

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

Now Proposition 4 is proved.

□

9 Appendix C: connection between the coefficients $r_{j,n}$ and the Touchard–Riordan numbers $t_{n,j}$ (Riordan, 1975)

Equation (1) in Riordan (1975) states an identity similar to Lemma 3:

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

Here,

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

and

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

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 3 as

$$r_{j,k+1} = (-1)^{j-1} \frac{t_{k,j-1}}{C_k}, \tag{21}$$

$$a_{i,k+1} = \frac{T_{k,i}}{C_k}, \tag{22}$$

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 3 provides an alternative algebraic derivation of the same identity.

10 Appendix D: conditional intensity derivation

According to the definition of conditional intensity (Rasmussen, 2018), for $t \in (t_{k+1}, t_k]$ and $k = 2, 3, \ldots, n$, we obtain

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

The term in the denominator of line 5 $\mathrm{P}(T_2 \le \tau \mid T_k > t)$ is equivalent to $\mathrm{P}(T_2 \le \tau \mid T_{k+1} = t)$ based on Proposition 2. As a minor notational correction, when $k = 2$ we have $t \in (t_3, \tau]$.

Next, we show that

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

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

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

11 Appendix E

When $N_e(t)$ is modeled by a transformed Gaussian process, only $N_e(t)$ and $\Lambda(t)$ are evaluable. Assume $L \le \frac{1}{N_e(t)} \le M \quad \forall t \in [0, \tau]$,

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

We construct a new upper bound for $\lambda^B(t)$, denoted as $\lambda^U(t)$, where

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

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

Step 1 Inverse transformation.

We derive $\Lambda^U(t) = \int_0^t \lambda^U(s)ds = \frac{\binom{k}{2}M}{L}\log\left(\frac{\exp(L\tau)-1}{\exp(L(\tau-t))-1}\right)$. Given the last sample $\tilde{t}_i$ from the dominating point process, we simulate a standard exponential distribution random variable W and solve the following equation

$$W = \Lambda^U(\tilde{t}_{i+1}) - \Lambda^U(\tilde{t}_i).$$

We obtain that

$$\tilde{t}_{i+1} = \tau - \frac{1}{L}\log\left(\exp\left\{L(\tau-\tilde{t}_i\right\} + \exp\left\{\frac{WL}{\binom{k}{2}M}\right\} - 1\right) + \frac{W}{\binom{k}{2}M}.$$

Step 2 Thinning.

We accept $\tilde{t}_{i+1}$ as a sample from the bounded coalescent point process with probability

$$\frac{g_k(\tilde{t}_{i+1},\tilde{t}_{i+1},\tau)\left(1-\exp\left\{L(\tilde{t}_{i+1}-\tau)\right\}\right)}{M\cdot N_e(\tilde{t}_{i+1})}.$$

Algorithm 2: Simulation of isochronous coalescent times by thinning -$N_e(t)$ and $\Lambda(t)$ are evaluable.

Input: $k=n$, $t_{n+1}=0$, $t=0$, $N_e(t)$, $\Lambda(t)$.
Output: $\{t_k\}_{k=n}^2$.
repeat
  Sample $E \sim Exponential\left(\binom{k}{2}\right)$ and $U\sim U(0,1)$;
  $t = \tau + \frac{E}{M} - \frac{1}{L}\log\left[\exp\left(\frac{LE}{M}\right) + \exp(L(\tau-t)) - 1\right]$;
  if $U \le \lambda^B(t)/\lambda^U(t)$ **then**
    $t_k \leftarrow t, k \leftarrow k-1$
  end
until $k<2$;

12 Appendix F

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 $d := \boldsymbol{l}'\boldsymbol{w}$. Starting from the definition of $\tilde{C}$ in Equation (5), and noting that the jitter term ϵ has already been set therein (typically $\epsilon = 10^{-16}$), we obtain

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

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 $(LL')^{-1}\boldsymbol{l}$, 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 $C\boldsymbol{w}$ 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 $\tilde{L} := \text{chol}(\tilde{C} + \epsilon_0 \boldsymbol{I})$. Rather than forming $\tilde{C}^{-1}$ explicitly, we compute products of the form $\tilde{C}^{-1}\boldsymbol{\lambda}$ (required by Equation (6)) via $\texttt{cho\_solve}(\tilde{L}, \boldsymbol{\lambda})$.

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 Appendix G: 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.

References

H. Albrecher and M. Bladt. Inhomogeneous phase-type distributions and heavy tails. *Journal of Applied Probability*, 56(4):1044–1064, 2019.

J. Carson, A. Ledda, L. Ferretti, M. Keeling, and X. Didelot. The bounded coalescent model: conditioning a genealogy on a minimum root date. *Journal of Theoretical Biology*, 548:111186, 2022.

J. Choi, W. Chen, A. Minkina, F. M. Chardon, C. C. Suiter, S. G. Regalado, S. Domcke, N. Hamazaki, C. Lee, B. Martin, et al. A time-resolved, multi-symbol molecular recorder via sequential genome editing. *Nature*, 608(7921):98–107, 2022.

X. Didelot, J. Gardy, and C. Colijn. Bayesian inference of infectious disease transmission from whole-genome sequence data. *Molecular biology and evolution*, 31(7):1869–1879, 2014.

X. Didelot, C. Fraser, J. Gardy, and C. Colijn. Genomic infectious disease epidemiology in partially sampled and ongoing outbreaks. *Molecular biology and evolution*, 34(4):997–1007, 2017.

A. Genz and F. Bretz. *Computation of multivariate normal and t probabilities*, volume 195. Springer Science & Business Media, 2009.

A. Genz and G. Trinh. Numerical computation of multivariate normal probabilities using bivariate conditioning. In *Monte Carlo and Quasi-Monte Carlo Methods: MCQMC, Leuven, Belgium, April 2014*, pages 289–302. Springer, 2016.

J. Hein, M. Schierup, and C. Wiuf. *Gene genealogies, variation and evolution: a primer in coalescent theory*. Oxford University Press, USA, 2004.

A. Hobolth, I. Rivas-González, M. Bladt, and A. Futschik. Phase-type distributions in mathematical population genetics: An emerging framework. *Theoretical Population Biology*, 2024.

A. Horváth and M. Telek. *Phase Type Distributions: Theory and Application*. John Wiley & Sons, 2024.

M. D. Karcher, J. A. Palacios, S. Lan, and V. N. Minin. phylodyn: an r package for phylodynamic simulation and inference. *Molecular ecology resources*, 17(1):96–100, 2017.

J. F. C. Kingman. The coalescent. *Stochastic processes and their applications*, 13(3):235–248, 1982.

P. W. Lewis and G. S. Shedler. Simulation of nonhomogeneous poisson processes by thinning. *Naval research logistics quarterly*, 26(3):403–413, 1979.

Q. Li, C. Scornavacca, N. Galtier, and Y.-B. Chan. The multilocus multispecies coalescent: a flexible new model of gene family evolution. *Systematic Biology*, 70(4):822–837, 2021.

I. Murray, R. Adams, and D. MacKay. Elliptical slice sampling. In *Proceedings of the thirteenth International Conference on Artificial Intelligence and Statistics*, pages 541–548. JMLR Workshop and Conference Proceedings, 2010.

I. Murray, Z. Ghahramani, and D. MacKay. Mcmc for doubly-intractable distributions. *arXiv preprint arXiv:1206.6848*, 2012.

J. A. Palacios and V. N. Minin. Integrated nested Laplace approximation for Bayesian nonparametric phylodynamics. In *Proceedings of the Twenty-Eighth Conference on Uncertainty in Artificial Intelligence*, UAI'12, pages 726–735, Arlington, Virginia, United States, 2012a. AUAI Press. ISBN 978-0-9749039-8-9.

J. A. Palacios and V. N. Minin. Integrated nested laplace approximation for bayesian nonparametric phylodynamics. *arXiv preprint arXiv:1210.4908*, 2012b.

J. A. Palacios and V. N. Minin. Gaussian process-based Bayesian nonparametric inference of population size trajectories from gene genealogies. *Biometrics*, 69(1):8–18, 2013a.

J. A. Palacios and V. N. Minin. Gaussian process-based bayesian nonparametric inference of population size trajectories from gene genealogies. *Biometrics*, 69(1):8–18, 2013b.

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

M. D. Rasmussen and M. Kellis. Unified modeling of gene duplication, loss, and coalescence using a locus tree. *Genome research*, 22(4):755–765, 2012.

J. Riordan. The distribution of crossings of chords joining pairs of 2 points on a circle. *Mathematics of Computation*, 29(129):215–222, 1975.

H. Rue and L. Held. *Gaussian Markov random fields: theory and applications*. CRC press, 2005.

S. Seabold, J. Perktold, et al. Statsmodels: econometric and statistical modeling with python. *SciPy*, 7(1): 92–96, 2010.

S. Seidel, A. Zwaans, S. Regalado, J. Choi, J. Shendure, and T. Stadler. Sciphy: A bayesian phylogenetic framework using sequential genetic lineage tracing data. *bioRxiv*, pages 2024–10, 2024.

M. Slatkin and R. R. Hudson. Pairwise comparisons of mitochondrial dna sequences in stable and exponentially growing populations. *Genetics*, 129(2):555–562, 1991.

M. Slatkin and W. Maddison. A cladistic measure of gene flow measured from the phylogenies of alleles. *Genetics*, 123(3):603–613, 1989.

B. Tang and J. Palacios. Exact bayesian gaussian cox processes using random integral. *arXiv preprint arXiv:2406.19722*, 2024.

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