OXFORD GENETICS

GENETICS, 2022, 221(1), iyac044
https://doi.org/10.1093/genetics/iyac044
Advance Access Publication Date: 25 March 2022
Investigation
Highlighted Article

Evaluation of methods for estimating coalescence times using ancestral recombination graphs

Débora Y. C. Brandt [1,*], Xinzhu Wei [2], Yun Deng [3], Andrew H. Vaughn [3], Rasmus Nielsen [1,3,4,5,*]

[1]Department of Integrative Biology, University of California Berkeley, Berkeley, CA 94720, USA,
[2]Department of Computational Biology, Cornell University, Ithaca, NY 14850, USA,
[3]Center for Computational Biology, University of California, Berkeley, CA 94720, USA,
[4]Department of Statistics, University of California Berkeley, Berkeley, CA 94720, USA,
[5]GLOBE Institute, University of Copenhagen, Copenhagen K 1350, Denmark

*Corresponding author: Department of Integrative Biology, University of California, 3040 Valley Life Sciences Building #3140, Berkeley, CA 94720-3140, USA. Email: deboraycb@berkeley.edu; Corresponding author: Department of Integrative Biology, University of California, 3040 Valley Life Sciences Building #3140, Berkeley, CA 94720-3140, USA. Email: rasmus_nielsen@berkeley.edu

Abstract

The ancestral recombination graph is a structure that describes the joint genealogies of sampled DNA sequences along the genome. Recent computational methods have made impressive progress toward scalably estimating whole-genome genealogies. In addition to inferring the ancestral recombination graph, some of these methods can also provide ancestral recombination graphs sampled from a defined posterior distribution. Obtaining good samples of ancestral recombination graphs is crucial for quantifying statistical uncertainty and for estimating population genetic parameters such as effective population size, mutation rate, and allele age. Here, we use standard neutral coalescent simulations to benchmark the estimates of pairwise coalescence times from 3 popular ancestral recombination graph inference programs: ARGweaver, Relate, and tsinfer+tsdate. We compare (1) the true coalescence times to the inferred times at each locus; (2) the distribution of coalescence times across all loci to the expected exponential distribution; (3) whether the sampled coalescence times have the properties expected of a valid posterior distribution. We find that inferred coalescence times at each locus are most accurate in ARGweaver, and often more accurate in Relate than in tsinfer+tsdate. However, all 3 methods tend to overestimate small coalescence times and underestimate large ones. Lastly, the posterior distribution of ARGweaver is closer to the expected posterior distribution than Relate's, but this higher accuracy comes at a substantial trade-off in scalability. The best choice of method will depend on the number and length of input sequences and on the goal of downstream analyses, and we provide guidelines for the best practices.

Keywords: ancestral recombination graph; ARGweaver; Relate; tsinfer; tsdate; simulation; calibration

Introduction

The full ancestral recombination graph (ARG) is a structure that encodes all coalescence and recombination events resulting from the stochastic process of the coalescent with recombination. Hudson (1983) first described a stochastic process that combines recombination and coalescence to generate genealogies. At each given site, the genealogy resulting from this process is equivalent to the one generated by the single-locus coalescent model (Kingman 1982), but because recombination breaks loci apart (Fig. 1a), the local genealogies can differ between sites.

Representations of the ARG

The full ARG can be represented as a directed graph with 2 types of nodes: (1) coalescence nodes, where 2 or more edges merge into one (backwards in time) and (2) recombination nodes, where 1 edge splits into 2 (backwards in time) (Fig. 1b). Alternatively, the full ARG can also be represented as an ordered collection of marginal coalescence trees, annotated with the recombination nodes. These marginal trees are embedded in the graph representation (Fig. 1, b and c).

In some representations, the collection of trees may or may not contain all the information from the full ARG, depending on whether the times of recombination events (red crosses in Fig. 1) are stored with the trees (Rasmussen et al. 2014), and whether the internal nodes of the tree are labeled so they can be explicitly shared between adjacent trees. Furthermore, in some cases only topology changing recombination events are represented, and thus information regarding recombination events that do not lead to topology changes can be lost (Kelleher et al. 2019). Finally, some representations of ARGs as a collection of local trees allow more than one recombination event between trees (Speidel et al. 2019). In the latter 2 cases, each tree will potentially be an average of multiple coalescence trees. Figure 1d shows an example of a collection of local trees that does not correspond to the underlying full ARG, since one of its local trees is an average of 2 adjacent trees with identical topologies.

Collections of local trees with labeled internal nodes, regardless of whether they represent a full ARG or not, can be

Received: November 14, 2021. **Accepted:** March 8, 2022

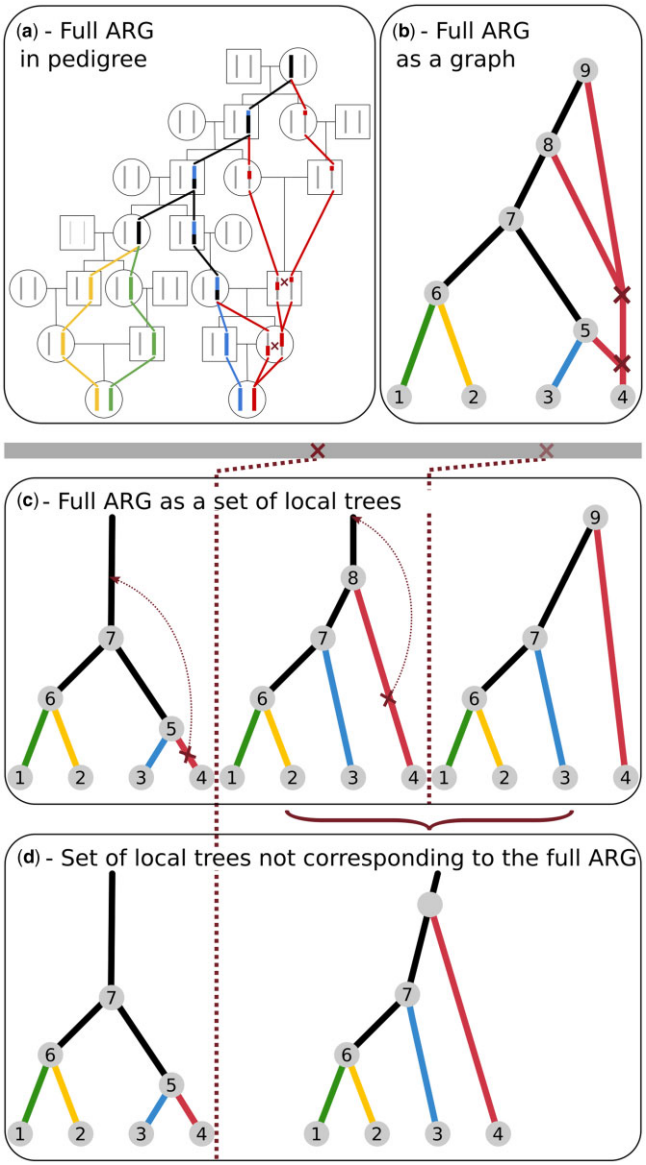

Fig. 1. Schematic representations of the genealogy of a sample of 2 diploid individuals. Colors denote the 4 haplotypes sampled, and black lines indicate lineages or sequence tracts where at least 1 coalescence has occurred. Dark red crosses indicate recombination events. a) The genealogy embedded in a pedigree. b) An ancestral recombination graph (ARG) that fully represents all genealogical relationships shown in (a), assuming that recombination events are annotated with the sequence coordinates. c) An equivalent representation of the full ARG as a set of local trees separated by a single recombination event. d) A set of trees that does not correspond to the full ARG. Instead, the second tree is an average of the local trees at that region. This set of trees is missing a recombination event that does not change topology, but changes the coalescence time. Other types of recombination events that could me missing in a partial ARG are: (1) recombination followed by coalescence in the same branch, which does not change topology or other coalescence times and (2) topology changing recombination events.

represented efficiently in computer memory by noting that each branch is part of many marginal trees (note repeated node numbers across trees in Fig. 1c). This property has been explored in the "tree sequence" format (Kelleher et al. 2018).

The full ARG contains all the information in a sample of DNA sequences regarding demography. Specifically, for a set of demographic parameters θ, parameters of the mutational process μ, sequence data x, and ARG G, $p(x|\theta,\mu,G) = p(x|\mu,G)$, i.e. if G is known there is no more information in the data about θ. A similar statement can be made for recombination and selection; if the leaf nodes of G are augmented with the allelic state at the selected loci. Therefore, the ARG is necessarily at least as informative as the combination of any and all summary statistics traditionally used to infer evolutionary processes (such as F_{ST}, π, Tajima's D, or EHH). Knowledge of the ARG is key for constructing powerful methods for extracting population genetic information from DNA sequencing data.

Inferring ARGs

Unfortunately, ARGs cannot be directly observed but must be inferred from the data. Together with an estimate of the ARG, it is desirable to quantify the uncertainty around the inferred ARG, for example by obtaining samples of ARGs according to their posterior probabilities under a given model (we discuss examples of these models in the next section). Such samples can be used to quantify uncertainty regarding ARG inferences in downstream analyses. Accurate sampling from the posterior distribution is especially relevant for downstream methods that rely on importance sampling to infer evolutionary parameters from ARGs. In essence, these methods weight parameter inference under each sampled ARG by the ARG probability and therefore require that the samples of ARGs accurately reflect their probability distribution. These types of methods can be used to infer population size history, selection (Stern et al. 2019), migration (Osmond and Coop 2021), mutation rates, and recombination rates.

Inferring full ARGs and quantifying inference uncertainty by sampling from the posterior distribution is a challenging problem computationally. It requires navigating a high-dimensional distribution of ARGs, which are themselves a complicated data structure. For this reason, inferring ARGs and sampling from their posterior distribution seemed like a nearly impossible endeavor some years ago, but important methodological developments now allow us to do so. Today, there are several methods available to estimate the full ARG or approximations of it, including ARGweaver (Rasmussen et al. 2014), Relate (Speidel et al. 2019), and tsinfer + tsdate (Kelleher et al. 2019; Wohns et al. 2022).

Approximations of the coalescent with recombination

The classical way to include recombination in coalescence models is to consider the temporal process of lineage splitting caused by recombination and lineage merger caused by coalescences as one moves backwards in time (Hudson 1983; Griffiths and Marjoram 1997) (Fig. 1, a and b). Wiuf and Hein (1999) considered instead the spatial process of recombination along a sequence. In this formulation, the ARG is constructed as a sequence of local coalescent trees along a genome, where each tree is separated from adjacent trees by recombination events (Fig. 1c). At each recombination breakpoint, a new tree is formed from the immediately preceding tree. To form the next tree, first one of the branches in the current tree is detached. Next, a point earlier than the detachment point is randomly chosen from any of the branches in any of the previous trees in the sequence. Finally, the detached branch coalesces to this chosen point.

To improve the computational efficiency in simulations, McVean and Cardin (2005) proposed approximating the spatial process as a Markovian process called the Sequentially Markovian Coalescent (SMC). In the SMC, when a lineage is detached from a tree at a recombination event, it can only coalesce back to one of the other lineages present at the current tree. Marjoram and Wall (2006) proposed an improved approximation,

the SMC', in which the detached lineage can coalesce to any branch in the current tree, including the one it was detached from. This means that some recombination events in this model do not generate a different local coalescent tree. This simple modification significantly improves the model in terms of approximating the full coalescent (Marjoram and Wall 2006; Wilton et al. 2015).

A heuristic approximation to the coalescence with recombination proposed by Li and Stephens (2003), extending ideas from Stephens and Donnelly (2000), approximates the coalescent with recombination using a copying process where 1 sequence is modeled as a copy of other sequences in the sample, with errors representing mutations and switches in the copying template representing recombination events. While this model has disadvantages, such as a dependence on the input order of sequences, it has proved computationally convenient for many purposes, including demography inference, introgression detection, and more (Sheehan et al. 2013; Steinrücken et al. 2018, 2019).

The formulation of the coalescent with recombination approximated as a Markovian process generating tree sequences in the SMC (McVean and Cardin 2005) and SMC' (Marjoram and Wall 2006) and as a copying process of individual sequences by Li and Stephens (2003), paved the way for more scalable ARG inference methods. Notably, ARGweaver (Rasmussen et al. 2014) based on the SMC or SMC' model, and Relate (Speidel et al. 2019), and tsinfer + tsdate (Kelleher et al. 2019; Wohns et al. 2022) based on the model by Li and Stephens (2003).

ARGweaver

ARGweaver uses Markov Chain Monte Carlo (MCMC) to sample ARGs from the posterior distribution under the SMC or SMC'. It relies on a discretization of time (such that all recombination and coalescence events are only allowed to happen at a discrete set of time points) which makes the state space of ARGs finite countable and allows the use of discrete state-space Hidden Markov Models (HMMs). It then uses a lineage threading approach, which is a Gibbs sampling update, to sample the history of a single lineage or haplotype from the full conditional posterior distribution given the rest of the ARG connecting all other haplotypes.

Relate

Relate simplifies the problem of ARG inference by inferring marginal coalescence trees, instead of full ARGs. Inference is divided into 2 steps. First, the Li and Stephens (2003) haplotype copying model is used to calculate pairwise distances between samples in order to infer local tree topologies. Next, it uses MCMC under a coalescent prior to infer coalescence times on those local trees. Relate is able to output samples of coalescence times from the posterior distribution using this MCMC approach, but it does so for the same fixed sequence of tree topologies. This is different from the ARGweaver MCMC sampling, which also samples the tree topology space (Table 1).

Tsinfer, tsdate, and the tree sequence framework

Tsdate (Wohns et al. 2022) is a method that estimates coalescence times of tree sequences. Here, we used this method to date tree sequences inferred by tsinfer (Kelleher et al. 2019). Similarly to Relate, tsinfer is also based on the copying process from Li and Stephens (2003). A key innovation of tsinfer is a highly efficient tree sequence data structure which stores sequence data and genealogies (Kelleher et al. 2016, 2018, 2019; Ralph et al. 2020). Tsinfer performs inference in 2 steps. First, it recreates ancestral haplotypes based on allele sharing between samples. Next, it uses an HMM to infer the closest matches between ancestral haplotypes and the sampled haplotypes using an ancestral copying process modified from the classical Li and Stephens (2003) model to generate the tree topology. Finally, nodes in tree sequences inferred by tsinfer can be dated by tsdate. Tsdate uses a conditional coalescent prior, where the standard coalescent is conditioned on the number of descendants of each node on a local tree. Like ARGweaver, tsdate also discretizes time for computational efficiency. This framework infers a fixed topology and coalescence time, but it has the potential to sample coalescence times.

Benchmarking of ARG inference methods

Here, we use standard neutral coalescent simulations to benchmark coalescence time inferences in ARGweaver (Rasmussen et al. 2014), Relate (Speidel et al. 2019), and tsinfer + tsdate (Kelleher et al. 2019; Wohns et al. 2022). We focus mainly on ARGweaver and Relate because they report measures of uncertainty in inference by allowing the user to output multiple samples from the posterior distribution. Sampling from the posterior is not currently implemented in tsdate (Table 1), but we include it in this evaluation because it is a promising framework for very fast tree-sequence inference, and it will likely provide an option to output samples from the posterior distribution of tree-sequences in future updates.

We focus our analyses on coalescence times not only because they are a very informative statistic about evolutionary processes, but also because they can be fairly compared across all methods. More specifically, ARGweaver and tsdate allow for polytomies (i.e. more than 2 branches coalesce at the same node). Relate, on the other hand, does not allow polytomies. Comparing topologies with and without polytomies could bias our results depending on how we chose to deal with polytomies, so we decided to focus on coalescence times only.

We run coalescent simulations on msprime (Kelleher et al. 2016) and compare the true (simulated) ARGs to the ARGs inferred by ARGweaver, Relate, and tsinfer + tsdate. We compare the ARGs with respect to their pairwise coalescence times using 3 different types of evaluation (Fig. 2). First, we compare the true pairwise coalescence time at each site to the inferred time. Second, we compare the overall distribution of pairwise coalescence times across all sites and all MCMC samples to the

Table 1. Genome-wide genealogy inference programs compared.

Program	Samples topologies	Samples coalescence times	Supports demographic model	Scalability (number of genomes)	Outputs full ARG	Supports unphased data
ARGweaver	Yes	Yes	No	~50	Yes	Yes
ARGweaver-D[a]	Yes	Yes	Yes	~50	Yes	Yes
Relate	No	Yes	No	$\sim 10^3$	No	No
tsinfer+tsdate	No	No	No	$\sim 10^5$	No	No

[a] Hubisz et al. (2020).

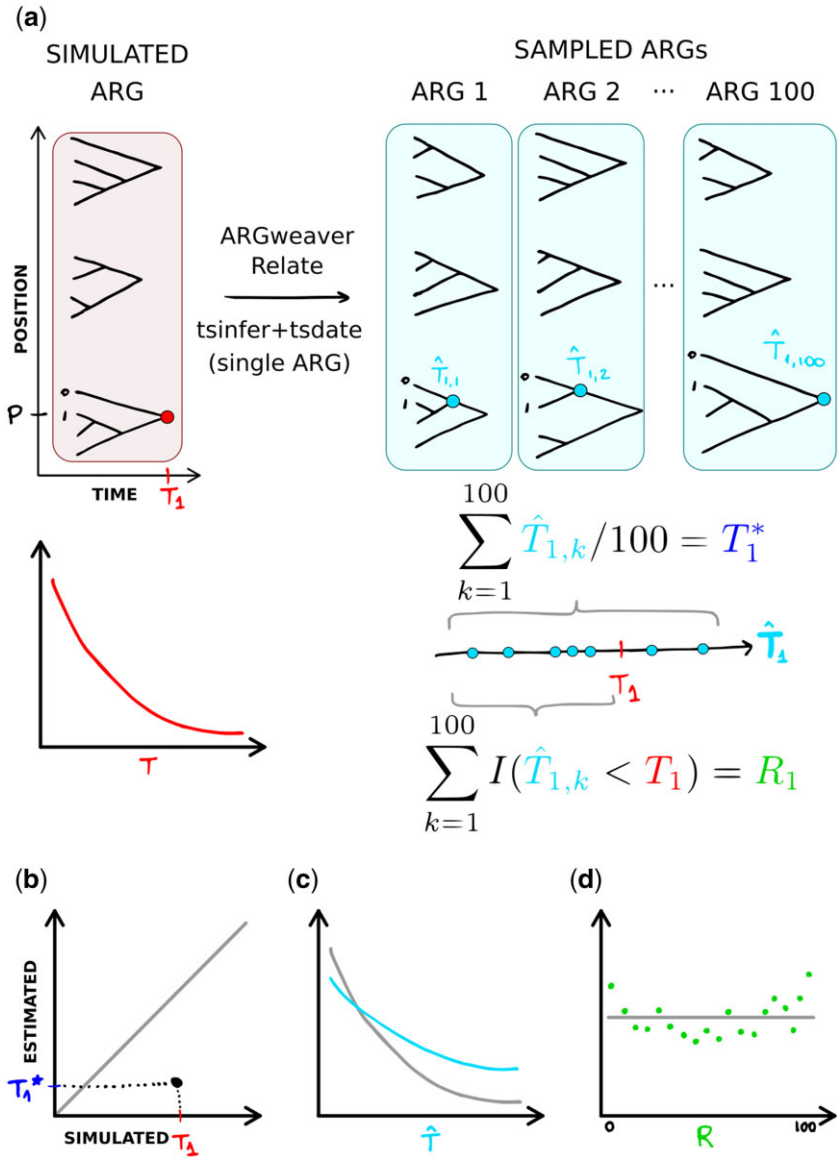

Fig. 2. Methods overview. a) Data (ARGs and DNA sequences) were simulated from the coalescent with recombination. In the model and simulated data, pairwise coalescence times (CT) are exponentially distributed (Supplementary Fig. 3). T_1 represents the CT between samples 0 and 1, at position P in the simulated data. $\hat{T}_{1,k}$ is the CT between samples 0 and 1 at position P, in each ARG sample k. Point estimates T_1^* are obtained as the mean of $\hat{T}_{1,k}$, and the rank statistic is computed as the number of $\hat{T}_{1,k}$ that are smaller than the true value T_1. b) We compare estimated to simulated values of the CT of each pair of samples, at each position of the genome. c) We compare the distribution of sampled CT across all sampled ARGs, all sites and all pairs of samples to the expected exponential distribution. d) We compare the distribution of ranks to the expected uniform distribution.

expected distribution. In Bayesian inference, the data averaged posterior distribution is equal to the prior. Since data are simulated under the standard coalescent with recombination the data averaged posterior should be exponential with rate 1 in coalescence time units ($2N_e$ generations, where N_e is the effective population size). Third, we used simulation-based calibration (SBC) (Cook et al. 2006; Talts et al. 2020) to evaluate if the posterior distributions sampled by ARGweaver and Relate are well calibrated (see details in *Methods*).

Methods

Simulations

We simulated tree sequences and SNP data with msprime version 0.7.4 (Kelleher et al. 2016). For simulations with Jukes and Cantor (1969) mutational model, we used msprime version 1.0.2 (Baumdicker et al. 2021) to add mutations to trees simulated under msprime 0.7.4, because the Jukes and Cantor (1969) model option was not available in msprime 0.7.4. Unless otherwise noted, simulations were done under the standard neutral coalescent (Hudson model in msprime) and using the following parameters: 4 diploid samples (i.e. 8 haplotypes), total map length $R = 20{,}000$ and mutation to recombination rate ratio $\mu/\rho = 1$. In practice, we used the following parameter values in msprime: effective population size of 10,000 diploids ($2N_e = 20{,}000$), mutation rate and recombination rate of 2×10^{-8} per base pair per generation and a total sequence length of 100 Mb.

We varied these standard simulation scenarios in several ways: using SMC and SMC' models, different numbers of samples (4, 16, 32, and 80 haplotypes), a multiplying the mutation to recombination ratio by 10 or dividing it by 10 (in each case changing either the mutation or the recombination rates), and changing

the total length of input sequence from 100 to 5 Mb and 250 kb. These simulated sequences were then divided into 20 equally sized segments, so that ARGweaver could be run on each in parallel (see below). The minimum length of total simulated sequence (250 kb) was chosen such that the average number of pairwise differences between each of the 20 segments was 10, given a mutation rate of 2×10^{-8}.

We extracted coalescence times at all sites in the simulated trees in BED format (columns: chromosome, start position, end position, and coalescence time), with 1 BED file for each pair of samples. Figure 2 shows an overview of the metrics extracted from simulated ARGs and from ARGs estimated by tsinfer + tsdate or sampled from the posterior by ARGweaver and Relate.

ARGweaver

VCF files from msprime were converted to ARGweaver sites format using a custom python script. We ran ARGweaver's *arg-sample* program to sample ARGs. This was done in parallel on 20 segments of equal size, using the *–region* option. We used the same values used in the msprime simulations (*–mutrate* and *–recombrate 2e-8* and *–popsize 10000*) and except where otherwise noted, we ran ARGweaver using the SMC' model (*–smcprime* option). We ran ARGweaver with 1,200 or 2,200 iterations (*–iters*) (with burn-in of the first 200 or 1,200 iterations, respectively), depending on how long it took to converge. Assessment of convergence is described below and in the Supplementary Materials, Evaluating MCMC Convergence. We extracted 100 MCMC samples from every 10th iteration among the last 1,000 iterations (default *–sample-step 10*).

We extracted all pairwise coalescence times in BED format with the program *arg-summarize* using options *–tmrca* and *–subset*, and we used bedops [version 2.4.35 (Neph et al. 2012)] to match the times sampled by ARGweaver to the simulated ones at each sequence segment. Finally, we used a custom Python script to calculate the ranks of simulated pairwise coalescence times on ARGweaver MCMC samples per site.

Time discretization

In ARGweaver, time is discretized such that recombination and coalescence events are only allowed to happen at a user-defined number of time points, K (default value is 20) (Rasmussen et al. 2014). These time points s_j (for $0 <= j <= K-1$) are given by the function

$$s_j = g(j) = \frac{1}{\delta}\left\{e^{\frac{j}{K-1}\log(1+\delta s_{K-1})} - 1\right\}, \quad (1)$$

where δ is a parameter determining the degree of clustering of points in recent times. Small values of δ lead to a distribution of points that is closer to uniform between 0 and s_{K-1}, and higher values increase the density of points at recent times (default value is 0.01) (Hubisz and Siepel 2020). Equation (1) ensures that s_0 is always 0, and s_{K-1} (or s_{max}) is user defined by the parameter *–maxtime* (default value is 200,000).

Rounding of continuous times into these K time points is done by defining bins with breakpoints between them, such that the breakpoint between times s_j and s_{j+1} is $s_{j+\frac{1}{2}} = g(j+\frac{1}{2})$. All continuous values in the bin between $s_{j-\frac{1}{2}}$ and $s_{j+\frac{1}{2}}$ are assigned the value s_j. We note that for the first and last intervals, the values assigned (s_0 and s_{K-1}) do not correspond to a midpoint in the time interval but rather to its minimum ($s_0 = 0$) or maximum ($s_{K-1} = s_{max}$)

Here, when reporting results in bins, we use the same time discretization as defined by the ARGweaver breakpoints ($s_{j+\frac{1}{2}}$). However, we change the value assigned to times in these bins: instead of using s_j, we define t_j as the median of the exponential distribution with rate 1 at the interval between $s_{j-\frac{1}{2}}$ and $s_{j+\frac{1}{2}}$. To this end, we first calculate the cumulative probability of the exponential distribution with rate 1 up to the median of the jth interval

$$p_j = \int_0^{s_{j-\frac{1}{2}}} e^{-x}dx + \frac{1}{2}\int_{s_{j-\frac{1}{2}}}^{s_{j-\frac{1}{2}}} e^{-x}dx = 1 - \left(\frac{e^{-s_{j-\frac{1}{2}}} + e^{-s_{j+\frac{1}{2}}}}{2}\right). \quad (2)$$

We then take the inverse CDF of the exponential distribution with rate 1, at the point p_j, to find the time $t_j = -\ln(1-p_j)$ corresponding to the median value for the interval.

This step is relevant for the simulation-based calibration (see below), where we take the rank of true (simulated) coalescence times relative to the values sampled by ARGweaver. If we used s_j, coalescence times in the first or last ARGweaver time interval would not be represented by a midpoint. We correct for that by using t_j, so that all time intervals are comparable.

Relate does not use time discretization, and tsdate uses a discretization scheme where the time points are the quantiles of the lognormal prior distribution on node ages (Wohns et al. 2022). Here, we always apply the ARGweaver time discretization scheme when comparing results in bins.

Relate

VCF files generated with msprime were converted to Relate haps and sample files using *RelateFileFormats –mode ConvertFromVcf* and Relate's *PrepareInputFiles* script. We ran Relate (version 1.1.2) using *–mode All* with the same mutation rate (*-m 2e-8*) and effective population size (*-N 20000*) used in the msprime simulations, as well as a recombination map with constant recombination rate along the genome, with the same rate used in msprime (*2e-8*).

We used Relate's *SampleBranchLengths* program to obtain 1,000 MCMC samples of coalescence times for the local trees inferred in the previous step in anc/mut output format (*–num-samples 1000 –format a*). Similarly to the ARGweaver analysis, we also performed this step in 20 sequence segments of 5 Mb, and we thinned the results to keep only every 10th MCMC sample. Finally, we extract pairwise coalescence times and calculate the ranks of true pairwise coalescence times relative to the 100 MCMC samples. Due to the large number of pairwise coalescence times, for the simulations with 80 and 200 samples, we extracted coalescence times from a subset of 210 pairs of samples. We extracted coalescence times for every 4th vs every 4 + 1th sample in the case of 80 samples, and 10th vs every 10 + 1th sample in the case of $n = 200$.

tsinfer and tsdate

VCF files generated by msprime were provided as input to the python API using *cyvcf2.VCF* and converted to tsinfer *samples* input object using the *add_diploid_sites* function described in the tsinfer tutorial (https://tsinfer.readthedocs.io/en/latest/tutorial.html#reading-a-vcf). Genealogies were inferred with tsinfer [version 0.2.0 (Kelleher et al. 2019)] with default settings and dated with tsdate [version 0.1.3 (Wohns et al. 2022)] using

the same parameter values as in the simulations ($Ne = 10{,}000$, *mutation_rate* = *2e-8*), with a prior grid of 20 timepoints.

Pairwise coalescence times were extracted from the tree sequences using the function *tmrca()* from tskit [version 0.3.4 (Kelleher et al. 2018)], and output in BED format, with 1 file for each pair of samples. Finally, coalescence times at each site, for each pair of samples were matched to the simulated ones (also in BED format) using bedops (Neph et al. 2012).

MCMC convergence

We evaluated MCMC convergence of Relate and ARGweaver through (1) visual inspection of trace plots, (2) autocorrelation plots, (3) effective sample sizes, and (4) the Gelman–Rubin convergence diagnostics based on potential scale reduction factor (Gelman and Rubin 1992; Brooks and Gelman 1998). Trace plots were also used to determine the number of burn-in samples, and autocorrelation plots were used to determine thinning of the samples. See evaluating MCMC convergence in Supplementary Materials for details.

Point estimates of pairwise coalescence times

We estimated pairwise coalescence times from the MCMC samples from Relate and ARGweaver by taking the average of 100 samples at each site (Fig. 2). Since tsdate does not output multiple samples of node times, we use its point estimate of pairwise coalescence times directly. Point estimates of coalescence times were compared with the simulated values for each pair of samples, at each site along the sequence.

Mean squared error (MSE) of point estimates was calculated from each point estimate of coalescence time (for each pair of samples, at each site), as well as per bin of size 0.1 of the simulated coalescence times (in units of $2N_e$ generations) for Supplementary Fig. 2. We also report Spearman's rank correlation (r_2) of the point estimates of pairwise times in each tree against the simulated tree, averaged over all positions in the genome.

Simulation-based calibration

In addition to comparing MCMC point estimates to the true simulated values, we use simulation methods proposed by Cook et al. (2006) and Talts et al. (2020) to assess whether Bayesian methods are sampling correctly from the true posterior distribution. Cook et al. (2006) proposed simulating data using parameters sampled from the prior. The posterior, when averaged over multiple simulated data sets, should then equal the prior.

In our case, we sample ARGs, G, from the full coalescence process with recombination with a known implicit prior of pairwise coalescence times, $P(t) = e^{-t}$. We simultaneously simulate sequence data, x, on the simulated ARGs from the distribution $p(x) = \int p(x|G)\mathrm{d}P(G)$. The distribution of the averaged posterior of G, $p_{\mathrm{ave}}(G) = \int p(G|X)\mathrm{d}P(x)$ should then equal the prior for G (Talts et al. 2020), and hence the prior distribution for the pairwise coalescence times, t, should equal the averaged posterior distribution for t. Here, all population parameters relating to mutation, effective population sizes, etc. are kept fixed and suppressed in the notation. One way we will examine the accuracy of the posterior inferences is, therefore, to compare the average of the posterior of t to the exponential distribution. In practice, we simulate data using msprime (Kelleher et al. 2016) and pipe the data to the MCMC samplers (ARGweaver and Relate) for inference of the posterior distribution. ARGweaver uses an approximation (SMC') of the model (coalescent with recombination) used in the data simulations, and Relate uses a heuristic method based on the Li and Stephens model. Thus, inadequacies of the fit of the posteriors could potentially be caused by this discrepancy between the model used in simulations and the models used for inference.

However, even if the averaged posterior resembles an exponential, the inferences for any particular value of t may have a posterior that is too narrow or too broad. For a closer examination of the accuracy of the posterior, we use a method proposed by Cook et al. (2006) and Talts et al. (2020) that compares each posterior to the true value. To this end, we compare each true (simulated) pairwise coalescence time to the corresponding posterior for the same pair of haplotypes. If the posterior is correctly calculated, the rank of the true value relative to the samples from the posterior should be uniformly distributed (Cook et al. 2006; Talts et al. 2020). We use 100 MCMC samples from ARGweaver and Relate for each data set, meaning our ranks take values from 0 to 100. Deviations from the uniform distribution of ranks quantifies inaccuracies in estimation of the posterior. For example, an excess of low and high ranks indicates that the inferred posterior distribution is underdispersed relative to the true posterior.

Results

Comparison of simulated to estimated coalescence time per site

We compared coalescence times estimated by ARGweaver, Relate, and tsinfer + tsdate to the true values known from msprime simulations. In all 3 methods, estimates of coalescence time per site are biased (Fig. 3; Supplementary Fig. 2). Small values of coalescence times are generally overestimated, while large values tend to be underestimated (Supplementary Fig. 2). In tsinfer+tsdate, point estimates are apparently bounded to a narrow range (Fig. 3g). The MSE of point estimates is larger in Relate (MSE = 0.625) and tsinfer + tsdate (MSE = 1.631) than in ARGweaver (MSE = 0.397), showing that point estimates of pairwise coalescence times at each site are closer to the true value in ARGweaver. Spearman's rank correlation is also highest in ARGweaver ($r_s = 0.761$), but in this metric tsinfer + tsdate ($r_s = 0.705$) perform better than Relate ($r_s = 0.669$).

For ARGweaver and Relate, the point estimates of coalescence times are obtained as the means of samples from the posterior. These Bayesian estimates are not designed to be unbiased and unbiasedness of the point estimator is arguably not an appropriate measure of performance for a Bayesian estimator. Therefore, we also evaluate the degree to which the posterior distributions reported by ARGweaver and Relate are well calibrated, i.e. represent distributions that can be interpreted as valid posteriors, and the degree to which the data-averaged posterior distributions of coalescence times equals the prior exponential distribution.

Posterior distribution of coalescence times

We simulated data under the standard coalescent model, where the distribution of pairwise coalescence times (in units of $2N_e$ generations, where N_e is the diploid effective population size) follows an exponential distribution with rate parameter 1 (Supplementary Fig. 3). As argued in the *Methods*, the same is true for the data-averaged posterior.

We compared the expected exponential distribution of coalescence times with the observed distribution of coalescence times across all sites inferred by ARGweaver, Relate, and tsinfer + tsdate (Fig. 4). For ARGweaver and Relate, we output 100 MCMC samples from the posterior distribution and plot the

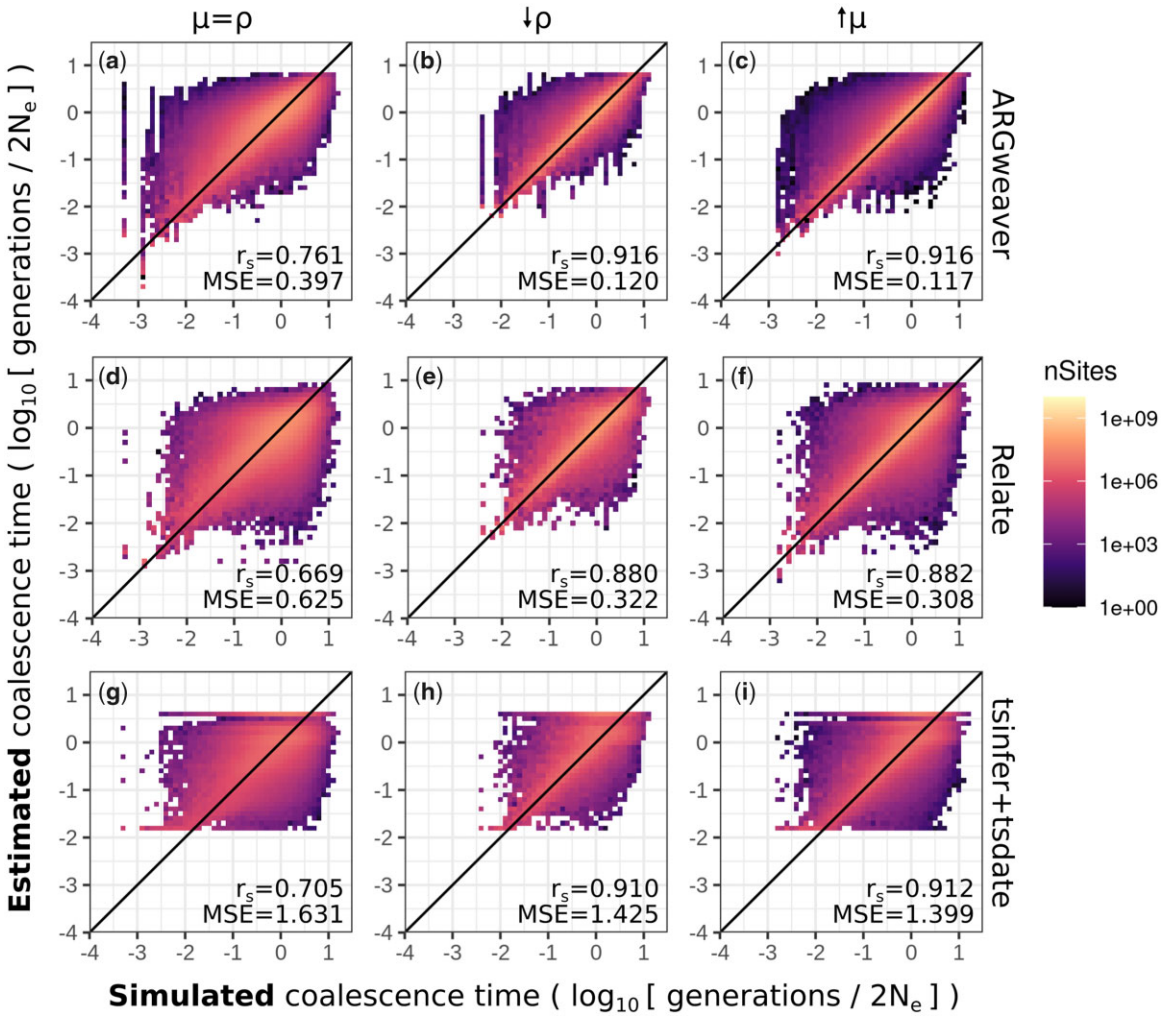

Fig. 3. Point estimates of coalescence times in ARGweaver (a–c), Relate (d–f), and tsinfer+tsdate (g–i). Left column: $\mu = \rho = 2 \times 10^{-8}$; middle column: $\mu/\rho = 10$, $\rho = 2 \times 10^{-9}$; and right column: $\mu/\rho = 10$, $\mu = 2 \times 10^{-7}$. For ARGweaver and Relate, point estimates are the means of 100 MCMC iterations. Note that axes are in log scale. See Supplementary Fig. 1 for the data in plots (a), (d), and (g) plotted in linear axes. Diagonal line shows $x = y$. MSE, mean squared error; r_s, Spearman's rank correlation.

distribution of pairwise coalescence times across all sites and MCMC samples.

To facilitate visual comparison of the distributions between methods, we discretized Relate and tsinfer + tsdate coalescence times into the same bins as ARGweaver (Fig. 4, d and g; see distributions without discretization in Supplementary Fig. 4; and see *Methods* for a description of ARGweaver time discretization). Because the time discretization breakpoints are regularly spaced on a log scale, we use a log scale on the x-axis for better visualization.

Distributions of coalescence times from ARGweaver and Relate (Fig. 4, a and d) show an excess around 1, when compared with the expected exponential distribution. However, that bias is more pronounced in Relate than ARGweaver. In tsinfer + tsdate, the distribution is truncated at 1.6, and it deviates more strongly from the expected exponential distribution (Fig. 4g). We note that the plots from ARGweaver and Relate are not directly comparable to those of tsinfer + tsdate, since there are 100 coalescence time samples at each site from the former 2 programs but only 1 from tsdate.

Simulation-based calibration

In this section, we use simulation-based calibration to evaluate whether ARGweaver and Relate are generating samples from a valid posterior distribution of coalescence times (see *Methods*). To that end, we simulated coalescence times at multiple sites following the standard coalescent prior distribution, and we generated 100 MCMC samples from the posterior distribution using both ARGweaver and Relate. Finally, we analyze the distribution of the ranks of the simulated coalescence times relative to the 100 sampled values at each site.

In the previous section, we showed that the posterior distributions of ARGweaver and Relate are similar to the theoretically expected exponential distribution. However, in that analysis we have not evaluated the distribution of MCMC samples relative to each simulated value. The results of simulation-based calibration are informative about that distribution and can reveal if the posterior distribution is well calibrated.

The distribution of ranks from ARGweaver [Fig. 5a; Kullback–Leibler Divergence (KLD) = 0.027] is closer to uniform than that of Relate (Fig. 5d, KLD = 0.602). However, both show an excess of low and high ranks. The excess of low and high ranks indicates that the sampled posterior distribution is underdispersed (Talts et al. 2020), i.e. the posterior has too little variance and does not represent enough uncertainty regarding the coalescence times.

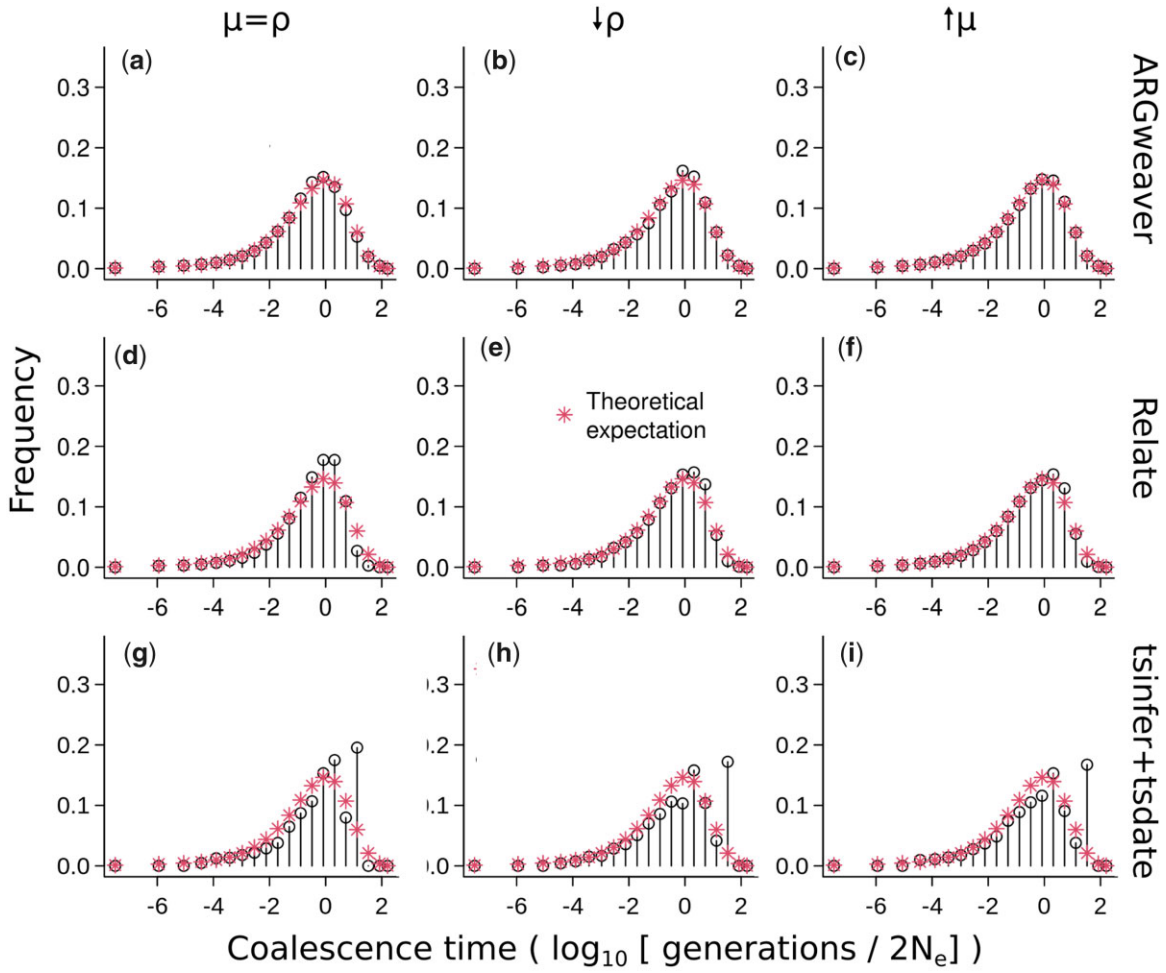

Fig. 4. Distribution of coalescence times inferred by ARGweaver (a–c), Relate (d–f), and tsinfer+tsdate (g–i). Left column: $\mu = \rho = 2 \times 10^{-8}$; middle column: $\mu/\rho = 10$, $\rho = 2 \times 10^{-9}$; and right column: $\mu/\rho = 10$, $\mu = 2 \times 10^{-7}$. Plots (d) and (g) show the same data as in Supplementary Fig. 4, using different binning.

One possible cause for this type of deviation from the uniform distribution could be MCMC convergence, i.e. samples being autocorrelated, resulting in effective sample size is lower than the number of samples taken, the MCMC chain not mixing well and/or the MCMC chain not being run long enough to achieve convergence.

We show detailed results for MCMC convergence in Relate and ARGweaver in the Supplementary Materials. Briefly, we have not found these types of convergence issues in ARGweaver or Relate with simulations of 8 haplotypes and mutation to recombination ratio of 1. Potential scale reduction factor (PSRF) from Gelman–Rubin convergence diagnostic statistics are all close to 1 (Supplementary Tables 2 and 3), and effective sample sizes are almost all larger than 100. Therefore, MCMC convergence does not seem to explain why the rank distributions are not uniform.

Increased mutation to recombination ratio

When inferring an ARG from sequence data, the information for inference comes from mutations that cause variable sites in the sequence data. The lower the recombination rate, the longer the span of local trees will be and the more mutations will be available to provide information about each local tree. More generally, an increased mutation to recombination ratio is expected to increase the amount of information available to infer the ARG.

In our standard simulations presented so far, the mutation to recombination ratio is 1 ($\mu = \rho = 2 \times 10^{-8}$). We increased the simulated mutation to recombination ratio to 10, both by dividing the recombination rate (ρ) by 10 and also by multiplying the mutation rate (μ) by 10. We expected that these scenarios would improve inference of ARGs, and consequently the estimates of pairwise coalescence times. Point estimates are better with increased mutation to recombination ratio in ARGweaver (Fig. 3, b and c), Relate (Fig. 3, e and f), and tsinfer + tsdate (Fig. 3, h and i).

The coalescence times distribution in Relate (Fig. 4, e and f) are closer to the expected with $\mu/\rho = 10$ relative to $\mu/\rho = 1$ (Fig. 4d), and the simulation-based calibration also improved (Fig. 5, d–f, KLD = 0.492 and 0.498 compared with KLD = 0.602).

The results from ARGweaver with $\mu/\rho = 10$ were more surprising, with the simulation-based calibration showing a more pronounced underdispersion of the posterior distribution (Fig. 5, b and c, KLD = 0.286 and 0.350, compared with KLD = 0.027 for $\mu/\rho = 1$). The overall distribution of coalescence times, however, showed little change (Fig. 4, b and c). One possible explanation for ARGweaver results being worse with higher mutation to recombination ratio might be that MCMC mixing is worse under those conditions, leading to convergence issues not observed for the previous scenario. Examining convergence diagnostics seems

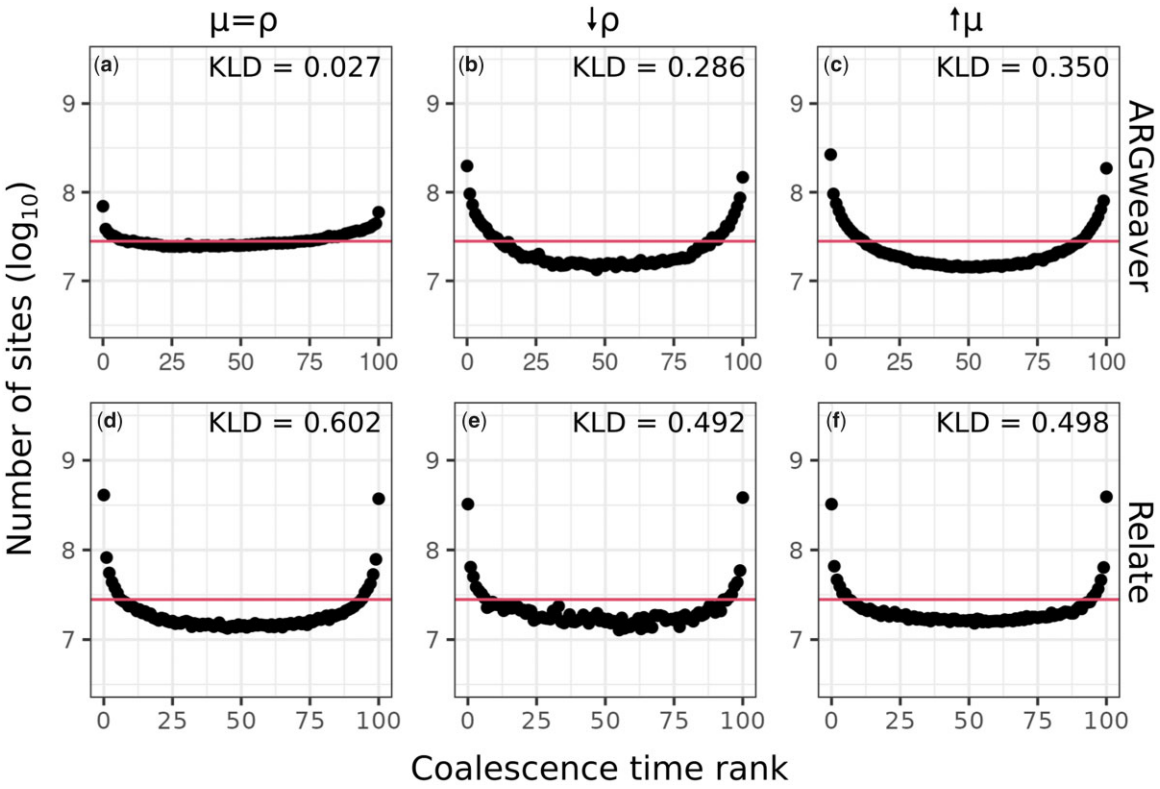

Fig. 5. Counts of ranks from simulation-based calibration in ARGweaver (a–c) and Relate (d–f). Horizontal line shows expected uniform distribution. Left column: $\mu = \rho = 2 \times 10^{-8}$; middle column: $\mu/\rho = 10$ decreasing recombination rate ($\rho = 2 \times 10^{-9}$); right column: $\mu/\rho = 10$ increasing mutation rate ($\mu = 2 \times 10^{-7}$). Horizontal line shows expected uniform distribution.

to confirm this with more coalescence times showing low effective sample size, and with a potential scale reduction factor showing evidence of lack of convergence of some coalescence times (see Evaluating MCMC convergence in Supplementary Materials).

We show additional simulation results in the Supplementary Materials, including simulations with reduced μ/ρ, which could be a realistic scenario around recombination hotspots (Supplementary Figs. 5 and 6) and ARGweaver results on simulations with intermediate values of μ/ρ (2 and 4), under the SMC and SMC' genealogy models, and with the Jukes–Cantor mutation model in the Supplementary Materials.

Number of samples

Next, we evaluate ARG inference with simulations with different sample sizes. Our standard sample size used so far was 8 haplotypes, and here we change it to 4, 16, and 32. For Relate and tsinfer + tsdate, which are scalable to larger sample sizes, we also evaluated inference with 80 and 200 sampled haplotypes.

For ARGweaver, increasing sample sizes decreased the MSE of point estimates (Fig. 6, a–c), distributions of coalescence times remained similar (Fig. 7, a–c), but underdispersion of the posterior distribution increased (Fig. 8, a–c). As mentioned in the previous section, this could be caused by an MCMC mixing problem. In particular, a larger number of samples will contribute to an increasing number of states for ARGweaver to explore, possibly leading to poor MCMC convergence (see *Evaluating MCMC convergence*).

With a smaller sample size ($n = 4$ haplotypes), the coalescence time distribution from Relate showed an excess around the mean value (coalescence time of 1) (Fig. 7d). With increasing sample sizes, it became more similar to the expected distribution (Fig. 7, e–h). Calibration of the posterior distribution improved with increasing sample sizes up to 32 haplotypes (Fig. 8, d–h).

Both the point estimates and posterior distribution of coalescence times in tsinfer + tsdate do not consistently improve or worsen with increasing sample sizes in the range tested here (Figs. 6, l–m and 7, l–m).

Length of input sequence

Point estimates of all programs remained similarly accurate when a much shorter input sequence was provided (5 mb and 250 kb, Supplementary Figs. 7, a–c and 8, a–c, compared with 100 Mb in previous analyses). The distribution of coalescence times with 5 Mb input sequence remained similar to the ones inferred with 100 Mb input sequence (Supplementary Fig. 7, d–f). However, distributions from simulations with only 250 kb input sequence are visibly more deviated from the expected exponential distribution (Supplementary Fig. 8, d–f). Distributions of ranks are noisier with decreasing input sequence length, but KLD remained similar (Supplementary Figs. 7, h and g and 8).

Runtime

We point out that runtimes differ widely among the programs compared here, and this factor should be taken into account for users making decisions on what method to use for their applications. For example, in the simulations with mutation rate equal to recombination rate, with sample size of 8 haplotypes and taking 1,000 MCMC samples, ARGweaver took a total of 641 computing hours while Relate took 17 h. The clock time was reduced by running both programs in parallel for segments of 5 Mb of the total 100 Mb sequence, meaning that ARGweaver took

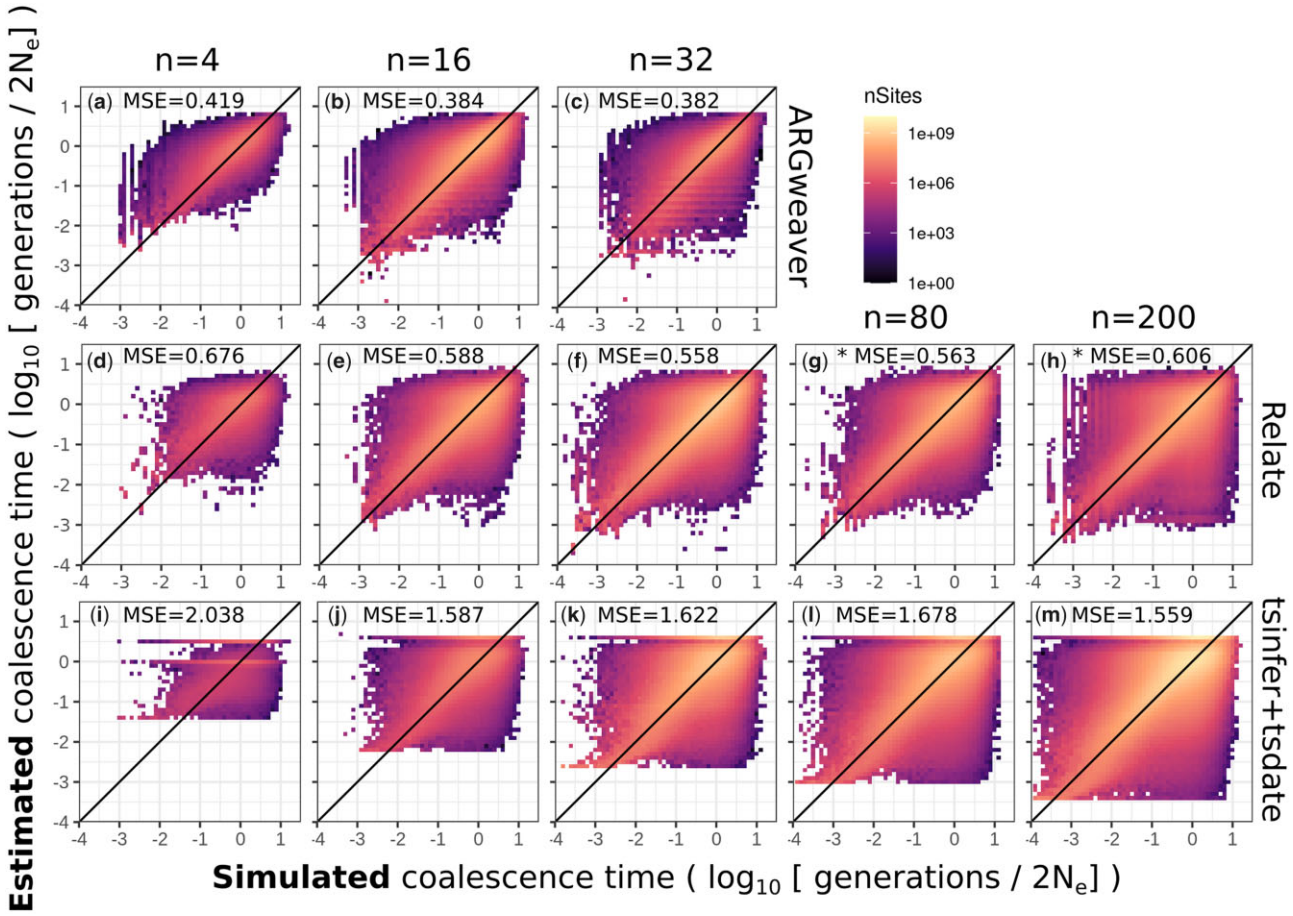

Fig. 6. Point estimates of ARGweaver (a–c), Relate (d–h), and tsinfer+tsdate (i–m). Columns show different number of simulated samples 4, 16, 32, 80, or 200 haplotypes. Mean squared error (MSE) is shown for each plot. Note that ARGweaver is not scalable for simulations with larger sample sizes. *Results for a subset of 210 pairs of samples, instead of all pairwise coalescence times.

approximately 35 h. However, this still could be a significant amount of time for the user, depending on their utilization of the algorithm. For a systematic comparison of runtimes between Relate and ARGweaver, see Speidel et al. (2019). Impressively, tsinfer and tsdate took only 5 min.

Discussion

ARG inference promises to be a tremendously useful tool for inferences of evolutionary history, such as natural selection or demography. However, it is also a very hard computational problem. We compared methods that use different approaches to this problem and evaluated their accuracy using simulated data and comparisons of 3 aspects of coalescence time estimates: (1) individual point estimates of each pairwise coalescence time; (2) the overall distribution of coalescence times across all sites; (3) the calibration of the reported posterior distributions.

ARGs are extremely rich in information, including topological information of individual coalescence trees and information regarding the distribution of recombination events. We have not evaluated these aspects of inferred ARGs but have instead only focused on pairwise coalescence times. However, pairwise coalescence times are extremely informative statistics about many population-level processes and pairwise relationships between individuals, and they are also indirectly informative about tree topologies. Other research has compared the accuracy of tree topology inference (Rasmussen et al. 2014; Kelleher et al. 2019) and recombination rates (Deng et al. 2021) among ARG inference methods. We opted to focus on coalescence times not only because they are a very informative statistic about evolutionary processes, but also because they can be fairly compared across all methods. As described in the *Introduction*, comparisons of tree topologies could be confounded by the presence of polytomies in ARGweaver and tsinfer + tsdate and the absence of polytomies in Relate.

We found a strong speed-accuracy trade-off in ARG inference. ARGweaver performs best in our 3 tests: point estimates, the overall distribution of coalescence times, and the quality of sampling from the posterior. Importantly, it is also the only method we compared that resamples both topologies and node times (Table 1). This likely leads to a better exploration of ARG space and is 1 reason why it provides better samples from the posterior. On the other hand, it also contributes to making ARGweaver much slower than the other methods and not scalable for genome-wide inference of 50 or more genomes.

Relate largely undersamples tree topologies (Deng et al. 2021), and thus every marginal tree estimate is only as good as an average over a series of true trees (Fig. 1d). This will naturally lead to a more centered, under-dispersed distribution, as shown by the larger deviations from the uniform distribution in simulation-based calibration (Figs. 5 and 8, where ARGweaver KLD values range from 0.008 to 0.350, and Relate range from 0.429 to 0.938). Despite not performing as well as ARGweaver in our evaluation criteria, Relate seems sufficient for comparisons of average trees across different regions in the genome.

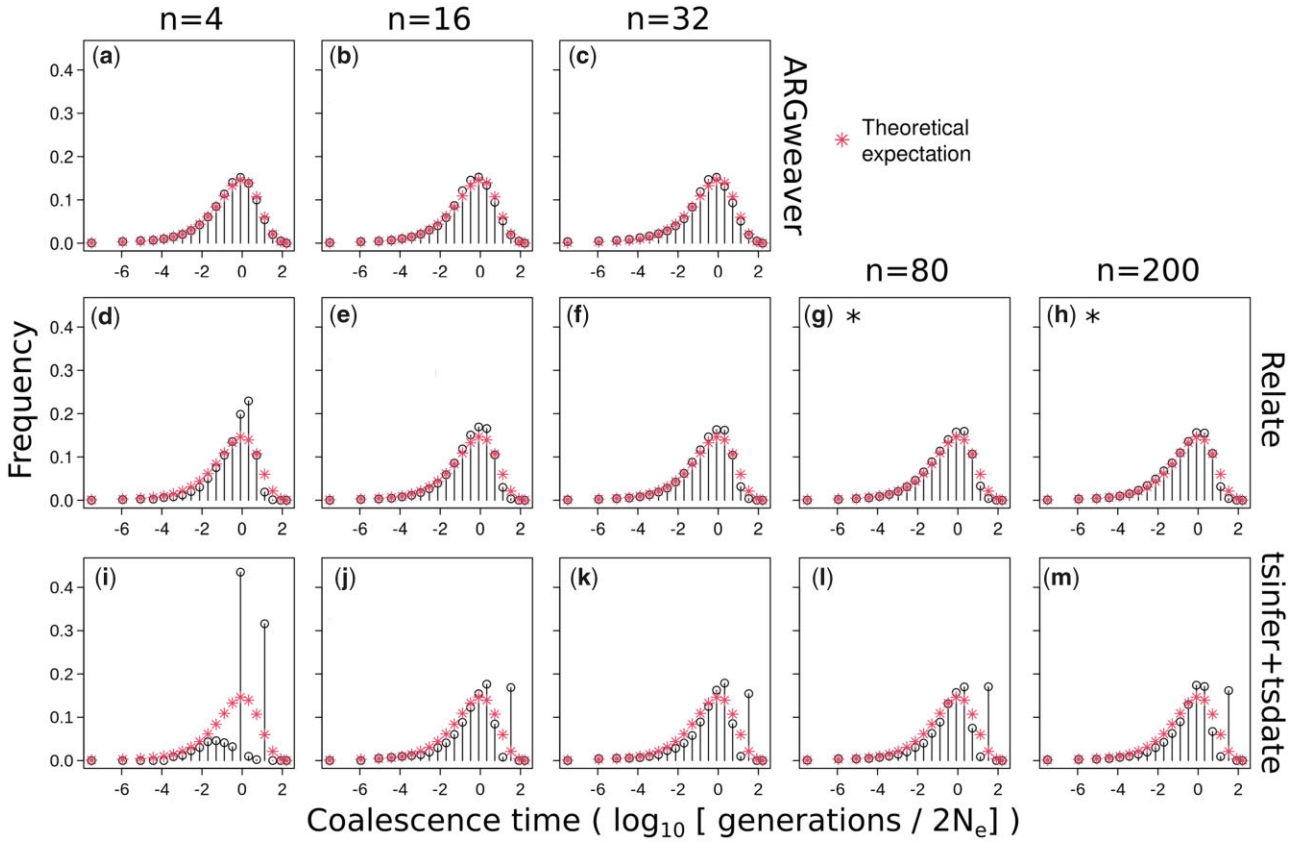

Fig. 7. Distribution of coalescence times in ARGweaver (a–c), Relate (d–h), and tsinfer+tsdate (i–m). Columns: sample sizes of 4, 16, 32, 80, and 200 haplotypes. *Results for a subset of 210 pairs of samples, instead of all pairwise coalescence times.

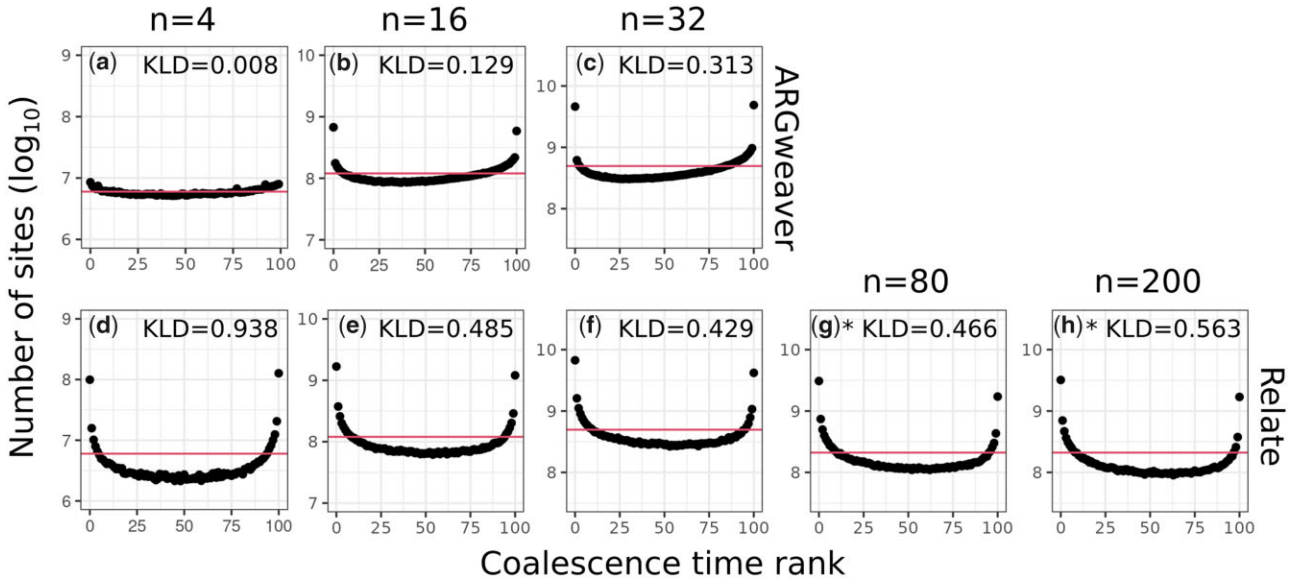

Fig. 8. Simulation-based calibration for ARGweaver (a–c) and Relate (d–h). Columns: sample sizes of 4, 16, 32, 80, and 200 haplotypes. Horizontal line shows expected uniform distribution. Note that the y-axis is centralized on different values but always has the same length. *Results for a subset of 210 pairs of samples, instead of all pairwise coalescence times.

Additionally, we showed that Relate's inferences generally improve with sample size (Figs 6–8). This is expected from inference using the Li and Stephens (2003) copy algorithm, which tends to better approximate the genealogical process with larger samples sizes (Hubisz et al. 2020). Because Relate is fast enough, even for thousands of samples, it is preferred for large numbers of genomes—not only because ARGweaver is not scalable for such large sample sizes but also because Relate inference tends to improve with larger sample sizes (Hubisz and Siepel 2020).

The framework of tsinfer and tsdate is also based on the Li and Stephens (2003) model, and it additionally takes advantage of the succinct tree sequence data structure that makes it scalable to even larger sample sizes than Relate, and at least an order of magnitude larger than tested here (Wohns et al. 2022). Although we did not find an improvement of tsinfer + tsdate estimates with increasing sample sizes in the range we tested (4–200 haplotypes), our analyses cannot rule out the possibility of better tsinfer + tsdate inference at larger sample sizes.

Increasing the mutation to recombination ratio in simulations improved point estimates from ARGweaver but did not improve posterior calibration (Fig. 5). This lack of improvement of the posterior sampling can be explained by lack of convergence and could potentially be improved by increasing the number of MCMC iterations. Although the statistics recorded by ARGweaver at each iteration (likelihood, number of recombinations, etc.) show convergence (Supplementary Fig. 9 and Supplementary Table 1), we observed that certain pairwise coalescence times did not converge in the simulations with increased mutation to recombination ratio (Supplementary Table 2, see more discussion in ARGweaver in Supplementary Materials).

Limitations of our analyses and future directions

The focus of this study is the inference of coalescence times under the standard neutral coalescent, assuming all parameter values of this model are known and correctly provided to the programs performing inference. In other words, our goal was to investigate the performance of the ARG inference methods when the underlying assumptions are met. We have not explored how the methods perform under more complex demographic models and in the presence of natural selection, when the underlying assumptions are not met, but this is clearly an important future direction.

We also restrict our analyses to small sample sizes relative to what is possible for Relate and tsinfer + tsdate. However, increasing sample sizes up to 200 samples does not consistently improve performances of these methods. We also note that interesting discoveries have been made by applying ARG-based methods with similarly small sample sizes, e.g. Hubisz et al. (2020) analyzed gene flow between archaic and modern humans using 5 genomes: 2 Neanderthals, 1 Denisovan, and 2 modern humans.

Other factors not explored here could also be relevant for applications to real data. For example, sequencing or phasing errors could reduce the performance of all methods. Each of the methods compared here deal with these problems in a different way. Both Relate and tsinfer require phased data. While Speidel et al. (2019) argue that Relate is robust to errors in computational phasing, Kelleher et al. (2019) acknowledge that phasing errors could reduce the performance of tsinfer. ARGweaver is the only method of the 3 that supports unphased data, by integrating over all possible phases. However, the performance of the program on unphased data has not been evaluated in this study.

Relate takes sequencing errors into account by allowing some mutations that are incompatible with the tree topology in its tree building algorithm. Some robustness to error is shown in Speidel et al. (2019, Supplementary Fig. 3). Tsdate also uses heuristics in the ancestral haplotype reconstruction stage to increase its robustness to genotyping errors (Kelleher et al. 2019), and its newest version also accounts for recurrent mutation. ARGweaver can deal with genotyping errors statistically, using genotype likelihoods and integrating over all possible genotypes (Hubisz and Siepel 2020). In addition, it can take into account local variation in coverage and mapping quality, all of which are features not tested here. ARGweaver can also incorporate a map of variable mutation rates. ARGweaver, Relate, and tsinfer can all incorporate maps of variable recombination rates across the genome, a feature which was not used in our constant rate simulations.

In our standard simulations, we use mutation rate equal to recombination rate, which is believed to be approximately true for humans. In reality, even if average recombination and mutation rates are similar, the average recombination rate is not distributed equally along the genome in humans and other mammals but is concentrated in recombination hotspots. Therefore, it is possible that ARG inference could be more accurate with real data, since local trees could span longer sequences separated by recombination hotspots.

Recommendations for usage

Given that ARGweaver provides the most accurate coalescence times estimates and the most well-calibrated samples from the posterior distribution of coalescence times, we recommend using it whenever computationally feasible. However, it is highly computationally demanding and its usage can become unfeasible with sample sizes close to 100. Running ARGweaver on small segments of sequence (5 Mb or 250 kb; Supplementary Figs. 7 and 8) gave similar results to applications on 100 Mb segments, making the program highly parallelizable, at least for the purpose of estimating pairwise coalescence times.

When ARGweaver is computationally prohibitive, Relate, and tsinfer + tsdate are viable alternative options. However, we emphasize that we have only examined coalescence time estimates, and for other downstream uses of ARG inference that do not rely mostly on coalescence times, the tradeoffs between these methods could be different. See Deng et al. (2021) for a comparison of these methods in the context of estimating recombination rates.

Data availability

The data underlying this article are available in GitHub https://github.com/deboraycb/ARGsims.

Supplemental material is available at *GENETICS* online.

Acknowledgments

We would like to thank Leo Speidel for identifying an error in the extraction of coalescence times from the Relate output in a previous version of the manuscript. We also thank Nicholas Barton, Adam Siepel, Ziyi Mo, Jerome Kelleher, Yan Wong, Wilder Wohns, and other reviewers for constructive and detailed reviews that greatly improved this paper.

Funding

This material is based upon work supported by National Science Foundation Graduate Research Fellowship No. 2146752 awarded to Andrew Vaughn and by National Institutes of Health grant R01GM138634 awarded to Rasmus Nielsen.

Conflicts of interest

None declared.

Literature cited

Baumdicker F, Bisschop G, Goldstein D, Gower G, Ragsdale AP, Tsambos G, Zhu SEB, Ellerman, CE Galloway, JG, *et al.* Efficient ancestry and mutation simulation with msprime 1.0. bioRxiv 17:2021.08.31.457499; 2021. https://doi.org/10.1093/genetics/iyab229

Brooks SP, Gelman A. General methods for monitoring convergence of iterative simulations. J Comput Graph Stat. 1998;7:434–455.

Cook SR, Gelman A, Rubin DB. Validation of software for Bayesian models using posterior quantiles. J Comput Graph Stat. 2006; 15(3):675–692.

Deng Y, Song YS, Nielsen R. The distribution of waiting distances in ancestral recombination graphs. Theor Popul Biol. 2021;141: 34–43. https://doi.org/10.1016/j.tpb.2021.06.003

Gelman A, Rubin DB. Inference from iterative simulation using multiple sequences. Statist Sci. 1992;7(4):457–472.

Griffiths RC, Marjoram P. An ancestral recombination graph. In: P Donnelly, S Tavare, editors. Progress in Population Genetics and Human Evolution, IMA Volumes in Mathematics and Its Applications, Vol. 87. New York, NY: Springer; 1997. p. 257–270.

Hubisz M, Siepel A. Inference of ancestral recombination graphs using ARGweaver. In: JY Dutheil, editor. Statistical Population Genomics, Vol. 2090, Chapter 10; New York, NY: Humana; 2020. p. 231–266.

Hubisz MJ, Williams AL, Siepel A. Mapping gene flow between ancient hominins through demography-aware inference of the ancestral recombination graph. PLoS Genet. 2020;16(8):e1008895.

Hudson RR. Properties of a neutral allele model with intragenic recombination. Theor Popul Biol. 1983;23(2):183–201.

Jukes TH, Cantor CR. Evolution of protein molecules. In: HN Munro, editor. Mammalian Protein Metabolism, Chapter 24. New York, NY: Academic Press; 1969. p. 21–132.

Kelleher J, Etheridge AM, McVean G. Efficient coalescent simulation and genealogical analysis for large sample sizes. PLoS Comput Biol. 2016;12(5):e1004842.

Kelleher J, Thornton KR, Ashander J, Ralph PL. Efficient pedigree recording for fast population genetics simulation. PLoS Comput Biol. 2018;14(11):e1006581.

Kelleher J, Wong Y, Wohns AW, Fadil C, Albers PK, McVean G. Inferring whole-genome histories in large population datasets. Nat Genet. 2019;51(9):1330–1338.

Kingman JFC. On the Genealogy of Large Populations. Technical Report; 1982.

Li N, Stephens M. Modelling linkage disequilibrium using single nucleotide polymorphism data. 2003;2233:2213–2233.

Marjoram P, Wall JD. Fast "coalescent" simulation. BMC Genet. 2006; 7:16.

McVean GAT, Cardin NJ. Approximating the coalescent with recombination. Philos Trans R Soc Lond B Biol Sci. 2005;360(1459): 1387–1393.

Neph S, Kuehn MS, Reynolds AP, Haugen E, Thurman RE, Johnson AK, Rynes E, Maurano MT, Vierstra J, Thomas S, *et al.* BEDOPS: high-performance genomic feature operations. Bioinformatics. 2012;28(14):1919–1920.

Osmond M, Coop G. Estimating dispersal rates and locating genetic ancestors with genome-wide genealogies. bioRxiv 2021.07.13. 452277; 2021. https://doi.org/10.1101/2021.07.13.452277

Ralph P, Thornton K, Kelleher J. Efficiently summarizing relationships in large samples: a general duality between statistics of genealogies and genomes. Genetics. 2020;215(3):779–797.

Rasmussen MD, Hubisz MJ, Gronau I, Siepel A. Genome-wide inference of ancestral recombination graphs. PLoS Genet. 2014;10(5): e1004342.

Sheehan S, Harris K, Song YS. Estimating variable effective population sizes from multiple genomes: a sequentially Markov conditional sampling distribution approach. Genetics. 2013;194(3): 647–662.

Speidel L, Forest M, Shi S, Myers SR. A method for genome-wide genealogy estimation for thousands of samples. Nat Genet. 2019; 51(9):1321–1329.

Steinrücken M, Kamm J, Spence JP, Song YS. Inference of complex population histories using whole-genome sequences from multiple populations. Proc Natl Acad Sci U S A. 2019;116(34): 17115–17120.

Steinrücken M, Spence JP, Kamm JA, Wieczorek E, Song YS. Model-based detection and analysis of introgressed Neanderthal ancestry in modern humans. Mol Ecol. 2018;27(19):3873–3888.

Stephens M, Donnelly P. Inference in molecular population genetics. J Roy Stat Soc B. 2000;62(4):605–635.

Stern AJ, Wilton PR, Nielsen R. An approximate full-likelihood method for inferring selection and allele frequency trajectories from DNA sequence data. PLoS Genet. 2019;15(9):e1008384.

Talts S, Betancourt M, Simpson D, Vehtari A, Gelman A. Validating Bayesian inference algorithms with simulation-based calibration. arXiv 2020;1–19. https://doi.org/10.48550/arXiv.1804.06788

Wilton PR, Carmi S, Hobolth A. The SMC' is a highly accurate approximation to the ancestral recombination graph. Genetics. 2015; 200(1):343–355.

Wiuf C, Hein J. Recombination as a point process along sequences. Theor Popul Biol. 1999;55(3):248–259.

Wohns AW, Wong Y, Jeffery B, Akbari A, Mallick S, Pinhasi R, Patterson N, Reich D, Kelleher J, McVean G. A unified genealogy of modern and ancient genomes. Science. 2022;375(6583):1–9.

Communicating editor: N. Barton