IMPROVED VARIATIONAL BAYESIAN PHYLOGENETIC INFERENCE USING MIXTURES

Anonymous authors
Paper under double-blind review

ABSTRACT

We present VBPI-Mixtures, an algorithm designed to enhance the accuracy of phylogenetic posterior distributions, particularly for tree-topology and branch-length approximations. Despite the Variational Bayesian Phylogenetic Inference (VBPI), a leading-edge black-box variational inference (BBVI) framework, achieving remarkable approximations of these distributions, the multimodality of the tree-topology posterior presents a formidable challenge to sampling-based learning techniques such as BBVI. Advanced deep learning methodologies such as normalizing flows and graph neural networks have been explored to refine the branch-length posterior approximation, yet efforts to ameliorate the posterior approximation over tree topologies have been lacking. Our novel VBPI-Mixtures algorithm bridges this gap by harnessing the latest breakthroughs in mixture learning within the BBVI domain. As a result, VBPI-Mixtures is capable of capturing distributions over tree-topologies that VBPI fails to model. We deliver state-of-the-art performance on difficult density estimation tasks across numerous real phylogenetic datasets.

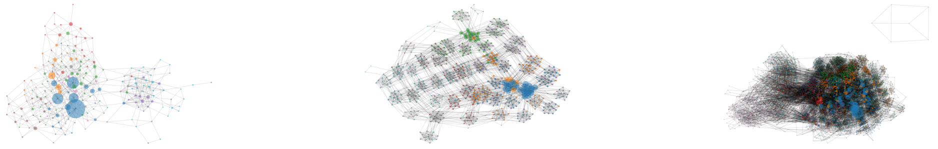
1 INTRODUCTION

Phylogenetic inference has a wide range of applications in various fields, such as molecular evolution, epidemiology, ecology, and tumor progression, making it an essential tool for modern evolutionary research. Bayesian phylogenetics allows researchers to reason about uncertainty in their findings about the evolutionary relationship between species.

The posterior distribution over phylogenetic trees given the species data is, however, challenging to infer, since the latent space is a Cartesian product of the discrete tree-topology space and the continuous branch-length space. Furthermore, the cardinality of the tree-topology space grows as a double factorial of the number of species (taxa), making the marginal likelihood computationally intractable in most interesting problem settings.

For over two decades, Markov Chain Monte Carlo (MCMC) approaches have been the go-to approaches for Bayesian phylogenetic analysis, where the MrBayes software (Huelsenbeck & Ronquist, 2001) has been particularly popular. However, random walk Metropolis-Hastings MCMC methods (Huelsenbeck & Ronquist, 2001; Höhna et al., 2016) rely on local operations to explore the tree-topology posterior, a limitation which is known to require long MCMC runs in order to visit posterior regions which are separated by low-probability tree topologies (Whidden & Matsen IV, 2015). Sequential Monte Carlo methods for Bayesian phylogeny (Bouchard-Côté et al., 2012; Wang et al., 2015; 2020) have been proposed to avoid these local operations, but the resampling mechanism can filter out important trees in early steps of the algorithm as well as cause degeneracy, necessitating many particles.

More recently, variational inference (VI) has been applied to Bayesian phylogenetics. In general, VI is often promoted over sampling-based approaches in high-dimensional problems as a variational approximation of the posterior is obtained from optimization, making VI less vulnerable to the curse of dimensionality. However, in practice, it can be challenging to do VI without utilizing sampling. For example, in Koptagel et al. coordinate-ascent update equations are derived in the phylogenetic setting, but these are evaluated using importance sampling.



(a)

(b)

(c)

Figure 1: Visualization of samples from the tree-topology posterior using a 1,000,000,000 iterations long MCMC run on (a) DS4, (b) DS7 and (c) DS8. Nodes represent unique tree-topologies and are colored based on cluster assignments, illustrating the multimodality of the tree-topology posterior. More details in Sec. 2.

In black-box VI (Ranganath et al., 2014), the gradients are instead taken of Monte-Carlo integrated estimates of the objective, typically the evidence lower bound (ELBO), using samples taken from the variational approximation. This approach has been successfully applied in the Variational Phylogenetic Bayesian Inference (VBPI; Zhang & Matsen IV (2019)) framework, along with its extensions (Zhang, 2020; 2023). In these extensions, more complicated branch-length approximations have been proposed using normalizing flows (NFs; Rezende & Mohamed (2015)) and graph-neural nets (GNNs; Kipf & Welling (2017)).

BBVI allows the practitioner to learn posterior approximations without deriving update equations or closed-form gradient formulations, the samples are taken from a distribution that is commonly known to concentrate on high-probability regions of the posterior, resulting in a learning procedure that does not sufficiently explore the posterior distribution. This was addressed for continuous distributions in Ruiz et al. (2016), where samples were instead drawn from an overdispersed proposal distribution. In discrete hierarchical models, similar to the targets approximated by VBPI, however, insufficient exploration may result in low-level states not being properly modeled. As the tree-topology posterior is known to typically be multimodal with many “subpeaks” (Whidden & Matsen IV, 2015) (visualized in Fig. 1), it is thus of significant importance to encourage the tree-topology approximation to explore the posterior.

We propose VBPI-Mixtures, a novel combination of two recent advances, VBPI and mixture learning in BBVI (Kviman et al., 2023). The mixture components cooperatively explore the tree-topology posterior during learning, and the increased flexibility of the VBPI-Mixtures results in approximations that can model posteriors intractable for the vanilla VBPI (see Sec. 3). Using a toy experiment where we design complicated hierarchical categorical target distributions, we first show that the mixture components specialize in different parts of the solution space and achieve smaller Kullback-Leibler divergences to the targets than a single-component approximation which uses more samples. We then apply VBPI-Mixtures on eight popular real datasets, outperforming the state-of-the-art algorithms. The learned tree-topology approximations are then visualized and compared numerically with MCMC “golden runs”, illustrating the joint exploration of the tree-topology space by the mixture components. Additionally, we derive the VIMCO estimator (Mnih & Rezende, 2016) for mixture approximations. Our contributions can be summarized as follows:

- We propose VBPI-Mixtures, a novel algorithm for Bayesian phylogenetics.
- We show that mixtures of subsplit Bayesian nets (SBNs) can approximate distributions that a single SBN cannot, making a persuasive case for VBPI-Mixtures.
- We derive the VIMCO estimator for mixtures.
- We visualize a two-component mixture of SBNs on real data, verifying that the components jointly explore the tree-topology space.
- Experimentally, we achieve state-of-the-art results on eight popular real phylogenetics datasets, and show that mixtures of SBNs offer more accurate approximations of the tree-topology posterior.

2 Background

Let $\mathcal{B} = \{b(e) : e \in E(\tau)\}$ denote the set of branch lengths for a topology, τ, and X is the data. The posterior distribution over leaf-labeled phylogenetic trees,

$$p(\mathcal{B}, \tau | X) = \frac{p(X|\tau, \mathcal{B})p(\mathcal{B}|\tau)p(\tau)}{p(X)}, \tag{1}$$

is intractable due to its normalizing constant, $p(X)$, i.e., the marginal likelihood. Furthermore, the data, $X = \{X_1, ..., X_N\} \in \Omega^{M \times N}$, are observed sequences of length M on the N leaves of the phylogenetic tree. Each entry in $X_{m,n}$ is a character in the alphabet Ω, e.g., $X_{m,n} \in \{A, C, G, T\}$ if DNA sequences are considered.

Although $p(X)$ is intractable, the three terms in the generative model in Eq. 1 can be computed efficiently. Typically, $p(\tau)$ is a uniform distribution over the (rooted or unrooted) tree-topology space, and the branch-length prior is an exponential distribution with rate λ, such that $p(\mathcal{B}|\tau) = \prod_{e \in E(\tau)} \lambda e^{-\lambda b(e)}$. The likelihood, $p(X|\tau, \mathcal{B})$, can be evaluated in linear (in N) time using the standard dynamic programming algorithm proposed by Felsenstein (2003).

Let a *clade* be a non-empty subset of the set of the N leaf labels, $\mathcal{X}$. A *subsplit* is a partition of this clade into two lexicographically ordered, disjoint clades, while a *split* is simply a root subsplit—a bipartition of $\mathcal{X}$. Furthermore, a primary subsplit pair (PSP) is a subsplit conditioned on a split—a tripartition of $\mathcal{X}$. In Appendix A we additionally give a brief introduction to Bayesian phylogenetic inference for machine learning researchers.

Variational Inference in Bayesian Phylogenetics The VI-based approach to Bayesian phylogenetics is to approximate $p(\mathcal{B}, \tau|X)$ using a simpler distribution, $q_{\psi,\phi}(\mathcal{B}, \tau) = q_\psi(\mathcal{B}|\tau)q_\phi(\tau)$. Generally, in VI, the approximations are learned by maximizing the evidence lower bound (ELBO),

$$\mathcal{L}(X) = \mathbb{E}_{q_{\psi,\phi}(\mathcal{B},\tau)} \left[\log \frac{p(X|\tau, \mathcal{B})p(\mathcal{B}|\tau)p(\tau)}{q_\psi(\mathcal{B}|\tau)q_\phi(\tau)} \right], \tag{2}$$

implicitly minimizing the Kullback-Leibler (KL) divergence from $p(\mathcal{B}, \tau|X)$ to $q_{\psi,\phi}(\mathcal{B}, \tau)$.

Subsplit Bayesian Networks Given a set of tree topologies, $\mathcal{T}$,[1] it is straightforward to form a look-up table of all subsplits in $\mathcal{T}$. The SBN uses the look-up table to define support over possible tree topologies and learns the probabilities of the subsplits in the table via, for example, stochastic optimization. As the look-up table contains probabilities of subsplits, it is referred to as a *conditional probability table* (CPT). When the CPT has been learned, the SBN provides a tractable probability distribution over tree topologies from which it is possible to sample. See Zhang & Matsen IV (2018) or Zhang & Matsen IV (2022) for the original, more in-depth accounts of SBNs.

VBPI In VBPI (Zhang & Matsen IV, 2019), the posterior approximations are learned by maximizing a multi-sample (Burda et al., 2015) version of $\mathcal{L}(X)$,

$$\mathcal{L}(X; K) = \mathbb{E}_{\mathcal{B}^k, \tau^k \sim q_{\psi,\phi}(\mathcal{B},\tau)} \left[\log \frac{1}{K} \sum_{k=1}^{K} \frac{p(X|\tau^k, \mathcal{B}^k)p(\mathcal{B}^k|\tau^k)p(\tau^k)}{q_\psi(\mathcal{B}^k|\tau^k)q_\phi(\tau^k)} \right], \tag{3}$$

where $q_\phi(\tau)$ is an SBN with a learnable CPT, ϕ, and $q_\psi(\mathcal{B}|\tau)$ is a multivariate LogNormal distribution with a diagonal covariance matrix, such that

$$q_\psi(\mathcal{B}|\tau) = \prod_{e \in E(\tau)} q(b(e)|\mu(e, \tau), \sigma^2(e, \tau)). \tag{4}$$

Two different parameterizations of $\mu(e, \tau)$ and $\sigma(e, \tau)$ have previously been proposed. The simpler approach is to let $\mu(e, \tau) = \psi^\mu_{e/\tau}$ and $\sigma(e, \tau) = \psi^\sigma_{e/\tau}$, where e/τ denotes a split of τ in edge e. The parameters $\psi^\mu_{e/\tau}$ and $\psi^\sigma_{e/\tau}$ are shared among all tree topologies where e/τ exists, resulting in an

[1] $\mathcal{T}$ is in practice obtained from some efficient tree-topology sampling algorithm, typically UFBoot (Minh et al., 2013).

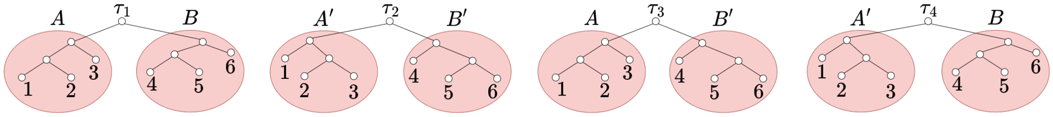


Figure 2: Mixtures of SBNs increase the flexibility of the tree-topology approximation. For instance, they can exactly capture a target distribution that assigns its probability uniformly across τ_1 and τ_2, leaving zero probability to τ_3 and τ_4. Meanwhile, this is not possible for a single SBN. See Example 3.1 for details.

amortized mapping from a tree topology to the parameters of $q_\psi(\mathcal{B}|\tau)$. Additional local information about the given τ can be added into the parameterization of the approximation by using PSPs,

$$\mu(e,\tau) = \psi^{\mu}_{e/\tau} + \sum_{i \in e /\!\!/ \tau} \gamma^{\mu}_i, \quad \sigma(e,\tau) = \psi^{\sigma}_{e/\tau} + \sum_{i \in e /\!\!/ \tau} \gamma^{\sigma}_i, \tag{5}$$

where $e /\!\!/ \tau$ denotes the set of PSPs neighboring to the split e/τ, and γ^{μ}_i is a learnable parameter associated with the i-th pair.

Multimodality of the tree-topology posterior In Whidden & Matsen IV (2015), modes are referred to as clusters of MCMC samples that are densely grouped in the tree-topology space and have high posterior density compared to their neighbors. They proposed a method for detecting and quantifying peaks by calculating the reversible subtree pruning and regrafting distance between topologies, combined with the MrBayes MCMC posterior probability. By applying this method, it was identified that certain datasets had a high number of modes (e.g., DS1, DS4, DS5, DS6, DS7), which shows the complexity of the tree-topology space. In Fig. 1 we visualize the multimodality of the posterior on three datasets.

Mixtures in Black-Box VI Learning mixtures of approximations in (black-box; Ranganath et al. (2014)) VI (Nalisnick et al., 2016; Morningstar et al., 2021; Kviman et al., 2023) is a compelling off-the-shelf technique to increase the flexibility of a variational approximation. Mixtures can be applied to any variational approximation, including an NF-based approximation or one for discrete latent variables, with little overhead.

The objective function, the ELBO for mixtures, is estimated by sampling from each mixture component in a stratified manner (Morningstar et al., 2021), or via multiple importance sampling techniques (Kviman et al., 2022), why the objective is often referred to as *MISELBO*. Maximizing MISELBO encourages the mixture components to cooperatively cover the target distribution, which is thought to be the key ingredient for their success in density estimation tasks (Kviman et al., 2023). In the next section, we formulate MISELBO for VBPI and explain why mixtures of SBNs are beneficial for exploring the complex tree-topology space.

3 Variational Bayesian Phylogenetic Inference using Mixtures

Here we present our proposed method, VBPI-Mixtures. We derive the VIMCO estimator for learning mixtures of SBNs, and show how to combine mixtures of branch length models with an expressive NF model. We start by providing the MISELBO formulation for VBPI with K importance samples,

$$\mathcal{L}(X;K,S) = \frac{1}{S}\sum_{s=1}^{S} \mathbb{E}_{\mathcal{B}_s^{1:K},\tau_s^{1:K} \sim q_{\psi_s,\phi_s}(\mathcal{B},\tau)}\left[\log \frac{1}{K}\sum_{k=1}^{K} \frac{p(X|\tau_s^k,\mathcal{B}_s^k)p(\mathcal{B}_s^k|\tau_s^k)p(\tau_s^k)}{\frac{1}{S}\sum_{j=1}^{S} q_{\psi_j}(\mathcal{B}_s^k|\tau_s^k)q_{\phi_j}(\tau_s^k)}\right]. \tag{6}$$

To evaluate $\mathcal{L}(X;K,S)$, we approximate the s-th expectation by Monte-Carlo integration using simulations from $q_{\psi_s,\phi_s}(\mathcal{B},\tau)$.

Mixtures promote exploration Note that minimizing the denominator in Eq. (6) corresponds to diversifying the mixture distribution, i.e., promoting the mixture components to jointly explore the latent space. This exploratory behavior is crucial in black-box VI as the ELBO is only evaluated in sampled (visited) latent variables (states).

More specifically, the samples are proposed from the same distribution that attempts to maximize the ELBO. Consequently, for $S = 1$, there is a risk that, for a fixed K, less probable regions will not be sufficiently explored during learning and thus will be poorly modeled. In fact, Zhang & Matsen IV (2019) showed that the vanilla VBPI does not benefit from more importance samples during training. Fortunately, the MISELBO objective offers a promising solution, as the mixture components are promoted to spread out and efficiently explore the multimodal phylogenetic posterior.

Mixtures increase the flexibility of the approximations A mixture of LogNormal pdfs is clearly more flexible than a single LogNormal pdf. For mixtures of SBNs, this is also true but may be less clear. An SBN constructs a tree by stochastically partitioning the available clades. The partition of a clade is sampled *independently* of the partitions sampled in the other clades. Mixtures of SBNs allow for modeling correlations in the sampling of the partitions, and thus increase the flexibility of the approximation. We explain this feature with a simple example, which trivially generalizes to larger trees and also applies to unrooted trees.

Example 3.1 *Consider the four rooted tree topologies in Fig. 2 where A, B, A' and B' are four different subtrees for the two clades (in red), and a target density, such that $p(\tau_1) = p(\tau_2) = 0.5$. A uniformly weighted mixture of two SBNs can easily approximate p exactly by letting $q_1(\tau_1) = 1$ and $q_2(\tau_2) = 1$, resulting in $\frac{1}{2}q_1(\tau_1) + 0 = 0 + \frac{1}{2}q_2(\tau_2) = 0.5$. However, as a single SBN, q, samples A or A' and B or B' independently, and in order to achieve $q(\tau_1), q(\tau_2) > 0$, it will have to assign non-zero probability to all four subtrees. Specifically, say $q(A) = \alpha$ and, consequently, $q(A') = 1-\alpha$, while $q(B) = \beta$ and, consequently, $q(B') = 1-\beta$. It follows that $q(\tau_3) = \alpha(1-\beta)$ and $q(\tau_4) = (1-\alpha)\beta$. Finally, $\alpha\beta = q(\tau_1) = q(\tau_2) = (1-\alpha)(1-\beta)$ implies $1 = \alpha + \beta$, which, in turn, implies that $q(\tau_3) = \alpha^2$ and $q(\tau_4) = \beta^2$. That is, all four trees will either have probability $1/4$ under q, or one of τ_3 and τ_4 will have a higher probability than τ_1 and τ_2. So, a single SBN can only yield a distribution that is very different from p.*

The above example exemplifies that there are tree-topology distributions which cannot be modeled using a single SBN, but which can be modeled by a mixture of SBNs. There is no converse example, as a mixture can trivially model a single SBN by letting $q_1(\tau) = q_2(\tau)$ for all τ. In Appendix B we construct an example which shows that these conflicting tree-topology posteriors can indeed occur for real DNA data.

3.1 VIMCO for Mixtures of Tree-Topology Approximations

Here we derive the VIMCO estimator of Eq. (6). Although the notation in this section is specific for Bayesian phylogenetics, our result is applicable to any mixture approximation. We purposefully follow the derivations in Mnih & Rezende (2016) closely.

3.1.1 Gradient Analysis

The gradients of Eq. (6) are studied first. Let

$$f(x, \mathcal{B}_s^k, \tau_s^k) = \frac{p(\mathcal{B}_s^k, \tau_s^k, X)}{\frac{1}{S}\sum_{j=1}^S q_{\psi_j}(\mathcal{B}_s^k|\tau_s^k) q_{\phi_j}(\tau_s^k)}, \tag{7}$$

and $\hat{L}_s^K = \log \frac{1}{K}\sum_{k=1}^K f(x, \mathcal{B}_s^k, \tau_s^k)$, where $\mathcal{B}_s^k, \tau_s^k$ are simulated from $q_{\psi_s,\phi_s}(\mathcal{B}, \tau)$. Note that $f(x, \mathcal{B}_s^k, \tau_s^k)$, and so also $\hat{L}_s^K$, is a function of the SBN parameters for all mixture components, i.e., $\phi_1, ..., \phi_S$. However, we omit these as arguments to the function in order to avoid cluttered notation.

We are interested in the gradient of Eq. (6) with respect to the SBN parameters for one of the mixture components, say i,

$$\nabla_{\phi_i}\mathcal{L}(X; K, S) = \nabla_{\phi_i}\frac{1}{S}\sum_{s=1}^S \mathbb{E}_{q_{\psi_s,\phi_s}(\mathcal{B},\tau)}\left[\hat{L}_s^K\right]. \tag{8}$$

The full derivations are given in Appendix C and lead to the following expression

$$\begin{aligned}\nabla_{\phi_i}\mathcal{L}(X;K,S) = \frac{1}{S}\mathbb{E}_{q_{\psi_i,\phi_i}(\mathcal{B},\tau)}\Big[\hat{L}_i^K \sum_{k=1}^{K} \nabla_{\phi_i} \log q_{\phi_i}(\tau_i^k)\Big] - \\ \frac{1}{S}\sum_{s=1}^{S}\mathbb{E}_{q_{\psi_s,\phi_s}(\mathcal{B},\tau)}\Big[\sum_{k=1}^{K} \tilde{w}_s^k \nabla_{\phi_i} \log \frac{1}{S}\sum_{j=1}^{S} q_{\psi_j}(\mathcal{B}_s^k|\tau_s^k) q_{\phi_j}(\tau_s^k)\Big].\end{aligned} \tag{9}$$

We make three important observations, (i) for $S = 1$, we retrieve the gradients used to train VBPI, (ii) as the second term is negated, maximizing it corresponds to diversifying the mixture distribution w.r.t. ϕ_i, and (iii) the first term is merely a scaled (by $1/S$) version of the corresponding term in the $S = 1$ case. Connecting to observation (iii), we conclude that extending the VIMCO estimator to $S > 1$ cannot be trivially achieved without our derivation provided above.

Furthermore, the analysis of the gradients of the importance weighted lower bound in (Mnih & Rezende, 2016)—using our notation, $\mathcal{L}(X;K)$—applies here, too. That is, the gradients in the second term in Eq. (9) are multiplied by normalized weights, ensuring that the norm of the weighted sum over all K gradients is not greater than the norm of the largest term in the sum. This means that ϕ_i will be updated mainly according to gradients based on simulations scored highly by f, while mitigating the impact of gradients from lower-scoring simulations.

In the first term, on the other hand, all gradients are multiplied by the same $\hat{L}_i^K$, indicating that the gradients of high-scoring simulations will not receive more weight than low-scoring ones, causing high variance and slow learning.

3.1.2 The VIMCO Estimator

As concluded above, the second term in Eq. (9) is well-behaved, and we do not need variance-reduction techniques to use it for learning in practice. The first term, however, requires attention in order to facilitate efficient learning.

Fortunately, as the first term is merely a scaled version of the corresponding term in the VIMCO estimator when $S = 1$, we can directly apply the localized learning signal strategy from Mnih & Rezende (2016) to obtain the VIMCO estimator for $S \geq 1$,

$$\begin{aligned}\nabla_{\phi_i}\mathcal{L}(X;K,S) \simeq \frac{1}{S}\sum_{k=1}^{K} \hat{L}_{i(k|-k)}^K \nabla_{\phi_i} \log q_{\phi_i}(\tau_i^k) - \\ \frac{1}{S}\sum_{s=1}^{S}\sum_{k=1}^{K} \tilde{w}_s^k \nabla_{\phi_i} \log \frac{1}{S}\sum_{j=1}^{S} q_{\psi_j}(\mathcal{B}_s^k|\tau_s^k) q_{\phi_j}(\tau_s^k),\end{aligned} \tag{10}$$

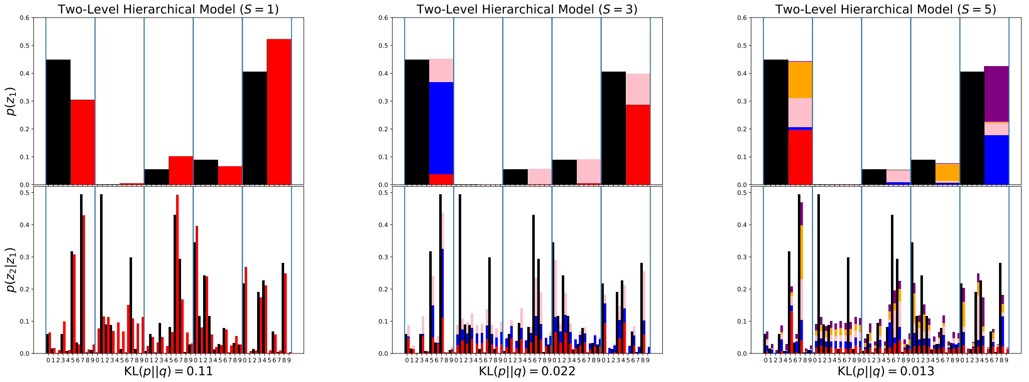
where $\tau_i^k, \mathcal{B}_i^k \sim q_{\phi_i,\psi_i}(\tau,\mathcal{B})$ and $\tau_s^k, \mathcal{B}_s^k \sim q_{\phi_s,\psi_s}(\tau,\mathcal{B})$. Here, $\hat{L}_{i(k|-k)}^K$ is the local learning signal for sample k, defined as

$$\hat{L}_{i(k|-k)}^K = \hat{L}_i^K - \log \frac{1}{K}\left(\sum_{k' \neq k} f\left(x, \tau_i^{k'}, \mathcal{B}_i^{k'}\right) + \hat{f}\left(x, \tau_i^{-k}, \mathcal{B}_i^{-k}\right)\right), \tag{11}$$

with $\hat{f}\left(x, \tau_i^{-k}, \mathcal{B}_i^{-k}\right)$ being an estimator of $f\left(x, \tau_i^k, \mathcal{B}_i^k\right)$, typically the geometric mean Mnih & Rezende (2016); Zhang & Matsen IV (2019); Zhang (2020), $\hat{f}\left(x, \tau_i^{-k}, \mathcal{B}_i^{-k}\right) = \exp\left(\frac{1}{K-1}\sum_{k' \neq k} \log f\left(x, \tau_i^{k'}, \mathcal{B}_i^{k'}\right)\right)$.

4 Experiments

In Sec. 3, we argued that a single-component approximation will struggle to properly model all parts of the target distribution when learned with black-box VI. Below, in Sec. 4.1, we experimentally verify this claim, and, furthermore, confirm that mixture components collaborate in order to jointly cover the target density, resulting in more accurate approximations and efficient exploration.



(a)

(b)

(c)

Figure 3: Approximations using (a) $S = 1$, (b) $S = 3$ and (c) $S = 5$ components of a two-level hierarchical target distribution with $n_1 = 5$ and $n_2 = 10$. The target is plotted in black. Each bin in the lower plot, delimited by the blue vertical lines, contains a CPD, conditioned on z_1 in the upper plot. The colors of the approximations represent the different mixture components, and their probabilities, scaled by $1/S$, are stacked. The approximations were trained using $K = \lfloor 20/S \rfloor$ importance samples, and $S = 5$ achieved the smallest KL divergence from the target distribution, printed below the plots. The components clearly cooperate in (b-c) as they explore complementary parts of the solution space.

We then, in Sec. 4.2, demonstrate that the increased model flexibility and promotion of exploration translates into better marginal log-likelihood estimates and more accurate tree-topology posterior approximations. We also visualise representations of VBPI-Mixtures on real data. Code for all experiments is provided at `github-url`.

4.1 Exploring a Discrete Two-Level Hierarchical Model using Black-Box VI

SBNs are hierarchical models with categorical distributions at each level. Here, we examine how mixtures explore discrete hierarchical target distribution, when learned via black-box VI.

Using a two-leveled hierarchical model of categorical distributions as the target distribution, $p(z_1, z_2) = p(z_2|z_1)p(z_1)$, we wish to minimize $\mathrm{KL}\left(\frac{1}{S}\sum_{s=1}^{S} q_{\phi_s}(z_1, z_2)\big\|p(z_1, z_2)\right)$. The CPD $p(z_2|z_1)$ is a categorical distribution with n_2 categories, conditioned on the sampled category in the previous level, z_1, and $p(z_1)$ is a categorical distribution with n_1 categories. Similarly, the s-th component in the mixture approximation, $q_{\phi_s}(z_1, z_2) = q_{\phi_s}(z_2|z_1)q_{\phi_s}(z_1)$, is a two-level hierarchical model with learnable probabilities, ϕ_s. The parameters of p are drawn from a Dirichlet distribution with all concentration parameters equal to 0.5, the approximations are trained using the VIMCO estimator derived in Sec. 3.1.2, and the learning rates are chosen based on a grid search on a different target distribution. All ϕ_s are initialized uniformly over the categories.

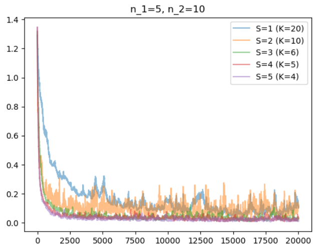


Figure 4: KL divergences from the target visualized in Fig. 3 and the approximations with $S = 1, ..., 5$ mixture components. On the x-axis are the number of training iterations.

In Fig. 3, three learned approximations are shown when $n_1 = 5$ and $n_2 = 10$, along with the corresponding KL divergences from the target to the approximation. For $S > 1$, the components have spread out, exploring complementary parts of the solution space. Note that the $p(z_1 = 2)$ has a negligible probability, resulting in approximations that do not capture $p(z_2|z_1 = 2)$.

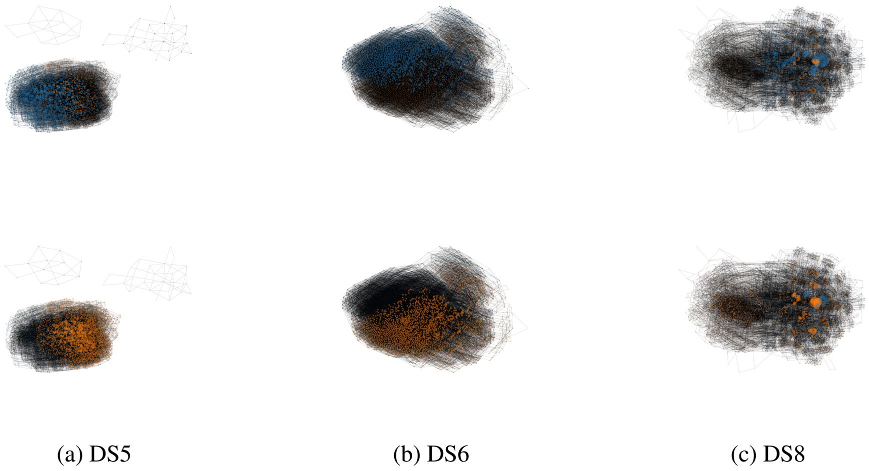

(a) DS5 (b) DS6 (c) DS8

Figure 5: Visualization of a uniformly weighted $S = 2$-component mixture of SBNs on (a) DS5, (b) DS6 and (c) DS8, where each node corresponds to a unique tree-topology. The upper row shows the distribution of five million sampled tree topologies from the first component, where a node, τ, is colored blue if $q_{\phi_1}(\tau) > q_{\phi_2}(\tau)$, or orange otherwise. Vice versa for the lower row. The size of a node is determined by its sampling frequency, which is why nodes with low frequency appear black. The components clearly spread out, exploring different parts of the space.

We include the curves of the KL divergences over the training iterations when $n_1 = 5$ and $n_2 = 10$ in Fig. 6, and, in Appendix D.1, we show the KL curves for other choices of K, n_1 and n_2. In all cases where $p(z_1)$ has multiple categories with non-negligible probabilities, (relating to multimodality in the tree-topology posterior) mixtures converge with fewer iterations and to smaller KL divergences.

4.2 Posterior Approximations using Real Data

We performed experiments on eight datasets (Hedges et al., 1990; Garey et al., 1996; Yang & Yoder, 2003; Henk et al., 2003; Lakner et al., 2008; Zhang & Blackwell, 2001; Yoder & Yang, 2004; Rossman et al., 2001) which we will refer to as DS1-8. These are popular datasets for evaluating Bayesian phylogenetics methods, and, as in Zhang & Matsen IV (2019); Zhang (2020); Moretti et al. (2021); Koptagel et al.; Zhang & Matsen IV (2022); Zhang (2023), we focus on learning the approximations of branch-length and tree-topology distributions. Following the referenced works, we assume the exponential branch-length prior has rate 10 and a uniform prior over all unrooted trees (see Sec. 2 for details about the generative model). The substitution model is the Jukes-Cantor 69 model (Jukes et al., 1969), and the candidate trees, $\mathcal{T}$, are gathered from ten replicates of 10000 ultrafast maximum likelihood bootstrap trees (Minh et al., 2013). The implementation is based on the code provided by Zhang & Matsen IV (2022), and we trained all VBPI models during 400,000 iterations, using the same hyperparameter settings as Zhang & Matsen IV (2019); Zhang (2020). Based on the study in Zhang & Matsen (2022), we let $K = 10$ during training.

4.2.1 Visualizing the Explorative Behaviour of Mixtures of SBNs

To graphically confirm the power of employing mixtures of SBNs, we in Fig. 5 visualize representations of the learned tree-topology posterior approximations for a subset of the different real datasets. The subset was selected based on the datasets where the explorative behavior of the approximations is most clear. The representations for the other datasets, along with more implementation details, are included in the Appendix D.2. The SBNs correspond to VBPI-Mixtures without NFs. The components (upper vs. lower row) have jointly explored the space, partly specializing on disjoint sets of tree-topologies, verifying that the MISELBO objective promotes coordinated exploration of the discrete latent space, as discussed in Sec. 3.

Table 1: Illustrating the impact of VBPI-Mixtures in terms of $\mathrm{KL}(p(\tau|X)\|q_\phi(\tau))$. Lower is better.

	DS1	DS2	DS3	DS4	DS5	DS6	DS7	DS8
VBPI-NF	0.0726	0.0110	0.0540	0.2093	2.2117	1.2842	0.2544	0.6018
Mix$_{\text{NF},S\,=\,2}$	0.0631	0.0059	0.0475	0.0965	2.0337	1.0883	0.1183	0.5199
Mix$_{\text{NF},\,S\,=\,3}$	**0.0598**	**0.0051**	**0.0377**	**0.0769**	**1.9526**	**1.0461**	**0.0847**	**0.4567**

Table 2: Marginal log-likelihood estimates on DS1-8. All VBPI methods use 1000 importance samples, and the results are averaged over 100 runs and three independently trained models. Following Zhang & Matsen IV (2019); Zhang (2020); Zhang & Matsen IV (2022); Zhang (2023), we bold font the results with the lowest standard deviations (shown in parentheses). Details in Sec. 4.2.2. Mixtures monotonically improve with S.

Data	DS1	DS2	DS3	DS4	DS5	DS6	DS7	DS8
# Taxa	27	29	36	41	50	50	59	64
# Sites	1949	2520	1812	1137	378	1133	1824	1008
VBPI with Mixtures								
VBPI	-7108.50 (0.23)	-26367.70 (0.09)	-33735.10 (0.14)	-13330.03 (0.23)	-8214.80 (0.50)	-6724.59 (0.53)	-37332.12 (0.45)	-8652.39 (0.71)
Mix$_{S\,=\,2}$	-7108.44 (0.12)	-26367.71 (0.06)	-33735.10 (0.07)	-13330.00 (0.17)	-8214.75 (0.36)	-6724.54 (0.31)	-37332.04 (0.24)	-8651.68 (0.49)
Mix$_{S\,=\,3}$	-7108.42 (0.11)	-26367.71 (0.04)	-33735.10 (0.06)	-13329.97 (0.17)	-8214.73 (0.26)	-6724.51 (0.28)	-37332.03 (0.18)	-8650.83 (0.46)
VBPI with NFs and Mixtures								
VBPI-NF	-7108.42 (0.15)	-26367.72 (0.06)	-33735.10 (0.07)	-13330.00 (0.23)	-8214.70 (0.47)	-6724.50 (0.45)	-37332.01 (0.27)	-8650.68 (0.46)
Mix$_{\text{NF},S\,=\,2}$	-7108.40 (0.10)	-26367.71 (0.04)	-33735.10 (0.05)	-13329.95 (0.15)	-8214.62 (0.26)	-6724.44 (0.32)	-37331.96 (0.19)	-8650.56 (0.33)
Mix$_{\text{NF},S\,=\,3}$	**-7108.40 (0.06)**	**-26367.70 (0.03)**	**-33735.09 (0.04)**	**-13329.94 (0.11)**	**-8214.56 (0.22)**	**-6724.40 (0.23)**	**-37331.96 (0.15)**	**-8650.54 (0.30)**
MCMC and VBPI with GNNs (scores from Zhang & Matsen IV (2019) and Zhang (2023))								
MrBayes$_{\text{ss}}$	-7108.42 (0.18)	-26367.57 (0.48)	-33735.44 (0.50)	-13330.06 (0.54)	-8214.51 (0.28)	-6724.07 (0.86)	-37332.76 (2.42)	-8649.88 (1.75)
GGNN	-7108.40 (0.19)	-26367.73 (0.10)	-33735.11 (0.09)	-13329.95 (0.19)	-8214.67 (0.36)	-6724.38 (0.42)	-37332.03 (0.30)	-8650.68 (0.48)
EDGE	-7108.41 (0.14)	-26367.73 (0.07)	-33735.12 (0.09)	-13329.94 (0.19)	-8214.64 (0.38)	-6724.37 (0.40)	-37332.04 (0.26)	-8650.65 (0.45)

4.2.2 Quantitative Evaluation of the Approximations

We quantitatively evaluate the approximations by, first, computing their KL divergences to the true posterior, and by, secondly, benchmarking their marginal log-likelihood estimates. The results are averaged over five independently trained models with different parameter initializations. All MrBayes (Huelsenbeck & Ronquist, 2001) results were produced using ten million long MCMC runs with four chains, sampling every 100 iterations. Our methods are denoted Mix$_S$, Mix$_{\text{NF},S}$, representing VBPI-Mixtures with PSPs or NFs, respectively. Mixtures that employ NFs share flow models, as described in Kviman et al. (2023).

Statistical distances to the tree-topology posterior Here, we compare statistical distances to the tree-topology posterior obtained from MrBayes as described above. In Table 1, the KL divergence from the posterior to the approximations, i.e., $\mathrm{KL}(p(\tau|X)\|q_\phi(\tau))$, is computed, where q_ϕ represents a mixture of SBNs, or a single SBN, from VBPI-NF. Lower is better, and, notably, VBPI-Mixtures consistently produce KL divergences across all datasets that monotonically decrease with S.

Marginal log-likelihood estimates In terms of marginal log-likelihood estimates, we benchmark our methods against the existing VBPI algorithms: VBPI with PSP parameterization (Zhang & Matsen IV, 2019), VBPI-NF (Zhang, 2020) with ten RealNVPs (Dinh et al., 2016), and VBPI-GNN (Zhang (2023); EDGE and GGNN). Additionally, we compare our results with the stepping-stone (SS; Xie et al. (2011)) method applied to MrBayes. The results are given in Table 2. Following Zhang & Matsen IV (2019); Zhang (2020); Zhang & Matsen IV (2022); Zhang (2023), we bold font the results with the lowest standard deviations. Rewarding low-variance estimates is motivated, as they imply, for instance, more reliable Bayesian model selections for downstream tasks. Increasing the number of mixture components results in significant improvements in terms of lower standard deviations (on all datasets) and higher mean log-likelihood scores (especially apparent on the more complex datasets, e.g. DS5-8).

5 Conclusion

We introduced VBPI-Mixtures, a novel algorithm that increases the flexibility of the phylogenetic posterior approximation by utilizing recent advances in mixtures for black-box VI. We showed that mixtures of SBNs can approximate distributions that a single SBN cannot, making a persuasive case for VBPI-Mixtures. Experimentally, we achieved state-of-the-art results in terms of marginal log-likelihood estimation and produced more accurate approximations of the tree-topology posterior.

REFERENCES

Alexandre Bouchard-Côté, Sriram Sankararaman, and Michael I Jordan. Phylogenetic inference via sequential monte carlo. *Systematic biology*, 61(4):579–593, 2012.

Yuri Burda, Roger Grosse, and Ruslan Salakhutdinov. Importance weighted autoencoders. September 2015.

Laurent Dinh, Jascha Sohl-Dickstein, and Samy Bengio. Density estimation using real nvp. *arXiv preprint arXiv:1605.08803*, 2016.

Joseph Felsenstein. *Inferring Phylogenies*. Sinauer, October 2003.

J R Garey, T J Near, M R Nonnemacher, and others. Molecular evidence for acanthocephala as a subtaxon of rotifera. *Journal of Molecular*, 1996.

Aric Hagberg, Pieter Swart, and Daniel S Chult. Exploring network structure, dynamics, and function using networkx. Technical report, Los Alamos National Lab. (LANL), Los Alamos, NM (United States), 2008.

S B Hedges, K D Moberg, and others. Tetrapod phylogeny inferred from 18S and 28S ribosomal RNA sequences and a review of the evidence for amniote relationships. *Mol. Biol.*, 1990.

Daniel A Henk, Alex Weir, and Meredith Blackwell. Laboulbeniopsis termitarius, an ectoparasite of termites newly recognized as a member of the laboulbeniomycetes. *Mycologia*, 95(4):561–564, 2003.

Sebastian Höhna, Michael J Landis, Tracy A Heath, Bastien Boussau, Nicolas Lartillot, Brian R Moore, John P Huelsenbeck, and Fredrik Ronquist. Revbayes: Bayesian phylogenetic inference using graphical models and an interactive model-specification language. *Systematic biology*, 65 (4):726–736, 2016.

John P Huelsenbeck and Fredrik Ronquist. Mrbayes: Bayesian inference of phylogenetic trees. *Bioinformatics*, 17(8):754–755, 2001.

Thomas H Jukes, Charles R Cantor, and Others. Evolution of protein molecules. *Mammalian protein metabolism*, 3:21–132, 1969.

Thomas N Kipf and Max Welling. Semi-supervised classification with graph convolutional networks. *ICLR*, 2017.

Hazal Koptagel, Oskar Kviman, Harald Melin, Negar Safinianaini, and Jens Lagergren. Vaiphy: a variational inference based algorithm for phylogeny. In *Advances in Neural Information Processing Systems*.

Oskar Kviman, Harald Melin, Hazal Koptagel, Victor Elvira, and Jens Lagergren. Multiple importance sampling elbo and deep ensembles of variational approximations. In *International Conference on Artificial Intelligence and Statistics*, pp. 10687–10702. PMLR, 2022.

Oskar Kviman, Ricky Molén, Alexandra Hotti, Semih Kurt, Vıctor Elvira, and Jens Lagergren. Cooperation in the latent space: The benefits of adding mixture components in variational autoencoders. In *International Conference on Machine Learning*, pp. 18008–18022. PMLR, 2023.

Clemens Lakner, Paul van der Mark, John P Huelsenbeck, Bret Larget, and Fredrik Ronquist. Efficiency of markov chain monte carlo tree proposals in bayesian phylogenetics. *Syst. Biol.*, 57(1): 86–103, February 2008.

Bui Quang Minh, Minh Anh Thi Nguyen, and Arndt von Haeseler. Ultrafast approximation for phylogenetic bootstrap. *Mol. Biol. Evol.*, 30(5):1188–1195, May 2013.

Andriy Mnih and Danilo Rezende. Variational inference for monte carlo objectives. In Maria Florina Balcan and Kilian Q Weinberger (eds.), *Proceedings of The 33rd International Conference on Machine Learning*, volume 48 of *Proceedings of Machine Learning Research*, pp. 2188–2196, New York, New York, USA, 2016. PMLR.

Antonio Khalil Moretti, Liyi Zhang, Christian A Naesseth, Hadiah Venner, David Blei, and Itsik Pe'er. Variational combinatorial sequential monte carlo methods for bayesian phylogenetic inference. In *Uncertainty in Artificial Intelligence*, pp. 971–981. PMLR, 2021.

Warren Morningstar, Sharad Vikram, Cusuh Ham, Andrew Gallagher, and Joshua Dillon. Automatic differentiation variational inference with mixtures. In *International Conference on Artificial Intelligence and Statistics*, pp. 3250–3258. PMLR, 2021.

Eric Nalisnick, Lars Hertel, and Padhraic Smyth. Approximate inference for deep latent gaussian mixtures. In *NIPS Workshop on Bayesian Deep Learning*, volume 2, pp. 131, 2016.

Rajesh Ranganath, Sean Gerrish, and David Blei. Black box variational inference. In *Artificial intelligence and statistics*, pp. 814–822. PMLR, 2014.

Danilo Rezende and Shakir Mohamed. Variational inference with normalizing flows. In *International conference on machine learning*, pp. 1530–1538. PMLR, 2015.

Amy Y Rossman, John M McKemy, Rebecca A Pardo-Schultheiss, and Hans-Josef Schroers. Molecular studies of the bionectriaceae using large subunit rDNA sequences. *Mycologia*, 93 (1):100–110, January 2001.

Francisco JR Ruiz, Michalis K Titsias, and David M Blei. Overdispersed black-box variational inference. *Uncertainty in Artificial Intelligence*, 2016.

Liangliang Wang, Alexandre Bouchard-Côté, and Arnaud Doucet. Bayesian phylogenetic inference using a combinatorial sequential monte carlo method. *Journal of the American Statistical Association*, 110(512):1362–1374, 2015.

Liangliang Wang, Shijia Wang, and Alexandre Bouchard-Côté. An annealed sequential monte carlo method for bayesian phylogenetics. *Systematic biology*, 69(1):155–183, 2020.

Chris Whidden and Frederick A Matsen IV. Quantifying mcmc exploration of phylogenetic tree space. *Systematic biology*, 64(3):472–491, 2015.

Wangang Xie, Paul O Lewis, Yu Fan, Lynn Kuo, and Ming-Hui Chen. Improving marginal likelihood estimation for bayesian phylogenetic model selection. *Systematic biology*, 60(2):150–160, 2011.

Z Yang and A D Yoder. Comparison of likelihood and bayesian methods for estimating divergence times using multiple gene loci and calibration points, with application to a radiation of cute-looking mouse lemur species. *Syst. Biol.*, 2003.

Anne D Yoder and Ziheng Yang. Divergence dates for malagasy lemurs estimated from multiple gene loci: geological and evolutionary context. *Mol. Ecol.*, 13(4):757–773, April 2004.

Cheng Zhang. Improved variational bayesian phylogenetic inference with normalizing flows. *Advances in neural information processing systems*, 33:18760–18771, 2020.

Cheng Zhang. Learnable topological features for phylogenetic inference via graph neural networks. *ICLR*, 2023.

Cheng Zhang and Frederick A Matsen, IV. A variational approach to bayesian phylogenetic inference. April 2022.

Cheng Zhang and Frederick A Matsen IV. Generalizing tree probability estimation via bayesian networks. *Advances in neural information processing systems*, 31, 2018.

Cheng Zhang and Frederick A Matsen IV. Variational bayesian phylogenetic inference. In *ICLR*, 2019.

Cheng Zhang and Frederick A Matsen IV. A variational approach to bayesian phylogenetic inference. *arXiv preprint arXiv:2204.07747*, 2022.

N Zhang and M Blackwell. Molecular phylogeny of dogwood anthracnose fungus (discula destructiva) and the diaporthales. *Mycologia*, 2001.

A A Brief Introduction to Bayesian Phylogenetic Inference for Machine Learning Researchers

This introduction aims to briefly explain the basic concepts required to understand the generative model provided in the main text.

Phylogenetic trees capture evolutionary relationships among species and provides valuable insights into life's evolutionary history. Within this domain, phylogenies are often depicted as bifurcating tree graphs, where nodes represent common ancestors, and branches (edges) signify evolutionary events and genetic distances between species. This framework enables an understanding of species relatedness, ancestry, and the evolutionary processes governing life's diversity.

Bayesian phylogenetic inference builds upon this framework by applying Bayesian statistical methods to infer the evolutionary history. It allows for a probabilistic approach to model uncertainty and variation, considering prior beliefs about evolutionary parameters and updating these beliefs as new data is incorporated. It is common to use DNA or protein sequences as data since it describes different attributes of the species, and the edges represent a mutation between species. Through sampling from a posterior distribution and utilizing tools like Markov Chain Monte Carlo (MCMC) methods or variational inference, Bayesian phylogenetics offers a robust and nuanced view of evolutionary relationships, integrating multiple sources of information and providing a rigorous statistical foundation for evolutionary hypotheses.

Mainly, two latent variables are regarded as important in Bayesian phylogenetic inference. First, the tree topology, τ, a binary tree with the observations assigned to its leaves. The tree-topology space grows as $(2n-3)!!$ for rooted and $(2n-5)!!$ for unrooted trees, where n is the number of leaves (observations/taxa). Furthermore, each edge, e, of the topology is associated with a positive continuous variable, the branch lengths, $b(e)$. The Cartesian product of discrete and continuous spaces makes inference in phylogenetics a challenging task.

B Conflicting Tree Topologies

Here we construct a realistic scenario where DNA sequences induce conflicting tree-topologies in the posterior that cannot be modelled by the vanilla SBN. On the other hand, they can be captured by VBPI-Mixtures.

It is well-known that DNA data sometimes has conflicting signals. Here we construct a toy example to demonstrate how τ_1 and τ_2 can have higher posterior support than τ_3 and τ_4. For simplicity, we use the connection between a lower parsimony score and a higher likelihood when branch lengths are short (to appear in the appendix). First, consider that the leaves in Fig. 2 have DNA sequences with nucleotides at sites i and j specified in the table below (the first and second columns represent i and j, respectively).

Table 3: Nucleotide assignments to sites i and j in the observations.

Sites	(1)	(2)	(3)	(4)	(5)	(6)
i	A	C	C	A	G	G
j	C	C	A	G	G	A

We compute the parsimony scores for each clade (A, A', B, B') and the cost of joining two clades to form $\tau_1, ..., \tau_4$. The cost of transitioning from one nucleotide (A, C, G, T) to another is 1.

In the first table below, we start by calculating all possible parsimony scores for each clade. Here X→Y denotes the parsimony score, Y, if the nucleotide at site i or j in the ancestral sequence in the root is X. Bolded is the best (lowest) score.

Table 4: Parsimony scores for the four different subtrees

MP	$A = ((1,2),3)$	$B = ((4,5),6)$	$A' = (1,(2,3))$	$B' = (4,(5,6))$
i:th	A→2,**C→1**,G→3,T→3	A→2,C→3,**G→1**,T→3	**A→1**,**C→1**,G→2,T→2	**A→1**,C→2,**G→1**,T→2
j:th	**A→1**,**C→1**,G→2,T→2	**A→1**,C→3,**G→1**,T→3	A→2,**C→1**,G→3,T→3	A→2,C→3,**G→1**,T→3

Table 5: Parsimony scores for all possible tree topologies

MP	$\tau_1(A \wedge B)$	$\tau_2(A' \wedge B')$	$\tau_3(A \wedge B')$	$\tau_4(A' \wedge B)$
i:th	A→4,**C→3**,**G→3**,T→4	**A→2**,C→3,G→3,T→4	**A→3**,**C→3**,**G→3**,T→4	**A→3**,**C→3**,**G→3**,T→4
j:th	**A→2**,C→3,G→3,T→4	A→4,**C→3**,**G→3**,T→4	**A→3**,**C→3**,**G→3**,T→4	**A→3**,**C→3**,**G→3**,T→4

Now we can see that τ_1 and τ_2 give us a better parsimony score, demonstrating that the scenario exemplified in Fig. 2 can occur in biological data when there are conflicting signals.

We direct readers who want to learn more about parsimony scores to Chapter 1 in Felsenstein (2003).

C Gradient Derivation

Here we show the full derivations of the gradients w.r.t. ϕ_i. For completeness, recall that

$$f(x, \mathcal{B}_s^k, \tau_s^k) = \frac{p(\mathcal{B}_s^k, \tau_s^k, X)}{\frac{1}{S}\sum_{j=1}^{S} q_{\psi_j}(\mathcal{B}_s^k|\tau_s^k) q_{\phi_j}(\tau_s^k)}, \tag{12}$$

and $\hat{L}_s^K = \log \frac{1}{K}\sum_{k=1}^{K} f(x, \mathcal{B}_s^k, \tau_s^k)$, where $\mathcal{B}_s^k, \tau_s^k$ are simulated from $q_{\psi_s,\phi_s}(\mathcal{B}, \tau)$.

That is, we are interested in the gradient of Eq. (6) w.r.t. the SBN parameters for one of the mixture components, say i,

$$\nabla_{\phi_i}\mathcal{L}(X; K, S) = \nabla_{\phi_i}\frac{1}{S}\sum_{s=1}^{S}\mathbb{E}_{q_{\psi_s,\phi_s}(\mathcal{B},\tau)}\left[\hat{L}_s^K\right]. \tag{13}$$

There are two cases to take into account in the sum, either $i = s$ or $i \neq s$. Starting with $i = s$ and using the product rule,

$$\begin{aligned}
\nabla_{\phi_i}\frac{1}{S}\mathbb{E}_{\mathcal{B}_i^{1:K},\tau_i^{1:K}\sim q_{\psi_i,\phi_i}(\mathcal{B},\tau)}\left[\hat{L}_i^K\right] &= \nabla_{\phi_i}\frac{1}{S}\sum_{\tau_i^{1:K}} q_{\phi_i}(\tau_i^{1:K})\mathbb{E}_{\mathcal{B}_i^{1:K}\sim q_{\psi_i}(\mathcal{B}|\tau_i^{1:K})}\left[\hat{L}_i^K\right] && (14)\\
&= \frac{1}{S}\sum_{\tau_i^{1:K}}\mathbb{E}_{\mathcal{B}_i^{1:K}\sim q_{\psi_i}(\mathcal{B}|\tau_i^{1:K})}\left[\nabla_{\phi_i}q_{\phi_i}(\tau_i^{1:K})\hat{L}_i^K\right] && (15)\\
&= \frac{1}{S}\sum_{\tau_i^{1:K}}\mathbb{E}_{q_{\psi_i}(\mathcal{B}|\tau_i^{1:K})}\left[\hat{L}_i^K\nabla_{\phi_i}q_{\phi_i}(\tau_i^{1:K}) + q_{\phi_i}(\tau_i^{1:K})\nabla_{\phi_i}\hat{L}_i^K\right]. && (16)
\end{aligned}$$

Recalling the identity that $\nabla_\phi g_\phi(z) = g_\phi(z)\nabla_\phi \log g_\phi(z)$, we start by rewriting the first term inside the expectation in Eq. (16) as

$$\begin{aligned}
\hat{L}_i^K\nabla_{\phi_i}q_{\phi_i}(\tau_i^{1:K}) &= q_{\phi_i}(\tau_i^{1:K})\hat{L}_i^K\nabla_{\phi_i}\log q_{\phi_i}(\tau_i^{1:K}) && (17)\\
&= q_{\phi_i}(\tau_i^{1:K})\hat{L}_i^K\sum_{k=1}^{K}\nabla_{\phi_i}\log q_{\phi_i}(\tau_i^k), && (18)
\end{aligned}$$

and then the second term

$$\begin{aligned}
q_{\phi_i}(\tau_i^{1:K})\nabla_{\phi_i}\hat{L}_i^K &= q_{\phi_i}(\tau_i^{1:K})\nabla_{\phi_i}\log\frac{1}{K}\sum_{k=1}^{K} f(x, \mathcal{B}_i^k, \tau_i^k) && (19)\\
&= q_{\phi_i}(\tau_i^{1:K})\sum_{k=1}^{K}\tilde{w}_i^k\nabla_{\phi_i}\log f(x, \mathcal{B}_i^k, \tau_i^k) && (20)\\
&= -q_{\phi_i}(\tau_i^{1:K})\sum_{k=1}^{K}\tilde{w}_i^k\nabla_{\phi_i}\log\frac{1}{S}\sum_{j=1}^{S} q_{\psi_j}(\mathcal{B}_i^k|\tau_i^k)q_{\phi_j}(\tau_i^k), && (21)
\end{aligned}$$

where $\tilde{w}_i^k = \frac{f(x,\mathcal{B}_i^k,\tau_i^k)}{\sum_{k'=1}^{K} f(x,\mathcal{B}_i^{k'},\tau_i^{k'})}$. Exchanging the two terms in Eq. (16) with Eq. (18) and (21), respectively, we get

$$\nabla_{\phi_i}\frac{1}{S}\mathbb{E}_{q_{\psi_i,\phi_i}(\mathcal{B},\tau)}\left[\hat{L}_i^K\right] = \frac{1}{S}\mathbb{E}_{q_{\psi_i,\phi_i}(\mathcal{B},\tau)}\left[\hat{L}_i^K\sum_{k=1}^{K}\nabla_{\phi_i}\log q_{\phi_i}(\tau_i^k)\right] - \tag{22}$$

$$\frac{1}{S}\mathbb{E}_{q_{\psi_i,\phi_i}(\mathcal{B},\tau)}\Big[\sum_{k=1}^{K}\tilde{w}_i^k\nabla_{\phi_i}\log\frac{1}{S}\sum_{j=1}^{S}q_{\psi_j}(\mathcal{B}_i^k|\tau_i^k)q_{\phi_j}(\tau_i^k)\Big]. \tag{23}$$

For $i \neq s$ we may move the gradient operator into the expectation directly and reuse the derivation of Eq. (21),

$$\begin{aligned}\nabla_{\phi_i}\frac{1}{S}\sum_{s\neq i}\mathbb{E}_{q_{\psi_s,\phi_s}(\mathcal{B},\tau)}\left[\hat{L}_s^K\right] &= \frac{1}{S}\sum_{s\neq i}\mathbb{E}_{q_{\psi_s,\phi_s}(\mathcal{B},\tau)}\left[\nabla_{\phi_i}\hat{L}_s^K\right] \\ &= -\frac{1}{S}\sum_{s\neq i}\mathbb{E}_{q_{\psi_s,\phi_s}(\mathcal{B},\tau)}\Big[\sum_{k=1}^{K}\tilde{w}_s^k\nabla_{\phi_i}\log\frac{1}{S}\sum_{j=1}^{S}q_{\psi_j}(\mathcal{B}_s^k|\tau_s^k)q_{\phi_j}(\tau_s^k)\Big].\end{aligned} \tag{24}$$

Considering both cases, we return to Eq. (13),

$$\begin{aligned}\nabla_{\phi_i}\mathcal{L}(X;K,S) = \frac{1}{S}&\mathbb{E}_{q_{\psi_i,\phi_i}(\mathcal{B},\tau)}\left[\hat{L}_i^K\sum_{k=1}^{K}\nabla_{\phi_i}\log q_{\phi_i}(\tau_i^k)\right] - \\ &\frac{1}{S}\sum_{s=1}^{S}\mathbb{E}_{q_{\psi_s,\phi_s}(\mathcal{B},\tau)}\Big[\sum_{k=1}^{K}\tilde{w}_s^k\nabla_{\phi_i}\log\frac{1}{S}\sum_{j=1}^{S}q_{\psi_j}(\mathcal{B}_s^k|\tau_s^k)q_{\phi_j}(\tau_s^k)\Big],\end{aligned} \tag{25}$$

which is the expression for the gradient we need in order to apply the VIMCO estimator (see Sec. 3.1.2).

D Additional Experimental Results and Implementation Details

D.1 The Two-Level Hierarchical Model

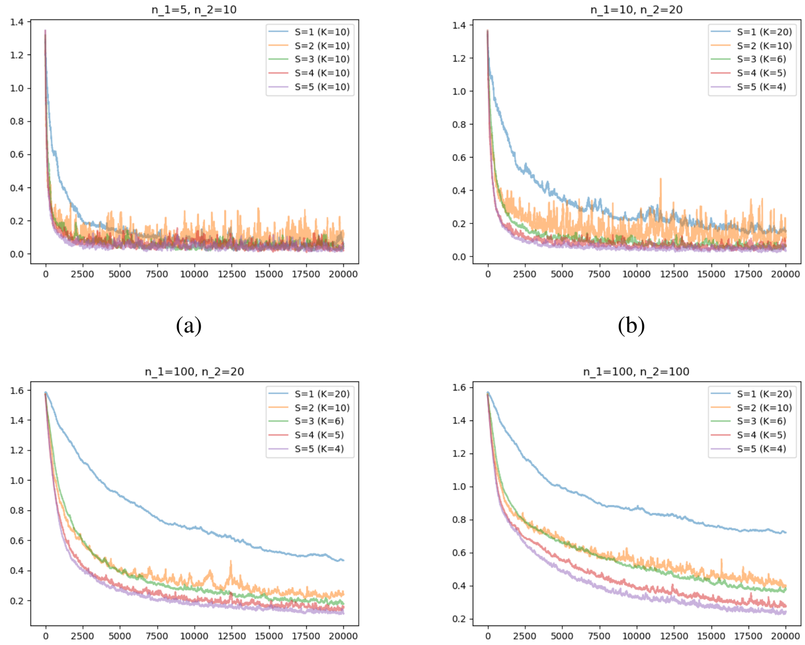
Running a grid search for all five algorithms ($S = 1, ..., 5$), using $n_1 = 5$ and $n_2 = 10$, we found that the learning rates 0.01, 0.1, 0.1, 0.2 and 0.25, respectively, were optimal. That is, these achieved the smallest KL divergences. For larger learning rates, $S = 1$ did not converge, or converged to worse KL divergences. The optimal learning rates found in the grid searches were used in all subsequent experiments.

In Fig. 6, we visualize the KL curves as functions of the number of training iterations. For all configurations of K, n_1 and n_2, the $S = 5$ model performs best. The pattern—$S = 1$ converges slower and to worse KL divergences than $S > 2$—holds also when all models use the same number of importance samples, K (shown in Fig 6a).

D.2 Visualization Details and More Plots

We employed the software as in Whidden & Matsen IV (2015), i.e. rSPR was used to determine the distances between the topologies. Additionally, we adopted the same methodology for cluster creation. This involved assigning the most probable peak to a cluster and subsequently assigning all unassigned trees to the same cluster if their distance from the peak tree was within one standard deviation below the mean distance of all unassigned trees. This iterative process continued until all topologies were assigned, or until eight clusters were reached.

For graph creation, we employ the Graphviz layout known as Scalable Force-Directed Placement (SFDP), in conjunction with the NetworkX library Hagberg et al. (2008). The clusters are represented by different colors, and the size of each node is determined by the normalized sampling frequency in Fig 1, 7. Moreover, edges between the topologies are only displayed if their distance is



(a)

(b)

(c)

(d)

Figure 6: KL curves for the two-level hierarchical model using different configurations of K, n_1 and n_2.

exactly one. It's worth noting that the rSPR distance measure is utilized, which counts the number of changes similar to the approach used in the MCMC method.

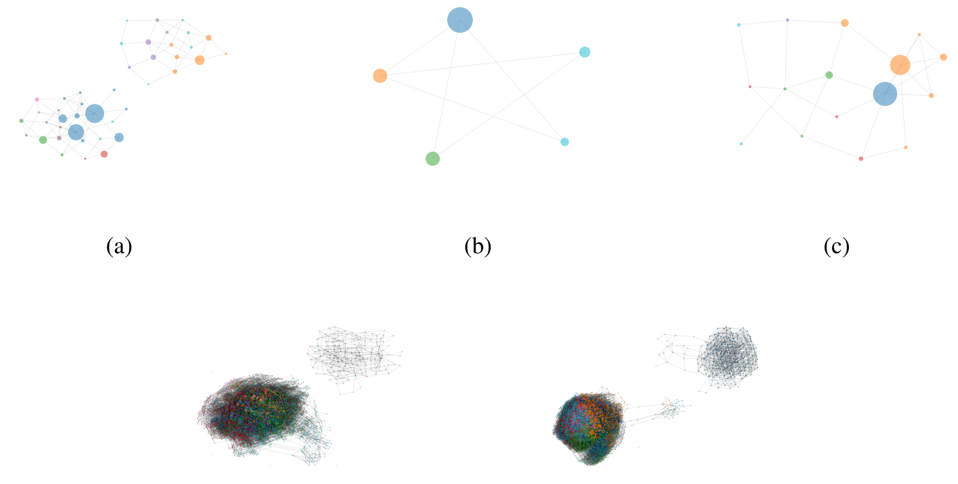
To ensure that the visualization focused on the most credible information, we imposed a constraint by limiting the nodes to the 95% most credible set. This ensured that only the most reliable nodes were included. Additionally, to manage computational resources effectively, we set a maximum limit of 4096 nodes for the graphs.

A similar approach was used for Fig. 5, 8, with the main difference being that we sampled from the components and displayed the joint set of topologies. The colors in this figure were based on which component sampled the topology the most. This representation was considered an approximation of the posterior.

E Limitations

We use S SBNs and branch length models to form our mixture approximations. This introduces a larger number of model parameters. In our current implementation, the training time was prolonged as we did not parallelize the parameter updates of the parameters of the mixture components. This can, however, be done, in order to heavily decrease training time.

Additionally, using shared parameters for the mixture components can also be utilized, if the practitioner is running on a limited memory budget. However, with modern compute engines and laptop computers, this is seldomly an issue. Nonetheless, devising clever modifications to reduce the number of parameters and training times for mixtures in black-box VI is an exciting future research field, out of the scope of this work.



(a)

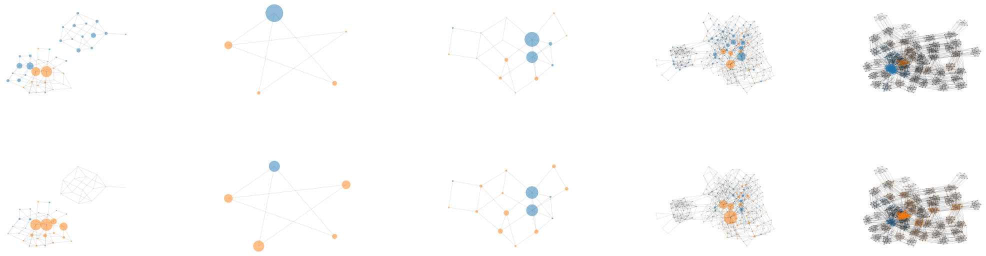
(b)

(c)

(d)

(e)

Figure 7: Visualization of samples from the tree-topology posterior using a 1,000,000,000 iterations long MCMC run on (a) DS1, (b) DS1, (c) DS3, (d) DS5 and (e) DS6. Nodes represent unique tree-topologies and are colored based on cluster assignments, illustrating the multimodality of the tree-topology posterior. More details in Sec. 2.



(a) DS1

(b) DS2

(c) DS3

(d) DS4

(e) DS7

Figure 8: Visualization of a uniformly weighted $S = 2$-component mixture of SBNs on (a) DS1, (b) DS2 and (c) DS3, (d) DS4 and (e) DS7, where each node corresponds to a unique tree-topology. The upper row shows the distribution of five million sampled tree topologies from the first component, where a node, τ, is colored blue if $q_{\phi_1}(\tau) > q_{\phi_2}(\tau)$, or orange otherwise. Vice versa for the lower row. The size of a node is determined by its sampling frequency, which is why nodes with low frequency appear black. The components clearly spread out, exploring different parts of the space.

F BROADER IMPACT

Bayesian phylogenetic inference algorithms are crucial for researchers to reason about uncertainty in their evolutionary findings. Variational inference algorithms provide a compelling alternative to MCMC-based algorithms as a parametric approximation is obtained. This implies that VI, and VBPI specifically, can be used in settings where the applcation of MCMC is less straightforward, for instance in out-of-distribution detection, or evaluation on held-out data. Also, as we have shown in our experiments in this paper, model evaluation can be more robust when using VI over MCMC, resulting in smaller variance of the estimator of the marginal log-likelihood. This is an important feature for downstream tasks.

G Compute Infrastructure

Most computations have been conducted on an AMD EPYC 7742 where two cores have been used per run. Final runtimes are shown in Table 6. The table shows the joint run time for both training and testing. Also worth noting is that multiple mixture components also multiply the number of particles, so the majority of the time increase is due to the Felsenstein pruning algorithm for the likelihood model evaluation, which grows linearly.

Finally, and crucially, the code used was not optimized for run time, and so a wall-clock time is not an apt metric for comparisons.

Table 6: Compute time reported in minutes for fitting the model as well as evaluating the marginal likelihood and continuous estimate of ELBO while training every 5000 iterations using 1000 samples with a single particle

Data	DS1	DS2	DS3	DS4	DS5	DS6	DS7	DS8
VBPI	735.4	813.95	962.67	1087.98	1273.57	1299.55	1606.03	1660.18
$\text{Mix}_{S\,=\,2}$	1984.47	2172.18	2598.17	2888.58	3322.77	3389.5	4127.52	4234.98
$\text{Mix}_{S\,=\,3}$	3224.57	3425.35	4056.07	4493.87	5614.58	5604.43	6705.18	7004.28
VBPI-NF	900.32	953.8	1150.78	1278.27	1585.87	1523.02	1869.23	1974.17
$\text{Mix}_{\text{NF},\,S\,=\,2}$	2385.38	2442.75	2893.15	3300.35	3800.88	3728.75	4647.48	4754.22
$\text{Mix}_{\text{NF},\,S\,=\,3}$	3669.8	3939.73	4576.08	5180.75	6206.52	6220.4	7568.85	7776.37