Variational Bayesian Phylogenetic Inference

Anonymous authors
Paper under double-blind review

Abstract

Bayesian phylogenetic inference is currently done via Markov chain Monte Carlo with simple mechanisms for proposing new states, which hinders exploration efficiency and often requires long runs to deliver accurate posterior estimates. In this paper we present an alternative approach: a variational framework for Bayesian phylogenetic analysis. We approximate the true posterior using an expressive graphical model for tree distributions, called a subsplit Bayesian network, together with appropriate branch length distributions. We train the variational approximation via stochastic gradient ascent and adopt multi-sample based gradient estimators for different latent variables separately to handle the composite latent space of phylogenetic models. We show that our structured variational approximations are flexible enough to provide comparable posterior estimation to MCMC, while requiring less computation due to a more efficient tree exploration mechanism enabled by variational inference. Moreover, the variational approximations can be readily used for further statistical analysis such as marginal likelihood estimation for model comparison via importance sampling. Experiments on both synthetic data and real data Bayesian phylogenetic inference problems demonstrate the effectiveness and efficiency of our methods.

1 Introduction

Bayesian phylogenetic inference is an essential tool in modern evolutionary biology. Given an alignment of nucleotide or amino acid sequences and appropriate prior distributions, Bayesian methods provide principled ways to assess the phylogenetic uncertainty by positing and approximating a posterior distribution on phylogenetic trees (Huelsenbeck et al., 2001). In addition to uncertainty quantification, Bayesian methods enable integrating out tree uncertainty in order to get more confident estimates of parameters of interest, such as factors in the transmission of Ebolavirus (Dudas et al., 2017). Bayesian methods also allow complex substitution models (Lartillot & Philippe, 2004), which are important in elucidating deep phylogenetic relationships (Feuda et al., 2017).

Ever since its introduction to the phylogenetic community in the 1990s, Bayesian phylogenetic inference has been dominated by random-walk Markov chain Monte Carlo (MCMC) approaches (Yang & Rannala, 1997; Mau et al., 1999; Huelsenbeck & Ronquist, 2001). However, this approach is fundamentally limited by the complexities of tree space. A typical MCMC method for phylogenetic inference involves two steps in each iteration: first, a new tree is proposed by randomly perturbing the current tree, and second, the tree is accepted or rejected according to the Metropolis-Hastings acceptance probability. Any such random walk algorithm faces obstacles in the phylogenetic case, in which the high-posterior trees are a tiny fraction of the combinatorially exploding number of trees. Thus, major modifications of trees are likely to be rejected, restricting MCMC tree movement to local modifications that may have difficulty moving between multiple peaks in the posterior distribution (Whidden & Matsen IV, 2015). Although recent MCMC methods for distributions on Euclidean space use intelligent proposal mechanisms such as Hamiltonian Monte Carlo (Neal, 2011), it is not straightforward to extend such algorithms to the composite structure of tree space, which includes both tree topology (discrete object) and branch lengths (continuous positive vector) (Dinh et al., 2017).

Variational inference (VI) is an alternative approximate inference method for Bayesian analysis which is gaining in popularity (Jordan et al., 1999; Wainwright & Jordan, 2008; Blei et al., 2017). Unlike MCMC methods that sample from the posterior, VI seeks the best candidate from a family of tractable distributions that minimizes a statistical distance measure to the target posterior, usually
the Kullback-Leibler (KL) divergence. By reformulating the inference problem into an optimization problem, VI tends to be faster and easier to scale to large data (via stochastic gradient descent) (Blei et al., 2017). However, VI can also introduce a large bias if the variational distribution is insufficiently flexible. The success of variational methods, therefore, largely relies on the existence of appropriate tractable variational distributions and efficient training procedures.

To our knowledge, there have been no previous variational formulations of Bayesian phylogenetic inference. This has been due to the lack of an appropriate family of distributions on phylogenetic trees that can be used as an approximating distribution. However the prospects for variational inference have changed recently with the introduction of subsplit Bayesian networks (SBNs) (Zhang & Matsen IV, 2018), which provide a family of flexible distributions on tree topologies (i.e. trees without branch lengths). SBNs build on previous work (Hohna & Drummond, 2012; Larget, 2013), but in contrast to these previous efforts, SBNs are sufficiently flexible for real Bayesian phylogenetic posteriors (Zhang & Matsen IV, 2018).

In this paper, we develop a general variational inference framework for Bayesian phylogenetics. We show that SBNs, when combined with appropriate approximations for the branch length distribution, can provide flexible variational approximations over the joint latent space of phylogenetic trees with branch lengths. We use recently-proposed unbiased gradient estimators for the discrete and continuous components separately to enable efficient stochastic gradient ascent. We also leverage the similarity of local structures among trees to reduce the complexity of the variational parameterization for the branch length distributions and provide an extension to better capture the between-tree variation. Finally, we demonstrate the effectiveness and efficiency of our methods on both synthetic data and a benchmark of challenging real data Bayesian phylogenetic inference problems.

2 BACKGROUND

Phylogenetic Posterior A phylogenetic tree is described by a tree topology $\tau$ and associated non-negative branch lengths $q$. The tree topology $\tau$ is used to model the evolutionary diversification of the species. It is a bifurcating tree with $N$ leaves, each of which has a label corresponding to one of the observed species. The internal nodes of $\tau$ represent the unobserved characters (e.g. DNA bases) of the ancestral species. A continuous-time Markov model is often used to describe the transition probabilities of the characters along the branches of the tree. A Bayesian approach to phylogenetic analysis requires a likelihood model for sequence evolution through a phylogenetic tree, and prior distributions on the tree topologies and the branch lengths. Let $Y = \{Y_1, Y_2, \ldots, Y_M\} \in \Omega^{N \times M}$ be the observed sequences (with characters in $\Omega$) of length $M$ over $N$ species. The probability of each site observation $Y_i$ is defined as the marginal distribution over the leaves

$$p(Y_i | \tau, q) = \sum_{a^i} \eta(a^i_\rho) \prod_{(u,v) \in E(\tau)} P_{a^i_u a^i_v}(q_{uv})$$

where $\rho$ is the root node (or any internal node if the tree is unrooted and the Markov model is time reversible), $a^i$ ranges over all extensions of $Y_i$ to the internal nodes with $a^i_u$ being the assigned character of node $u$, $E(\tau)$ denotes the set of edges of $\tau$, $P_{ij}(t)$ denotes the transition probability from character $i$ to character $j$ across an edge of length $t$ and $\eta$ is the stationary distribution of the Markov model. Assuming different sites are identically distributed and evolve independently, the likelihood of observing the sequence set $Y$ has the form

$$p(Y | \tau, q) = \prod_{i=1}^M \sum_{a^i} \eta(a^i_\rho) \prod_{(u,v) \in E(\tau)} P_{a^i_u a^i_v}(q_{uv})$$

(1)

The phylogenetic likelihood in equation (1) can be evaluated efficiently through the pruning algorithm (Felsenstein, 2003), also known as the sum-product algorithm in probabilistic graphical models (Strimmer & Moulton, 2000; Koller & Friedman, 2009; Hohna et al., 2014). Given a proper prior distribution with density $p(\tau, q)$ imposed on the tree topologies and the branch lengths, the posterior $p(\tau, q | Y)$ is proportional to the joint density

$$p(\tau, q | Y) \propto p(Y | \tau, q)p(\tau, q)$$

where $p(Y)$ is the intractable normalizing constant.
Subsplit Bayesian Networks  We now review subsplit Bayesian networks [Zhang & Matsen IV, 2018], powerful probabilistic models that can provide a rich family of flexible distributions on tree topologies. Let \( \mathcal{X} \) be the set of leaf labels. We call a nonempty subset of \( \mathcal{X} \) a clade. Let \( \succ \) be a total order on clades (e.g., lexicographical order). A subsplit \((W, Z)\) of a clade \( X \) is an ordered pair of disjoint subclades of \( X \) such that \( W \cup Z = X \) and \( W \succ Z \). A subsplit Bayesian network \( B_X \) on a leaf set \( \mathcal{X} \) of size \( N \) is a Bayesian network whose nodes take on subsplit values or singleton clade values that represent the local topological structure of trees (Figure 1). Following the splitting processes (see the solid dark subgraphs in Figure 1 middle right panel), rooted trees have unique subsplit decompositions and hence can be uniquely represented as compatible SBN assignments (Figure 1). Given the subsplit decomposition of a rooted tree \( \tau = \{s_1, s_2, \ldots\} \), where \( s_1 \) is the root subsplit and \( \{s_i\}_{i \succ 1} \) are other local subsplits, the SBN tree probability takes the following form

\[
p_{\text{sbn}}(T = \tau) = p(S_1 = s_1) \prod_{i > 1} p(S_i = s_i | S_{\pi_i} = s_{\pi_i})
\]

where \( S_i \) denotes the subsplit- or singleton-clade-valued random variables at node \( i \) and \( \pi_i \) is the index set of the parents of \( S_i \). The Bayesian network formulation of SBNs enjoys many benefits: i) flexibility. The expressiveness of SBNs is freely adjustable by changing the dependency structures between nodes, allowing for a wide range of flexible distributions; ii) normality. SBN-induced distributions are all naturally normalized if the associated conditional probability tables (CPTs) are consistent, which is a common property of Bayesian networks. The SBN framework also generalizes to unrooted trees, which are the most common type of trees in phylogenetics. Concretely, unrooted trees can be viewed as rooted trees with unobserved roots. Marginalizing out the unobserved root node \( S_1 \), we have the SBN probability estimates for unrooted trees

\[
p_{\text{sbn}}(T^u = \tau) = \sum_{s_1 \sim \tau} p(S_1 = s_1) \prod_{i > 1} p(S_i = s_i | S_{\pi_i} = s_{\pi_i})
\]

where \( \sim \) means all root subsplits that are compatible with \( \tau \).

To reduce model complexity and encourage generalization, the same set of CPTs for parent-child subsplit pairs is shared across the SBN network, regardless of their locations. Similar to weight sharing used in convolutional networks (LeCun et al., 1998) for detecting translationally-invariant structure of images (e.g., edges, corners), this heuristic parameter sharing used in SBNs is for identifying conditional splitting patterns of phylogenetic trees. Prior to this work, SBNs were introduced as an extension of previous work (Holmes & Drummond, 2012; Larget, 2013) to further improve the posterior estimation of phylogenetic trees based on MCMC samples. See [Zhang & Matsen IV] (2018) for more detailed discussion on SBNs.

3 Variational Phylogenetic Inference via SBNs

The flexible and tractable tree topology distributions provided by SBNs also serve as an essential building block to perform variational inference [Jordan et al., 1999] for phylogenetics. Suppose that
we have a family of approximate distributions \( Q_{\phi}(\tau) \) (e.g., SBNs) over phylogenetic tree topologies, where \( \phi \) denotes the corresponding variational parameters (e.g., CPTs for SBNs). For each tree \( \tau \), we posit another family of densities \( Q_{\psi}(q|\tau) \) over the branch lengths, where \( \psi \) is the branch length variational parameters. We then combine these distributions and use the product

\[
Q_{\phi,\psi}(\tau, q) = Q_{\phi}(\tau)Q_{\psi}(q|\tau)
\]

as our variational approximation. Inference now amounts to finding the member of this family that minimizes the Kullback-Leibler (KL) divergence to the exact posterior,

\[
\phi^*, \psi^* = \arg \min_{\phi, \psi} D_{KL}(Q_{\phi,\psi}(\tau, q)\| p(\tau, q|Y))
\]

which is equivalent to maximizing the evidence lower bound (ELBO),

\[
L(\phi, \psi) = \mathbb{E}_{Q_{\phi,\psi}(\tau, q)} \log \left( \frac{p(Y|\tau, q)p(\tau, q)}{Q_{\phi}(\tau)Q_{\psi}(q|\tau)} \right) \leq \log p(Y).
\]

As the ELBO is based on a single-sample estimate of the evidence, it heavily penalizes samples that fail to explain the observed sequences. As a result, the variational approximation tends to cover only the high-probability areas of the true posterior. This effect can be minimized by averaging over \( K > 1 \) samples when estimating the evidence (Burda et al., 2016; Mnih & Rezende, 2016), which leads to tighter lower bounds

\[
L^K(\phi, \psi) = \mathbb{E}_{Q_{\phi,\psi}(\tau^K, q^K)} \log \left( \frac{1}{K} \sum_{i=1}^{K} p(Y|\tau^i, q^i)p(\tau^i, q^i)}{Q_{\phi}(\tau^i)Q_{\psi}(q^i|\tau^i)} \right) \leq \log p(Y)
\]

where \( Q_{\phi,\psi}(\tau^K, q^K) \equiv \prod_{i=1}^{K} Q_{\phi,\psi}(\tau^i, q^i) \); the tightness of the lower bounds improves as the number of samples \( K \) increases (Burda et al., 2016). We will use multi-sample lower bounds in the sequel and refer to them as lower bounds for short.

3.1 Variational Parameterization

The CPTs in SBNs are, in general, associated with all possible parent-child subsplit pairs. Therefore, a full parameterization requires a combinatorially increasing number of parameters. In practice, however, we can find a sufficiently large subsplit support of CPTs (i.e., where the associated conditional probabilities are allowed to be nonzero) that covers favorable subsplit pairs from trees in the high-probability areas of the true posterior. Compared to estimating the true posterior probability of the trees that often involves an expensive long MCMC run, locating the high-probability areas of the true posterior probability are diffuse), we use a \( \phi \)-based structured parameterization that exploits the similarity of local structures among trees to reduce the number of parameters. A split is a bipartition \( (X_1, X_2) \) of the leaf labels \( \mathcal{X} \) (i.e. \( X_1 \cup X_2 = \mathcal{X}, X_1 \cap X_2 = \emptyset \)) and each edge \( e \) of a phylogenetic tree \( \tau \) represents a split, the bipartition that consists of the leaf labels from both sides of the edge. Note that a split can be viewed as a root subsplit. We then assign \( \mu(\cdot, \cdot), \sigma(\cdot, \cdot) \) for each split \((\cdot, \cdot)\) in \( S_\tau \). We denote the corresponding split of edge \( e \) of tree \( \tau \) as \( e/\tau \).
\textbf{A Simple Independent Approximation} \quad Given a phylogenetic tree $\tau$, we use a simple model that assumes the branch lengths for the edges of the tree are independently distributed. The approximate density $Q_{\psi}(q|\tau)$, therefore, has the form

\begin{equation}
Q_{\psi}(q|\tau) = \prod_{e \in E(\tau)} p^{\text{Lognormal}}(q_e | \mu(e, \tau), \sigma(e, \tau)), \quad \mu(e, \tau) = \psi^{\mu}_{e/\tau}, \quad \sigma(e, \tau) = \psi^{\sigma}_{e/\tau}
\end{equation}

where $p^{\text{Lognormal}}(\cdot | \mu, \sigma)$ is the probability density function of $\text{Lognormal}(\mu, \sigma^2)$.

\textbf{Capturing Between-Tree Branch Length Variation} \\
We assign additional parameters for each primary subsplit pair. Denoting the primary subsplit pair(s) of edge $e$ in tree $\tau$ as $e/\tau$, we then simply sum all variational parameters associated with $e$ to form the mean and variance parameters for the corresponding branch length (Figure 2):

\begin{equation}
\mu(e, \tau) = \psi^{\mu}_{e/\tau} + \sum_{s \in e/\tau} \psi^{\mu}_s, \quad \sigma(e, \tau) = \psi^{\sigma}_{e/\tau} + \sum_{s \in e/\tau} \psi^{\sigma}_s.
\end{equation}

This modifies the density in equation 4 by adding contributions from primary subsplit pairs and hence allows for more flexible between-tree approximations. Note that the above structured parameterizations of branch length distributions also enable joint learning across tree topologies.

\textbf{3.2 Stochastic Gradient Estimators and The VBPI Algorithm} \\
In practice, the lower bound is usually maximized via stochastic gradient ascent (SGA). However, the naive stochastic gradient estimator obtained by differentiating the lower bound has very large variance and is impractical for our purpose. Fortunately, various variance reduction techniques have been introduced in recent years including the control variate (Paisley et al., 2012) [Ranganath et al., 2014] [Mnih & Gregor, 2014] [Mnih & Rezende, 2016] for general latent variables and the reparameterization trick (Kingma & Welling, 2014) for continuous latent variables. In the following, we apply these techniques to different components of our latent variables and derive efficient gradient estimators with much lower variance, respectively. In addition, we also consider a stable gradient estimator based on an alternative variational objective.

\textbf{The VIMCO Estimator} \quad Let $f_{\phi, \psi}(\tau, q) = \frac{p^{Y}/q^{\psi}(\tau,q)}{Q_{\phi}(\tau)|Q_{\psi}(q|\tau)}$. The stochastic lower bound with $K$ samples is $\hat{L}^K(\phi, \psi) = \log \left( \frac{1}{K} \sum_{i=1}^{K} f_{\phi, \psi}(\tau^i, q^i) \right)$. Mnih & Rezende (2016) propose a localized learning signal strategy that significantly reduces the variance of the naive gradient estimator by utilizing the independence between the multiple samples and the regularity of the learning signal, which estimates the gradient as follows (derivation in Appendix A):

\begin{equation}
\nabla_{\phi} L^K(\phi, \psi) = \mathbb{E}_{Q_{\phi, \psi}(\tau^{1:K}, q^{1:K})} \sum_{j=1}^{K} \left( \hat{L}^K_{j-1}(\phi, \psi) - \hat{w}^j \right) \nabla_{\phi} \log Q_{\phi}(\tau^j)
\end{equation}
where
\[
\hat{L}_{i,j-1}^K(\phi, \psi) := \hat{L}_j(\phi, \psi) - \log \frac{1}{K} \left( \sum_{i \neq j} f_{\phi, \psi}(\tau^i, q^j) + \hat{f}_{\phi, \psi}(\tau^{-j}, q^{-j}) \right)
\]
is the per-sample local learning signal, with \(\hat{f}_{\phi, \psi}(\tau^{-j}, q^{-j})\) being some estimate of \(f_{\phi, \psi}(\tau^i, q^j)\) for sample \(j\) using the rest of samples (e.g., the geometric mean), and \(\hat{w}^j = \frac{\sum_{i=1}^K f_{\phi, \psi}(\tau^i, q^j)}{\sum_{i=1}^K f_{\phi, \psi}(\tau^i, q^i)}\) is the self-normalized importance weight. This gives the following VIMCO estimator
\[
\nabla_\phi L^K(\phi, \psi) \simeq \sum_{j=1}^K \left( \hat{L}_{i,j-1}^K(\phi, \psi) - \hat{w}^j \right) \nabla_\phi \log Q(\tau^j) \text{ with } \tau^j, q^j \overset{iid}{\sim} Q(\tau, q) \tag{6}
\]

**The Reparameterization Trick** The above VIMCO estimator also works for the branch length gradient. However, as branch lengths are continuous latent variables, we can use an alternative approach, the reparameterization trick, to estimate the gradient. Note that the Log-normal distribution has a simple reparameterization: \(q \sim \text{Lognormal}(\mu, \sigma^2) \iff q = \exp(\mu + \sigma \epsilon), \epsilon \sim N(0, 1)\), we can rewrite the lower bound as follows
\[
L^K(\phi, \psi) = \mathbb{E}_{Q(\tau, \epsilon) \mid \epsilon \sim \mathcal{N}(0, 1)} \left( \frac{1}{K} \sum_{j=1}^K p(Y_j \mid \tau^j, g_{\phi}(\epsilon_j \mid \tau^j)) p(\tau^j, g_{\phi}(\epsilon_j \mid \tau^j)) \right) \tag{3}
\]
where \(g_{\phi}(\epsilon \mid \tau) = \exp(\mu_{\phi, \tau} + \sigma_{\phi, \tau} \epsilon)\). Then the gradient of the lower bound w.r.t. \(\psi\) is (see Appendix A for derivation)
\[
\nabla_\psi L^K(\phi, \psi) = \mathbb{E}_{Q(\tau^j, \epsilon) \mid \epsilon \sim \mathcal{N}(0, 1)} \left( \sum_{j=1}^K \hat{w}^j \nabla_\psi \log f_{\phi, \psi}(\tau^j, g_{\phi}(\epsilon_j \mid \tau^j)) \right) \tag{7}
\]
where \(\hat{w}^j = \sum_{i=1}^K f_{\phi, \psi}(\tau^i, g_{\phi}(\epsilon_j \mid \tau^i)) \sum_{i=1}^K f_{\phi, \psi}(\tau^i, g_{\phi}(\epsilon_j \mid \tau^i))\) is the same normalized importance weight as in equation equation5. Therefore, we can form the Monte Carlo estimator of the gradient
\[
\nabla_\psi L^K(\phi, \psi) \simeq \sum_{j=1}^K \hat{w}^j \nabla_\psi \log f_{\phi, \psi}(\tau^j, g_{\phi}(\epsilon_j \mid \tau^j)) \text{ with } \tau^j, \epsilon^j \overset{iid}{\sim} p(\epsilon) \tag{8}
\]

**Self-normalized Importance Sampling Estimator** In addition to the standard variational formulation equation2 one can reformulate the optimization problem by minimizing the reversed KL divergence, which is equivalent to maximizing the likelihood of the variational approximation \(Q(\tau, q)\), where \(\phi^*, \psi^* = \arg \max_{\phi, \psi} \hat{L}(\phi, \psi), \hat{L}(\phi, \psi) = \mathbb{E}_{p(\tau, q \mid Y)} \log Q(\tau, q)\). \tag{9}

We can use an importance sampling estimator to compute the gradient of the objective
\[
\nabla_\phi \hat{L}(\phi, \psi) = \mathbb{E}_{p(\tau, q \mid Y)} \nabla_\phi \log Q(\phi, \psi) = \frac{1}{p(Y)} \mathbb{E}_{Q(\phi, \psi) \mid \epsilon \sim \mathcal{N}(0, 1)} \frac{p(Y, \epsilon \mid \tau, q) p(\tau, q)}{Q(\tau) Q(\epsilon \mid \tau)} \nabla_\phi \log Q(\tau)
\]
\[
\simeq \sum_{j=1}^K \hat{w}^j \nabla_\phi \log Q(\tau^j) \text{ with } \tau^j, q^j \overset{iid}{\sim} Q(\tau, q) \tag{10}
\]
with the same importance weights \(\hat{w}^j\) as in equation5 (a detailed derivation can be found in Appendix A). This can be viewed as a multi-sample generalization of the wake-sleep algorithm (Hinton et al., 1995) and was first used in the reweighted wake-sleep algorithm (Bornschein & Bengio, 2015) for training deep generative models. We therefore call the gradient estimator in equation10 the RWS estimator. Like the VIMCO estimator, the RWS estimator also provides gradients for branch lengths. However, we find in practice that equation10 that uses the reparameterization trick is more useful and often leads to faster convergence, although it uses a different optimization objective. A better understanding of this phenomenon would be an interesting subject of future research.

All stochastic gradient estimators introduced above can be used in conjunction with stochastic optimization methods such as SGA or some of its adaptive variants (e.g., Adam (Kingma & Ba, 2015)) to maximize the lower bounds. See algorithm1 in Appendix B for a basic variational Bayesian phylogenetic inference (VBPI) approach.
Figure 3: Comparison of multi-sample objective on approximating a challenging distribution over unrooted phylogenetic trees with 8 leaves using VIMCO and RWS gradient estimators. **Left:** Evidence lower bound. **Middle:** KL divergence. **Right:** Variational approximations vs ground truth probabilities. The number in brackets specifies the number of samples used in the training objective.

4 Experiments

Throughout this section we evaluate the effectiveness and efficiency of our variational framework for inference over phylogenetic trees. The simplest SBN (the one with a full and complete binary tree structure) is used for the phylogenetic tree topology variational distribution; we have found it to provide sufficiently accurate approximation. For real datasets, we estimate the CPT supports from ultrafast maximum likelihood phylogenetic bootstrap trees using UFBoot (Minh et al., 2013), which is a fast approximate bootstrap method based on efficient heuristics. We compare the performance of the VIMCO estimator and the RWS estimator with different variational parameterizations for the branch length distributions, while varying the number of samples in the training objective to see how these affect the quality of the variational approximations. For VIMCO, we use Adam for stochastic gradient ascent with learning rate 0.001 (Kingma & Ba, 2015). For RWS, we also use AMSGrad (Sashank et al., 2018), a recent variant of Adam, when Adam is unstable. Results were collected after 200,000 parameter updates. The KL divergences reported are over the discrete collection of phylogenetic tree structures, from trained SBN distribution to the ground truth, and a low KL divergence means a high quality approximation of the distribution of trees.

4.1 Simulated Scenarios

To empirically investigate the representative power of SBNs to approximate distributions on phylogenetic trees under the variational framework, we first conduct experiments on a simulated setup. We use the space of unrooted phylogenetic trees with 8 leaves, which contains 10395 unique trees in total. Given an arbitrary order of trees, we generate a target distribution \( p_0(\tau) \) by drawing a sample from the symmetric Dirichlet distributions \( \text{Dir}(\beta_1) \) of order 10395, where \( \beta \) is the concentration parameter. The target distribution becomes more diffuse as \( \beta \) increases; we used \( \beta = 0.008 \) to provide enough information for inference while allowing for adequate diffusion in the target. Note that there are no branch lengths in this simulated model and the lower bound is

\[
L^K(\phi) = E_{Q(\tau^1:K)} \log \left( \frac{1}{K} \sum_{i=1}^{K} \frac{p_0(\tau^i)}{Q(\tau^i)} \right) \leq 0
\]

with the exact evidence being \( \log(1) = 0 \). We then use both the VIMCO and RWS estimators to optimize the above lower bound based on 20 and 50 samples \( (K) \). We use a slightly larger learning rate (0.002) in AMSGrad for RWS. In addition to the lower bound estimate, KL divergence from the variational approximation to the target distribution is also provided to measure the performance of different methods.

Figure 3 shows the empirical performance of different methods. From the left plot, we see that the lower bounds converge rapidly and the gaps between lower bounds and the exact model evidence are close to zero, demonstrating the expressive power of SBNs on phylogenetic tree probability estimations. The evolution of KL divergences (middle plot) is consistent with the lower bounds. All methods benefit from using more samples, with VIMCO performing better in the end and RWS
learning slightly faster at the beginning. The slower start of VIMCO is partly due to the regularization term in the lower bounds, which turns out to be beneficial for the overall performance since the regularization encourages the diversity of the variational approximation and leads to more sufficient exploration in the starting phase, similar to the exploring starts (ES) strategy in reinforcement learning (Sutton & Barto, 1998). The right plot compares the variational approximations obtained by VIMCO and RWS, both with 50 samples, to the ground truth. We see that VIMCO slightly underestimates trees in high-probability areas as a result of the regularization effect. While RWS provides better approximations for trees in high-probability areas, it tends to underestimate trees in low-probability areas which deteriorates the overall performance. We expect the biases in both approaches to be alleviated with more samples.

4.2 Real Data Phylogenetic Posterior Estimation

In the second set of experiments we evaluate the proposed variational Bayesian phylogenetic inference (VBPI) algorithms at estimating unrooted phylogenetic tree posteriors on 8 real datasets commonly used to benchmark phylogenetic MCMC methods (Lakner et al., 2008; Hohna & Drummond, 2012; Larget, 2013; Whidden & Matsen IV, 2015) (Table 1). We concentrate on the most challenging part of the phylogenetic model: joint learning of the tree topologies and the branch lengths. We assume a uniform prior on the tree topology, an i.i.d. exponential prior (Exp(10)) for the branch lengths and the simple Jukes & Cantor (1969) substitution model. We consider two different variational parameterizations for the branch length distributions as introduced in section 3.1. In the first case, we use the simple split-based parameterization that assigns parameters to the splits associated with the edges of the trees. In the second case, we assign additional parameters for the primary subsplit pairs (PSP) to better capture the between-tree variation. We form our ground truth posterior from an extremely long MCMC run of 100000 iterations. We train the variational approximations with VIMCO and RWS using 10 and 20 sample objectives and use Adam with learning rate 0.001 for both methods.

Figure 4: Performance on DS1. **Left:** KL divergence for methods that use the simple split-based parameterization for the branch length distributions. **Middle:** KL divergence for methods that use PSP. **Right:** Per-tree marginal likelihood estimation (in nats): VBPI vs GSS. The number in brackets specifies the number of samples used in the training objective. MCMC results are averaged over 10 independent runs. The results for VBPI were obtained using 1000 samples and the error bar shows one standard deviation over 100 independent runs.

Although we use larger learning rates for RWS in this experiment, we found RWS generally learns slightly faster than VIMCO at the beginning. See Figure 4 for the real data phylogenetic inference problems in section 4.2 where we use Adam with learning rate 0.001 for both methods.
Table 1: Data sets used for variational phylogenetic posterior estimation, and marginal likelihood estimates of different methods across datasets. The marginal likelihood estimates of all variational methods are obtained by importance sampling using 1000 samples. We run stepping-stone in MrBayes using default settings with 4 chains for 10,000,000 iterations and sampled every 100 iterations. The results are averaged over 10 independent runs with standard deviation in brackets.

<table>
<thead>
<tr>
<th>DATA SET</th>
<th>REFERENCE</th>
<th>(ETA, PISIES)</th>
<th>MARGINAL LIKELIHOOD (NATS)</th>
</tr>
</thead>
<tbody>
<tr>
<td>DS1</td>
<td>Hedges et al. (1990)</td>
<td>(27, 1949)</td>
<td>-7108.43(0.26) -7108.35(0.21) -7108.41(0.14) -7108.42(0.10) -7108.42(0.18)</td>
</tr>
<tr>
<td>DS2</td>
<td>Hedges et al. (1997)</td>
<td>(29, 2529)</td>
<td>-26367.73(0.08) -26367.70(0.10) -26367.70(0.10) -26367.70(0.10) -26367.70(0.10)</td>
</tr>
<tr>
<td>DS3</td>
<td>van Vleuten et al. (2014)</td>
<td>(36, 1812)</td>
<td>-33735.06(0.11) -33735.06(0.11) -33735.06(0.11) -33735.06(0.11) -33735.06(0.11)</td>
</tr>
<tr>
<td>DS4</td>
<td>Hedges et al. (2001)</td>
<td>(50, 378)</td>
<td>-8214.74(0.38) -8214.74(0.38) -8214.74(0.38) -8214.74(0.38) -8214.74(0.38)</td>
</tr>
<tr>
<td>DS5</td>
<td>van Vleuten et al. (2014)</td>
<td>(50, 378)</td>
<td>-6723.70(0.65) -6723.70(0.65) -6723.70(0.65) -6723.70(0.65) -6723.70(0.65)</td>
</tr>
<tr>
<td>DS6</td>
<td>Zhang &amp; Rambout (2011)</td>
<td>(50, 378)</td>
<td>-37332.00(0.43) -37332.00(0.43) -37332.00(0.43) -37332.00(0.43) -37332.00(0.43)</td>
</tr>
<tr>
<td>DS7</td>
<td>Hedges et al. (1999)</td>
<td>(59, 1824)</td>
<td>-37332.00(0.43) -37332.00(0.43) -37332.00(0.43) -37332.00(0.43) -37332.00(0.43)</td>
</tr>
<tr>
<td>DS8</td>
<td>Hedges et al. (1999)</td>
<td>(64, 1008)</td>
<td>-8653.34(0.55) -8651.54(0.80) -8650.63(0.42) -8650.55(0.46) -8649.88(1.75)</td>
</tr>
</tbody>
</table>

Figure 4 (left and middle plots) shows the resulting KL divergence to the ground truth on DS1 as a function of the number of parameter updates. The results for methods that adopt the simple split-based parameterization of variational branch length distributions are shown in the left plot. We see that the performance of all methods improves significantly as the number of samples is increased. As before, RWS learns faster at the beginning while VIMCO performs better in the end. The middle plot, containing the results using PSP for variational parameterization, clearly indicates that a better modeling of between-tree variation of the branch length distributions is beneficial for all method / number of samples combinations. Specifically, PSP enables more flexible branch length distributions across trees which makes the learning task much easier, evidenced by the considerably smaller gaps between the methods. To benchmark the learning efficiency of VBPI, we also compare to MrBayes 3.2.5 [Ronquist et al., 2012], a standard MCMC implementation. We run MrBayes with 4 chains and 10 runs for two million iterations, sampling every 100 iterations. For each run, we compute the KL divergence to the ground truth every 50000 iterations with the first 25% discarded as burn-in. For a relatively fair comparison (in terms of the number of likelihood evaluations), we compare 10 (i.e. 2·4) times the number of MCMC iterations with the number of 20-sample objective VBPI iterations. Although MCMC converges faster at the start, we see that VBPI methods (especially those with PSP) quickly surpass MCMC and arrive at good approximations with much less computation. This is because VBPI iteratively updates the approximate distribution of trees (e.g., SBNs) which in turn allows guided exploration in the phylogenetic tree space. As a commonly used summarization of tree samples, we also compare the majority-rule consensus trees obtained from VBPI and the ground truth MCMC run (Figure 5 in Appendix D), and find they are identical.

The variational approximations provided by VBPI can be readily used to perform importance sampling for phylogenetic inference (see more details in Appendix C). Thanks to the structured variational parameterization for the branch length distributions, VBPI can provide variational approximations on branch lengths for any tree that is covered by the subsplit support. We, therefore, can use them to estimate the marginal likelihood of trees via importance sampling. The right plot of figure 4 compares VBPI using VIMCO with 20 samples and PSP to the state-of-the-art generalized stepping-stone (GSS) [Fan et al., 2011] algorithm for estimating the marginal likelihood of trees in the 95% credible set of DS1. For GSS, we use 50 power posteriors and for each power posterior we run 1,000,000 MCMC iterations, sampling every 1000 iterations with the first 10% discarded as burn-in. The reference distribution for GSS was obtained from an independent Gamma approximation using the maximum a posterior estimate. We see that the estimates given by VBPI closely match GSS estimates and have small variance in general. Similarly, we can use VBPI to estimate the marginal likelihood of the data (i.e., the model evidence) via importance sampling. Table 1 shows the marginal likelihood estimates using different VIMCO approximations and one of the state-of-the-art methods, the stepping-stone (SS) algorithm [Xie et al., 2011], which unlike GSS is able to also integrate out tree topologies. For each data set, all methods provide estimates for the same marginal likelihood, with better approximation leading to lower variance. We see that VBPI using 1000 samples is already competitive with SS using 100000 samples and provides estimates with much less variance (hence more reproducible and reliable). Again, the extra flexibility enabled by PSP alleviates the demand for larger number of samples used in the training objective, making it possible to achieve high quality variational approximations with less samples.
5 DISCUSSION

In this work we introduced VBPI, a general variational framework for Bayesian phylogenetic inference. By combining subsplit Bayesian networks, a recent framework that provides flexible distributions of trees, and efficient structured parameterizations for branch length distributions, VBPI exhibits guided exploration (enabled by SBNs) in tree space and provides competitive performance to MCMC methods with less computation. Moreover, variational approximations provided by VBPI can be readily used for further statistical analysis such as marginal likelihood estimation for model comparison via importance sampling, which, compared to MCMC based methods, dramatically reduces the cost at test time. We report promising numerical results demonstrating the effectiveness and efficiency of VBPI on a benchmark of real data Bayesian phylogenetic inference problems.

When the data are weak and posteriors are diffuse, support estimation of CPTs becomes challenging. However, compared to classical MCMC approaches in phylogenetics that need to traverse the enormous support of posteriors on complete trees to accurately evaluate the posterior probabilities, the SBN parameterization in VBPI has a natural advantage in that it alleviates this issue by factorizing the uncertainty of complete tree topologies into local structures.

Many topics remain for future work: constructing more flexible approximations for the branch length distributions (e.g., using normalizing flow [Rezende & Mohamed, 2015] for within-tree approximation and deep networks for the modeling of between-tree variation), deeper investigation of support estimation approaches in different data regimes, and efficient training algorithms for general variational inference on discrete / structured latent variables.

REFERENCES


We start with the lower bound. In this section we will derive the gradient for the multi-sample objectives introduced in section 3.

Using the product rule and noting that
\[ \nabla \phi \psi \left( \tau^j, q^j \right) = \nabla \phi \log f_{\phi, \psi} \left( \tau^j, q^j \right) = -\nabla \phi \log Q_{\phi} \left( \tau^j \right), \]

we have
\[ \nabla \phi L^K \left( \phi, \psi \right) = \mathbb{E}_{Q_{\phi, \psi} \left( \tau^{1:K}, q^{1:K} \right)} \nabla \phi \log \left( \frac{1}{K} \sum_{j=1}^{K} f_{\phi, \psi} \left( \tau^j, q^j \right) \right) + \mathbb{E}_{Q_{\phi, \psi} \left( \tau^{1:K}, q^{1:K} \right)} \sum_{j=1}^{K} \nabla \phi Q_{\phi} \left( \tau^j \right) \log \left( \frac{1}{K} \sum_{j=1}^{K} f_{\phi, \psi} \left( \tau^j, q^j \right) \right) \]

\[ = \mathbb{E}_{Q_{\phi, \psi} \left( \tau^{1:K}, q^{1:K} \right)} \sum_{j=1}^{K} f_{\phi, \psi} \left( \tau^j, q^j \right) \nabla \phi \log f_{\phi, \psi} \left( \tau^j, q^j \right) + \mathbb{E}_{Q_{\phi, \psi} \left( \tau^{1:K}, q^{1:K} \right)} \sum_{j=1}^{K} \log \left( \frac{1}{K} \sum_{j=1}^{K} f_{\phi, \psi} \left( \tau^j, q^j \right) \right) \nabla \phi \log Q_{\phi} \left( \tau^j \right) \]

This gives the naive gradient of the lower bound w.r.t. \( \phi \).

Using the reparameterization trick, the lower bound has the form
\[ L^K \left( \phi, \psi \right) = \mathbb{E}_{Q_{\phi, \epsilon} \left( \tau^{1:K}, \epsilon^{1:K} \right)} \log \left( \frac{1}{K} \sum_{j=1}^{K} \frac{p_Y \left( \tau^j, g_{\phi} \left( \epsilon^j | \tau^j \right) \right) p \left( \tau^j, g_{\phi} \left( \epsilon^j | \tau^j \right) \right)}{Q_{\phi} \left( \tau^j \right) Q_{\psi} \left( \epsilon^j | \tau^j \right) | \tau^j} \right) \]

\[ = \mathbb{E}_{Q_{\phi, \epsilon} \left( \tau^{1:K}, \epsilon^{1:K} \right)} \log \left( \frac{1}{K} \sum_{j=1}^{K} f_{\phi, \psi} \left( \tau^j, g_{\phi} \left( \epsilon^j | \tau^j \right) \right) \right) \]

Since \( \psi \) is not involved in the distribution with respect to which we take expectation,
\[ \nabla \psi L^K \left( \phi, \psi \right) = \mathbb{E}_{Q_{\phi, \epsilon} \left( \tau^{1:K}, \epsilon^{1:K} \right)} \nabla \psi \log \left( \frac{1}{K} \sum_{j=1}^{K} f_{\phi, \psi} \left( \tau^j, g_{\phi} \left( \epsilon^j | \tau^j \right) \right) \right) \]

\[ = \mathbb{E}_{Q_{\phi, \epsilon} \left( \tau^{1:K}, \epsilon^{1:K} \right)} \sum_{j=1}^{K} f_{\phi, \psi} \left( \tau^j, g_{\phi} \left( \epsilon^j | \tau^j \right) \right) \nabla \psi \log f_{\phi, \psi} \left( \tau^j, g_{\phi} \left( \epsilon^j | \tau^j \right) \right) \]

\[ = \mathbb{E}_{Q_{\phi, \epsilon} \left( \tau^{1:K}, \epsilon^{1:K} \right)} \sum_{j=1}^{K} \tilde{u}^j \nabla \psi \log f_{\phi, \psi} \left( \tau^j, g_{\phi} \left( \epsilon^j | \tau^j \right) \right). \]
Next, we derive the gradient of the multi-sample likelihood objective used in RWS
\[ \hat{L}(\phi, \psi) = \mathbb{E}_{p(\tau,q|Y)} \log Q_{\phi,\psi}(\tau, q). \]
Again, \( p(\tau, q|Y) \) is independent of \( \phi, \psi \), and we have
\[
\nabla_\phi \hat{L}(\phi, \psi) = \mathbb{E}_{p(\tau,q|Y)} \nabla_\phi \log Q_{\phi,\psi}(\tau, q) \\
= \mathbb{E}_{Q_{\phi,\psi}(\tau,q)} \frac{p(\tau, q|Y)}{Q_{\phi}(\tau)Q_{\psi}(q|\tau)} \nabla_\phi \log Q_{\phi,\psi}(\tau, q) \\
= \frac{1}{p(Y)} \mathbb{E}_{Q_{\phi,\psi}(\tau,q)} f_{\phi,\psi}(\tau, q) \nabla_\phi \log Q_{\phi}(\tau) \\
\simeq \sum_{j=1}^{K} \frac{f_{\phi,\psi}(\tau^j, q^j)}{\sum_{k=1}^{K} f_{\phi,\psi}(\tau^j, q^i)} \nabla_\phi \log Q_{\phi}(\tau^j) \text{ with } \tau^j, q^j \overset{iid}{\sim} Q_{\phi,\psi}(\tau, q).
\]

The second to last step uses self-normalized importance sampling with \( K \) samples. \( \nabla_\psi \hat{L}(\phi, \psi) \) can be computed in a similar way.

B The Variational Bayesian Phylogenetic Inference Algorithm

**Algorithm 1** The variational Bayesian phylogenetic inference (VBPI) algorithm.

1: \( \phi, \psi \leftarrow \text{Initialize parameters} \)
2: \( \text{while not converged do} \)
3: \( \tau^1, \ldots, \tau^K \leftarrow \text{Random samples from the current approximating tree distribution } Q_{\phi}(\tau) \)
4: \( e^1, \ldots, e^K \leftarrow \text{Random samples from the multivariate standard normal distribution } \mathcal{N}(0, I) \)
5: \( g \leftarrow \nabla_\phi \hat{L}(\phi, \psi; \tau^{1:K}, e^{1:K}) \quad \text{(Use any gradient estimator from section 3.2)} \)
6: \( \phi, \psi \leftarrow \text{Update parameters using gradients } g \text{ (e.g. SGA)} \)
7: \( \text{end while} \)
8: \( \text{return } \phi, \psi \)

C Importance Sampling for Phylogenetic Inference via Variational Approximations

In this section, we provide a detailed importance sampling procedure for marginal likelihood estimation for phylogenetic inference based on the variational approximations provided by VBPI.

C.1 Estimating Marginal Likelihood of Trees

For each tree \( \tau \) that is covered by the subsplit support,
\[
Q_{\psi}(q|\tau) = \prod_{e \in E(\tau)} p_{\text{Lognormal}}(q_e | \mu(e, \tau), \sigma(e, \tau))
\]
can provide accurate approximation to the posterior of branch lengths on \( \tau \), where the mean and variance parameters \( \mu(e, \tau), \sigma(e, \tau) \) are gathered from the structured variational parameters \( \psi \) as introduced in section 3.1. Therefore, we can estimate the marginal likelihood of \( \tau \) using importance sampling with \( Q_{\psi}(q|\tau) \) being the importance distribution as follows
\[
p(Y|\tau) = \mathbb{E}_{Q_{\psi}(q|\tau)} \frac{p(Y|\tau, q)p(q)}{Q_{\psi}(q|\tau)} \simeq \frac{1}{M} \sum_{j=1}^{M} \frac{p(Y|\tau, q^j)p(q^j)}{Q_{\psi}(q^j|\tau)} \text{ with } q^j \overset{iid}{\sim} Q_{\psi}(q|\tau)
\]
C.2 Estimating Model Evidence

Similarly, we can estimate the marginal likelihood of the data as follows

\[ p(Y) = \mathbb{E}_{Q_\phi, \psi}(\tau, q) p(Y|\tau, q) \frac{1}{K} \sum_{j=1}^{K} p(Y|\tau^j, q^j) p(\tau^j, q^j) \]

with \( \tau^j, q^j \sim Q_\phi, \psi(\tau, q) \).

In our experiments, we use \( K = 1000 \). When taking a log transformation, the above Monte Carlo estimate is no longer unbiased (for the evidence \( \log p(Y) \)). Instead, it can be viewed as one sample Monte Carlo estimate of the lower bound

\[ L^K(\phi, \psi) = \mathbb{E}_{Q_\phi, \psi(\tau_1^K, q_1^K)} \log \left( \frac{1}{K} \sum_{i=1}^{K} p(Y|\tau^i, q^i) p(\tau^i, q^i) \right) \leq \log p(Y) \] (11)

whose tightness improves as the number of samples \( K \) increases. Therefore, with a sufficiently large \( K \), we can use the lower bound estimate as a proxy for Bayesian model selection.

D Consensus Tree Comparison on DS1

Figure 5: A comparison of majority-rule consensus trees obtained from VBPI and ground truth MCMC run on DS1. Left: Ground truth MCMC. Right: VBPI (10000 sampled trees). The plot is created using the treesspace (Jombart et al., 2017) R package.