Nature Communications

https://doi.org/10.1038/s41467-026-71057-z

Article in Press

Scaling up Bayesian population phylogenomics through virtual dimension reduction

Received: 30 July 2025

Accepted: 10 March 2026

Cite this article as: Flouri, T., Jiao, X., Huang, J. *et al.* Scaling up Bayesian population phylogenomics through virtual dimension reduction. *Nat Commun* (2026). https://doi.org/10.1038/s41467-026-71057-z

Tomáš Flouri, Xiyun Jiao, Jun Huang, Bruce Rannala & Ziheng Yang

Scaling up Bayesian population phylogenomics through virtual dimension reduction

Tomáš Flouri[1*†], Xiyun Jiao[2*†], Jun Huang[3], Bruce Rannala[4*], Ziheng Yang[1*]

[1*]Department of Genetics, Evolution, and Environment, University College London, Gower Street, London WC1E 6BT, UK.
[2]Department of Statistics and Data Science, China Southern University of Science and Technology, Shenzhen, Guangdong 518055, China.
[3]School of Biomedical Engineering, Capital Medical University, Beijing, 100069, China.
[4]Department of Evolution and Ecology, University of California, Davis, CA 95616, USA .

*Corresponding author(s). E-mail(s): t.flouris@ucl.ac.uk; jiaoxy@sustech.edu.cn; brannala@ucdavis.edu; z.yang@ucl.ac.uk;
†These authors contributed equally to this work.

Abstract

Population phylogenomics uses sampled genomes to jointly infer population genetic processes (ancestral and contemporary population sizes, historical gene flow) and a phylogenetic tree relating species or populations including species split times. This challenging problem has been tackled most successfully in the Bayesian framework under the multispecies coalescent (MSC) model via Markov chain Monte Carlo (MCMC) computational algorithms. However, MCMC methods suffer from two serious problems: (i) mixing difficulties due to the high-dimensional state space with complex constraints, and (ii) the intrinsically serial nature of MCMC algorithms that defies parallelisation. To deal with both issues, we develop a new method, called Virtual Dimension Reduction allowing Parallelisation (VDRoP), that achieves the same MCMC mixing efficiency as dimension reduction through analytical integration of parameters, but without sacrificing parallel computation and without the restriction to conjugate priors. We implement the new method in the Bayesian program BPP and apply it to genomic datasets from *Adansonia* baobab trees, *Anopheles* mosquitoes, and *Heliconius* butterflies. The new algorithms reduce the run-time of MCMC analyses by 3 to 8 fold and improve the mixing efficiency by up to 50 fold for representative empirical datasets.

Keywords: BPP, gene flow, introgression, migration, multispecies coalescent, MSC-I, MSC-M

Introduction

In population phylogenomics, genomic samples are analysed to infer phylogenetic relationships among species or populations accounting for local population genetic processes (such as genetic drift and selection) and potential gene flow. This is a challenging problem of fundamental importance in understanding the evolution of all organisms. Methods for inferring key parameters such as population split times, rates of migration, and effective population sizes under the multispecies coalescent [1, 2] rely heavily on Bayesian Markov chain Monte Carlo (MCMC) algorithms [3, 4]. Those algorithms generate a Markov chain of samples from the posterior distribution of parameters, combining information from the genomic sequence data and prior distributions.

Two significant challenges have slowed progress in scaling population phylogenomic methods to handle large genomes and complex patterns of gene flow between populations [5–7]. First, MCMC algorithms are difficult to parallelise; because each MCMC step can start only after the previous step finishes, the algorithm is intrinsically serial. Second, the massive size of genomic datasets and large number of highly correlated parameters makes it difficult to achieve good mixing in MCMC algorithms; poor mixing occurs when sampled parameter values in the Markov chain are highly correlated reducing the effective number of samples from the posterior distribution. Under the MSC model, gene trees (at potentially thousands of loci) are unobserved latent variables and are part of the state space for the Markov chain. They constitute a large highly complex space with both discrete structures (the gene tree topology) and continuous variables (the branch lengths or coalescent times), and place stringent constraints on the species tree and MSC model parameters.

The first challenge, parallelisation, can be addressed by exploiting conditional independence. In a Bayesian hierarchical model, conditional independence allows a subset of variables in the state space of the Markov chain to be updated in parallel when other variables are fixed. Most high-dimensional MCMC algorithms use multiple proposal steps, each updating a subset of parameters, while collectively the steps ensure that every dimension of the state space is updated in each MCMC iteration. While the MCMC iterations are sequential, proposals updating subsets of parameters within each MCMC iteration can be applied in parallel by relying on conditional independence.

In algorithms under the MSC model, gene trees at different loci are independent given the species tree and parameters and can thus be updated in parallel. Parallelising MCMC updates of gene trees is particularly important because these are computationally the most expensive proposals in the algorithm. Indeed under the MSC model (with and without gene flow), sequence data at different loci are independently and identically distributed (i.i.d.), so that the number of loci is the sample size in the inference problem [1, 2]. Among many factors that affect the information content in the data (such as the number of loci, the number of sequences sampled per species, the sequence length, and the mutation rate), the number of loci is often the most important [e.g., 8–10]. For example, to estimate the rates of gene flow reliably, thousands of loci are often needed and it is important to develop algorithms that can accommodate many loci even if for a relatively small phylogeny.

The second challenge, poor mixing in high dimensions, can be addressed by analytically integrating some parameters out of the model, reducing the dimension of the parameter space. Analytical integration of population size parameters and migration rates have been used to reduce dimensionality and improve mixing in population phylogenomic models [5, 11, 12]. However, this approach has two serious limitations. First, it is tractable only under conjugate priors, but conjugate priors may not be the most appropriate and may cause inference problems as well as MCMC mixing problems. Second, integrating out parameters using conjugate priors often destroys conditional independence. For example, gene trees for different loci are independent given the parameters on the species tree, but they are not independent when those shared parameters are integrated out.

In this paper we propose a solution to the integration versus parallelisation tradeoff that allows their simultaneous use in an MCMC analysis of a hierarchical Bayesian model reaping the benefits of both. The method, which we refer to as Virtual Dimension Reduction allowing Parallelisation (VDRoP), proposes certain parameters from their conditional distribution (achieving the same mixing efficiency as analytical integration) but allows the updates of gene trees to be executed in parallel. Here, efficiency is measured by the variance of an estimate from an MCMC sample relative to the variance based on an independent sample. We solve the second problem of the integration approach, the need to use a conjugate prior, by using approximations to the conditional distribution for non-conjugate priors. We show that the approximation improves with increasing data size. Thus, the VDRoP algorithm removes the limitations of both the integration and parallelisation strategies for improving MCMC efficiency.

We describe MCMC algorithms under the MSC models emphasizing the conditional-independence structure of the hierarchical model. We implement new algorithms that leverage the conditional independence and exploit the use of (approximate) conditional distributions of parameters to design MCMC moves that achieve the same efficiency as analytical integration. We develop new methods to generate posterior summaries for parameters which are integrated out in the MCMC algorithm or whose conditionals are tractable or can be approximated. A simple example involving a bivariate Gaussian target is presented to illustrate the new algorithms. We then analyse three genomic datasets to assess the performance of the new algorithms in real-world applications.

Results

The multispecies coalescent model has been extended to account for gene flow between species [2]. Gene flow is modelled as either a discrete introgression/hybridization event which occurred at a time point in the past in the MSC-introgression (MSC-I) model [13–15] or as a continuous migration process in the MSC-migration (MSC-M) model [5, 7, 16]. Both models are implemented in BPP [7, 15], and have been applied to data of more than 10,000 loci (albeit for a small number of species), providing a powerful framework for testing gene flow and estimating its rate from genomic data [e.g., 18–20]. Previous studies have demonstrated the correctness and efficiency of our implementations [7, 15]. Our focus here is on improvement of computational and mixing efficiency of MCMC algorithms. Theories and algorithms developed in this study are implemented in BPP under the MSC, MSC-I (fig. 1**a**) and MSC-M (fig. 1**b**) models [21], detailed in Supplementary Note 1. Here we

briefly describe our implementation of the MSC-M model, while noting the differences from the simpler MSC and MSC-I models.

The gene-tree density under the MSC-M model

We consider an MSC-M model for two species (fig. 1**b**) to introduce the parameters in the model and the calculation of the density for gene trees. The probability density of the gene tree at each locus under MSC-M is given by the backwards-in-time process of coalescent and migration [22–25] and has been described previously [e.g., 5, 12]. Here we follow eqs. 2&3 in ref. [7].

There are three types of parameters in the MSC-M model: species/population split times ($\boldsymbol{\tau}$), population sizes ($\boldsymbol{\theta}$), and the *mutation-scaled migration rates* ($\boldsymbol{\varpi}$) (fig. 1**b**). For example, the MSC-M model for two species of figure 1**b** involves six parameters, $\Theta = (\boldsymbol{\tau}, \boldsymbol{\theta}, \boldsymbol{\varpi}) = (\tau, \theta_A, \theta_B, \theta_{AB}, \varpi_{AB}, \varpi_{BA})$. In analysis of sequence data, we measure time by mutations so that one time unit is the expected time to accumulate one mutation per site. Then both τ and θ are measured in the expected number of mutations per site. The mutation-scaled migration rate is then $\varpi_{AB} = m_{AB}/\mu$, where μ is the mutation rate per site per generation, and m_{AB} is the expected proportion of immigrants in the recipient population B from the donor population A in each generation. We use the real-world view with time running forward to define parameters (m, ϖ).

Let the multi-locus sequence data be $X = (X^{(i)})$, where $X^{(i)}$ denotes the sequence alignment at locus i. Let $\boldsymbol{G} = (G^{(i)})$ be the gene trees, where $G^{(i)}$ includes the rooted tree, the coalescent times, and the migration history at the locus (including the number, directions and timings of migration events). Given the species tree and parameters, the gene trees are assumed to be independent, while different sites in the sequence at the same locus are assumed to share the same genealogy. These assumptions hold if there is free recombination between loci and no recombination within a locus. Simulation suggests that inference under the MSC such as estimation of the rate of gene flow is robust to moderate levels of within-locus recombination, with rates at about ten times the human rate [10, 26, 27].

The MCMC algorithm samples from the joint conditional distribution of the parameters and the gene trees.

$$p(\Theta, \boldsymbol{G}|\boldsymbol{X}) \propto p(\Theta)p(\boldsymbol{G}|\Theta)p(\boldsymbol{X}|\boldsymbol{G}), \quad (1)$$

where $p(\Theta)$ is the prior, $p(\boldsymbol{G}|\Theta)$ is the probability density of the gene trees given the parameters in the MSC-M model, and $p(\boldsymbol{X}|\boldsymbol{G})$ is the probability of the sequence data given the gene trees or the phylogenetic likelihood [28]. The hierarchical model is illustrated in figure 2.

When we trace the history of the sample backwards in time in any population j at locus i, two kinds of events may occur as competing Poisson processes. Coalescence between any pair of sequences occurs at the rate of $\frac{2}{\theta_j h_i}$, where h_i is the heredity (ploidy) scalar for locus i (e.g., 1 for a diploid autosomal locus, $\frac{3}{4}$ for an X locus, and $\frac{1}{4}$ for a Y or mitochondrial locus). Migration for each sequence in population j from population s occurs at the rate of ϖ_{sj} (here an $s \to j$ migration means that a sequence in population j is traced back to population s).

The gene-tree density, $p(\boldsymbol{G}|\Theta)$, is a product over populations and over loci. For each population j and locus i we break the time period into K_{ij} time segments (defined by species divergence, coalescent and migration events) during which the coalescent and migration rates are constant (fig. 1**b**). Segment k ($k = 1, \cdots, K_{ij}$) has time duration t_{ijk} and n_{ijk} lineages ancestral to the sample. Let $\mathbb{I}_{sjk}$ be an indicator which takes the

value 1 if migration from s to j is possible during time segment k (that is, if both species s and j exist during time segment k and are permitted to exchange migrants in the model) and 0 otherwise. Let c_{ji} be the number of coalescent events in population j at locus i and w_{sji} be the number of migration events from population s to population j at locus i. The contribution to the gene tree density from population j and locus i is equal to the product of the Poisson rates for events that have occurred and the probability of no events during the time duration of the population, given by the variable-rate Poisson process [see, e.g. 29, p.322].

$$\begin{aligned} p(\boldsymbol{G}|\Theta) &= \prod_j \prod_i \left[\left(\frac{2}{\theta_j h_i}\right)^{c_{ji}} \cdot \exp\left\{ -\sum_{k=1}^{K_{ij}} \binom{n_{ijk}}{2} \frac{2}{\theta_j h_i} t_{ijk} \right\} \right] \\ &\times \prod_j \prod_i \prod_s \left[\varpi_{sj}^{w_{sji}} \cdot \exp\left\{ -\sum_{k=1}^{K_{ij}} \mathbb{I}_{sjk} n_{ijk} t_{ijk} \varpi_{sj} \right\} \right] \\ &= \prod_j \prod_i \left(\frac{2}{\theta_j h_i}\right)^{c_{ji}} \exp\left\{ -C_{ji} \cdot \frac{2}{\theta_j h_i} \right\} \\ &\times \prod_j \prod_s \varpi_{sj}^{w_{sj}} \exp\{-W_{sj}\varpi_{sj}\}, \end{aligned} \tag{2}$$

where $C_{ji} = \sum_{k=1}^{K_{ij}} \binom{n_{ijk}}{2} t_{ijk}$ is the total coalescent waiting time in population j at locus i, summed over time segments and lineage pairs, $W_{sj} = \sum_i \sum_{k=1}^{K_{ij}} \mathbb{I}_{sjk} n_{ijk} t_{ijk}$ is the total migration waiting time for $s \longrightarrow j$ migration in population j, summed over time segments, lineages and loci, and $w_{sj} = \sum_i w_{sji}$ is the total number of $s \longrightarrow j$ migration events over all loci.

In ref. [7] we used the *population migration rate*, $M_{AB} = m_{AB} N_B = \varpi_{AB} N_B \mu = \varpi_{AB} \theta_B / 4$, which is the expected number of $A \longrightarrow B$ migrants per generation. Here we reparametrize the model using the mutation-scaled migration rate (ϖ) as this leads to simpler algorithms. The change of parametrization from M to ϖ means a change to the prior on migration rates while the likelihood model stays the same. Also there is an error in eq. 2 of ref. [7]: the mutation-scaled migration rate from species s to j should be $\frac{4M_{sj}}{\theta_j}$ instead of $\frac{4M_{sj}}{\theta_j h_i}$. The error affects analyses in which loci with different heredity scalars are included in the dataset (e.g., a mixture of autosomal and X-linked loci), and has no effect when all loci have the same heredity scalar (for example when all loci are autosomal).

Let G_j be the part of the gene trees in species j over all loci, with $G = (G_j)$. We rewrite the gene-tree density of eq. 2 as

$$p(G|\Theta) = \prod_j p(G_j|\Theta) = \prod_j p_j^{(c)} p_j^{(m)}, \tag{3}$$

where

$$\begin{aligned} p_j^{(c)} &= \prod_i \left[\left(\frac{2}{\theta_j h_i}\right)^{c_{ji}} \exp\left\{ -C_{ji} \cdot \frac{2}{\theta_j h_i} \right\} \right], \\ p_j^{(m)} &= \prod_s \varpi_{sj}^{w_{sj}} \exp\{-W_{sj}\varpi_{sj}\}. \end{aligned} \tag{4}$$

This is the same factorisation as in equation 2 of ref. [12] [see also 11]. The contribution to the gene-tree density from each population j consists of two components. The coalescent component, $p_j^{(c)}$, is the coalescent rates multiplied by the probability of no coalescent events over the total time period in the population when coalescence was possible, while the migration component, $p_j^{(m)}$, is the migration rates multiplied by the probability of no migration events over the total time period available for migration in the population.

Inverse-gamma priors on θ are conjugate

The gene-tree density (eq. 3) has the form of an inverse-gamma density for any θ_j for population j. We can use inverse-gamma priors to integrate out θs to reduce the space of the Markov chain [6, 12, 21, 30]. Previously Hey and Nielsen [11] integrated out θs by using uniform priors with user-specified

upper bounds; see ref. [10] for a discussion of issues arising from the use of uniform priors in the IMa3 program [5]. Note that from the inverse-gamma density,

$$\tilde{g}(y|\alpha,\beta) = \frac{\beta^\alpha}{\Gamma(\alpha)} y^{-\alpha-1}\, \mathrm{e}^{-\beta/y}, \qquad (5)$$

we have

$$\int_0^\infty y^{-\alpha-1}\, \mathrm{e}^{-\beta/y}\, \mathrm{d}y = \frac{\Gamma(\alpha)}{\beta^\alpha}. \qquad (6)$$

Let $\theta_j \sim \mathrm{IG}(\alpha_c, \beta_c)$. If we integrate out θ_j from the gene-tree density, the coalescent-component of the gene tree density becomes

$$\begin{aligned} \bar{p}_j^{(c)} &= \int \tilde{g}(\theta_j|\alpha_c,\beta_c) p_j^{(c)}\, \mathrm{d}\theta_j \\ &= \left[\prod_i \left(\frac{2}{h_i}\right)^{c_{ji}}\right] \cdot \frac{\beta_c^{\alpha_c}}{\Gamma(\alpha_c)} \cdot \frac{\Gamma(\alpha_c + c_j)}{(\beta_c + C_j^*)^{\alpha_c + c_j}}, \end{aligned} \qquad (7)$$

where

$$\begin{aligned} c_j &= \sum_i c_{ji}, \\ C_j^* &= \sum_i \tfrac{2}{h_i} C_{ji}. \end{aligned} \qquad (8)$$

In other words, when all population size parameters are integrated out, the gene-tree density becomes $p(G|\tau,\varpi) = \prod_j \bar{p}_j^{(c)} p_j^{(m)}$ instead of eq. 3.

However, when population size parameters are integrated out, gene trees at different loci are not independent, and $\bar{p}_j^{(c)}$ for population j is not a product over loci (i). Thus MCMC steps updating gene-tree node ages or gene-tree topologies (via subtree pruning regrafting or SPR) [1] cannot be parallelised. Instead we use the conditional distributions of θs to propose new values (i.e., the Gibbs sampler) and modify MCMC steps to make use of the knowledge of the conditional. Given the species tree (S) and the gene trees (G), the conditional distribution of θ_j is inverse-gamma $\mathrm{IG}(\alpha_j^{(c)*}, \beta_j^{(c)*})$, with

$$\alpha_j^{(c)*} = \alpha_c + c_j, \quad \beta_j^{(c)*} = \beta_c + C_j^*. \qquad (9)$$

The Gibbs sampler can thus be used to update θ_j. The algorithm works also under models which assume that several populations share the same θ; one simply counts the coalescent events (c_j) and calculates the total coalescent waiting time (C_j^*) by summing over all populations that share the same θ.

Gamma priors for θ are not conjugate but approximate conditionals can be used to achieve similar performance to integration

The inverse-gamma prior (on θs) is a heavy-tailed distribution, which sometimes makes unreasonably large θ values appear during the MCMC. Heterozygosity of $\theta > 10\%$ is rare in extant species of plants and animals and is expected to be rare in extinct ancestral species as well. Thus the inverse-gamma prior may be a poor prior and may adversely impact inference under the model or cause MCMC mixing problems. The light-tailed gamma distribution appears to be a better choice. However, the gamma prior is not conjugate, and the conditional distribution $p(\theta_j|X,\tau,G)$ is intractable, so that the Gibbs sampler is impossible.

We have developed gamma and inverse-gamma approximations to the conditional and implemented Metropolised Gibbs algorithms to update θs in BPP. We describe the approach in Supplementary Note 2 in the context of Bayesian inference under a Poisson process in which a gamma prior is assigned to the expected waiting time for the event. We evaluated the approximation under different scenarios (e.g., with the prior being consistent or conflicting with the data) and found that the approximation is good even

with only 2 or 10 events (fig. S1). Both the gamma and inverse-gamma approximations converge to the correct conditional when the amount of data (i.e., the total number of coalescent events over all loci in that population) approaches infinity, so that the approximation is more accurate in larger datasets with more loci and more sequences.

We also note an interesting difference between the gamma and inverse-gamma approximations when the Markov chain is far out in the tail (fig. S2). In such a case, the light tail of the gamma may lead to rejection of the proposal, causing convergence issues (fig. S2), and the inverse-gamma should be preferable. The situation is similar to rejection sampling or importance sampling, in which the sampling density should have a heavier tail than the target [e.g., 31, p.60–63, p.122–3]. When the chain has reached stationarity, both the gamma and inverse-gamma approximations are effective.

Gamma priors on migration rates (ϖ_{sj}) are conjugate

The gene-tree density of eq. 3 has the form of the gamma density for any migration rate ϖ_{sj} from population s to population j [11, 12]. Note that from the gamma density

$$g(y|\alpha,\beta) = \frac{\beta^\alpha}{\Gamma(\alpha)} y^{\alpha-1} \mathrm{e}^{-\beta y}, \quad y > 0, \quad (10)$$

we have

$$\int_0^\infty y^{\alpha-1} \mathrm{e}^{-\beta y} \,\mathrm{d}y = \frac{\Gamma(\alpha)}{\beta^\alpha}. \quad (11)$$

Let $\varpi_{sj} \sim \mathrm{G}(\alpha_m, \beta_m)$. If we integrate out ϖ_{sj} from the gene-tree density (eq. 3), the migration component of the gene tree density in eq. 3 becomes

$$\begin{aligned}\bar{p}_j^{(m)} &= \int g(\varpi_{sj}|\alpha_m,\beta_m) p_j^{(m)} \,\mathrm{d}\varpi_{sj} \\ &= \prod_s \left[\frac{\beta_m^{\alpha_m}}{\Gamma(\alpha_m)} \cdot \frac{\Gamma(\alpha_m + w_{sj})}{(\beta_m + W_{sj})^{\alpha_m + w_{sj}}} \right].\end{aligned} \quad (12)$$

In other words, when all migration rates are integrated out, the gene-tree density becomes $p(G|\boldsymbol{\tau},\boldsymbol{\theta}) = \prod_j p_j^{(c)} \bar{p}_j^{(m)}$ instead of eq. 3.

Furthermore, if we assign the gamma prior $\varpi_{sj} \sim \mathrm{G}(\alpha_m, \beta_m)$, the conditional distribution of ϖ_{sj} will be gamma $\mathrm{G}(\alpha_{sj}^{(m)*}, \beta_{sj}^{(m)*})$, with

$$\alpha_{sj}^{(m)*} = \alpha_m + w_{sj}, \quad \beta_{sj}^{(m)*} = \beta_m + W_{sj}. \quad (13)$$

Thus one can sample ϖ_{sj} from its conditional distribution, given the species split times (τs) and the gene trees.

Integrating out both $\boldsymbol{\theta}$ and $\boldsymbol{\varpi}$

Under inverse-gamma priors on population sizes, $\theta \sim \mathrm{IG}(\alpha_c, \beta_c)$, and gamma priors on migration rates, $\varpi \sim \mathrm{G}(\alpha_m, \beta_m)$, we see from eq. 2 that $\boldsymbol{\theta}$ and $\boldsymbol{\varpi}$ are conditionally independent given the coalescent and migration histories in the gene trees. We can integrate out both $\boldsymbol{\varpi}$ and $\boldsymbol{\theta}$ from eq. 3, to give the gene-tree density

$$p(G|\boldsymbol{\tau}) = \prod_j \bar{p}_j^{(c)} \bar{p}_j^{(m)}, \quad (14)$$

with $\bar{p}_j^{(c)}$ and $\bar{p}_j^{(m)}$ given in eqs. 7 & 12.

To allow parallelisation, we keep $\boldsymbol{\theta}$ and $\boldsymbol{\varpi}$ in the MCMC and instead sample from their conditional distributions given the species split times $\boldsymbol{\tau}$ and the gene trees, $\theta_j \sim \mathrm{IG}(\alpha_j^*, \beta_j^*)$ and $\varpi_{sj} \sim \mathrm{G}(\alpha_{sj}^*, \beta_{sj}^*)$. In the case of gamma priors on θs, either approximate gamma or inverse-gamma conditionals are used instead.

Improved MCMC algorithms

While our theory above concerning the treatment of the population size parameters (θs) is developed under the MSC-M model, it applies to all MSC models including the simple MSC model with no gene flow and the discrete MSC-I model [15]. We note that in the MSC-I model the beta distribution is a conjugate prior for the introgression probability (φ), and φ can be treated similarly to the migration rate ϖ in the MSC-M model.

Modifications to the MCMC steps in BPP [1, 7, 15, 32] are described in Supplementary Note 1 (fig. 3). Here we provide an overview.

Step 1 updates coalescent times and migration times on gene trees. If there are L loci, with n sequences per locus, there will be $L \cdot (n-1)$ coalescent times, while the number of migration times on gene trees is arbitrary depending on the migration rates. Step 2 applies gene-tree SPR or simulation to update gene tree topologies (and coalescent and migration times), looping over loci and for each locus over branches on the gene tree. With thousands of loci and dozens of sequences per locus, the gene trees constitute a complex space of huge dimensions. A major effort has been to parallelise those two steps which update gene trees at all loci.

Step 4 updates species split times (τs) using the rubber-band proposal under the MSC, MSC-I, and MSC-M models, with coordinated deterministic changes to coalescent times and migration times on gene trees [1, 7]. Step 5 is a scaling or mixing move that multiplies all species split times on the species tree ($\boldsymbol{\tau}$) and coalescent and migration times on all gene trees by the same scale factor. In both steps, we sample $\boldsymbol{\theta}$, $\boldsymbol{\varphi}$ (in MSC-I) and $\boldsymbol{\varpi}$ (in MSC-M) from their (approximate) conditionals given the newly proposed species split times and the gene trees.

Other steps are used to update $\boldsymbol{\theta}$ and $\boldsymbol{\varpi}$ under MSC-M or $\boldsymbol{\theta}$ and $\boldsymbol{\varphi}$ under MSC-I.

Simple example for a bivariate Gaussian target to illustrate the algorithms

In this section, we define the mixing efficiency (E) for an MCMC algorithm and use a simple problem involving a bivariate Gaussian target to illustrate the new algorithms implemented in BPP and tested in the Results section using three empirical datasets (table S1).

Let $(\phi_1, \phi_2, \cdots, \phi_N)$ be a sample for any parameter ϕ from the MCMC, and we estimate the expectation of any function of the parameter, $h(\phi)$, by the sample mean $\tilde{h} = \frac{1}{N}\sum_i h(\phi_i)$. The mixing efficiency of the MCMC is then defined as the variance ratio

$$E = \lim_{N\to\infty} \frac{\mathbb{V}(\bar{h})}{\mathbb{V}(\tilde{h})} = \frac{1}{1 + 2(\rho_1 + \rho_2 + \cdots)}, \tag{15}$$

where $\tilde{h}$ is the estimate based on the MCMC sample and $\bar{h}$ is the estimate based on an independent sample of the same size, and where $\rho_k = \mathrm{corr}(h(\phi_i), h(\phi_{i+k}))$ is the lag-k autocorrelation [e.g., 33–35]. Thus efficiency is measured by the ratio of the variance for an estimator based on an independent sample to the variance based on the MCMC sample of the same size. The effective sample size (ESS) is given as $N \cdot E$. We also use the lag-1 autocorrelation ρ_1, which is simpler to compute.

Besides the mixing efficiency E (eq. 15), the computational cost for each MCMC iteration may differ among algorithms. However unless stated otherwise explicitly, the algorithms tested in this study have similar running time per iteration. Also the mixing efficiency may differ widely for different parameters in the model, and as a rule of thumb, the MCMC should be run long enough so that ESS exceeds 100 or 200 for every parameter. All runs in this study exceed this expectation.

Consider MCMC algorithms sampling from a two-dimensional target,

$$\pi(x, y) = \pi(x)\pi(y|x), \tag{16}$$

to estimate μ_x; that is, the function $h(\mu_x) = \mu_x$ in eq. 15. We use the bivariate Gaussian (N_2) target so that $\pi(x) = \phi(x)$, $\pi(y|x) = \phi(y; \rho x, 1 - \rho^2)$, and

$$\pi(x, y) = \pi(x)\pi(y|x) = \phi_2\left(\begin{bmatrix} x \\ y \end{bmatrix}; \begin{bmatrix} 0 \\ 0 \end{bmatrix}, \begin{bmatrix} 1 & \rho \\ \rho & 1 \end{bmatrix}\right), \tag{17}$$

where $\phi(x; \mu, \sigma^2)$ is the probability density function (PDF) for $N(\mu, \sigma^2)$, with $\phi(x; 0, 1) \equiv \phi(x)$, and where $\phi_k(\boldsymbol{x}; \boldsymbol{\mu}, \boldsymbol{\Sigma})$ is the k-variate Gaussian density with the mean vector $\boldsymbol{\mu}$ and variance-covariance matrix $\boldsymbol{\Sigma}$. Note that for the algorithms discussed here, mixing efficiency and acceptance rate depend on the correlation ρ but not the means or variances of the N_2 target (fig. 4).

In algorithm A1, we integrate out y, and sample from the marginal distribution $\pi(x) = \phi(x)$ using a Gaussian sliding-window proposal, with $q(x'|x) = \phi(x'; x, s^2)$. It is well-known that the optimal step length is $s = 2.5$, at which efficiency (for estimating the target mean μ_x) is $E = 0.228$ (with the acceptance rate 43%) [35, 36].

In algorithm A2, we generate x' from the sliding window, $q(x'|x) = \phi(x'; x, 2.5^2)$ as in A1, and also sample y' from its conditional, with $q(y'|x', x) = \pi(y'|x')$, with the $(x, y) \to (x', y')$ proposal accepted with probability

$$\alpha = \min\left\{1, \frac{q(x|x')q(y|x, x')}{q(x'|x)q(y'|x', x)} \times \frac{\pi(x', y')}{\pi(x, y)}\right\} \tag{18}$$

$$= \min\left\{1, \frac{q(x|x')}{q(x'|x)} \times \frac{\pi(x')}{\pi(x)}\right\}. \tag{19}$$

Efficiency E (for estimating μ_x) and the acceptance rate are then independent of ρ (fig. 4, A1 or A2), and indeed algorithms A1 and A2 are equivalent as far as sampling of x is concerned.

Algorithms A1 and A2 here are analogous to algorithms A1 and A2 of table S1 implemented in BPP; the variable x represents τs and gene trees in the MSC models while y represents $\boldsymbol{\theta}$, $\boldsymbol{\varpi}$ or $\boldsymbol{\varphi}$. As the inverse-gamma priors are conjugate for θs, we can integrate $\boldsymbol{\theta}$ out (in A1) or sample them while updating the species split times and gene trees (in A2), and both algorithms have equivalent efficiency.

When the conditional $\pi(y|x)$ is intractable, we may use a proposal density that is close to the conditional, $q(y'|x', x) = q(y'|x') \approx \pi(y'|x')$. Then eq. 18 holds but eq. 19 does not. This mimics algorithms A4 and A5 under MSC-I, and B4 and B5 under MSC-M (table S1), in which the gamma priors on θs are not conjugate, and the conditionals for θ are approximated using the gamma (A4 and B4) or inverse-gamma (A5 and B5) distributions. If the approximation is good, the algorithm may achieve nearly the same efficiency and acceptance rate as if the exact conditionals were used.

In algorithm 3, we use two 1-D Gaussian sliding-window proposals, updating x and y separately. With the step lengths adjusted to have the acceptance rate of 40%, the efficiency (for estimating μ_x) decreases with the increase of correlation, dropping to 0.024 at $\rho = 0.9$ and 0.012 at $\rho = 0.95$ (fig. 4). This mimics algorithms A3, A6 and B6 of table S1, algorithms in earlier versions of BPP.

More efficient estimation of posterior means for parameters by use of conditionals

Population sizes (θs), introgression probabilities (φ), and migration rates (ϖ) are of biological interest, so it is useful to generate their posterior distributions, in particular, posterior means and highest probability density (HPD) credibility intervals (CIs), even

when they are integrated out in the MCMC to improve efficiency. Furthermore even if those parameters are sampled during the MCMC (as in the algorithms of this paper), their conditionals contain more information than the sample values and can be used to produce more precise posterior summaries than the MCMC sample.

Suppose we have an MCMC algorithm operating on X with the variable Y integrated out. Let $(x_1, x_2, \cdots, x_N)$ be a sample from this algorithm. Given each x_i, $i = 1, \cdots, N$, the conditional distribution of Y is available. Thus one can sample from the conditional distribution $y_i \sim p(Y|x_i)$ and use the sample mean $\tilde{y} = \frac{1}{N}\sum y_i$ to estimate μ_y. However, we can also calculate the conditional means, $u_i = E(Y|x_i)$, and use the sample mean $\tilde{u} = \frac{1}{N}\sum_{i=1}^{N} u_i$ to estimate μ_y, with a reduced variance relative to $\tilde{y}$. The efficiency of $\tilde{u}$ relative to $\tilde{y}$ (or the ratio of the variances for estimating μ_y) is

$$E_{\tilde{u},\tilde{y}} \equiv \lim_{N\to\infty} \frac{\mathbb{V}(\tilde{y})}{\mathbb{V}(\tilde{u})} = \frac{1}{1-\left(1-\frac{1}{c}\right)E_{\tilde{y}}} > 1, \tag{20}$$

where $E_{\tilde{y}}$ is the efficiency of $\tilde{y}$ relative to an independent sample, and where

$$c = \frac{\sigma_y^2}{\sigma_u^2} > 1 \tag{21}$$

is the variance ratio with $\sigma_y^2 = \mathbb{E}(Y-\mu_y)^2$ and $\sigma_u^2 = \int (u(X)-\mu_y)^2 p(x)\,\mathrm{d}x$. The theory is developed in Supplementary Note 3. Note that here we are treating μ_y as the expectation of a deterministic function of X: $\mu_y = \mathbb{E}_X(u)$ where $u = u(X) = \mathbb{E}(Y|X)$ is a function of X.

In the statistics literature, the conditional expectation is commonly used to improve an estimator, a technique known as Rao-Blackwellisation [37]. For an estimate based on an independent sample the conditional expectation always leads to reduced variance, but this may not be true for a dependent sample [38, 39]. Here our use of the conditional, $u_i = \mathbb{E}(Y|x_i)$, is guaranteed to reduce the variance in the estimate, with $\mathbb{V}(\tilde{u}) < \mathbb{V}(\tilde{y})$. See Supplementary Note 3. Furthermore, the improvement in mixing efficiency requires virtually no extra computational cost after the MCMC sample (x_i) has been generated.

The improvement of $\tilde{u}$ over $\tilde{y}$, measured by $E_{\tilde{u},\tilde{y}}$, is a monotonically increasing function of c and $E_{\tilde{y}}$ (eq. 20). Here c is determined by the inference problem or the posterior distribution, and it is an interesting question how c is influenced by the model, the parameters and features of the data (such as the number of loci, the number of sequences per locus, &c.). In contrast, $E_{\tilde{y}}$ is affected by the MCMC algorithm operating on X. In this regard, *the more efficient $\tilde{y}$ already is, the greater will be the improvement achieved by switching to $\tilde{u}$* (eq. 20).

Consider for instance two algorithms for sampling from $X \sim \mathrm{G}(10, 100)$ and $Y|X \sim \mathrm{IG}(2+10, 0.02+x)$. We have $c = \sigma_y^2/\sigma_u^2 = 2.54$. In the first algorithm, we use a sliding-window on $\log(x)$ to update x, and y is sampled from $p(y|x)$. The estimated efficiency for $\tilde{y}$ is $E_{\tilde{y}} = 0.43$, while $\tilde{u}$ based on the sample of $u = \frac{0.02+x}{11}$ gave $E_{\tilde{u}} = 0.58$, with an improvement of $E_{\tilde{u},\tilde{y}} = 1.35$ folds (eq. S21). In the second algorithm we use the mirror move to update $\log(x)$ [40], with y sampled from $p(y|x)$. Then $E_{\tilde{y}} = 1.13$, while the efficiency for the u sample is $E_{\tilde{u}} = 3.58$, with an improvement of $E_{\tilde{u},\tilde{y}} = 3.59$ folds (eq. S21). See Supplementary Note 3 for details.

The theory also works for a function of the variables, such as $\mathbb{E}(Y_1Y_2)$, where both Y_1 and Y_2 may be integrated out in the MCMC. This is useful for generating the posterior summaries for the population migration rate $M_{AB} = \varpi_{AB}\theta_B/4$ in the MSC-M model, when both ϖ_{AB} and θ_B are integrated out or sampled from their conditionals.

Besides the posterior mean, $\mu_y = \mathbb{E}(Y)$ or $\mathbb{E}(Y_1Y_2)$, the whole posterior distribution, or the probability density function (PDF) for Y can be recovered from the conditionals sampled throughout the MCMC, $p(Y|x_i)$. We also generated the 95% HPD CI as well as the 95% equal-tail CI for Y. The detailed analyses and proofs are presented in Supplementary Note 3 (fig. S3, table S2).

Three datasets for evaluating the mixing efficiency of the new algorithms

We used three real datasets with different features to examine the mixing efficiency of our new algorithms. The first dataset consists of 344 target-enrichment loci from eight baobab species in the genus *Adansonia* (fig. 5**a**). Previous analyses of genetic data found considerable uncertainty in the species phylogeny [41, 42]. We inferred the species phylogeny under the MSC model using BPP [43, 44]. The maximum *a posteriori* (MAP) tree (fig. 5**a**) has a posterior of ~ 100%, possibly because the full likelihood method uses the information in the data more efficiently than summary methods used in previous studies. This species tree differs from those inferred in ref. [41] using subsampled data and summary methods such as ASTRAL, and is similar to the maximum likelihood tree from plastid data [41, fig. 6]. The phylogeny, with the monophyly of the six Malagasy species and strong support for the two Malagasy sections (Longitubae and Brevitubae), appears to simplify the interpretation of the evolution of morphological characters such as flower colours and pollination [45]. We introduced potential gene-flow events onto the species phylogeny based on previous tests of ref. [41] using the D-statistic [46] and SNAQ [47]. Only one of the two events of figure 5**a** was supported in our analysis. Below we run MSC-I and MSC-M models with the $x \to y$ gene flow (fig. 5**a**).

The second dataset consists of 4,133 noncoding loci from chromosome 2L1 from six species of African mosquitoes in the *Anopheles gambiae* species complex: *A. gambiae* (G), *A. coluzzii* (C), *A. arabiensis* (A), *A. melas* (L), *A. merus* (R), and *A. quadriannulatus* (Q) [48, 49]. The species tree is shown in figure 6**a**&**b**, with two gene-flow events (A → GC and R → Q) accounted for using the MSC-I and MSC-M models.

The third dataset consists of 5341 noncoding loci from chromosome 1 for three species of *Heliconius* butterflies: *Heliconius hecale* (H), *H. cydno* (C), and *H. Melpomene* (M) [17, 50]. The species tree is shown in figure7**a**&**b**, with C→M gene flow modeled using either the discrete MSC-I model or the continuous MSC-M model.

Mixing efficiency of different algorithms applied to empirical data

We analysed the three empirical datasets using six MCMC algorithms under the MSC-I model (A1–A6) and three algorithms under the MSC-M model (B4–B6) (table S1). The mixing efficiency E (and ρ_1) were used to compare algorithms. The results are summarised in figures 5–7. Computational tests were conducted on a Lenovo ThinkSystem SR850 server with 4x Intel Xeon Gold 6154 18C processors. Apart from algorithm A1, all other algorithms involve similar amount of computation, so the improvements are mostly due to reduced variance or improved mixing efficiency. We discuss the results for the baobabs (fig. 5) in detail, and the results for the *Anopheles* and *Heliconius* (figs. 6&7) follow the same format.

First in algorithm A1 (I-IG-int), we integrate out $\boldsymbol{\theta}$ using the conjugate inverse-gamma priors, while algorithm A2 (I-IG-gIG) under the same MSC-I model and same

inverse-gamma priors uses a Gibbs sampler to sample θ from the inverse-gamma conditional. Here the key 'I-IG-int' means the MSC-I model, with inverse-gamma priors on θs, and with θs integrated out. In general, our key to algorithms is in the format 'model-prior-algorithm', with model to be 'I' for MSC-I and 'M' for MSC-M; with prior for θs to be 'IG' for inverse-gamma and 'G' for gamma (beta for φ in MSC-I and gamma for ϖ in MSC-M are always used); and with algorithm to be 'slide' for sliding-windows, 'it' for integrating out θs, 'gIG' for (Metropolised) Gibbs based on inverse-gamma conditional for θs, 'gG' for Metropolised Gibbs based on the gamma conditional for θs.

Figure 5**b** shows the posterior means and 95% HPD CIs for parameters produced by the nine algorithms. The posterior should be the same under the same model and prior, that is, among algorithms A1 (I-IG-int), A2 (I-IG-gIG), and A3 (I-IG-slide) for the MSC-I model under the inverse-gamma priors on θs; among algorithms A4 (I-G-gG), A5 (I-G-gIG), and A6 (I-G-slide) under the MSC-I model with gamma priors on θs; and among B4 (M-G-gG), B5 (M-G-gIG), B6 (M-G-slide) under the MSC-M model with gamma priors on θs. The gamma and inverse-gamma priors on θs produced virtually identical posteriors (fig. 5**b**), as the dataset is large. Also species split times &c. are very similar between the two models (MSC-I and MSC-M). In particular, the posterior mean (and 95% HPD CI) for the rate of gene flow is φ_{xy} = 0.428 (0.352, 0.503) under the MSC-I model, and ϖ_{xy} = 20.0 (4.0, 37.0) under MSC-M (fig. 5**b**, table S3).

Next we examine the mixing efficiency of different algorithms. Algorithm A2 (I-IG-gIG) is expected to have the same mixing efficiency as A1 (I-IG-int). This is reflected by the efficiency E and autocorrelation ρ_1 (eq. 15) for the two algorithms falling on the diagonal line for all parameters in figure 5**c** (A2 = A1). Note that good mixing is indicated by high E and low ρ_1, and there is large variation in mixing efficiency among parameters. While $\boldsymbol{\theta}$ are integrated out in A1, in A2 we keep $\boldsymbol{\theta}$ and $\boldsymbol{\varpi}$ as part of the Markov-chain state and sample them from their conditional distributions given the species tree and the gene trees. This allows parallelisation of MCMC proposals that update the gene trees at all loci. Thus A2 has a large computational advantage on multi-core computers over A1, achieving a 8.2-fold reduction in running time (fig. 8).

The same idea works even under the gamma priors on θs, for which analytical integration over θs (as in algorithm A1) is impossible. By using approximate gamma or inverse-gamma conditionals, algorithms A4 (I-G-gG) and A5 (I-G-gIG) under the gamma priors on $\boldsymbol{\theta}$ achieve similar efficiency as A2 (I-IG-gIG) for the inverse-gamma priors (fig. 5**c**, A4 ≈ A5, A5 ≈ A2). See also similar results in figure 5**c**, B4 ≈ B5 under the MSC-M model.

In algorithm A6 (I-G-slide) for the MSC-I model with gamma priors on $\boldsymbol{\theta}$, parameters $\boldsymbol{\theta}$ are updated using sliding windows, and no change of $\boldsymbol{\theta}$ is made in the rubber-band or mixing (scaler) steps which updates the species split times with coordinated changes to the gene-tree node ages. This is effectively the algorithm in the previous version of BPP (version 4.7). We expect the new algorithm A5 (I-G-gIG) to have higher mixing efficiency than the old algorithm A6 (I-G-slide), and this is indeed the case (fig. 5**c**, A5 ≫ A6). Similarly algorithm B5 (M-G-gIG) is much better than B6 (M-G-slide) under the MSC-M model (fig. 5**c**, B5 ≫ B6). Note that the gamma priors on $\boldsymbol{\theta}$ are not conjugate, so it is impossible to integrate out $\boldsymbol{\theta}$ analytically. Our use of the approximate conditionals thus make it possible to achieve the impossible.

Similarly in analyses of the *Anopheles* and *Heliconius* datasets, all algorithms produced the same posterior under the same

model and priors (figs. 6&7), and the gamma and inverse-gamma priors on $\boldsymbol{\theta}$ had minimal impact on the posteriors of parameters. For *Anopheles* (fig. 6**d**, table S4) the posterior means and 95% HPD CIs for the rate of gene flow are $\varphi_{A\to GC} = 0.963\ (0.950, 0.975)$ and $\varphi_{R\to Q} = 0.016\ (0.008, 0.024)$ under the MSC-I model, and $\varpi_{R\to Q} = 0.137\ (0.007, 0.291)$ and $\varpi_{A\to GC} = 203\ (191, 215)$ under MSC-M. For *Heliconius* the posterior means and 95% HPD CIs are $\varphi_{cm} = 0.465\ (0.430, 0.500)$ under MSC-I and $\varpi = 2.25\ (0.05, 5.38)$ under MSC-M (fig. 7**d**, table S5).

Similarly in the *Anopheles* and *Heliconius* datasets our expectations for the mixing efficiency of different algorithms are also confirmed (figs. 6&7). The mixing efficiency of algorithms under the MSC-I model is expected to be in the order: A2 (I-IG-gIG) = A1 (I-IG-int), A2 (I-IG-gIG) $\gg$ A3 (I-IG-slide), A2 (I-IG-gIG) $\approx$ A4 (I-G-gG) $\approx$ A5 (I-G-gIG), A5 (I-G-gIG) $\gg$ A6 (I-G-slide). For example A2 (I-IG-gIG) has the same mixing efficiency (E and ρ_1) as A1 (I-IG-int) (figs. 6**d**&7**d**), but a large computational advantage over A1 on multithreaded computers, achieving a reduction in running time of 3.1× and 3.4× in the two datasets (table 1). Similarly under the MSC-M model, with gamma prior on θs, the mixing efficiency of the algorithms is expected to be in the order B4 (M-G-gG) $\approx$ B5 (M-G-gIG), and B5 (M-G-gIG) $\gg$ B6 (M-G-slide). These expectations are confirmed in all datasets (fig. 6**d**, and fig. 7**d**). Relative to version 4.7 of BPP [7, 15] which uses the sliding window moves to update θs and M (i.e., algorithms A3 and A6 for MSC-I and B6 for MSC-M), the improvement in mixing efficiency of the new algorithms (A2, A4 and A5 for MSC-I and B4 and B5 for MSC-M) is up to 4-fold for baobabs (table S3), 51-fold for *Anopheles* (table S4) and 19-fold for *Heliconius* (table S5). Note that the improvement in mixing efficiency varies among parameters.

Discussion

In this study we introduce three improvements to current MCMC algorithms in Bayesian phylogenomics. First, we implement new moves that are equivalent to reducing the dimension of the MCMC algorithm by integration but still allow parallelisation of computations and inference of all parameters. It is known that conjugate priors may be used to integrate out certain parameters (such as θ under the MSC, MSC-I, and MSC-M models, φ under the MSC-I models, and ϖ in the MSC-M models) to reduce the dimension of the state space for the Markov chain, leading to improved mixing of the MCMC algorithm. However analytical integration destroys the conditional independence of the hierarchical model, so that most of the computational tasks involved in the MCMC algorithm cannot be parallelised. We make use of the fact that if we keep those parameters in the MCMC but sample them from their conditional distributions, the MCMC algorithm remains equivalent in terms of the parameters that are not integrated out, including species split times τs, gene trees and coalescent times in the MSC models. Keeping those parameters in the MCMC (algorithm A2) allows the MCMC moves that update the gene trees and coalescent times to be executed in parallel. In the three empirical datasets tested, this strategy produces a 3 to 8 fold reduction of running time (fig. 8, table 1).

Second, we develop methods for dimension reduction by virtual integration with non-conjugate priors. This approach relies on approximate conditionals when non-conjugate priors are used for some parameters (θs). Note that conjugate priors, while computationally convenient, may not be the most appropriate biologically. Under MSC models, the heavy-tailed inverse gamma prior on θ has been observed to cause the chain to visit implausibly large θ values, causing both mixing and inference problems. In such cases,

the gamma prior has been noted to ameliorate the problem. However the gamma prior is not conjugate, and an algorithm that integrates out θs is beyond reach. By using gamma and inverse-gamma approximations to the conditional, we have achieved mixing efficiency previously possible only if the parameters were integrated out. The approximations are more accurate in larger datasets (with more loci and/or more sequences leading to more coalescent events). Most phylogenomic datasets should be large enough to achieve good approximation. In analyses of empirical datasets such as those tested in this study, the Metropolised gibbs move updating θs was noted to have acceptance rate close to 100%, indicating the effectiveness of the approximate conditional (note that acceptance rate should be exactly 1 if the true conditional is used). Furthermore, algorithms A2 under the inverse-gamma prior (which uses the true conditional) and algorithms A4 and A5 under the gamma prior (which use approximate conditionals) achieved the same mixing efficiency. We envisage that similar approximate conditionals may be useful in other expensive MCMC algorithms.

Third, we develop a theory to use the conditionals or approximate conditionals to estimate the posterior distributions for parameters $\boldsymbol{\theta}$, $\boldsymbol{\varphi}$, and $\boldsymbol{\varpi}$, including posterior means, PDFs and CDFs. This allows the estimation of the posterior for parameters that are integrated out in the algorithm. It also improves on the efficiency of inferences when variables are not integrated out but conditionals or approximations of conditionals are available. We quantify the relative improvement (or reduction in variance) achieved by using the conditional mean $u_i = \mathbb{E}(y_i|X)$ rather than sampled value (y_i).

Together these algorithmic advances lead to improvements in mixing efficiency of MCMC for inference using genomic data under the MSC models with and without gene flow. Currently BPP is the only program that appears to implement the MSC-M model correctly; see figures 4 & 5 in ref. [7] on our tests of IMa3 [5] and MIGRATE [51], respectively, and table S1 and Supplementary Note 1 in the same paper for our tests of G-PhoCS [16]. Bpp is also the only program that can handle datasets with thousands of loci under either the MSC-I or MSC-M models, datasets that are large and informative enough to allow meaningful inference of gene flow [e.g. 7, 15, 17, 49]. However, only within-model algorithms for estimating parameters such as the rates of gene flow are implemented in BPP, while cross-model MCMC moves for searching in the space of gene-flow models are lacking. For species triplets, a model-comparison approach to selecting models of gene flow (including ghost introgression, inflow, and outflow) is developed in BPP with the Savage-Dickey density ratio used to calculate Bayes factors [52, 53], but there is a need to develop model-comparison strategies or cross-model MCMC algorithms that work on larger phylogenies. Algorithms for searching in the space of gene-flow models are available in the MSC-M framework in IMa3 [5] and in the MSC-I framework in PhyloNet [13] and *BEAST [14]; however those programs can handle only small datasets with $\leq$ 100 loci.

Another advantage of the BPP program is that it implements both the continuous MSC-M and discrete MSC-I models [7, 15], making it straightforward to apply both models to the same genomic datasets to test their goodness of fit and to learn about the mode of gene flow. We conclude that algorithmic improvements such as those described in this study have made BPP a practically useful tool for analysing genomic data to test for gene flow and to estimate its rate.

We note that the algorithms developed in this paper may be useful for other similar Bayesian implementations under the MSC

models [5, 6]. For example, in the saturated migration model of ref. [5], population size parameters and migration rates (θs and ϖs) are integrated out. In particular, migration rates are specified in the saturated model even for migration events that are not supported by the data. Our theory (Supplementary Note 3) allows estimation of the posterior distributions of those important parameters, including Bayesian tests of migration under the MSC-M model. Similarly, the virtual dimension reduction algorithms and approximate conditionals may have general applicability in other major applications of hierarchical models with conditional independence.

Methods

We use three empirical datasets to examine the mixing efficiency of MCMC algorithms under the MSC-I and MSC-M models.

Baobab species in the genus *Adansonia*

We analyse the dataset of 344 target-enrichment loci (i.e., putative single-copy nuclear genes) from eight baobab species in the genus *Adansonia* generated in ref. [41]. We removed the two outgroups *Bombax ceiba* and *Pseudobombax croizatii* as they are very distant [54] and kept the close outgroup *Scleronema micrantha*. Multiple samples for the same species are kept in the same alignment at each locus without subsampling. There are 28–38 sequences per locus, and the sequence length ranges over 759–9574 sites (median 1679). We note that Wan et al. [42] generated genome assemblies for the eight species of baobabs, but only one sample is sequenced per species so that the data may not be ideal for inferring gene flow. There is considerable uncertainty in the species phylogeny and gene flow. In particular, the relationships among *A. digitata*, *A. gregorii*, and the clade of six Malagasy species is highly uncertain, as is the position of *A. rubrostipa* (fig. 5**a**) [41, 42].

We conducted species tree estimation under the MSC model with no gene flow (the A01 analysis, 55) using different starting trees. The inferred species tree is in figure 5**a**. We then included gene-flow events on the phylogeny based on previous analyses [41]. Only one gene-flow event in figure 5**a** ($x \to y$) was supported in our analysis. Thus we fit MSC-I and MSC-M models with the $x \to y$ gene flow (fig. 5**a**).

We used the gamma prior $\theta \sim \mathrm{G}(2, 200)$ or inverse-gamma prior IG(3, 0.02) with mean 0.01. The age of the root was assigned the gamma prior $\tau_r \sim \mathrm{G}(2, 100)$ with mean 0.02. Under the MSC-I, we assigned the prior $\varphi \sim \mathrm{U}(0, 1)$ on the introgression probability. We used the `linked-msci` option so that a branch before and after an introgression event are assigned the same θ. Under the MSC-M model, we assigned the gamma prior $\varpi \sim \mathrm{G}(2, 1)$ with mean 2. The MSC-I model was implemented using MCMC algorithms A1–A6, and the MSC-M model was implemented using algorithms B4–B6 (table S1). We used a burn-in of 4×10^4 iterations and took 5×10^5 samples, sampling every iteration.

African mosquitoes in the *Anopheles gambiae* group

We analyse a dataset of noncoding loci from chromosome arm L1 from six species of African mosquitoes in the *Anopheles gambiae* species complex: *A. gambiae* (G), *A. coluzzii* (C), *A. arabiensis* (A), *A. melas* (L), *A. merus* (R), and *A. quadriannulatus* (Q) (fig. 6**a**&**b**). The data consist of 4133 noncoding loci, with 12 sequences per locus (two sequences per species). The data consist of "haploid consensus sequences", with heterozygotes resolved into the majority nucleotide [56]. The data were originally published and analysed in ref. [48], and recompiled by Thawornwattana et al. [49],

who analysed coding and noncoding data from all chromosome arms (see also 7, 15). Here we used noncoding loci from L1 only.

We fitted the MSC-I and MSC-M models (fig. 6**a**&**b**), using the algorithms of table S1 with similar settings to the above. We used the gamma priors $\theta \sim \mathrm{G}(2, 100)$ or $\mathrm{IG}(3, 0.04)$ with prior mean 0.02, and $\tau_r \sim \mathrm{G}(2, 20)$ with prior mean 0.1. Under the MSC-I, we assigned the prior $\varphi \sim \mathrm{U}(0, 1)$ on the introgression probability. Under the MSC-M model, we assigned the gamma prior $\varpi \sim \mathrm{G}(2, 1)$. We used a burn-in of 4×10^4 iterations and took 2×10^5 samples, sampling every iteration. The results under both the MSC-I and MSC-M models match those published in table 1 of ref. [7]. Here we focus on the mixing or computational efficiency of the algorithms.

Heliconius butterflies

We analysed a dataset of 5341 noncoding loci from chromosome 1 from three species of *Heliconius* butterflies: *Heliconius hecale* (H), *H. cydno* (C), and *H. Melpomene* (M) (fig. 7**a**&**b**), from ref. [17]. The species tree is (H, (C, M)), with gene flow between C and M. Thawornwattana et al. ([17], tables 2&3) detected ongoing gene flow from C → M but rejected gene flow in the opposite direction. See tables S4–S6 and figure S8 in ref. [17] for analyses of both coding and noncoding data from all 21 chromosomes. Here we use the MSC-I and MSC-M models to account for the C → M gene flow to analyse the noncoding data from chromosome 1 only.

We use the gamma prior for the age of the root, $\tau_r \sim \mathrm{G}(2, 200)$. We used the priors $\mathrm{G}(2, 200)$ and $\mathrm{IG}(3, 0.04)$ for θ, both with prior mean 0.01. For the rate of gene flow, we assigned the uniform prior $\varphi \sim \mathrm{U}(0, 1)$ under MSC-I and the gamma prior $\varpi \sim \mathrm{G}(2, 1)$ under MSC-M. We used a burn-in of 2×10^4 iterations and took 5×10^5 samples, sampling every iteration.

Data availability

The three empirical datasets analysed in this study and BPP scripts are available at https://figshare.com/articles/dataset/bppMigration-algorithms-data_tgz/30032800. Accession codes for sequences in the baobabs dataset are in table S6 [41]. For the *Anopheles* dataset, 12 genomes (two per species) were from the assemblies of refs. [57] and [48]; GenBank accession numbers and NCBI BioSample numbers are in table S7 [7, 49]. The BioSample accession numbers for the three *Heliconius* genomes are SAMN11398304, SAMN11398291, and SAMN11398301 [50] (see table S1 in ref. [17]).

Code Availability

The VDRoP algorithms are implemented in the BPP software, available at https://github.com/bpp/bpp.

References

[1] Rannala, B. & Yang, Z. Bayes estimation of species divergence times and ancestral population sizes using DNA sequences from multiple loci. *Genetics* **164**, 1645–1656 (2003).

[2] Jiao, X., Flouri, T. & Yang, Z. Multispecies coalescent and its applications to infer species phylogenies and cross-species gene flow. *Nat. Sci. Rev.* (2021).

[3] Metropolis, N., Rosenbluth, A., Rosenbluth, M., Teller, A. & Teller, E. Equations of state calculations by fast computing machines. *J. Chem. Physi.* **21**, 1087–1092 (1953).

[4] Hastings, W. Monte Carlo sampling methods using Markov chains and their

application. *Biometrika* **57**, 97–109 (1970).

[5] Hey, J. *et al.* Phylogeny estimation by integration over isolation with migration models. *Mol. Biol. Evol.* **35**, 2805–2818 (2018).

[6] Douglas, J., Jimenez-Silva, C. L. & Bouckaert, R. StarBeast3: Adaptive parallelised Bayesian inference under the multispecies coalescent. *Syst. Biol.* (2022).

[7] Flouri, T., Jiao, X., Huang, J., Rannala, B. & Yang, Z. Efficient Bayesian inference under the multispecies coalescent with migration. *Proc. Nat. Acad. Sci. U.S.A.* **120**, e2310708120 (2023).

[8] Huang, J., Flouri, T. & Yang, Z. A simulation study to examine the information content in phylogenomic datasets under the multispecies coalescent model. *Mol. Biol. Evol.* **37**, 3211–3224 (2020).

[9] Thawornwattana, Y., Flouris, T., Mallet, J. & Yang, Z. Inference of gene flow between species from genomic data when the mode, direction and lineages are misspecified. *Mol. Biol. Evol.* **42**, 1–18 (2025).

[10] Thawornwattana, Y., Rannala, B. & Yang, Z. On the robustness of Bayesian inference of gene flow to intragenic recombination and natural selection. *Mol. Biol. Evol.* **43**, 1–22 https://doi.org/10.1093/molbev/msaf327 (2026).

[11] Hey, J. & Nielsen, R. Integration within the Felsenstein equation for improved Markov chain Monte Carlo methods in population genetics. *Proc Natl Acad Sci U.S.A.* **104**, 2785–2790 (2007).

[12] Jones, G. R. Divergence estimation in the presence of incomplete lineage sorting and migration. *Syst. Biol.* **68**, 19–31 (2019).

[13] Wen, D. & Nakhleh, L. Coestimating reticulate phylogenies and gene trees from multilocus sequence data. *Syst. Biol.* **67**, 439–457 (2018).

[14] Zhang, C., Ogilvie, H. A., Drummond, A. J. & Stadler, T. Bayesian inference of species networks from multilocus sequence data. *Mol. Biol. Evol.* **35**, 504–517 (2018).

[15] Flouri, T., Jiao, X., Rannala, B. & Yang, Z. A Bayesian implementation of the multispecies coalescent model with introgression for phylogenomic analysis. *Mol. Biol. Evol.* **37**, 1211–1223 (2020).

[16] Gronau, I., Hubisz, M. J., Gulko, B., Danko, C. G. & Siepel, A. Bayesian inference of ancient human demography from individual genome sequences. *Nature Genet.* **43**, 1031–1034 (2011).

[17] Thawornwattana, Y., Huang, J., Flouris, T., Mallet, J. & Yang, Z. Inferring the direction of introgression using genomic sequence data. *Mol. Biol. Evol.* **40**, msad178 (2023).

[18] Thawornwattana, Y., Seixas, F. A., Mallet, J. & Yang, Z. Full-likelihood genomic analysis clarifies a complex history of species divergence and introgression: the example of the erato-sara group of *Heliconius* butterflies. *Syst. Biol.* **71**, 1159–1177 (2022).

[19] Thawornwattana, Y., Seixas, F. A., Yang, Z. & Mallet, J. Major patterns in the introgression history of *Heliconius* butterflies. *eLife* **12**, RP90656,

DOI:10.7554/eLife.90656 (2023).

[20] Santos, S. H. D. *et al.* Massive inter-species introgression overwhelms phylogenomic relationships among jaguar, lion, and leopard. *Syst. Biol.* 10.1093/sysbio/syaf021 (2025).

[21] Flouri, T., Jiao, X., Rannala, B. & Yang, Z. Species tree inference with BPP using genomic sequences and the multispecies coalescent. *Mol. Biol. Evol.* **35**, 2585–2593 (2018).

[22] Notohara, M. The coalescent and the genealogical process in geographically structured populations. *J. Math. Biol.* **29**, 59–75 (1990).

[23] Beerli, P. & Felsenstein, J. Maximum-likelihood estimation of migration rates and effective population numbers in two populations using a coalescent approach. *Genetics* **152**, 763–773 (1999).

[24] Beerli, P. & Felsenstein, J. Maximum likelihood estimation of a migration matrix and effective population sizes in *n* subpopulations by using a coalescent approach. *Proc. Natl. Acad. Sci. U.S.A.* **98**, 4563–4568 (2001).

[25] Nielsen, R. & Wakeley, J. Distinguishing migration from isolation: a Markov chain Monte Carlo approach. *Genetics* **158**, 885–896 (2001).

[26] Zhu, T., Flouri, T. & Yang, Z. A simulation study to examine the impact of recombination on phylogenomic inferences under the multispecies coalescent model. *Mol. Ecol.* **31**, 2814–2829 (2022).

[27] Yan, Z., Ogilvie, H. A. & Nakhleh, L. Comparing inference under the multispecies coalescent with and without recombination. *Mol. Phylogenet. Evol.* **181**, 107724 (2023).

[28] Felsenstein, J. Evolutionary trees from DNA sequences: a maximum likelihood approach. *J. Mol. Evol.* **17**, 368–376 (1981).

[29] Yang, Z. *Molecular Evolution: A Statistical Approach* (Oxford University Press, Oxford, England, 2014).

[30] Jones, G. Algorithmic improvements to species delimitation and phylogeny estimation under the multispecies coalescent. *J. Math. Biol.* **74**, 447—-467 (2017).

[31] Ripley, B. *Stochastic Simulation* (Wiley, New York, 1987).

[32] Rannala, B. & Yang, Z. Improved reversible jump algorithms for Bayesian species delimitation. *Genetics* **194**, 245–253 (2013).

[33] Peskun, P. Optimum Monte-Carlo sampling using Markov chains. *Biometrika* **60**, 607–612 (1973).

[34] Green, P. J. & Han, X. L. in *Metropolis methods, Gaussian proposals and antithetic variables* (eds Barone, P., Frigessi, A. & Piccioni, M.) *Stochastic Models, Statistical Methods and Algorithms in Image Analysis* 142–164 (Springer, New York, 1992).

[35] Gelman, A., Roberts, G. & Gilks, W. in *Efficient Metropolis jumping rules* (eds Bernardo, J., Berger, J., Dawid, A. & Smith, A.) *Bayesian Statistics 5*, Vol. 5 599–607 (Oxford University Press, Oxford, 1996).

[36] Yang, Z. & Rodríguez, C. E. Searching for efficient Markov chain Monte Carlo proposal kernels. *Proc. Natl. Acad. Sci. U.S.A.* **110**, 19307–19312 (2013).

[37] Robert, C. P. & Roberts, G. Rao–blackwellisation in the Markov chain Monte Carlo era. *Int. Statist. Rev.* **89**, 237–249 (2021).

[38] Liu, J. S., Wong, W. H. & Kong, A. Covariance structure of the Gibbs sampler with applications to the comparisons of estimators and augmentation schemes. *Biometrika* **81**, 27–40 (1994).

[39] Geyer, C. J. Conditioning in Markov chain Monte Carlo. *J. Comput. Graph. Statist.* **4**, 148–154 (1995).

[40] Thawornwattana, Y., Dalquen, D. & Yang, Z. Designing simple and efficient Markov chain Monte Carlo proposal kernels. *Bayesian Analysis* **13**, 1033–1059 (2018).

[41] Karimi, N. *et al.* Reticulate evolution helps explain apparent homoplasy in floral biology and pollination in Baobabs (*Adansonia; Bombacoideae; Malvaceae*). *Syst. Biol.* **69**, 462–478 (2020).

[42] Wan, J. N. *et al.* The rise of baobab trees in Madagascar. *Nature* **629**, 1091–1099 (2024).

[43] Yang, Z. & Rannala, B. Unguided species delimitation using DNA sequence data from multiple loci. *Mol. Biol. Evol.* **31**, 3125–3135 (2014).

[44] Rannala, B. & Yang, Z. Efficient bayesian species tree inference under the multispecies coalescent. *Syst. Biol.* **66**, 823–842 (2017).

[45] Baum, D. A. The comparative pollination and floral biology of baobabs (*Adansonia*-Bombacaceae). *Ann. Missouri Bot. Gard.* **82**, 322–348 (1995).

[46] Patterson, N. *et al.* Ancient admixture in human history. *Genetics* **192**, 1065–1093 (2012).

[47] Solis-Lemus, C. & Ane, C. Inferring phylogenetic networks with maximum pseudolikelihood under incomplete lineage sorting. *PLoS Genet.* **12**, e1005896 (2016).

[48] Fontaine, M. C. *et al.* Extensive introgression in a malaria vector species complex revealed by phylogenomics. *Science* **347**, 1258524 (2015).

[49] Thawornwattana, Y., Dalquen, D. & Yang, Z. Coalescent analysis of phylogenomic data confidently resolves the species relationships in the *Anopheles gambiae* species complex. *Mol. Biol. Evol.* **35**, 2512–2527 (2018).

[50] Edelman, N. B. *et al.* Genomic architecture and introgression shape a butterfly radiation. *Science* **366**, 594–599 (2019).

[51] Beerli, P. Comparison of bayesian and maximum-likelihood inference of population genetic parameters. *Bioinformatics* **22**, 341–345 (2006).

[52] Ji, J., Jackson, D. J., Leache, A. D. & Yang, Z. Power of Bayesian and heuristic tests to detect cross-species introgression with reference to gene flow in the *Tamias quadrivittatus* group of North American chipmunks. *Syst. Biol.* **72**, 446–465 (2023).

[53] Cheng, S., Flouris, T., Zhu, T. & Yang, Z. The impact of taxon sampling on

inference of gene flow by summary and Bayesian methods using genomic sequence data. *Syst. Biol.* (2026).

[54] Carvalho-Sobrinho, J. G. *et al.* Revisiting the phylogeny of Bombacoideae (Malvaceae): Novel relationships, morphologically cohesive clades, and a new tribal classification based on multilocus phylogenetic analyses. *Mol. Phylogenet. Evol.* **101**, 56–74 (2016).

[55] Yang, Z. The BPP program for species tree estimation and species delimitation. *Curr. Zool.* **61**, 854–865 (2015).

[56] Huang, J., Bennett, J., Flouri, T. & Yang, Z. Phase resolution of heterozygous sites in diploid genomes is important to phylogenomic analysis under the multispecies coalescent model. *Syst. Biol.* **71**, 334–352 (2022).

[57] Neafsey, D. E. *et al.* Mosquito genomics. highly evolvable malaria vectors: the genomes of 16 *Anopheles* mosquitoes. *Science* **347**, 1258522 (2015).

[58] Yang, Z. & Rannala, B. Bayesian species delimitation using multilocus sequence data. *Proc. Natl. Acad. Sci. U.S.A.* **107**, 9264–9269 (2010).

[59] Bull, V. *et al.* Polyphyly and gene flow between non-sibling *Heliconius* species. *BMC Biol.* **4**, 11 (2006).

Acknowledgments

We are grateful to Dr Nisa Karimi for preparing the HapHunt baobab dataset to include multiple samples per species. This study has been supported by Biotechnology and Biological Sciences Research Council grants (BB/T003502/1 and BB/X007553/1), a Natural Environment Research Council grant (NSFDEB-NERC NE/X002071/1) to Z.Y., an SNSF scientific visit grant (IZSEZ0_232434/1) to Z.Y., a Natural Science Foundation of China (NSFC) grant (12101295), a Guangdong Natural Science Foundation grant (2022A1515011767), and a Shenzhen Training Project of Excellent Scientific & Technological Talents grant (RCYX20221008093033012) to X.J., an NSFC grant (32200490) and a grant from Fundamental Research Funds for Beijing Municipal Universities (XJJS202523) to J.H., and an NIH Grant (GM123306) to B.R. The study has also been supported by a Swiss National Science Foundation scientific visit grant (IZSEZ0_232434/1) to Z.Y. and Prof. Maria Anisimova.

Author contributions

T.F., X.J., B.R. and Z.Y. designed and tested the `bpp` algorithms and prepared the documentation. J.H. analysed the empirical datasets. Z.Y. and B.R. supervised the research. All authors interpreted data and edited the manuscript.

Competing interests

The authors declare no competing interests.

Table 1: Running time of algorithms A1 and A2 applied to three datasets

Dataset	A1	A2	ratio
Adansonia	43h15m	5h18m	8.16
Anopheles	30h27m	9h57m	3.06
Heliconius	25h49m	7h35m	3.40

Note.— Each algorithm is run using 18 threads on the same CPU in a server while no other jobs are running. CPU usage on the threads for the *Adansonia* dataset is shown in figure 8.

Fig. 1: **MSC models of interspecific gene flow.** (**a**) MSC-introgression (MSC-I) model on a phylogeny for three species (A, B, C) with introgression/hybridisation between B and C at time $\tau_X = \tau_Y$. There are three types of parameters in the MSC-I model: species split and introgresion times (τ_R, τ_S, τ_Y), population sizes $(\theta_A, \theta_B, \theta_C, \theta_R, \theta_S, \theta_X, \theta_Y)$ and introgression probabilities (φ_X, φ_Y). (**b**) MSC-migration (MSC-M) model with migration between B and C. There are three types of parameters in the MSC-M model: species split times (τ_R, τ_S), population sizes $(\theta_A, \theta_B, \theta_C, \theta_R, \theta_S)$ and population migration rates $(\varpi_{BC}, \varpi_{CB})$.

Fig. 2: **Directed Acyclic Graph (DAG) representation of the MSC-M model.** The parameter vector is $\Theta = (\boldsymbol{\tau}, \boldsymbol{\theta}, \boldsymbol{\varpi})$, while the state of the Markov chain is $(\Theta, \boldsymbol{G}) = (\boldsymbol{\tau}, \boldsymbol{\theta}, \boldsymbol{\varpi}, \boldsymbol{G})$. Species split times $(\boldsymbol{\tau})$ are assigned the gamma-Dirichlet prior [58, eq. 2], population size parameters $(\boldsymbol{\theta})$ are assigned inverse-gamma or gamma priors, and mutation-scaled migration rates $(\boldsymbol{\varpi})$ are assigned gamma priors. The MSC-I model is similar, with introgression probabilities $(\boldsymbol{\varphi}$, with beta priors) replacing migration rates. The conditional independence of the model is exploited in the MCMC algorithms.

Fig. 3: **MCMC proposal algorithms implemented in BPP.** Species split times (step 4) are updated one after another, and within each calculations of the gene-tree density and likelihood for all loci are executed in parallel.

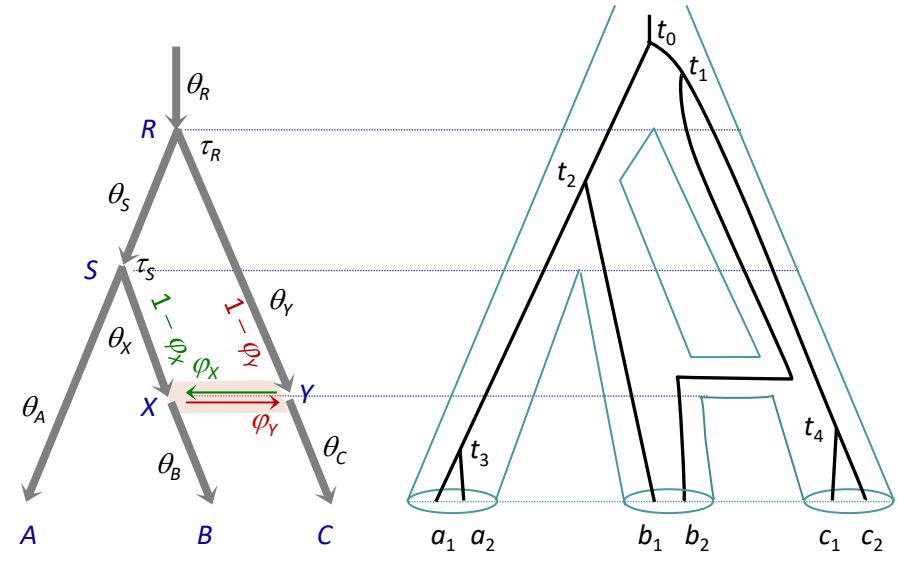
Fig. 4: **Efficiency for three MCMC algorithms in a bivariate Gaussian target.** Efficiency for estimating μ_x in the N_2 target of eq. 17 is measured by E of eq. 15. In algorithm A1, y is integrated out and a sliding-window move is used to sample x from its marginal. This has the same efficiency as algorithm A2, which samples y' from its conditional for the newly proposed x': $\pi(y'|x')$. Note that E does not depend on ρ for A1 and A2. In Algorithm A3, two one-dimensional sliding-window moves are used to update x and y, respectively. Efficiency is calculated for 1,000 values of ρ by running the algorithm over 10^7 MCMC iterations. A3 suffers from the correlation between x and y: $E = 0.228$ at $\rho = 0$ (equivalent to A1 and A2) but drops to 0.024 at $\rho = 0.9$ and 0.012 at $\rho = 0.95$.

Fig. 5: **Analysis of the *Adansonia* baobab dataset.** (**a**) Species tree for eight baobabs species in the genus *Adansonia* and the outgroup *Scleronema micranthaum*, inferred in BPP analysis of the *Adansonia* data under the MSC model assuming no gene flow. Picture of *Adansonia rubrostipa* from SW Madagascar courtesy of Dr Nisa Karimi. (**b**) Parameter estimates (posterior means and 95% HPD CIs) for nine algorithms of table S1: A1 (I-IG-int), A2 (I-IG-gIG), A3 (I-IG-slide), A4 (I-G-gG), A5 (I-G-gIG), and A6 (I-G-slide) under the MSC-I model; and B4 (M-G-gG), B5 (M-G-gIG), and B6 (M-G-slide) under the MSC-M model. Note that θs are integrated out in A1, so that there are eight estimates for θs (for A2–A6, B4–B6) but nine estimates for τs (for A1–A6, B4–B6). (**c**) Scatter-plots of mixing efficiency E and autocorrelation ρ_1 (eq. 15) for pairs of algorithms. The labels for the panels indicate our expectations, so that $y = x$ means that the algorithm on the y-axis is expected to have identical (equivalent) performance to the algorithm on the x-axis, '$y \approx x$' means similar performance, and '$y \gg x$' means y has much better performance than x (with larger E and smaller ρ_1). See table S1 for detailed descriptions and BPP settings for the algorithms.

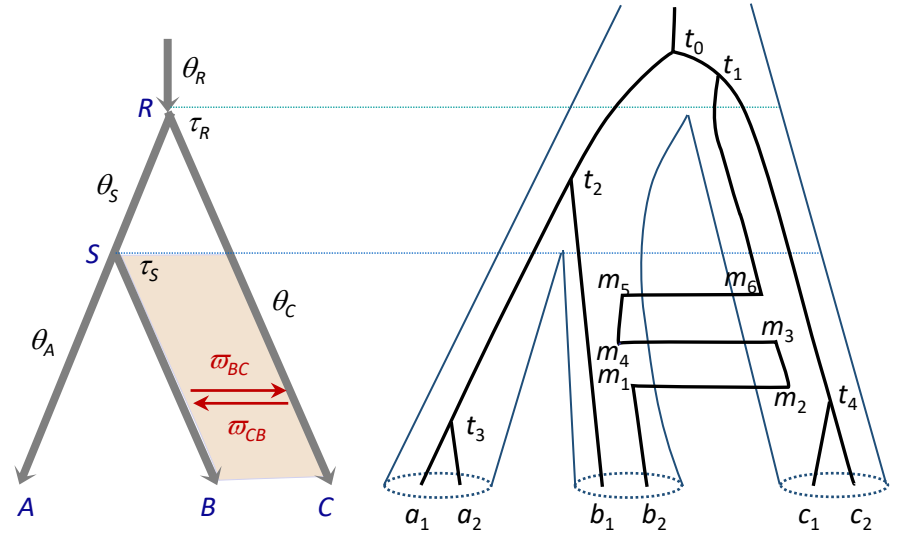
Fig. 6: **Analysis of the *Anopheles* mosquito dataset.** (**a**) MSC-I and (**b**) MSC-M models for six species of African mosquitoes in the *Anopheles gambiae* species complex: *A. gambiae* (G), *A. coluzzii* (C), *A. arabiensis* (A), *A. melas* (L), *A. merus* (R), and *A. quadriannulatus* (Q). Redrawn according to figure 6 in ref. [7]. (**c**) Posterior means and 95% HPD CIs for parameters and (**d**) mixing efficiency (E) and autocorrelation (ρ_1) for nine MCMC algorithms (table S1). See legend to figure 5.

Fig. 7: **Analysis of the *Heliconius* dataset.** (**a**) MSC-I and (**b**) MSC-M models for three *Heliconius* species: *H. hecale* (H), *H. cydno* (C), and *H. melpomene* (M), with potential gene flow between *H. cydno* and *H. melpomene*. Models with the $C \rightarrow M$ gene flow are used to analyse genomic sequence data. Picture of *Heliconius melpomene melpomene* from French Guiana [59] courtesy of Dr James Mallet. (**c**) Posterior means and 95% HPD CIs for parameters and (**d**) mixing efficiency (E) and autocorrelation (ρ_1) for nine MCMC algorithms (table S1). See legend to figure 5.

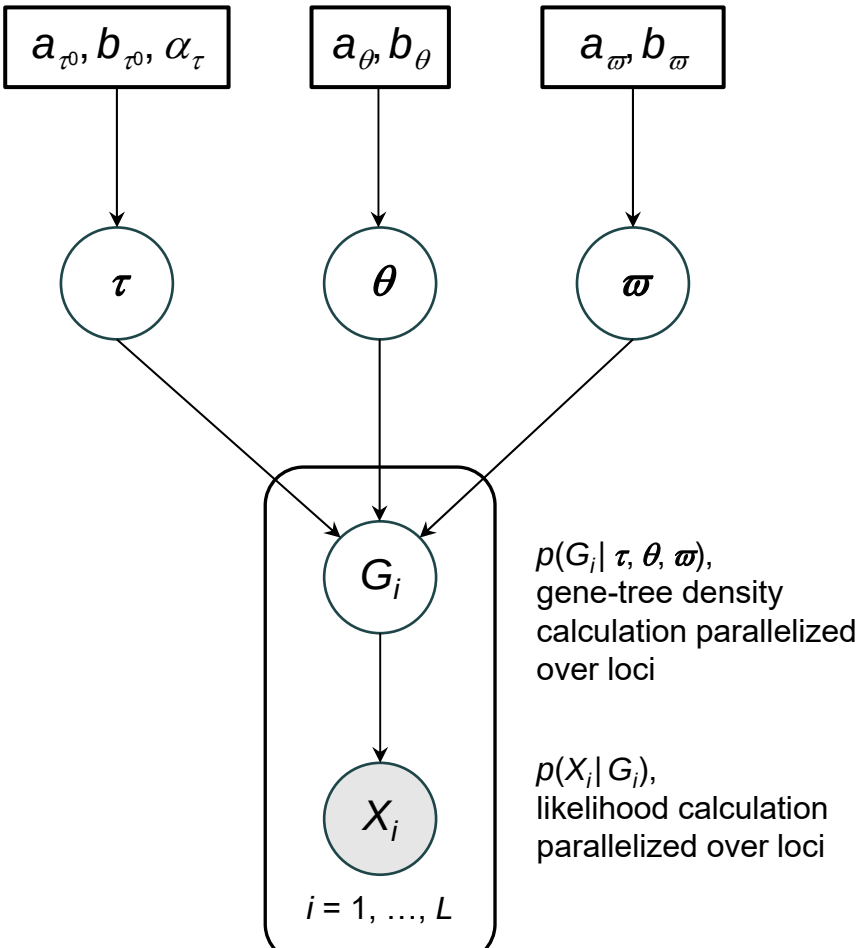
Fig. 8: Screenshot for the `htop` command showing the CPU load for algorithms A1 (integrating out θs) and A2 (sampling θs from their conditionals) applied to the *Adansonia* dataset under the MSC-I model (fig. 5**a**). The two algorithms were run on two CPUs in the same server, each using 18 threads, taking 5×10^5 samples after a burn-in of 5×10^4. With algorithm A1, all MCMC steps including the gene-tree node age and gene-tree SPR proposals are conducted serially (although likelihood calculation is parallelised by loci), so that the main thread has CPU usage close to 100% while the worker threads spend most time waiting with low CPU usage of $\leq 1\%$. With algorithm A2, the gene-tree node age and gene-tree SPR proposals are conducted in parallel so that the worker threads achieve high CPU usage of $\sim 75\%$. Running time was 43h15m for A1 and 5h18m for A2, with an 8.2-fold speedup.

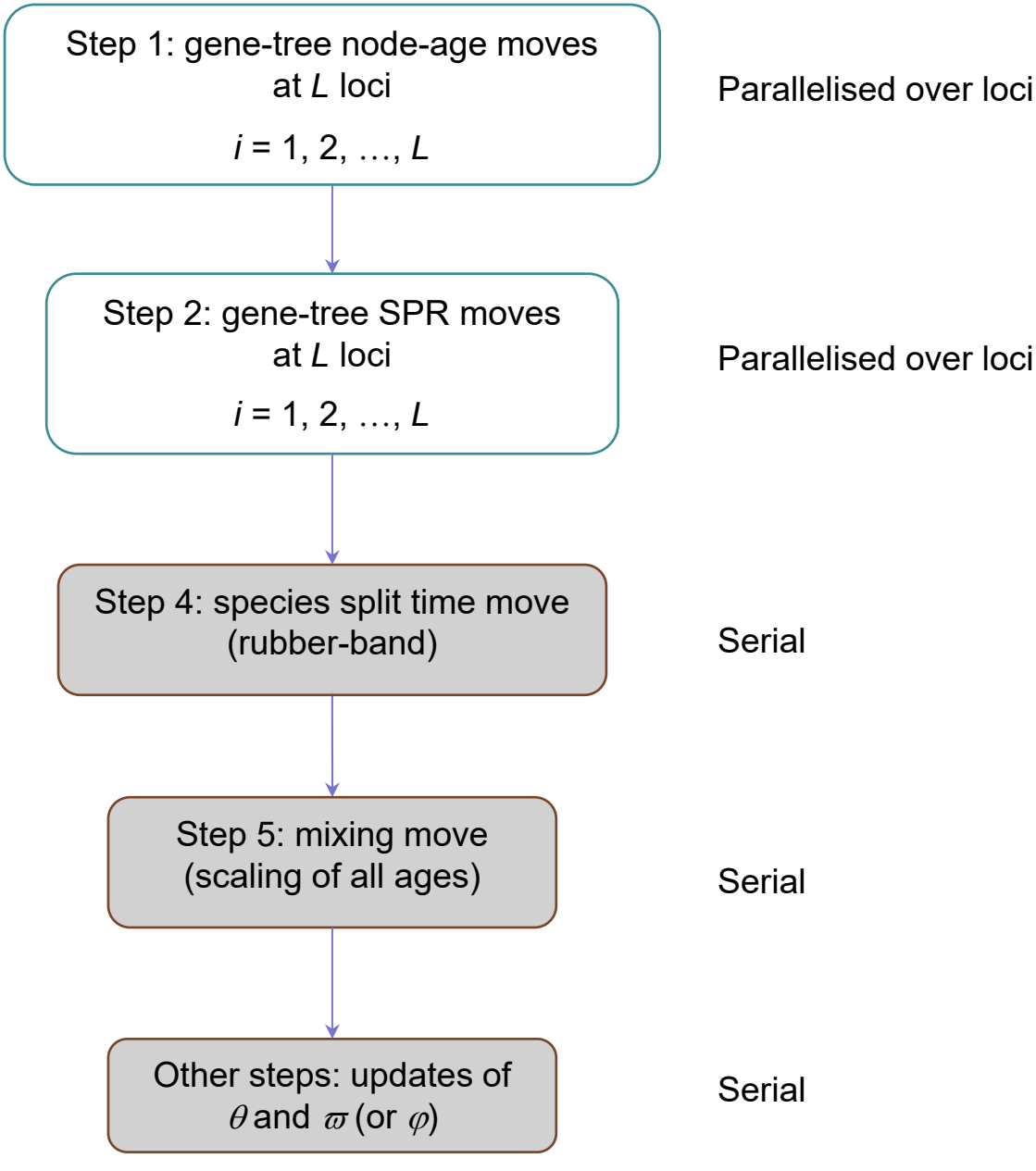


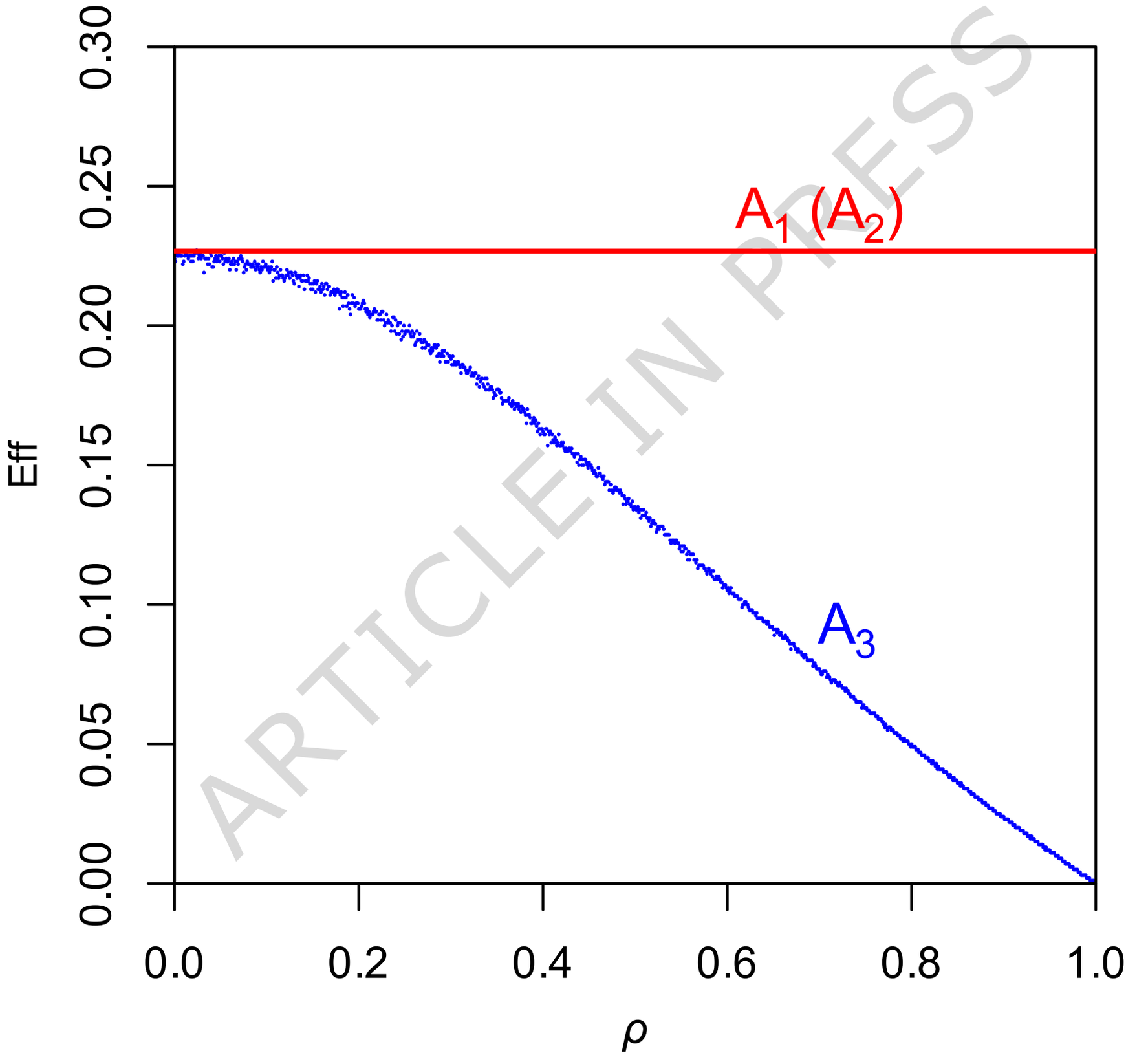
(**a**) An MSC-I model with a gene tree

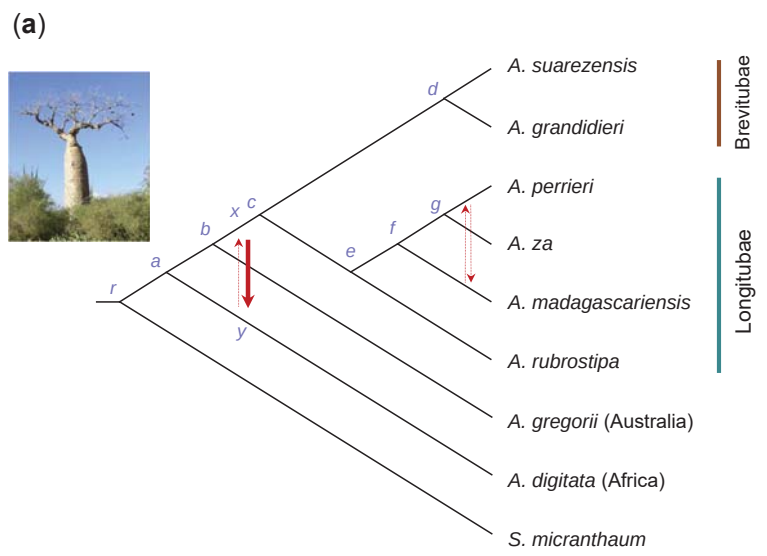


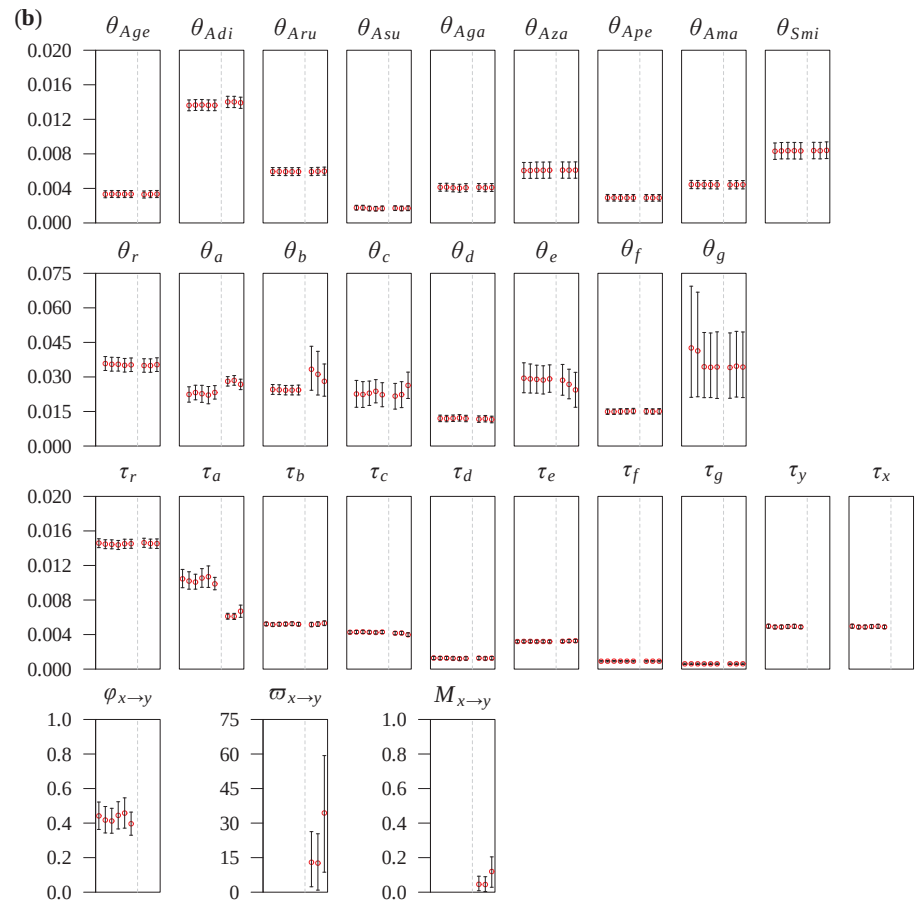
(**b**) An MSC-M (or IM) model with a gene tree

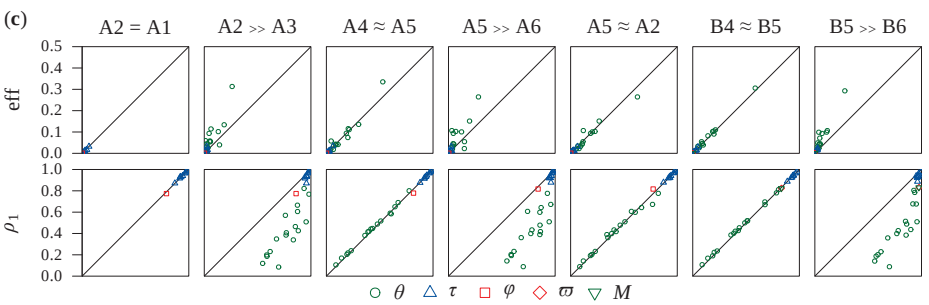

$a_{\tau 0}, b_{\tau 0}, \alpha_{\tau}$
a_{θ}, b_{θ}
a_{ϖ}, b_{ϖ}
τ
θ
ϖ
G_i
X_i
$i = 1, \ldots, L$
$p(G_i | \tau, \theta, \varpi)$, gene-tree density calculation parallelized over loci
$p(X_i | G_i)$, likelihood calculation parallelized over loci

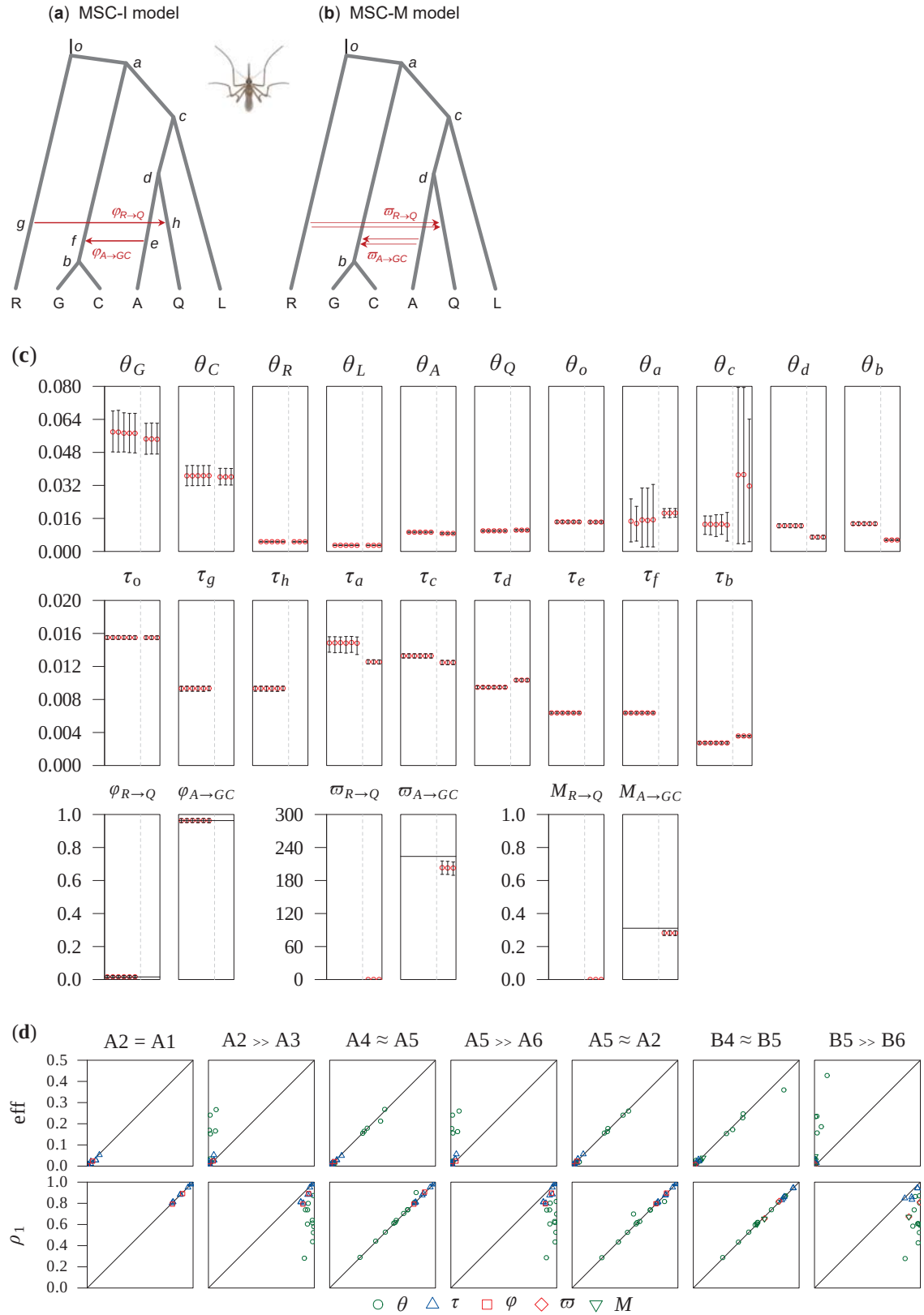

Step 1: gene-tree node-age moves at L loci
i = 1, 2, …, L
Parallelised over loci
Step 2: gene-tree SPR moves at L loci
i = 1, 2, …, L
Parallelised over loci
Step 4: species split time move (rubber-band)
Serial
Step 5: mixing move (scaling of all ages)
Serial
Other steps: updates of θ and ϖ (or φ)
Serial

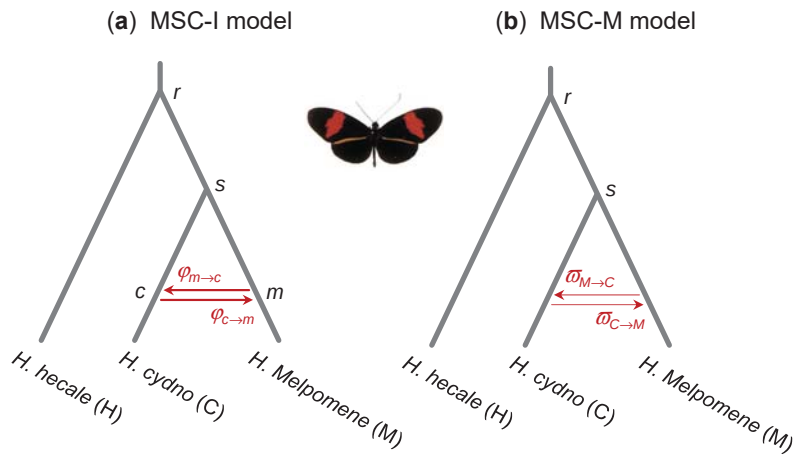


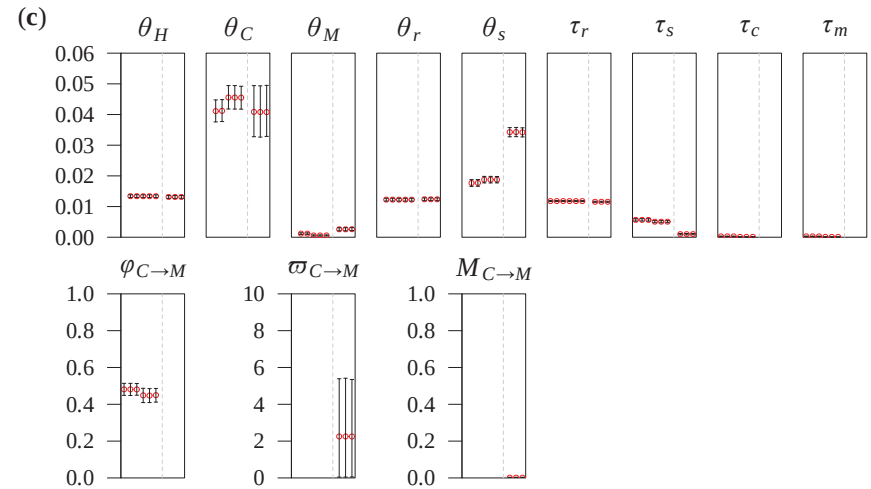

(a)
A. suarezensis
A. grandidieri
A. perrieri
A. za
A. madagascariensis
A. rubrostipa
A. gregorii (Australia)
A. digitata (Africa)
S. micranthaum
Brevitubae
Longitubae
r
a
b
x
c
d
e
f
g
y

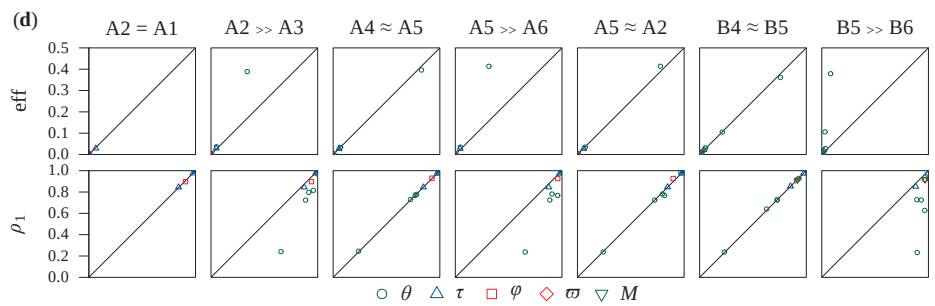

(b)
θAge θAdi θAru θAsu θAga θAza θApe θAma θSmi
θr θa θb θc θd θe θf θg
τr τa τb τc τd τe τf τg τy τx
φx→y ϖx→y Mx→y

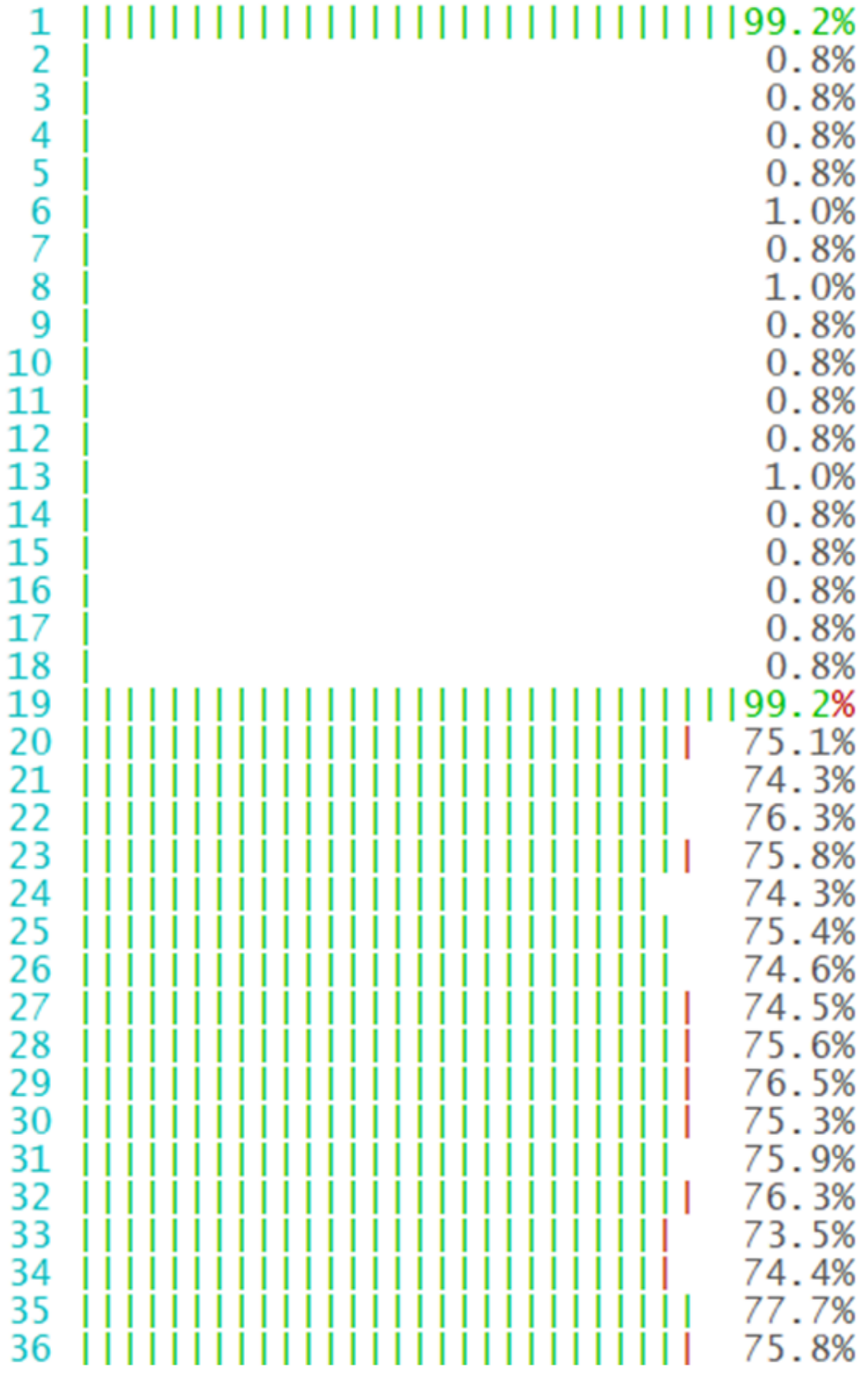

(c)
A2 = A1
A2 >> A3
A4 ≈ A5
A5 >> A6
A5 ≈ A2
B4 ≈ B5
B5 >> B6
eff
ρ1
θ τ φ ϖ M


(a) MSC-I model
(b) MSC-M model
o
a
c
d
g
h
f
e
b
R G C A Q L
$\varphi_{R\to Q}$
$\varphi_{A\to GC}$
$\varpi_{R\to Q}$
$\varpi_{A\to GC}$
(c)
θ_G θ_C θ_R θ_L θ_A θ_Q θ_o θ_a θ_c θ_d θ_b
τ_o τ_g τ_h τ_a τ_c τ_d τ_e τ_f τ_b
$\varphi_{R\to Q}$ $\varphi_{A\to GC}$ $\varpi_{R\to Q}$ $\varpi_{A\to GC}$ $M_{R\to Q}$ $M_{A\to GC}$
(d)
A2 = A1
A2 >> A3
A4 ≈ A5
A5 >> A6
A5 ≈ A2
B4 ≈ B5
B5 >> B6
eff
ρ_1
θ τ φ ϖ M


(a) MSC-I model
(b) MSC-M model
r
s
c
m
$\varphi_{m\to c}$
$\varphi_{c\to m}$
$\varpi_{M\to C}$
$\varpi_{C\to M}$
H. hecale (H)
H. cydno (C)
H. Melpomene (M)


(c)
θ_H θ_C θ_M θ_r θ_s τ_r τ_s τ_c τ_m
$\varphi_{C\to M}$ $\varpi_{C\to M}$ $M_{C\to M}$


(d)
A2 = A1
A2 >> A3
A4 ≈ A5
A5 >> A6
A5 ≈ A2
B4 ≈ B5
B5 >> B6
eff
ρ_1
θ τ φ ϖ M



Load on 18 threads for algorithm A1 (integrating out θs)

Load on 18 threads for algorithm A2 (sampling θs)