Calculating Higher-Order Moments of Phylogenetic Stochastic Mapping Summaries in Linear Time

Amrit Dhar[1] and Vladimir N. Minin[1,2]

[1]Department of Statistics, University of Washington, Seattle
[2]Department of Biology, University of Washington, Seattle

September 27, 2016

Abstract

Stochastic mapping is a simulation-based method for probabilistically mapping substitution histories onto phylogenies according to continuous-time Markov models of evolution. This technique can be used to infer properties of the evolutionary process on the phylogeny and, unlike parsimony-based mapping, conditions on the observed data to randomly draw substitution mappings that do not necessarily require the minimum number of events on a tree. Most stochastic mapping applications simulate substitution mappings only to estimate the mean and/or variance of two commonly used mapping summaries: the number of particular types of substitutions (labeled substitution counts) and the time spent in a particular group of states (labeled dwelling times) on the tree. Fast, simulation-free algorithms for calculating the mean of stochastic mapping summaries exist. Importantly, these algorithms scale linearly in the number of tips/leaves of the phylogenetic tree. However, to our knowledge, no such algorithm exists for calculating higher-order moments of stochastic mapping summaries. We present one such simulation-free dynamic programming algorithm that calculates prior and posterior mapping variances and scales linearly in the number of phylogeny tips. Our procedure suggests a general framework that can be used to efficiently compute higher-order moments of stochastic mapping summaries without simulations. We demonstrate the usefulness of our algorithm by extending previously developed statistical tests for rate variation across sites and for detecting evolutionarily conserved regions in genomic sequences.

Keywords: dynamic programming; posterior predictive diagnostics; evolutionary conservation

1 Introduction

Given a multiple sequence alignment of DNA nucleotides, scientists are often interested in reconstructing a phylogenetic tree to help them learn more about the ancestral relationships between the sequences and the underlying evolutionary process (Yang, 2006). However, in some cases, phylogeny estimation by itself does not provide all the needed information about sequence evolution because we observe data only at the tips of the phylogeny. We do not have much insight into the evolution taking place on the different tree branches other than through the estimated branch lengths, which are usually specified in terms of the expected number of substitutions per site (Felsenstein, 2004, Chapter 13). However, researchers are often interested in making inferences about the evolutionary process on the phylogeny because these inferences could be used to answer important scientific questions. For instance, estimates of non-synonymous/synonymous substitution rate ratios on a phylogeny are commonly used to test for positive selection on protein-coding genes (Nielsen and Yang, 1998). Stochastic mapping can be used to accurately estimate these ratios and, more generally, can help us make reliable inferences about the latent evolutionary process on the phylogeny (Nielsen, 2002; Huelsenbeck et al., 2003; Dimmic et al., 2005; Zhai et al., 2007; Lemey et al., 2012). Stochastic mapping is a simulation-based technique used to probabilistically map substitution histories onto phylogenies according to continuous-time Markov chain (CTMC) models of evolution. This approach was motivated by the need for alternatives to parsimony mapping, which focuses attention on mappings requiring the fewest substitutions.

Stochastic mapping was first introduced by Nielsen (2002), who described how to sample substitution mappings from the posterior probability distribution of mappings for a single trait or a site in a multiple sequence alignment. By using Nielsen's sampling procedure, one can compute Monte Carlo estimates for the posterior mean and/or variance, among other properties of the posterior distribution, of any mutational mapping summary random variable of interest. The two most popular stochastic mapping summaries are the number of particular types of substitutions (labeled substitution counts) and the time spent in a particular group of states (labeled dwelling times) on the tree. In most applications, substitution mappings are simulated only to estimate the mean and/or variance of the two mapping summaries discussed above (Minin and Suchard, 2008b). Recognizing this, Minin and Suchard (2008b) synthesized previous work of Hobolth and Jensen (2005), Dutheil et al. (2005), and Holmes and Rubin (2002), among others, and developed an efficient algorithm that analytically calculates the expectations of the aforementioned mapping summaries. The authors compute restricted expectations of CTMC labeled substitution counts (Ball and Milne, 2005; Minin and Suchard, 2008a) and labeled dwelling times (Neuts, 1995; Guindon et al., 2004; Minin and Suchard, 2008b) on each tree branch and propagate these expectations across the phylogeny using a generalized pruning algorithm (Felsenstein, 1981). Similarly to Felsenstein's pruning algorithm, the algorithm of Minin and Suchard (2008b) scales linearly in the number of phylogeny tips. Minin and Suchard (2008b) drew inspiration from the work of Schadt et al. (1998), who formulated a similar algorithm that computes first derivatives of phylogenetic likelihood functions. Unfortunately, it is not straightforward to extend the algorithm of Minin and Suchard (2008b) to efficiently calculate the variances of the previously mentioned mapping summaries; as a result, these stochastic mapping variances can only be approximated using Monte Carlo simulations.

In this paper, we present a simulation-free dynamic programming algorithm that calculates prior and posterior mapping variances and scales linearly in the number of phylogeny tips. We draw upon concepts introduced by Kenney and Gu (2012), who developed a recursive procedure for calculating second derivatives of phylogenetic likelihood functions. Our procedure suggests a general framework that can be used to efficiently compute higher-order moments of stochastic mapping summaries without simulations. The structure of the rest of the paper is as follows. Section 2 introduces notation that is used throughout the entire paper and discusses our research problem more formally. In Section 3, we give a description of our algorithm for efficiently calculating these stochastic mapping variances. In Section 4, we demonstrate the usefulness of our algorithm by extending previously developed statistical tests for rate variation across sites and for detecting evolutionarily conserved regions in genomic sequences. Concluding remarks are presented in Section 5.

2 Notation and Problem Background

We use much of the notation provided by Minin and Suchard (2008b). Suppose we have a discrete evolutionary trait X (i.e. DNA base) that takes on m distinct states and that evolves according to a CTMC on a phylogeny. This evolutionary process, call it $\psi_{\boldsymbol{\theta}}$, depends on the parameter set $\boldsymbol{\theta} = \{\tau, \mathbf{t}, \mathbf{Q}, \boldsymbol{\pi}\}$, which consists of a rooted tree topology τ with n tips and $B_n = 2n-2$ branches; branch lengths $\mathbf{t} = (t_1, ..., t_{B_n})$; a reversible CTMC rate matrix $\mathbf{Q} = \{q_{ij}\}$ for $i, j = 1, ..., m$; and a CTMC stationary distribution $\boldsymbol{\pi} = (\pi_1, ..., \pi_m)^T$. We assume that our evolutionary process starts at stationarity (i.e. we assume that the root distribution is equal to $\boldsymbol{\pi}$). While not necessary, this commonly used assumption ensures that the stochastic mapping moments will be invariant to the placement of the root (Minin and Suchard, 2008b). When this assumption is not used, as often is the case in analyses of morphological traits (Pagel, 1999), our methods still work without modification, but the root of the tree has to be specified by the user. Matrix $\mathbf{P}(t) = \{p_{ij}(t)\} = \exp(\mathbf{Q}t)$ represents the CTMC transition probability matrix for a branch of length t.

We define $\Theta = \{1, ..., B_n\}$ to be the set of branch indices of τ. Let $\Theta_b = \{b^* \in \Theta \mid b^* \preceq b\}$ denote the set of branch indices in the subtree relating all descendants of branch b, including b, where $b^* \preceq b$ for $b, b^* \in \Theta$ if either b^* is a descendant of b or b^* is equal to b. Let $\mathcal{I} \subset \Theta$ represent the set of internal branches (i.e. branches that connect two internal nodes) and $\mathcal{E} = \Theta \setminus \mathcal{I}$ represent the set of terminal branches (i.e. branches that connect an internal node to a tip node). Let $\mathbf{D} = (D_1, ..., D_n)$ denote the trait values observed at the n tips of τ, $\mathbf{D}_{1:L} = \{\mathbf{D}_1, ..., \mathbf{D}_L\}$ signify an alignment of length L, and $\mathbf{i} = (i_1, ..., i_{n-1})$ represent the unobserved internal node states of τ. In addition, we let $\mathbf{i}_b$ be the vector of internal node states for the subtree strictly beneath branch b. Note that the internal nodes of τ are labeled with integers $\{1, ..., n-1\}$ starting from the root of the tree; the corresponding labels of the branches and tips of τ are assigned arbitrarily. We also introduce $\mathbf{i}^* = (i_1^*, ..., i_{n-1}^*, i_n^*, ..., i_{2n-1}^*) = (i_1, ..., i_{n-1}, D_1, ..., D_n)$, which is the concatenation of $\mathbf{i}$ and $\mathbf{D}$. For each branch $b \in \Theta$, $p(b)$ and $c(b)$ represent the node labels (in $\mathbf{i}^*$) of the parent and child of branch b, respectively.

Most stochastic mapping applications infer properties about the evolutionary process on the phylogeny through the use of a summary measure H. We restrict attention to additive mapping summaries of the form:

$$H_\Omega \equiv H_\Omega(\mathbf{M}) = \sum_{b\in\Omega} h(\{X_{bt}\}), \tag{1}$$

where $\mathbf{M} = (\{X_{1t}\}, ..., \{X_{B_n t}\})$ denotes the collection of CTMC trajectories along the branches of τ, $\Omega \subseteq \Theta$ represents a predefined set of branch indices, and h signifies a summary measure applied to a single CTMC trajectory. Let $\mathcal{L} \subset \{1, ..., m\}^2$ be a set of state pairs that labels substitutions of trait X and $\mathbf{w} \subset \{0, 1\}^m$ be a set that labels states of trait X. For any given CTMC path $\{X_t\}$ in $[0, t)$, the two most popular choices of h are $h_1(\{X_t\})$, which counts the number of substitutions labeled by set $\mathcal{L}$, and $h_2(\{X_t\})$, which measures the dwelling time in states labeled by set $\mathbf{w}$ (Minin and Suchard, 2008b). In this paper, we work exclusively with the summary function h_1 as this summary measure is used in both of our scientific applications. However, we do note that our algorithmic results hold true regardless of the specific summary measure used.

Minin and Suchard (2008b) were able to calculate the posterior mapping expectation $\mathrm{E}(H_\Omega|\mathbf{D})$ for both h_1 and h_2 in $O(n)$ time and with $O(n)$ storage but were unable to achieve the same space-time complexity when calculating the posterior mapping variance $\mathrm{Var}(H_\Omega|\mathbf{D})$. Before we delve into the difficulties associated with computing $\mathrm{Var}(H_\Omega|\mathbf{D})$, we refresh our readers on two important quantities:

$$\mathrm{E}(H_\Omega^2 \mathbb{1}_{\mathbf{D}}) = \mathrm{E}(H_\Omega^2|\mathbf{D}) \times \mathrm{P}(\mathbf{D}), \tag{2}$$

$$e^{[k]}_{ij}(h, t) = \mathrm{E}\Big\{h(\{X_t\})\big[h(\{X_t\}) - 1\big]...\big[h(\{X_t\}) - k + 1\big]\mathbb{1}_{\{X_t=j\}}\Big| X_0 = i\Big\}, \tag{3}$$

where $\mathbb{1}_{\{.\}}$ represents the indicator function; $k = 1, 2, ...$; $i, j = 1, ..., m$; and $\mathrm{P}(\mathbf{D})$ denotes the phylogenetic likelihood defined as the probability of observing the tip sequence $\mathbf{D}$. Equation (2) connects the restricted mapping second moment $\mathrm{E}(H_\Omega^2 \mathbb{1}_{\mathbf{D}})$ to the posterior mapping second moment $\mathrm{E}(H_\Omega^2|\mathbf{D})$. As Minin and Suchard (2008b) state, the restricted expectation in equation (2) integrates over all evolutionary mappings consistent with $\mathbf{D}$ on the tips of τ. Since $\mathrm{P}(\mathbf{D})$ can be easily computed using the pruning algorithm (Felsenstein, 1981), we focus our attention on calculating $\mathrm{E}(H_\Omega^2 \mathbb{1}_{\mathbf{D}})$. Quantity $e^{[k]}_{ij}(h, t)$ denotes the kth restricted factorial moment of $h(\{X_t\})$ for a CTMC path $\{X_t\}$ in $[0, t)$ that starts in state i and ends in state j. We let $\mathbf{e}^{[k]}(h, t) = \{e^{[k]}_{ij}(h, t)\}$ represent the corresponding restricted factorial moment matrix. Minin and Suchard (2008a) derive a simple recurrence relation to calculate $\mathbf{e}^{[k]}(h_1, t)$ for $k = 1, 2, ...$; a similar relation exists for h_2 as well (Minin and Suchard, 2008b).

To help us illustrate the computational challenges inherent in calculating $\mathrm{Var}(H_\Omega|\mathbf{D})$, we express $\mathrm{E}(H_\Omega^2 \mathbb{1}_{\mathbf{D}})$ in the following manner (suppressing the fact that $b, b' \in \Omega$ for brevity):

$$\mathrm{E}(H_\Omega^2 \mathbb{1}_{\mathbf{D}}) = \mathrm{E}\left[\left(\sum_b h(\{X_{bt}\})\right)^2 \mathbb{1}_{\mathbf{D}}\right] \tag{4}$$

$$= \sum_b \mathrm{E}\big[h(\{X_{bt}\})^2 \mathbb{1}_{\mathbf{D}}\big] + \sum_{b\neq b'} \mathrm{E}\big[h(\{X_{bt}\})h(\{X_{b't}\})\mathbb{1}_{\mathbf{D}}\big] \tag{5}$$

$$= \sum_b \sum_{\mathbf{i}} \mathrm{E}\big[h(\{X_{bt}\})^2 |\ \mathbf{i}, \mathbf{D}\big]\mathrm{P}(\mathbf{i}, \mathbf{D}) + \sum_{b\neq b'} \sum_{\mathbf{i}} \mathrm{E}\big[h(\{X_{bt}\})h(\{X_{b't}\})|\ \mathbf{i}, \mathbf{D}\big]\mathrm{P}(\mathbf{i}, \mathbf{D}) \tag{6}$$

$$\begin{aligned} &= \sum_b \sum_{\mathbf{i}} \mathrm{E}\big[h(\{X_{bt}\})^2 |\ i^*_{p(b)}, i^*_{c(b)}\big]\pi_{i^*_1} \prod_{b^*\in\Theta} p_{i^*_{p(b^*)} i^*_{c(b^*)}}(t_{b^*}) \\ &+ \sum_{b\neq b'} \sum_{\mathbf{i}} \mathrm{E}\big[h(\{X_{bt}\})h(\{X_{b't}\})|\ i^*_{p(b)}, i^*_{c(b)}, i^*_{p(b')}, i^*_{c(b')}\big]\pi_{i^*_1} \prod_{b^*\in\Theta} p_{i^*_{p(b^*)} i^*_{c(b^*)}}(t_{b^*}) \end{aligned} \tag{7}$$

$$= \sum_b \sum_{\mathbf{i}} \Big[e^{[2]}_{i^*_{p(b)} i^*_{c(b)}}(h, t_b) + e^{[1]}_{i^*_{p(b)} i^*_{c(b)}}(h, t_b)\Big]\pi_{i^*_1} \prod_{b^*\in\Theta\backslash\{b\}} p_{i^*_{p(b^*)} i^*_{c(b^*)}}(t_{b^*}) \tag{8}$$

$$+ \sum_{b\neq b'} \sum_{\mathbf{i}} e^{[1]}_{i^*_{p(b)} i^*_{c(b)}}(h, t_b) e^{[1]}_{i^*_{p(b')} i^*_{c(b')}}(h, t_{b'})\pi_{i^*_1} \prod_{b^*\in\Theta\backslash\{b,b'\}} p_{i^*_{p(b^*)} i^*_{c(b^*)}}(t_{b^*}). \tag{9}$$

The single summation over b in (8) can be efficiently computed by utilizing a modified version of the generalized pruning algorithm presented in (Minin and Suchard, 2008b). However, the straightforward double summation over $b \neq b'$ in (9) requires at most $O(B_n^2) = O((2n-2)^2) = O(n^2)$ computations. In the next section, we describe how to overcome this computational roadblock via a post-order tree traversal algorithm that computes $\mathrm{E}(H_\Omega^2 \mathbb{1}_{\mathbf{D}})$ (and therefore $\mathrm{Var}(H_\Omega|\mathbf{D})$) in $O(n)$ time and with $O(n)$ storage.

3 Methods

3.1 Algorithm Setup

To help motivate our procedure and make it easier to understand, we utilize illustrations of various "colored" phylogenies. Figure 1 displays the example "uncolored" tree we use to create all these illustrations. Note that for a given $b \neq b'$ where $b, b' \in \Omega$, the corresponding summand in (9) is:

$$\sum_{\mathbf{i}} e^{[1]}_{i^*_{p(b)} i^*_{c(b)}}(h, t_b) e^{[1]}_{i^*_{p(b')} i^*_{c(b')}}(h, t_{b'}) \pi_{i^*_1} \prod_{b^* \in \Theta \setminus \{b, b'\}} p_{i^*_{p(b^*)} i^*_{c(b^*)}}(t_{b^*}). \tag{10}$$

If we replace the restricted first moments in (10) with the appropriate transition probabilities, then expression (10) would represent the phylogenetic likelihood. From equation (5), we know that expression (10) is equal to the restricted product moment $\mathrm{E}\big[h(\{X_{bt}\})h(\{X_{b't}\})\mathbb{1}_{\mathbf{D}}\big]$. In Figure 1, we present two "colored" phylogenies to help visualize the calculation of $\mathrm{E}\big[h(\{X_{bt}\})h(\{X_{b't}\})\mathbb{1}_{\mathbf{D}}\big]$. Thus, we can visualize the double sum over $b \neq b'$ in (9) by imagining the red and blue colorings being permuted across all branches in Ω. The cached vectors used in our procedure are described in a similar fashion.

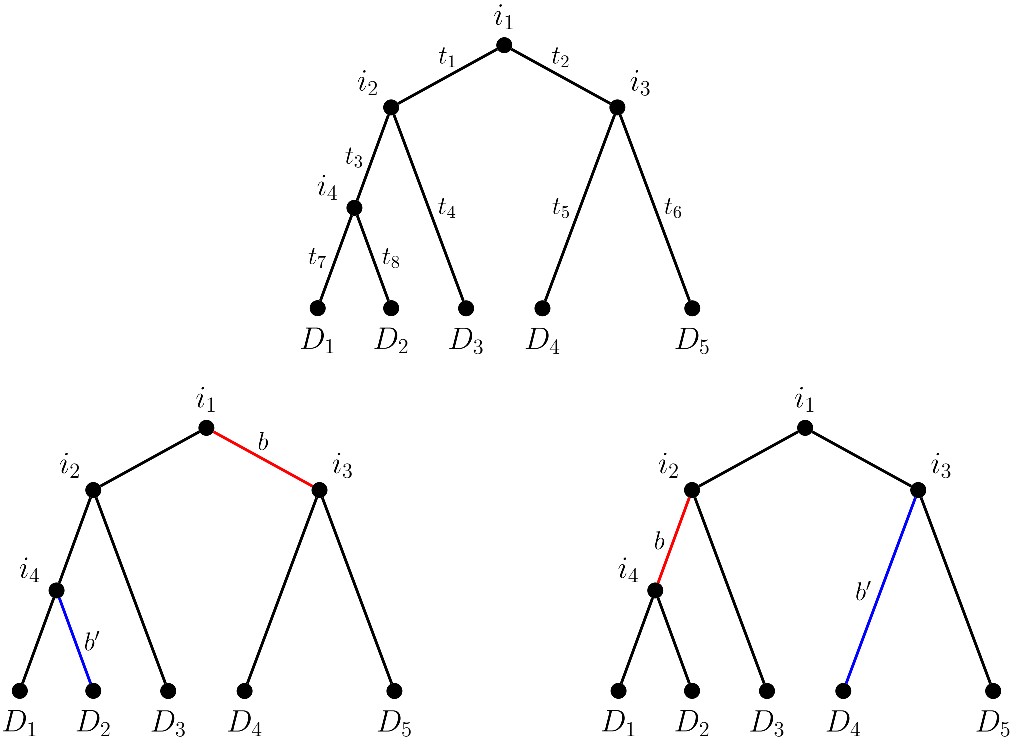


Figure 1: The example phylogenies we use to help motivate our algorithmic procedure. (Top) The "uncolored" tree used to create all the "colored" phylogenies. This tree has $n = 5$ tips, $B_n = 8$ branches with branch lengths $\mathbf{t} = (t_1, ..., t_8)$, internal node states $\mathbf{i} = (i_1, ..., i_4)$, and tip states $\mathbf{D} = (D_1, ..., D_5)$. In addition, $\Theta = \{1,2,3,4,5,6,7,8\}$, $\mathcal{I} = \{1,2,3\}$, and $\mathcal{E} = \{4,5,6,7,8\}$. (Bottom) Two "colored" phylogenies that illustrate the calculation of $\mathrm{E}\big[h(\{X_{bt}\})h(\{X_{b't}\})\mathbb{1}_{\mathbf{D}}\big]$. The colored branches specify the locations of the restricted first moments, while the uncolored branches determine the locations of the transition probabilities. The first tree (left) and second tree (right) visualize the calculations of $\mathrm{E}\big[h(\{X_{2t}\})h(\{X_{8t}\})\mathbb{1}_{\mathbf{D}}\big]$ and $\mathrm{E}\big[h(\{X_{3t}\})h(\{X_{5t}\})\mathbb{1}_{\mathbf{D}}\big]$, respectively, for the "uncolored" tree shown above.

Let $\mathbf{F}_u = (F_{u1}, ..., F_{um})^T$ be the vector of partial likelihoods at node u, where F_{ui} denotes the probability of the observed data at only the tips that descend from node u, given that the state of node u is i. We let $\mathbf{S}_b = (S_{b1}, ..., S_{bm})^T$ denote the vector of directional likelihoods at branch b, where S_{bi} represents the likelihood of the observed data at only the tips that descend from branch b, given that the state of parent node $p(b)$ is i. Minin and Suchard (2008b) utilize these $\mathbf{F}_u$ and $\mathbf{S}_b$ vectors within their algorithm for computing the posterior mapping expectation $\mathrm{E}(H_\Omega|\mathbf{D})$.

We also define vectors $\mathbf{V}_b^{[1]} = \big(V_{b1}^{[1]}, ..., V_{bm}^{[1]}\big)^T$, $\mathbf{V}_b^{[2]} = \big(V_{b1}^{[2]}, ..., V_{bm}^{[2]}\big)^T$, and $\mathbf{W}_b = (W_{b1}, ..., W_{bm})^T$. Elements of vector $\mathbf{V}_b^{[1]}$ are defined as follows:

$$V_{bi}^{[1]} = \sum_{b^\dagger} \sum_{\mathbf{i}_b} e^{[1]}_{i^*_{p(b^\dagger)} i^*_{c(b^\dagger)}}(h, t_{b^\dagger}) \prod_{b^* \in \Theta_b \setminus \{b^\dagger\}} p_{i^*_{p(b^*)} i^*_{c(b^*)}}(t_{b^*}), \tag{11}$$

where the state of parent node $p(b)$ is i and $b^\dagger \in \Omega_b$ for the set of branches of interest in the subtree that is "below" branch b (including b), $\Omega_b = \Omega \cap \Theta_b$. An illustration of $\mathbf{V}_b^{[1]}$ is given in Figure 2. We can interpret $\boldsymbol{\pi}^T \mathbf{V}_b^{[1]}$ as the restricted mapping first moment $\mathrm{E}(H_{\Omega_b} \mathbb{1}_{\mathbf{D}})$ for the subtree that is "below" branch b (including b). Kenney and Gu (2012) cache a vector similar to $\mathbf{V}_b^{[1]}$ within their algorithm for computing second derivatives of phylogenetic likelihood functions. If we replace the restricted first moment in expression (11) with the appropriate transition probability derivative, then we would recover this cached vector used in (Kenney and Gu, 2012). Similarly, $V_{bi}^{[2]}$ is defined the same as $V_{bi}^{[1]}$, except in the case of $V_{bi}^{[2]}$, we replace the restricted first moment in expression (11) with the corresponding second restricted factorial moment $e^{[2]}_{i^*_{p(b^\dagger)} i^*_{c(b^\dagger)}}(h, t_{b^\dagger})$. The visual depiction of $\mathbf{V}_b^{[2]}$ is analogous to that of $\mathbf{V}_b^{[1]}$ in Figure 2. Elements of vector $\mathbf{W}_b$ are defined as follows:

$$W_{bi} = \sum_{b^\dagger \neq b^{\dagger\dagger}} \sum_{\mathbf{i}_b} e^{[1]}_{i^*_{p(b^\dagger)} i^*_{c(b^\dagger)}}(h, t_{b^\dagger}) e^{[1]}_{i^*_{p(b^{\dagger\dagger})} i^*_{c(b^{\dagger\dagger})}}(h, t_{b^{\dagger\dagger}}) \prod_{b^* \in \Theta_b \setminus \{b^\dagger, b^{\dagger\dagger}\}} p_{i^*_{p(b^*)} i^*_{c(b^*)}}(t_{b^*}), \tag{12}$$

where the state of parent node $p(b)$ is i and $b^\dagger, b^{\dagger\dagger} \in \Omega_b$ for Ω_b defined as above. A pictorial representation of $\mathbf{W}_b$ is provided in Figure 2. $\boldsymbol{\pi}^T \mathbf{W}_b$ can be viewed as the sum of restricted product moments $\mathrm{E}\big[h(\{X_{b^\dagger t}\}) h(\{X_{b^{\dagger\dagger} t}\}) \mathbb{1}_{\mathbf{D}}\big]$ over $b^\dagger \neq b^{\dagger\dagger}$ ($b^\dagger, b^{\dagger\dagger} \in \Omega_b$) for the subtree that is "below" branch b (including b). In the next subsection, we describe our algorithm in full detail and provide some intuition behind the recursive formulas used in our procedure.

3.2 Algorithm Recursion

Our post-order tree traversal algorithm recursively computes $\mathbf{F}_u$, $\mathbf{S}_b$, $\mathbf{V}_b^{[1]}$, $\mathbf{V}_b^{[2]}$, and $\mathbf{W}_b$ at all nodes $u \in \{1, ..., n-1, n, ..., 2n-1\}$ and branches $b \in \Theta$. Like the pruning algorithm, this procedure starts at the tips of the tree and continues through all ancestral nodes until it arrives at the root of the tree. We start by describing how these vectors are initialized at the terminal nodes/branches of τ and then specify the recursive formulas used to calculate $\mathbf{F}_u$, $\mathbf{S}_b$, $\mathbf{V}_b^{[1]}$, $\mathbf{V}_b^{[2]}$, and $\mathbf{W}_b$ at the internal nodes/branches of τ.

First, we follow standard practice and set $F_{ui} = \mathbb{1}_{\{i = i^*_u\}}$ for all terminal nodes $u \in \{n, ..., 2n-1\}$ and $i = 1, ..., m$. Oftentimes, we have partially observed and/or missing data at the tips of τ and our initialization of F_{ui} can be adjusted to reflect this information (Felsenstein, 1981). For terminal branches $b \in \mathcal{E}$, we set:

$$S_{bi} = \sum_{j=1}^{m} p_{ij}(t_b) F_{c(b)j}, \tag{13}$$

for $i = 1, ..., m$. Using matrix notation, we express the equation in (13) as $\mathbf{S}_b = \mathbf{P}(t_b) \mathbf{F}_{c(b)}$. The initializations of $\mathbf{V}_b^{[1]}$ and $\mathbf{V}_b^{[2]}$ depend on whether or not $b \in \Omega$. For all terminal branches $b \in \mathcal{E}$, we define:

$$V_{bi}^{[1]} = \sum_{j=1}^{m} e_{ij}^{[1]}(h, t_b) F_{c(b)j} \mathbb{1}_{\{b \in \Omega\}}, \tag{14}$$

$$V_{bi}^{[2]} = \sum_{j=1}^{m} e_{ij}^{[2]}(h, t_b) F_{c(b)j} \mathbb{1}_{\{b \in \Omega\}}, \tag{15}$$

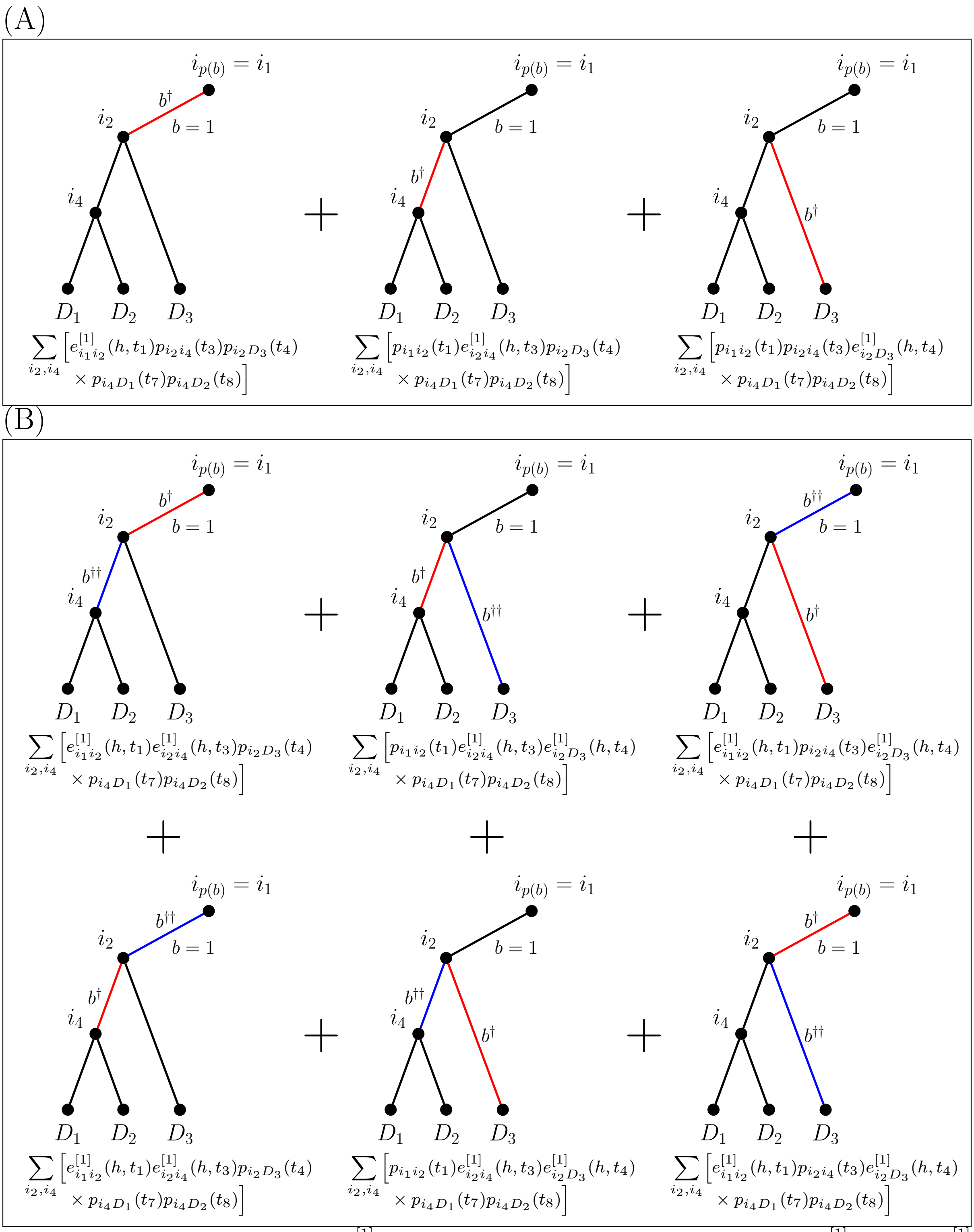

(A)
$i_{p(b)} = i_1$
$b^\dagger$
$b = 1$
i_2
i_4
D_1 D_2 D_3
$\sum_{i_2,i_4}\Big[e^{[1]}_{i_1i_2}(h,t_1)p_{i_2i_4}(t_3)p_{i_2D_3}(t_4) \times p_{i_4D_1}(t_7)p_{i_4D_2}(t_8)\Big]$
+
$\sum_{i_2,i_4}\Big[p_{i_1i_2}(t_1)e^{[1]}_{i_2i_4}(h,t_3)p_{i_2D_3}(t_4) \times p_{i_4D_1}(t_7)p_{i_4D_2}(t_8)\Big]$
+
$\sum_{i_2,i_4}\Big[p_{i_1i_2}(t_1)p_{i_2i_4}(t_3)e^{[1]}_{i_2D_3}(h,t_4) \times p_{i_4D_1}(t_7)p_{i_4D_2}(t_8)\Big]$
(B)
$b^{\dagger\dagger}$
$\sum_{i_2,i_4}\Big[e^{[1]}_{i_1i_2}(h,t_1)e^{[1]}_{i_2i_4}(h,t_3)p_{i_2D_3}(t_4) \times p_{i_4D_1}(t_7)p_{i_4D_2}(t_8)\Big]$
+
$\sum_{i_2,i_4}\Big[p_{i_1i_2}(t_1)e^{[1]}_{i_2i_4}(h,t_3)e^{[1]}_{i_2D_3}(h,t_4) \times p_{i_4D_1}(t_7)p_{i_4D_2}(t_8)\Big]$
+
$\sum_{i_2,i_4}\Big[e^{[1]}_{i_1i_2}(h,t_1)p_{i_2i_4}(t_3)e^{[1]}_{i_2D_3}(h,t_4) \times p_{i_4D_1}(t_7)p_{i_4D_2}(t_8)\Big]$
+
$\sum_{i_2,i_4}\Big[e^{[1]}_{i_1i_2}(h,t_1)e^{[1]}_{i_2i_4}(h,t_3)p_{i_2D_3}(t_4) \times p_{i_4D_1}(t_7)p_{i_4D_2}(t_8)\Big]$
+
$\sum_{i_2,i_4}\Big[p_{i_1i_2}(t_1)e^{[1]}_{i_2i_4}(h,t_3)e^{[1]}_{i_2D_3}(h,t_4) \times p_{i_4D_1}(t_7)p_{i_4D_2}(t_8)\Big]$
+
$\sum_{i_2,i_4}\Big[e^{[1]}_{i_1i_2}(h,t_1)p_{i_2i_4}(t_3)e^{[1]}_{i_2D_3}(h,t_4) \times p_{i_4D_1}(t_7)p_{i_4D_2}(t_8)\Big]$

Figure 2: Visual depictions of the $\mathbf{V}_b^{[1]}$ and $\mathbf{W}_b$ vectors. (A) An illustration of the $\mathbf{V}_b^{[1]}$ vector. $V_{bi}^{[1]}$ can be interpreted as the sum over all "single-colored" phylogenies for the subtree defined by Θ_b and the predefined set of "colored" branches $\Omega_b = \Omega \cap \Theta_b$, conditional on the state of parent node $p(b)$ being i. We illustrate $\mathbf{V}_1^{[1]}$ for the "uncolored" phylogeny given in Figure 1, where $\Omega_1 = \{1, 3, 4\}$. (B) An illustration of the $\mathbf{W}_b$ vector. W_{bi} can be interpreted as the sum over all "double-colored" phylogenies for the same subtree and set of "colored" branches as described in (A), conditional on the state of parent node $p(b)$ being i. We illustrate $\mathbf{W}_1$ for the "uncolored" tree displayed in Figure 1, where $\Omega_1 = \{1, 3, 4\}$.

for $i = 1, ..., m$. The vectorized representations of equations (14) and (15) are $\mathbf{V}_b^{[1]} = \mathbf{e}^{[1]}(h, t_b)\mathbf{F}_{c(b)}\mathbb{1}_{\{b\in\Omega\}}$ and $\mathbf{V}_b^{[2]} = \mathbf{e}^{[2]}(h, t_b)\mathbf{F}_{c(b)}\mathbb{1}_{\{b\in\Omega\}}$, respectively. Finally, $W_{bi} = 0$ for all terminal branches $b \in \mathcal{E}$ and $i = 1, ..., m$. Note that the definitions of $\mathbf{V}_b^{[1]}$, $\mathbf{V}_b^{[2]}$, and $\mathbf{W}_b$ for $b \in \mathcal{E}$ are consistent with the illustrations provided in Figure 2.

Now, we present the recursive formulas that compute $\mathbf{F}_u$, $\mathbf{S}_b$, $\mathbf{V}_b^{[1]}$, $\mathbf{V}_b^{[2]}$, and $\mathbf{W}_b$ for internal nodes $u \in \{1, ..., n-1\}$ and internal branches $b \in \mathcal{I}$. The recursion for the partial likelihood F_{ui} is the centerpiece of Felsenstein's pruning algorithm (Felsenstein, 1981):

$$F_{ui} = \underbrace{\left[\sum_{j=1}^{m} p_{ij}(t_{b_1})F_{c(b_1)j}\right]}_{S_{b_1 i}} \times \underbrace{\left[\sum_{j=1}^{m} p_{ij}(t_{b_2})F_{c(b_2)j}\right]}_{S_{b_2 i}}, \tag{16}$$

where b_1 and b_2 represent the two branches connecting node u to its two child nodes and $i = 1, ..., m$. In addition, as shown in the brackets above, equation (13) also denotes the recursion for S_{bi} at internal branches $b \in \mathcal{I}$ (Felsenstein, 1981). Thus, the recursive formula for F_{ui} in (16) can be compactly expressed as $\mathbf{F}_u = \mathbf{S}_{b_1} \circ \mathbf{S}_{b_2}$, where $\circ$ symbolizes element-wise multiplication between two vectors. The recursive equations used to calculate $V_{bi}^{[1]}$, $V_{bi}^{[2]}$, and W_{bi} at internal branches $b \in \mathcal{I}$ are:

$$V_{bi}^{[1]} = \sum_{j=1}^{m}\left[e_{ij}^{[1]}(h, t_b)F_{c(b)j}\mathbb{1}_{\{b\in\Omega\}} + p_{ij}(t_b)\Big(V_{b_1 j}^{[1]}S_{b_2 j} + V_{b_2 j}^{[1]}S_{b_1 j}\Big)\right], \tag{17}$$

$$V_{bi}^{[2]} = \sum_{j=1}^{m}\left[e_{ij}^{[2]}(h, t_b)F_{c(b)j}\mathbb{1}_{\{b\in\Omega\}} + p_{ij}(t_b)\Big(V_{b_1 j}^{[2]}S_{b_2 j} + V_{b_2 j}^{[2]}S_{b_1 j}\Big)\right], \tag{18}$$

$$\begin{aligned} W_{bi} = \sum_{j=1}^{m}\Bigg[&2 \times e_{ij}^{[1]}(h, t_b)\Big(V_{b_1 j}^{[1]}S_{b_2 j} + V_{b_2 j}^{[1]}S_{b_1 j}\Big)\mathbb{1}_{\{b\in\Omega\}} \\ &+ p_{ij}(t_b)\Big(2 \times V_{b_1 j}^{[1]}V_{b_2 j}^{[1]} + W_{b_1 j}S_{b_2 j} + W_{b_2 j}S_{b_1 j}\Big)\Bigg], \end{aligned} \tag{19}$$

respectively, where b_1 and b_2 represent the two branches that are "below" branch b and $i = 1, ..., m$. The recursive formulas presented in equations (17)-(19) can also be expressed as:

$$\mathbf{V}_b^{[1]} = \mathbf{e}^{[1]}(h, t_b)\mathbf{F}_{c(b)}\mathbb{1}_{\{b\in\Omega\}} + \mathbf{P}(t_b)\Big(\mathbf{V}_{b_1}^{[1]} \circ \mathbf{S}_{b_2} + \mathbf{V}_{b_2}^{[1]} \circ \mathbf{S}_{b_1}\Big), \tag{20}$$

$$\mathbf{V}_b^{[2]} = \mathbf{e}^{[2]}(h, t_b)\mathbf{F}_{c(b)}\mathbb{1}_{\{b\in\Omega\}} + \mathbf{P}(t_b)\Big(\mathbf{V}_{b_1}^{[2]} \circ \mathbf{S}_{b_2} + \mathbf{V}_{b_2}^{[2]} \circ \mathbf{S}_{b_1}\Big), \tag{21}$$

$$\begin{aligned} \mathbf{W}_b = {} & 2 \times \mathbf{e}^{[1]}(h, t_b)\Big(\mathbf{V}_{b_1}^{[1]} \circ \mathbf{S}_{b_2} + \mathbf{V}_{b_2}^{[1]} \circ \mathbf{S}_{b_1}\Big)\mathbb{1}_{\{b\in\Omega\}} \\ & + \mathbf{P}(t_b)\Big(2 \times \mathbf{V}_{b_1}^{[1]} \circ \mathbf{V}_{b_2}^{[1]} + \mathbf{W}_{b_1} \circ \mathbf{S}_{b_2} + \mathbf{W}_{b_2} \circ \mathbf{S}_{b_1}\Big), \end{aligned} \tag{22}$$

respectively. From Figure 2, we know that elements of $\mathbf{V}_b^{[1]}$ can be interpreted as the sum over all "single-colored" phylogenies for the subtree defined by Θ_b and for the predefined set of "colored" branches $\Omega_b = \Omega \cap \Theta_b$, conditional on the state of parent node $p(b)$. Equation (20) partitions this sum into three distinct pieces: 1) $\mathbf{e}^{[1]}(h, t_b)\mathbf{F}_{c(b)}\mathbb{1}_{\{b\in\Omega\}}$, 2) $\mathbf{P}(t_b)\Big(\mathbf{V}_{b_1}^{[1]} \circ \mathbf{S}_{b_2}\Big)$, and 3) $\mathbf{P}(t_b)\Big(\mathbf{V}_{b_2}^{[1]} \circ \mathbf{S}_{b_1}\Big)$. The first piece represents the "single-colored" tree with "colored" branch b (only if $b \in \Omega$); the second piece represents a sum over "single-colored" phylogenies, where the "colored" branch is permuted across all branches in Ω_{b_1}; and the third piece represents another sum over "single-colored" phylogenies, where the "colored" branch is permuted across all branches in Ω_{b_2}. This partitioning allows us to compute $\mathbf{V}_b^{[1]}$ as a function of previously cached vectors $\mathbf{F}_{c(b)}$, $\mathbf{S}_{b_1}$, $\mathbf{S}_{b_2}$, $\mathbf{V}_{b_1}^{[1]}$, and $\mathbf{V}_{b_2}^{[1]}$. The recursions for $\mathbf{V}_b^{[2]}$ and $\mathbf{W}_b$ have analogous interpretations. Note that our algorithm requires $O(n)$ storage because we cache a constant (with respect to n) number of vectors at all nodes $u \in \{1, ..., n-1, n, ..., 2n-1\}$ and branches $b \in \Theta$. In addition, our procedure utilizes $O(n)$ computations because there are $O(B_n) = O(n)$ iterations in the algorithm and each iteration involves a constant (with respect to n) number of operations.

3.3 Posterior Mapping Variance Computation

Our tips-to-root tree traversal procedure terminates after computing the vectors $\mathbf{F}_{root}$, $\mathbf{S}_{root_1}$, $\mathbf{S}_{root_2}$, $\mathbf{V}^{[1]}_{root_1}$, $\mathbf{V}^{[1]}_{root_2}$, $\mathbf{V}^{[2]}_{root_1}$, $\mathbf{V}^{[2]}_{root_2}$, $\mathbf{W}_{root_1}$, and $\mathbf{W}_{root_2}$, where *root* denotes the root node label and $root_1$ and $root_2$ represent the two branches connecting the root node to its children. We use these cached vectors to efficiently calculate the posterior mapping variance $\mathrm{Var}(H_\Omega|\mathbf{D})$.

We first describe how to compute the restricted mapping second moment $\mathrm{E}(H^2_\Omega \mathbb{1}_{\mathbf{D}})$. We remind our readers that the double sum over $b \neq b'$ $(b, b' \in \Omega)$ in (9) can be visualized as a sum over all "double-colored" phylogenies, where Ω denotes the predefined set of "colored" branches. There are four distinct cases of $b \neq b'$ that we consider in (9): 1) $b \preceq root_1, b' \preceq root_1$; 2) $b \preceq root_1, b' \preceq root_2$; 3) $b \preceq root_2, b' \preceq root_1$; and 4) $b \preceq root_2, b' \preceq root_2$. Cases 1) and 4) represent the "double-colored" phylogenies that have both "colored" branches on the same side of the root, while Cases 2) and 3) denote the "double-colored" phylogenies that have one "colored" branch on each side of the root. This decomposition of the double sum in (9) was suggested by Kenney and Gu (2012) in the context of computing second derivatives of phylogenetic likelihood functions. The sums over all "double-colored" phylogenies in Cases 1), 2), 3), and 4) are mathematically represented as $\boldsymbol{\pi}^T\Big(\mathbf{W}_{root_1} \circ \mathbf{S}_{root_2}\Big)$, $\boldsymbol{\pi}^T\Big(\mathbf{V}^{[1]}_{root_1} \circ \mathbf{V}^{[1]}_{root_2}\Big)$, $\boldsymbol{\pi}^T\Big(\mathbf{V}^{[1]}_{root_1} \circ \mathbf{V}^{[1]}_{root_2}\Big)$, and $\boldsymbol{\pi}^T\Big(\mathbf{W}_{root_2} \circ \mathbf{S}_{root_1}\Big)$, respectively. Thus, the double sum in (9) can be efficiently computed as:

$$\boldsymbol{\pi}^T\Big(2 \times \mathbf{V}^{[1]}_{root_1} \circ \mathbf{V}^{[1]}_{root_2} + \mathbf{W}_{root_1} \circ \mathbf{S}_{root_2} + \mathbf{W}_{root_2} \circ \mathbf{S}_{root_1}\Big). \tag{23}$$

The sum over $b \in \Omega$ in (8) can also be calculated using the cached vectors mentioned above. We can visualize the sum in (8) as a sum over all "single-colored" phylogenies, where Ω denotes the predefined set of "colored" branches. There are two cases of $b \in \Omega$ that we consider in (8): 1) $b \preceq root_1$ and 2) $b \preceq root_2$. Using logic similar to that described above, we can calculate the sum in (8) as:

$$\boldsymbol{\pi}^T\Big[\big(\mathbf{V}^{[1]}_{root_1} + \mathbf{V}^{[2]}_{root_1}\big) \circ \mathbf{S}_{root_2} + \big(\mathbf{V}^{[1]}_{root_2} + \mathbf{V}^{[2]}_{root_2}\big) \circ \mathbf{S}_{root_1}\Big]. \tag{24}$$

We obtain a simple formula for computing the restricted mapping second moment $\mathrm{E}(H^2_\Omega \mathbb{1}_{\mathbf{D}})$ by adding together the expressions in (23) and (24):

$$\begin{aligned}\mathrm{E}(H^2_\Omega \mathbb{1}_{\mathbf{D}}) = \boldsymbol{\pi}^T\Big[&2 \times \mathbf{V}^{[1]}_{root_1} \circ \mathbf{V}^{[1]}_{root_2} + \mathbf{W}_{root_1} \circ \mathbf{S}_{root_2} + \mathbf{W}_{root_2} \circ \mathbf{S}_{root_1} \\ &+ \big(\mathbf{V}^{[1]}_{root_1} + \mathbf{V}^{[2]}_{root_1}\big) \circ \mathbf{S}_{root_2} + \big(\mathbf{V}^{[1]}_{root_2} + \mathbf{V}^{[2]}_{root_2}\big) \circ \mathbf{S}_{root_1}\Big].\end{aligned} \tag{25}$$

Other quantities involved in the calculation of $\mathrm{Var}(H_\Omega|\mathbf{D})$ include the phylogenetic likelihood $\mathrm{P}(\mathbf{D})$ and the restricted mapping first moment $\mathrm{E}(H_\Omega \mathbb{1}_{\mathbf{D}})$. From (Felsenstein, 1981), we know that $\mathrm{P}(\mathbf{D}) = \boldsymbol{\pi}^T\mathbf{F}_{root}$. Minin and Suchard (2008b) express $\mathrm{E}(H_\Omega \mathbb{1}_{\mathbf{D}})$ in the following manner:

$$\mathrm{E}(H_\Omega \mathbb{1}_{\mathbf{D}}) = \sum_b \sum_{\mathbf{i}} e^{[1]}_{i^*_{p(b)} i^*_{c(b)}}(h, t_b)\pi_{i^*_1} \prod_{b^* \in \Theta \setminus \{b\}} p_{i^*_{p(b^*)} i^*_{c(b^*)}}(t_{b^*}), \tag{26}$$

where $b \in \Omega$. Notice that the right-hand side of equation (26) is virtually identical to the sum over $b \in \Omega$ in (8). Given our interpretations of the sums in (8) and (9), it is easy to see that:

$$\mathrm{E}(H_\Omega \mathbb{1}_{\mathbf{D}}) = \boldsymbol{\pi}^T\Big(\mathbf{V}^{[1]}_{root_1} \circ \mathbf{S}_{root_2} + \mathbf{V}^{[1]}_{root_2} \circ \mathbf{S}_{root_1}\Big). \tag{27}$$

Finally, the posterior mapping variance $\mathrm{Var}(H_\Omega|\mathbf{D})$ can be expressed as follows:

$$\mathrm{Var}(H_\Omega|\mathbf{D}) = \mathrm{E}(H^2_\Omega|\mathbf{D}) - \mathrm{E}(H_\Omega|\mathbf{D})^2 \tag{28}$$

$$= \frac{\mathrm{E}(H^2_\Omega \mathbb{1}_{\mathbf{D}})}{\mathrm{P}(\mathbf{D})} - \left[\frac{\mathrm{E}(H_\Omega \mathbb{1}_{\mathbf{D}})}{\mathrm{P}(\mathbf{D})}\right]^2. \tag{29}$$

We can compute $\mathrm{Var}(H_\Omega|\mathbf{D})$ by first calculating $\mathrm{P}(\mathbf{D})$, $\mathrm{E}(H_\Omega \mathbb{1}_{\mathbf{D}})$, and $\mathrm{E}(H^2_\Omega \mathbb{1}_{\mathbf{D}})$ using our post-order tree traversal algorithm and then substituting these quantities into equation (29).

3.4 Prior Mapping Variance Computation

Given our efficient calculation of the posterior mapping variance $\text{Var}(H_\Omega|\mathbf{D})$, it is natural to ask whether it is possible to extend the tree traversal algorithm described above to the computation of the prior mapping variance $\text{Var}(H_\Omega)$. There is a plethora of literature in evolutionary biology that discusses how to calculate the prior mapping variance for a single tree branch (Zheng, 2001; Bloom et al., 2007; Minin and Suchard, 2008a,b). Siepel et al. (2006) describe a dynamic programming procedure that approximates the probability distribution of H_Θ and use it to estimate $\text{Var}(H_\Theta)$.

It turns out that we can exactly compute prior variances using a modified version of our tree traversal algorithm. The only changes that need to be made are to the initializations of $\mathbf{F}_u$ at all terminal nodes u. If we set $F_{ui} = 1$ for all terminal nodes u and $i = 1, ..., m$, then our recursive procedure will be able to compute the prior variance $\text{Var}(H_\Omega)$. We also note that all elements of the vectors $\mathbf{F}_u$ and $\mathbf{S}_b$ for $u \in \{1, ..., n-1, n, ..., 2n-1\}$ and $b \in \Theta$ are equal to 1 as a result of these modified initializations. Remember that the initializations of $\mathbf{F}_u$ can account for partially observed and/or missing data at the tips of τ. These modified $\mathbf{F}_u$ initializations are intuitive because prior mapping moments average over all possible observed trait values at the tips of the phylogeny and thus every combination of observed tip states needs to be accounted for in our $\mathbf{F}_u$ initializations. As we will see, the prior mapping variance $\text{Var}(H_\Omega)$ is essential to one of the applications we present in this paper.

3.5 Prior and Posterior Mapping Covariances

Two other quantities of interest associated with stochastic mapping summaries are the prior mapping covariance $\text{Cov}(H_{\Omega_1}, H_{\Omega_2})$ and the posterior mapping covariance $\text{Cov}(H_{\Omega_1}, H_{\Omega_2}|\mathbf{D})$, where $\Omega_1, \Omega_2 \subseteq \Theta$ are predefined sets of branch indices. Efficient computations of the prior and posterior mapping covariances are interesting in their own right and utilized in one of the applications we present in this paper. In the Appendix, we describe another tree traversal algorithm for computing these covariances. This new algorithm is a generalization of the recursive procedure described above and much of the intuition provided for our original procedure carries over to this new algorithm. Furthermore, this new procedure runs in $O(n)$ time and with $O(n)$ storage.

3.6 Implementation

The efficient calculations of the prior and posterior mapping moments discussed above are implemented in the R package `phylomoments`, which is available at `https://github.com/dunleavy005/phylomoments`. This package also contains our implementation of the stochastic mapping simulation technique put forth by Nielsen (2002) along with other assorted functions. The computationally intensive parts of our methods are written in C++ and ported to R using the R packages `Rcpp` and `RcppArmadillo` (Eddelbuettel and François, 2011; Eddelbuettel and Sanderson, 2014). In the next section, we present two scientific applications that employ stochastic mapping variances.

4 Applications

4.1 Testing for Rate Variation Across Sites

Our first application is centered around an across-site rate variation test proposed by Nielsen (2002). Nielsen (2002) uses simulated posterior mapping variances within a posterior predictive approach to model diagnostics. In this subsection, we describe a posterior predictive testing framework that adheres to the principles outlined by Gelman et al. (1996) and test for across-site rate variation in two real datasets using exactly computed posterior mapping variances.

4.1.1 Overview of Posterior Predictive Tests

Conceptually, posterior predictive assessments can be seen as Bayesian analogues of classical frequentist model diagnostics and hypothesis tests. Unlike classical testing procedures, posterior predictive tests permit the use of "test statistics" that depend on both data and unknown parameters. These "test statistics" are otherwise known as discrepancy measures (Meng, 1994; Gelman et al., 1996). In this subsection, we denote the discrepancy measure as $T \equiv T(\mathbf{D}_{1:L}, \boldsymbol{\theta})$. Posterior predictive model testing is based on the following principle: if the assumed model adequately fits the observed data $\mathbf{D}_{1:L}^{obs}$, then simulated datasets $\mathbf{D}_{1:L}^{rep}$ from the assumed model should look like $\mathbf{D}_{1:L}^{obs}$. Similarity between $\mathbf{D}_{1:L}^{obs}$ and

$\mathbf{D}_{1:L}^{rep}$ is measured through the discrepancy T. We would like to compare the observed discrepancy $T(\mathbf{D}_{1:L}^{obs}, \boldsymbol{\theta})$ to a reference distribution induced by the hypothesized model. The reference distribution used in posterior predictive diagnostics is derived from the joint posterior distribution of $\mathbf{D}_{1:L}^{rep}$ and $\boldsymbol{\theta}$:

$$\mathrm{P}(\mathbf{D}_{1:L}^{rep}, \boldsymbol{\theta} | \mathbf{D}_{1:L}^{obs}) = \mathrm{P}(\mathbf{D}_{1:L}^{rep} | \boldsymbol{\theta}) \mathrm{P}(\boldsymbol{\theta} | \mathbf{D}_{1:L}^{obs}). \tag{30}$$

Intuitively, this distribution indicates which datasets and parameter values are most plausible if the assumed model holds true. Posterior predictive assessments are usually conducted using simulations as the joint distribution in (30) is often analytically intractable. We summarize posterior predictive simulations in the following three steps:

1. Sample $\boldsymbol{\theta}^* \sim \mathrm{P}(\boldsymbol{\theta} | \mathbf{D}_{1:L}^{obs})$;
2. Simulate $\mathbf{D}_{1:L}^{rep,*} \sim \mathrm{P}(\mathbf{D}_{1:L}^{rep} | \boldsymbol{\theta}^*)$;
3. Calculate $T(\mathbf{D}_{1:L}^{obs}, \boldsymbol{\theta}^*)$ and $T(\mathbf{D}_{1:L}^{rep,*}, \boldsymbol{\theta}^*)$.

We repeat these steps N times, where N is a large number, and then compare the N samples of $T(\mathbf{D}_{1:L}^{obs}, \boldsymbol{\theta}^*)$ and $T(\mathbf{D}_{1:L}^{rep,*}, \boldsymbol{\theta}^*)$ by constructing two empirical distributions; a small overlap between these distributions, which could be visualized with histograms, suggests a poor model fit. We can quantify the disagreement between the observed and predicted discrepancies by calculating the posterior predictive p-value (Meng, 1994; Gelman et al., 1996):

$$ppp = \mathrm{P}\big[T(\mathbf{D}_{1:L}^{rep}, \boldsymbol{\theta}) > T(\mathbf{D}_{1:L}^{obs}, \boldsymbol{\theta}) \big| \mathbf{D}_{1:L}^{obs}\big], \tag{31}$$

where the probability is computed with respect to the joint posterior distribution $\mathrm{P}(\mathbf{D}_{1:L}^{rep}, \boldsymbol{\theta} | \mathbf{D}_{1:L}^{obs})$. Given N posterior samples of $\mathbf{D}_{1:L}^{rep}$ and $\boldsymbol{\theta}$, we can estimate the posterior predictive p-value by computing:

$$ppp \approx \frac{1}{N} \sum_{i=1}^{N} \mathbb{1}_{\left\{T(\mathbf{D}_{1:L}^{rep,(i)}, \boldsymbol{\theta}^{(i)}) > T(\mathbf{D}_{1:L}^{obs}, \boldsymbol{\theta}^{(i)})\right\}}, \tag{32}$$

where $\mathbf{D}_{1:L}^{rep,(i)}$ and $\boldsymbol{\theta}^{(i)}$ represent the ith posterior samples of $\mathbf{D}_{1:L}^{rep}$ and $\boldsymbol{\theta}$, respectively, for $i = 1, ..., N$.

4.1.2 Posterior Predictive Rate Variation Tests

Now, we use the posterior predictive testing framework described above to formulate a test for rate variation across sites in an alignment. We assume that alignment sites evolve independently according to the same distribution provided by the continuous-time Markov process $\psi_{\boldsymbol{\theta}}$. We focus on selecting discrepancy measures that can gauge the variability in substitution rates across sites. One possible discrepancy measure is the variance of the total number of substitutions in the alignment (Nielsen, 2002):

$$T_{var} \equiv T_{var}(\mathbf{D}_{1:L}, \boldsymbol{\theta}) = \mathrm{Var}\left(\sum_{i=1}^{L} H_{\Theta}^{(i)} \middle| \mathbf{D}_{1:L} \right) = \sum_{i=1}^{L} \mathrm{Var}(H_\Theta | \mathbf{D}_i), \tag{33}$$

where $H_{\Theta}^{(i)}$ represents the number of substitutions at site i. Note that the second equality in (33) follows from conditional independence assumptions. Another discrepancy we consider in our analyses is the posterior dispersion index (i.e. posterior variance-to-mean ratio) for substitution counts:

$$\begin{aligned} T_{disp} \equiv T_{disp}(\mathbf{D}_{1:L}, \boldsymbol{\theta}) &= \mathrm{Var}\left(\sum_{i=1}^{L} H_{\Theta}^{(i)} \middle| \mathbf{D}_{1:L} \right) \Big/ \mathrm{E}\left(\sum_{i=1}^{L} H_{\Theta}^{(i)} \middle| \mathbf{D}_{1:L} \right) \\ &= \left[\sum_{i=1}^{L} \mathrm{Var}(H_\Theta | \mathbf{D}_i) \right] \Big/ \left[\sum_{i=1}^{L} \mathrm{E}(H_\Theta | \mathbf{D}_i) \right], \end{aligned} \tag{34}$$

where $H_{\Theta}^{(i)}$ is specified as above. In the presence of rate variation across sites, the prior dispersion index for substitution counts is often greater than 1 (Yang, 1996). One way to see this is by assuming that the number of substitutions occurring on a particular tree branch follows a Poisson distribution, where the Poisson rate parameter varies across sites according to a gamma distribution. A simple calculation shows that, marginally, the number of substitutions occurring on this branch follows a negative binomial distribution, which has a variance-to-mean ratio that is greater than 1. We can make a similar argument about the number of substitutions across an entire phylogeny. Even though the above explanation applies only to the prior dispersion index, it does motivate our use of the posterior dispersion index as a discrepancy measure for detecting across-site rate variation in an alignment.

The posterior predictive simulations for our rate variation test can be summarized using the same three steps described above. We can sample $\boldsymbol{\theta}^* \sim \mathrm{P}(\boldsymbol{\theta}|\mathbf{D}_{1:L}^{obs})$ using a Bayesian phylogenetic inference software package; in our examples, we use the computer program MrBayes (Huelsenbeck and Ronquist, 2001) to perform the posterior sampling of $\boldsymbol{\theta}$. For every posterior sample of $\boldsymbol{\theta}$, we can simulate a replicate alignment $\mathbf{D}_{1:L}^{rep,*} \sim \mathrm{P}(\mathbf{D}_{1:L}^{rep}|\boldsymbol{\theta}^*)$ by independently generating $\mathbf{D}_1^{rep,*}, ..., \mathbf{D}_L^{rep,*}$ according to $\psi_{\boldsymbol{\theta}^*}$; we simulate tip data from $\psi_{\boldsymbol{\theta}^*}$ using the standard discretized CTMC approach (Yang, 2006, Chapter 9). We can then analytically calculate the observed and predicted discrepancies (i.e. $T(\mathbf{D}_{1:L}^{obs}, \boldsymbol{\theta}^*)$ and $T(\mathbf{D}_{1:L}^{rep,*}, \boldsymbol{\theta}^*)$) for $T = T_{var}, T_{disp}$ using the algorithm presented in the previous section. Nielsen (2002) also used the discrepancy T_{var} in posterior predictive rate variation tests but could only obtain Monte Carlo estimates of T_{var}. Monte Carlo estimation of T_{var} is computationally intensive because it uses simulations to estimate $\mathrm{Var}(H_\Theta|\mathbf{D}_i)$ for $i = 1, ..., L$. The tree traversal algorithm proposed in this paper not only eliminates the Monte Carlo error associated with estimating the variance T_{var} but also speeds up the computation of this quantity.

4.1.3 Testing for Rate Variation in Two Sequence Alignments

We analyze the two sequence alignments used by Nielsen (2002) to demonstrate the effectiveness of our posterior predictive testing scheme. Our first dataset contains β-globin sequences for 17 vertebrate species, where each sequence is 432 base pairs long. Our second dataset comprises 28 sequences of the hemagglutinin (HA) gene of human influenza virus A; each sequence has 987 base pairs. In all our analyses, we use a general time-reversible (GTR) substitution model (Tavaré, 1986) with a Dirichlet(1,1,1,1,1,1) prior for the GTR exchangeability rates and a Dirichlet(1,1,1,1) prior for the base frequencies $\boldsymbol{\pi}$. Furthermore, we assume a uniform prior on all possible tree topologies τ and let all branch lengths in $\mathbf{t}$ be a priori uniformly distributed on the interval $[0, 100]$. For each dataset, we generate $N = 1000$ posterior samples of $\mathbf{D}_{1:L}^{rep}$ and $\boldsymbol{\theta}$ using the simulation procedure described previously and calculate the observed and predicted values of T_{var} and T_{disp}. The 1000 posterior samples of $\boldsymbol{\theta}$ are obtained by running the Markov chain Monte Carlo (MCMC) procedure in MrBayes for 1,100,000 iterations and storing values of $\boldsymbol{\theta}$ every 1000 iterations from iteration 100,001 to iteration 1,100,000. We use trace plots (not shown in this paper) to assess convergence of the MCMC samplers and find that using 1,100,000 MCMC iterations is sufficient for our purposes.

We summarize the observed and predicted discrepancies in Figure 3, which demonstrates that the observed distributions of T_{var} do not deviate much from the corresponding predicted distributions of T_{var}. The posterior predictive p-values that were computed using the discrepancy T_{var} are approximately 0.60 and 0.80 for the β-globin and influenza datasets, respectively. Thus, it seems that our posterior predictive test based on T_{var} fails to detect across-site rate variation in the two datasets. We note that our posterior predictive test results based on T_{var} do not corroborate the findings by Nielsen (2002), who concluded that there is rate variation across sites in the β-globin and influenza datasets. However, Nielsen (2002) used the posterior variance of substitution counts in a slightly different way without clearly specifying all predictive distributions. We believe our analyses are better aligned with the posterior predictive principles outlined by Gelman et al. (1996).

In Figure 3, the observed and predicted distributions of T_{disp} do not completely overlap and appear more separated than the observed and predicted distributions of T_{var}. For both datasets, the observed values of T_{disp} are, on average, greater than the predicted values of T_{disp}. This suggests that the discrepancy T_{disp} is able to detect observed rate variation that is not accounted for by our hypothesized model. The posterior predictive p-values that were computed using T_{disp} are approximately 0.17 and 0.031 for the β-globin and influenza datasets, respectively; note that these p-values are smaller than the corresponding p-values that were computed using T_{var} and as a result provide stronger evidence in support of the rate variation hypothesis.

A popular frequentist approach to modeling rate variation among sites employs a discrete gamma distribution with a fixed number of rate classes (Yang, 1994, 1996). We check our posterior predictive test results by performing likelihood ratio tests that compare discrete gamma models with one rate category (H_0) and four rate categories (H_a). We perform the likelihood ratio tests using the PhyML package (Guindon et al., 2010) and obtain p-values close to 0; these tests also suggest the presence of across-site rate variation in the β-globin and influenza datasets. The posterior predictive p-values that

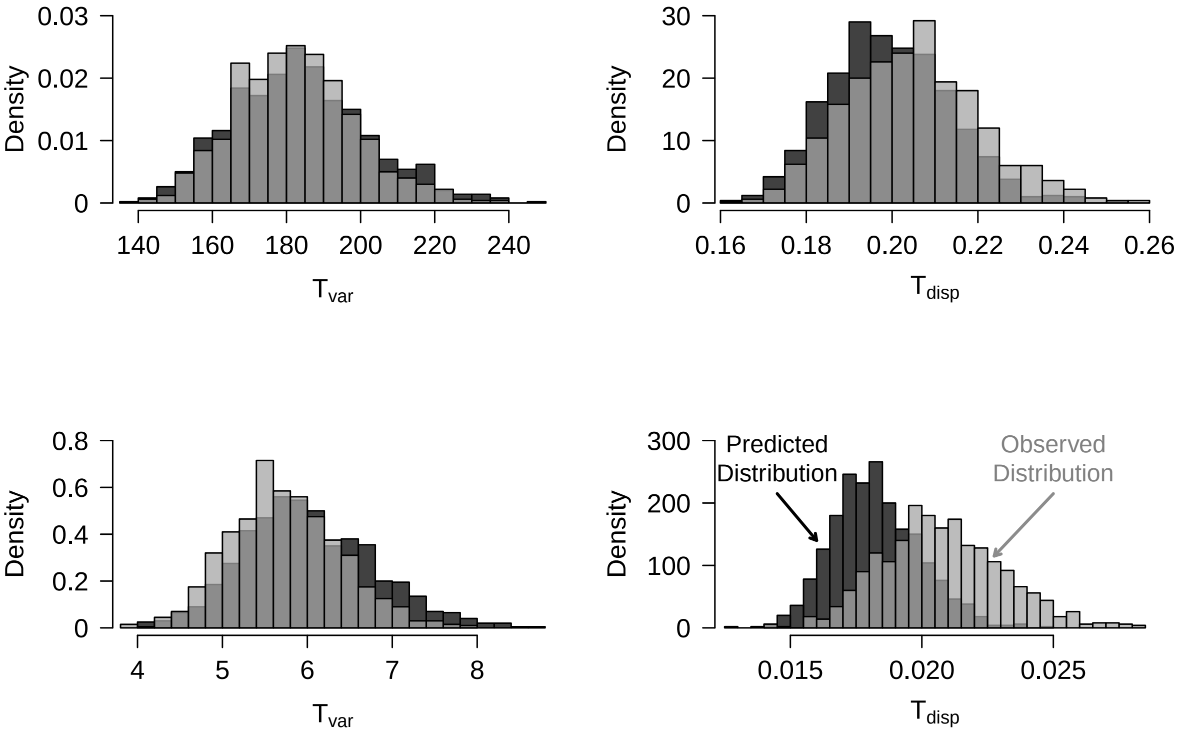


Figure 3: Observed and predicted distributions of the discrepancies T_{var} and T_{disp}. In each plot, we superimpose the observed distribution (grey) on top of the predicted distribution (black). The top two plots display the observed and predicted distributions for the β-globin dataset, while the bottom two plots show the observed and predicted distributions for the influenza dataset. These distributions were constructed using $N = 1000$ posterior samples of $\mathbf{D}_{1:L}^{rep}$ and $\boldsymbol{\theta}$.

were computed using T_{disp} are not as small as the likelihood ratio p-values, but this should not be surprising because posterior predictive p-values tend to be more conservative than classical frequentist p-values (Meng, 1994). Our posterior predictive analyses suggest that the posterior dispersion index T_{disp} is better than the posterior variance T_{var} at detecting observed rate variation among sites.

4.1.4 Monte Carlo Error and Timing Analyses

To assess the efficiency gains from using our tree traversal algorithm in this setting, we approximate Monte Carlo standard errors and running times associated with simulation-based T_{var} estimates. For each dataset, we compute these standard errors and running times on randomly subsampled alignments of length L using m Monte Carlo replicates per site, where $L = 50, 100, 200, 400$ and $m = 100, 500, 1000, 10000$; the Monte Carlo standard errors are approximated using well-known formulas for the moments of the sample variance (Mood et al., 1974, Chapter VI). We account for the posterior uncertainty in $\boldsymbol{\theta}$ by first calculating Monte Carlo errors and running times for 200 randomly subsampled posterior $\boldsymbol{\theta}$'s and then averaging these metrics across the $\boldsymbol{\theta}$ samples; the same 200 samples of $\boldsymbol{\theta}$ are used in all our simulations. For each setting of L, we also compute the exact values of T_{var} and track the corresponding running times for the 200 posterior samples of $\boldsymbol{\theta}$ and then average these results over the subsampled $\boldsymbol{\theta}$'s.

Tables 1 and 2 present the running time comparisons and Monte Carlo error approximations, respectively, for the β-globin and influenza datasets; Table 2 can be found in the Appendix. Based on the results shown in Table 2, it seems that the Monte Carlo error has a convergence rate of $O\left(\sqrt{L/m}\right)$. This can be justified using a Central Limit Theorem argument if L and m are large; remember that the Monte Carlo estimator of T_{var} is a sum of L independent, non-identically distributed sample variances, where each sample variance is calculated using m independent Monte Carlo replicates. Furthermore, the Monte Carlo error percentages range from 0.15% to 28% across the two datasets. Table 1 suggests that the Monte Carlo running time increases linearly in L and m as we might expect. We can also see that our exact computations of T_{var} are at least an order of magnitude faster than the Monte Carlo estimates of

T_{var}. Thus, it is clear that our tree traversal algorithm improves the computational efficiency of posterior predictive rate variation tests that utilize Monte Carlo discrepancy estimates.

	Running times (β-globin)				Running times (influenza)			
	$L = 50$	$L = 100$	$L = 200$	$L = 400$	$L = 50$	$L = 100$	$L = 200$	$L = 400$
$m = 100$	0.70	1.4	3.1	6.1	1.0	2.0	4.1	8.0
$m = 500$	3.3	6.5	15	28	4.7	9.3	19	37
$m = 1000$	7.2	13	28	57	9.0	18	37	74
$m = 10000$	72	140	290	570	93	180	370	740
Exact	0.0040	0.0059	0.0097	0.017	0.0061	0.0091	0.015	0.026

Table 1: Monte Carlo running times associated with simulation-based T_{var} estimates for the β-globin and influenza datasets. We compute these running times on randomly subsampled alignments of length L using m Monte Carlo replicates per site. Each table entry, excluding the entries on the bottom row, represents an averaged Monte Carlo running time (in seconds), where the averaging is done over 200 randomly subsampled posterior $\boldsymbol{\theta}$'s. Each table entry on the bottom row denotes an average over running times (in seconds) associated with exact calculations of T_{var}, where the averaging is done over the same 200 posterior samples of $\boldsymbol{\theta}$ mentioned previously. All table entries are rounded to two significant digits.

4.2 Detecting Evolutionarily Conserved Regions in Genomic Alignments

Our second application focuses on the detection of conserved elements in genomic alignments; in this setting, conservation refers to evolution that is slower than expected. Detection of conserved DNA sites is of prime interest in comparative genomics because most of the conserved elements in genome-wide sequence alignments are believed to be caused by negative selection and to have evolutionarily important biological functions (Siepel et al., 2005). Computational methods for detecting conserved genomic regions are essential because they are used to flag candidate functional elements, which can then be further examined experimentally (Siepel et al., 2006).

In this subsection, we analyze two statistical tests of conservation presented by Siepel et al. (2006). One test is used to detect conservation across all lineages in a phylogeny, while the other is used to identify lineage-specific conservation; both tests are referred to as SPH conservation tests and implemented in the computer program phyloP (Pollard et al., 2010). Our exact calculations of prior and posterior mapping moments can be used to make the SPH conservation tests more powerful. We present some modifications to these conservation tests and demonstrate the efficacy of our proposed changes via simulations.

4.2.1 Modifying the SPH Conservation Tests

Let $\psi_{\boldsymbol{\theta}_{\text{neut}}}$ denote the baseline neutral evolutionary model that is assumed to be given; neutral models are commonly estimated using fourfold degenerate sites extracted from genome-wide sequence alignments of interest (Pollard et al., 2010). As in (Pollard et al., 2010), we define $\psi(\rho, \lambda; \Theta_b)$ to be a scaled evolutionary model that is identical to $\psi_{\boldsymbol{\theta}_{\text{neut}}}$ except that it has all its branch lengths scaled by the factor $\rho \in [0, 1]$ and the branch lengths in the subtree defined by Θ_b additionally scaled by the factor $\lambda \in [0, 1]$. For convenience, we let $\psi(\rho) \equiv \psi(\rho, \lambda = 1; \Theta_b)$ for all $b \in \Theta$ and $\rho \in [0, 1]$. Throughout this subsection, we assume that $\mathbf{D}_1, ..., \mathbf{D}_L$ are independent and identically distributed according to $\psi(\cdot)$. The two tests of conservation described in (Siepel et al., 2006) reduce to tests of the models that constrain parameters ρ and λ to particular values.

The SPH "all-branch" conservation test examines conservation across all branches of the phylogeny. Specifically, it tests the null hypothesis $H_0 : \rho = 1$ against the alternative hypothesis $H_a : \rho < 1$ for the evolutionary model $\psi(\rho)$. Siepel et al. (2006) use the following test statistic to distinguish between the two hypotheses:

$$\mathcal{T}_{all} \equiv \mathcal{T}_{all}(\mathbf{D}_{1:L}) = \mathrm{E}\left(\sum_{i=1}^{L} H_{\Theta}^{(i)} \middle| \mathbf{D}_{1:L} \right) = \sum_{i=1}^{L} \mathrm{E}(H_\Theta | \mathbf{D}_i), \tag{35}$$

where $H_{\Theta}^{(i)}$ represents the number of substitutions at site i. The test statistic in (35) serves as a proxy for the "observed" number of substitutions in the genomic alignment $\mathbf{D}_{1:L}$. The prior distribution of $\sum_{i=1}^{L} H_{\Theta}^{(i)}$ is taken to be the null distribution, and the p-value of this test is obtained by first comparing the observed value of $\mathcal{T}_{all}$ to this prior distribution and then computing the corresponding left-tail probability. It turns out that the p-values obtained from this testing procedure are not uniformly distributed and tend to be conservative under the null hypothesis (Siepel et al., 2006).

To understand why this occurs, we must examine the null distribution chosen for this test. Notice first that $\mathcal{T}_{all}$ is a sum of independent and identically distributed random variables with mean $\mathrm{E}[\mathrm{E}(H_\Theta|\mathbf{D})] = \mathrm{E}(H_\Theta)$ and variance $\mathrm{Var}[\mathrm{E}(H_\Theta|\mathbf{D})] = \mathrm{Var}(H_\Theta) - \mathrm{E}[\mathrm{Var}(H_\Theta|\mathbf{D})]$. For L large, we can invoke the Central Limit Theorem and approximate the sampling distribution of $\mathcal{T}_{all}$ with a normal distribution having mean $L \times \mathrm{E}(H_\Theta)$ and variance $L \times \mathrm{Var}(H_\Theta) - L \times \mathrm{E}[\mathrm{Var}(H_\Theta|\mathbf{D})]$. Using a similar asymptotic argument as above, we can also approximate the null distribution of $\sum_{i=1}^{L} H_\Theta^{(i)}$ — an unobserved quantity — with a normal distribution having mean $L \times \mathrm{E}(H_\Theta)$ and variance $L \times \mathrm{Var}(H_\Theta)$. Siepel et al. (2006) use the exact version of this distribution, which results in the conservative nature of their all-branch test, because the correct null distribution of $\mathcal{T}_{all}$, at least asymptotically, has a variance that is smaller than the one assumed by the authors. For a given significance level, an overdispersed null distribution causes the critical value for rejecting H_0 to be more extreme than it would be for a proper null distribution. This is of practical importance because more extreme critical values make it harder to correctly flag conserved genomic elements.

The SPH all-branch conservation test can be corrected and made more powerful by using the correct asymptotic distribution of $\mathcal{T}_{all}$ as the null distribution. The dynamic programming algorithm discussed in this paper can be used to calculate the mean and variance of this asymptotic distribution. The quantities $\mathrm{E}(H_\Theta)$ and $\mathrm{Var}(H_\Theta)$ are easily computed using our results for prior mapping moments. Given our efficient computation of the posterior mapping variance $\mathrm{Var}(H_\Theta|\mathbf{D})$, we estimate $\mathrm{E}[\mathrm{Var}(H_\Theta|\mathbf{D})]$ using Monte Carlo simulation of sequence data $\mathbf{D}$. Even though we need Monte Carlo simulations to calculate the asymptotic variance of $\mathcal{T}_{all}$, our approach is still more efficient than directly estimating $\mathrm{Var}[\mathrm{E}(H_\Theta|\mathbf{D})]$ via Monte Carlo sampling.

Siepel et al. (2006) also describe two testing procedures that are used to detect lineage-specific conservation. Both procedures analyze conservation at the subtree level and are referred to as SPH "subtree" tests. Formally, these two approaches test the null hypothesis $H_0 : \lambda = 1, \rho \in [0,1]$ against the alternative hypothesis $H_a : \lambda < 1, \rho \in [0,1]$ for the evolutionary model $\psi(\rho, \lambda; \Theta_b)$. The following test statistic is used in both procedures:

$$\mathcal{T}_{sub} \equiv \mathcal{T}_{sub}(\mathbf{D}_{1:L}) = \mathrm{E}\left(\sum_{i=1}^{L} H_{\Theta_b}^{(i)} \middle| \mathbf{D}_{1:L}\right) = \sum_{i=1}^{L} \mathrm{E}(H_{\Theta_b}|\mathbf{D}_i), \tag{36}$$

where $H_{\Theta_b}^{(i)}$ denotes the number of substitutions in the subtree defined by Θ_b at site i. Note that the statistic in (36) is the subtree equivalent of that shown in (35). The SPH "marginal" subtree test compares the observed value of $\mathcal{T}_{sub}$ to the marginal distribution of $\sum_{i=1}^{L} H_{\Theta_b}^{(i)}$, while the SPH "conditional" subtree test compares the observed value of $\mathcal{T}_{sub}$ to the conditional distribution of $\sum_{i=1}^{L} H_{\Theta_b}^{(i)}$ given $\sum_{i=1}^{L} H_\Theta^{(i)}$ is equal to the observed value of $\mathcal{T}_{all}$. Conceptually, the marginal subtree test examines whether the number of substitutions in the subtree is less than would be expected under the null model, whereas the conditional subtree test analyzes whether the number of substitutions in the subtree is surprising given the total number of substitutions in the tree. Even though these subtree tests are intuitively appealing, they suffer from the same problems discussed previously for the all-branch test (Siepel et al., 2006).

The SPH marginal subtree test can be corrected and made more powerful by using the correct asymptotic distribution of $\mathcal{T}_{sub}$ as the null distribution; this asymptotic distribution is obtained using reasoning similar to that used for the asymptotic distribution of $\mathcal{T}_{all}$. We propose our own conditional subtree test based on the following test statistic:

$$\begin{aligned}\mathcal{T}_{ratio} \equiv \mathcal{T}_{ratio}(\mathbf{D}_{1:L}) = \frac{\mathcal{T}_{sub}(\mathbf{D}_{1:L})}{\mathcal{T}_{all}(\mathbf{D}_{1:L})} &= \mathrm{E}\left(\sum_{i=1}^{L} H_{\Theta_b}^{(i)} \middle| \mathbf{D}_{1:L}\right) \Big/ \mathrm{E}\left(\sum_{i=1}^{L} H_{\Theta}^{(i)} \middle| \mathbf{D}_{1:L}\right) \\ &= \left[\sum_{i=1}^{L} \mathrm{E}(H_{\Theta_b}|\mathbf{D}_i)\right] \Big/ \left[\sum_{i=1}^{L} \mathrm{E}(H_{\Theta}|\mathbf{D}_i)\right],\end{aligned} \tag{37}$$

where $H_{\Theta_b}^{(i)}$ and $H_\Theta^{(i)}$ are defined as previously. This test statistic serves as a proxy for the "observed" proportion of substitutions in the subtree defined by Θ_b across the alignment $\mathbf{D}_{1:L}$. Using the Central Limit Theorem and the Delta Method, we approximate the sampling distribution of $\mathcal{T}_{ratio}$ with a normal distribution whose moments depend on $\mathrm{E}(H_{\Theta_b})$, $\mathrm{E}(H_\Theta)$, $\mathrm{Var}[\mathrm{E}(H_{\Theta_b}|\mathbf{D})]$, $\mathrm{Var}[\mathrm{E}(H_\Theta|\mathbf{D})]$, and $\mathrm{Cov}[\mathrm{E}(H_{\Theta_b}|\mathbf{D}), \mathrm{E}(H_\Theta|\mathbf{D})]$; we omit the exact forms of the mean and variance of this asymptotic distribution for brevity. By the Laws of Total Variance and Covariance, we can express $\mathrm{Var}[\mathrm{E}(H_{\Theta_b}|\mathbf{D})]$, $\mathrm{Var}[\mathrm{E}(H_\Theta|\mathbf{D})]$, and $\mathrm{Cov}[\mathrm{E}(H_{\Theta_b}|\mathbf{D}), \mathrm{E}(H_\Theta|\mathbf{D})]$ as follows:

$$\mathrm{Var}[\mathrm{E}(H_{\Theta_b}|\mathbf{D})] = \mathrm{Var}(H_{\Theta_b}) - \mathrm{E}[\mathrm{Var}(H_{\Theta_b}|\mathbf{D})], \tag{38}$$

$$\text{Var}[\text{E}(H_\Theta|\mathbf{D})] = \text{Var}(H_\Theta) - \text{E}[\text{Var}(H_\Theta|\mathbf{D})], \tag{39}$$

$$\text{Cov}[\text{E}(H_{\Theta_b}|\mathbf{D}), \text{E}(H_\Theta|\mathbf{D})] = \text{Cov}(H_{\Theta_b}, H_\Theta) - \text{E}[\text{Cov}(H_{\Theta_b}, H_\Theta|\mathbf{D})]. \tag{40}$$

The prior moments $\text{E}(H_{\Theta_b})$, $\text{E}(H_\Theta)$, $\text{Var}(H_{\Theta_b})$, $\text{Var}(H_\Theta)$, and $\text{Cov}(H_{\Theta_b}, H_\Theta)$ are efficiently computed using the post-order tree traversal algorithm outlined in the Appendix. The quantities $\text{E}[\text{Var}(H_{\Theta_b}|\mathbf{D})]$, $\text{E}[\text{Var}(H_\Theta|\mathbf{D})]$, and $\text{E}[\text{Cov}(H_{\Theta_b}, H_\Theta|\mathbf{D})]$ are approximated using Monte Carlo replicates of $\mathbf{D}$ and our exact calculations of posterior mapping variances and covariances. For both of our lineage-specific conservation tests, we estimate the global scale parameter ρ by numerically maximizing the observed log-likelihood function. In the next subsubsection, we present simulation results that demonstrate the utility of our modified SPH conservation tests.

4.2.2 Simulation Experiments

We evaluate the power and false positive rates of the original and modified SPH conservation tests using simulated alignments. The neutral evolutionary model used by Pollard et al. (2010) is also employed in all our simulation experiments. This model was estimated using fourfold degenerate sites extracted from alignments of the 44 ENCODE regions (Birney et al., 2007) for 36 vertebrate species. For the all-branch and subtree tests, we simulate replicate alignments by independently generating L alignment columns according to $\psi(\rho)$ and $\psi(\rho, \lambda; \Theta_{primates})$, respectively, where *primates* denotes the branch above the primates subtree in the neutral phylogeny. We consider $L = 1, 2, 4, ..., 48, 50$ and $\rho = 0.1, 0.3, 0.5, 0.7, 0.9, 1$ in our all-branch simulations and $L = 1, 5, 10, ..., 45, 50$; $\rho = 0.1, 0.25, 0.4, ..., 1$; and $\lambda = 0.1, 0.25, 0.4, ..., 1$ in our subtree simulations. We keep the alignment length L relatively small because the primary application of SPH tests is scanning whole genomes in search of short ultra-conserved genetic elements. For each simulation setting under H_a, we generate 1000 replicate datasets, compute the conservation p-values for each dataset using the original and modified SPH tests, and estimate the power by calculating the proportion of p-values less than the given significance level; we construct power curves by varying the significance threshold between 0 and 1. False positive rates are similarly estimated for each simulation setting under H_0. Note that these power curves should not be confused with receiver operating characteristic (ROC) curves; in our simulations, we plot the power (i.e. true positive rates) against the significance levels, whereas ROC curves plot the power against the false positive rates.

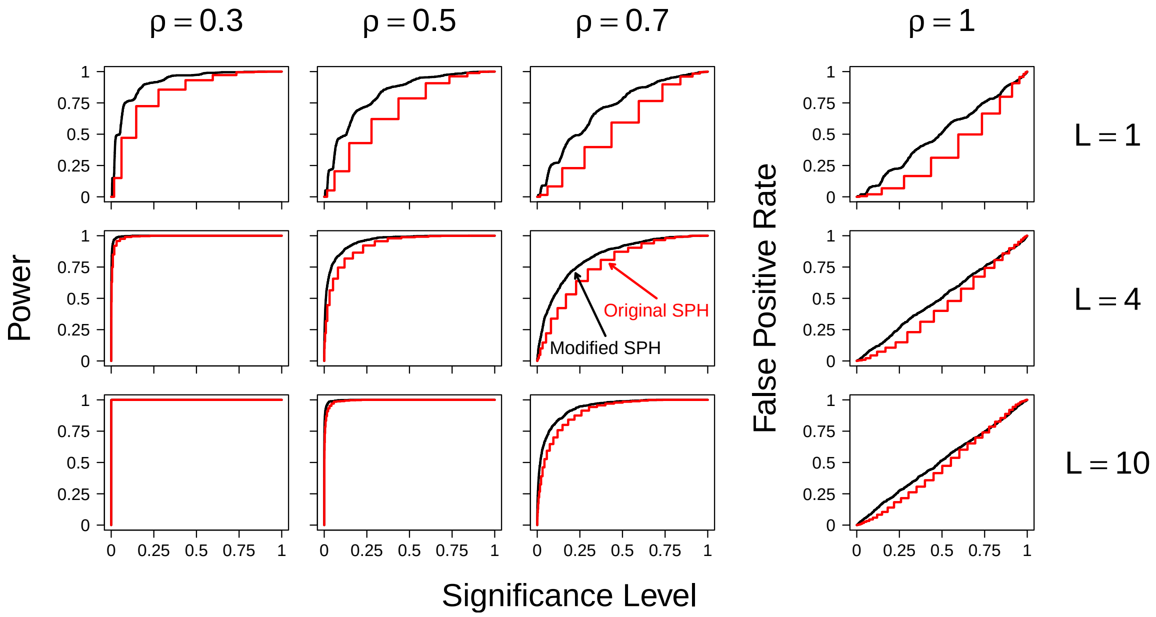
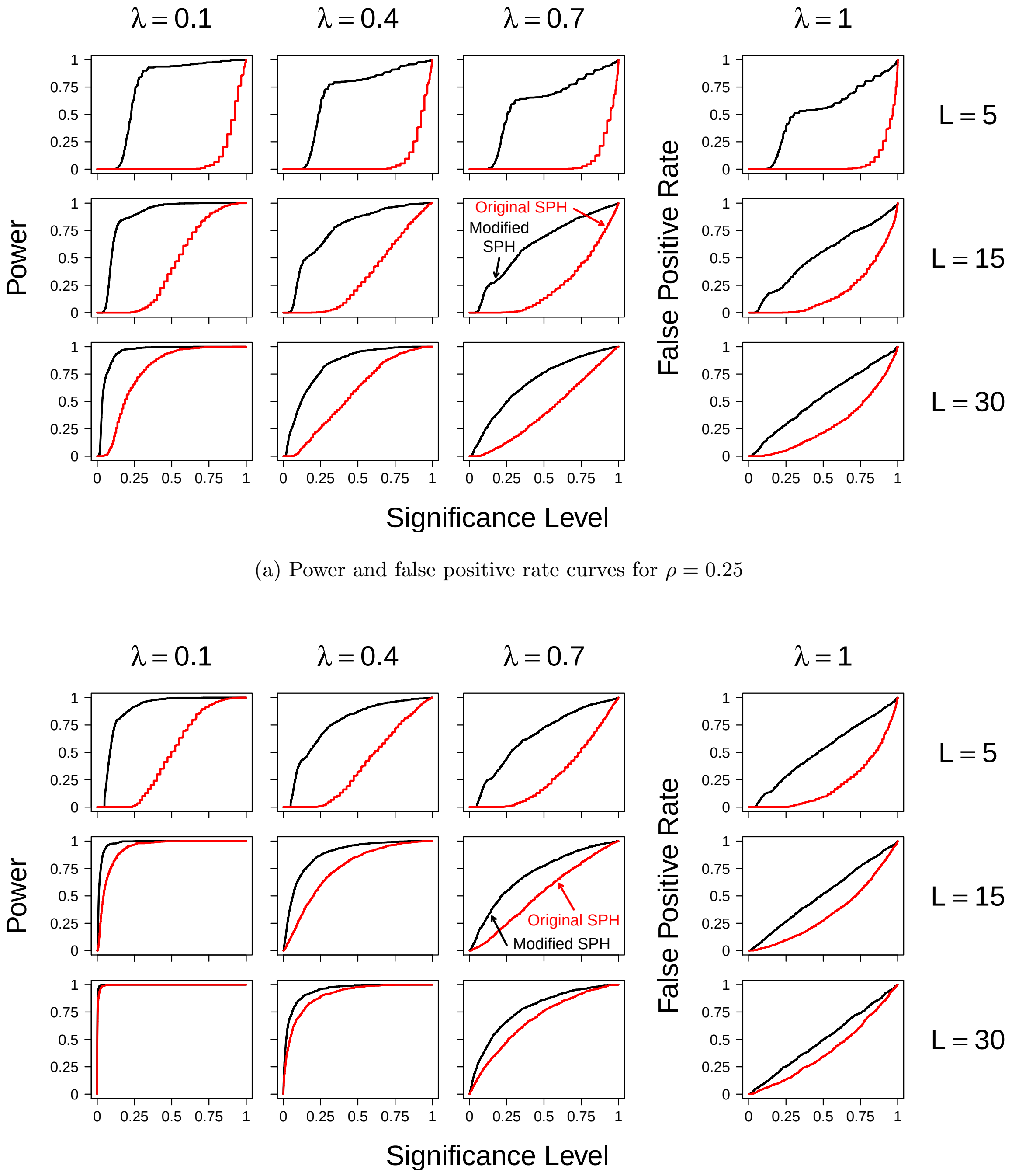


Figure 4: Power and false positive rate plots from our all-branch simulation experiments. The power and false positive rate curves for the original SPH all-branch test are shown in red, while the corresponding performance curves for the modified SPH all-branch test are displayed in black. In this figure, we present performance plots for $L = 1, 4, 10$ and $\rho = 0.3, 0.5, 0.7, 1$.

In Figure 4, we display some of the power and false positive rate plots from our all-branch simulation experiments. Specifically, we present performance plots for $L = 1, 4, 10$ and $\rho = 0.3, 0.5, 0.7, 1$. These plots suggest that the modified SPH all-branch test is more powerful than the original SPH all-branch test. The gap between the power curves for the two tests is negligible for large L and small ρ but increases as we examine shorter alignments with lower levels of conservation. This latter result is surprising because the null distribution used in the modified SPH all-branch test is based on an asymptotic approximation. The false positive rate plots seem to indicate that the p-values obtained from the modified all-branch test are approximately uniformly distributed under the null hypothesis, even for small L. In addition, it is apparent that the original SPH all-branch p-values are conservative under H_0, confirming the results found in (Siepel et al., 2006).



(a) Power and false positive rate curves for $\rho = 0.25$

(b) Power and false positive rate curves for $\rho = 0.85$

Figure 5: Power and false positive rate plots from our subtree simulation experiments. The power and false positive rate curves for the original SPH conditional subtree test are shown in red, while the corresponding performance curves for the modified SPH conditional subtree test are displayed in black. In this figure, we present performance plots for $L = 5, 15, 30$; $\rho = 0.25, 0.85$; and $\lambda = 0.1, 0.4, 0.7, 1$.

Figure 5 presents power and false positive rate plots from our subtree simulations. We provide performance plots for $L = 5, 15, 30$; $\rho = 0.25, 0.85$; and $\lambda = 0.1, 0.4, 0.7, 1$. The modified SPH subtree tests are more powerful than the original SPH subtree tests in all our simulation experiments. Furthermore, we find that the power curves for the two modified subtree tests are nearly identical; the power curves for the two original subtree tests are similar as well. We only display the power curves for the conditional subtree tests in Figure 5; the full set of power curves is shown in Figure 6 (see Appendix).

Our subtree simulation experiments suggest that the presence of strong phylogeny-wide conservation (as measured by ρ) makes it more difficult to correctly identify lineage-specific conservation. For ρ close to 1, the separation between the power curves for the original and modified subtree tests is minimal when L is large and λ is small but widens as we analyze shorter elements with lower levels of primate-specific conservation. However, for ρ close to 0, the power curves differ quite substantially for all settings of L and λ. The power of the modified subtree tests is more robust to changes in ρ, while the power of the original subtree tests diminishes greatly as ρ decreases. Thus, it appears that the modified subtree tests can more accurately detect lineage-specific conservation in the presence of strong phylogeny-wide conservation.

Our results show that testing for conservation in a subtree of interest is more difficult than testing for conservation across the entire phylogeny. This is not too surprising because we have to account for the uncertainty associated with estimating ρ in the subtree tests and the modified subtree test statistics achieve asymptotic normality under H_0 at a rate slower than is observed for the modified all-branch test statistic. The latter can be seen by comparing the false positive rate plots in Figures 4 and 5; additionally, these plots indicate that the original SPH subtree p-values are more conservative than the original SPH all-branch p-values.

5 Discussion

In this paper, we present a post-order tree traversal algorithm that computes prior and posterior stochastic mapping variances with space and time complexity linear in the number of tips on the phylogeny; prior and posterior mapping covariances are efficiently calculated using a generalized version of this algorithm (see Appendix). In many applications, including the ones presented in this paper, the posterior distribution of stochastic mapping summaries can be approximated by a normal distribution. Since the normal distribution is fully specified by its mean vector and covariance matrix, our new stochastic mapping (co)variance computation together with an already available efficient way of computing the stochastic mapping mean enables access to the full posterior distribution of stochastic mapping summaries without resorting to costly simulations. Our methodology builds upon the results of Minin and Suchard (2008b) and is inspired by the work of Kenney and Gu (2012), who devised a dynamic programming procedure for calculating second derivatives of phylogenetic likelihood functions.

In fact, our algorithm for computing prior and posterior mapping moments can be adapted to provide a more straightforward description of the Kenney and Gu (2012) algorithm. If we replace the restricted factorial moments in the recursive equations of our algorithm with the appropriate derivatives of CTMC transition probabilities, then we would obtain an algorithm that computes the second derivatives of interest. Even though this reformulation is equivalent to the approach of Kenney and Gu (2012), we believe our presentation is more streamlined and easier to follow. We also point out that our algorithm and the work by Kenney and Gu (2012) can be viewed as extensions of analogous calculations for hidden Markov models (Lystig and Hughes, 2002; Cappé and Moulines, 2005).

The exact calculations of prior and posterior mapping moments allow us to construct more efficient posterior predictive rate variation tests and more accurate tests of genomic conservation. From our analyses of the β-globin and influenza datasets, we find that the posterior dispersion index for substitution counts is a useful discrepancy measure for detecting observed rate variation across sites. We motivate our use of the posterior dispersion index as a discrepancy measure by alluding to the prior dispersion index for substitution counts often being greater than 1 in the presence of rate variation among sites. The observed and predicted distributions of the posterior dispersion index in Figure 3 do not have support outside the interval $(0, 1)$, but this is not entirely inconsistent with the reasoning given above because a simple limiting argument shows that the posterior dispersion index for long sequence alignments is less than or equal to the prior dispersion index, even in the presence of across-site rate variation. From our all-branch and subtree simulation experiments, we see that the modified SPH tests are better than the original SPH tests at correctly identifying phylogeny-wide and lineage-specific conservation in genomic sequence alignments. Specifically, we observe that the differences in power between the original and modified conservation tests are greatest when we analyze short elements with low levels of conservation.

Similarly to the techniques found in (Kellis et al., 2003), our modified tests of conservation could aid in the discovery of new transcriptional regulatory motifs in the human genome.

The work presented here can be extended in several different directions. One obvious extension is to generalize our post-order tree traversal algorithm to calculate higher-order moments of stochastic mapping summaries, such as the (co)skewness and (co)kurtosis. These higher-order moments could then be used to construct more complex discrepancy measures for posterior predictive model diagnostics. Similarly to the test of rate variation among sites, it is of interest to develop a posterior predictive approach to testing for rate variation among branches of the phylogeny. Posterior predictive tests of this type could serve as useful diagnostic tools to assess the appropriateness of relaxed molecular clock models (Drummond et al., 2006). In addition, it would also be beneficial to establish a posterior predictive framework for testing the stationarity and homogeneity assumptions implicit in reversible substitution models. These tests could be used to determine the suitability of nonreversible substitution models (Boussau and Gouy, 2006). Our dynamic programming algorithm can be easily altered to compute the posterior mapping variance of labeled dwelling times, which could be employed as a discrepancy measure in these posterior predictive tests. Finally, another potential avenue for future research is to improve the two tests of genomic acceleration discussed by Pollard et al. (2010), where acceleration refers to evolution that is faster than expected. Pursuing this line of research could lead to the detection of new "human accelerated regions" in our genome (Pollard et al., 2006a,b). Based on the results in this paper and the promising directions for further study, we think stochastic mapping is and will continue to be essential to making reliable inferences about the latent evolutionary process on the phylogeny.

Acknowledgments

We thank Melissa Hubisz for providing us access to the neutral evolutionary model used in (Pollard et al., 2010) and answering our questions about the R package `rphast`. VNM was supported in part by the National Institute of Health grant R01-AI107034.

References

Ball, F. and Milne, R. K. (2005). Simple derivations of properties of counting processes associated with Markov renewal processes. Journal of Applied Probability, 42(4):1031–1043.

Birney, E., Stamatoyannopoulos, J. A., Dutta, A., Guigó, R., Gingeras, T. R., Margulies, E. H., Weng, Z., Snyder, M., Dermitzakis, E. T., Thurman, R. E., et al. (2007). Identification and analysis of functional elements in 1% of the human genome by the ENCODE pilot project. Nature, 447(7146):799–816.

Bloom, J. D., Raval, A., and Wilke, C. O. (2007). Thermodynamics of neutral protein evolution. Genetics, 175(1):255–266.

Boussau, B. and Gouy, M. (2006). Efficient likelihood computations with nonreversible models of evolution. Systematic Biology, 55(5):756–768.

Cappé, O. and Moulines, E. (2005). Recursive computation of the score and observed information matrix in hidden Markov models. In IEEE/SP 13th Workshop on Statistical Signal Processing, 2005, pages 703–708. IEEE.

Dimmic, M. W., Hubisz, M. J., Bustamante, C. D., and Nielsen, R. (2005). Detecting coevolving amino acid sites using Bayesian mutational mapping. Bioinformatics, 21(1):i126–i135.

Drummond, A. J., Ho, S. Y. W., Phillips, M. J., and Rambaut, A. (2006). Relaxed phylogenetics and dating with confidence. PLoS Biology, 4(5):e88.

Dutheil, J., Pupko, T., Jean-Marie, A., and Galtier, N. (2005). A model-based approach for detecting coevolving positions in a molecule. Molecular Biology and Evolution, 22(9):1919–1928.

Eddelbuettel, D. and François, R. (2011). Rcpp: Seamless R and C++ integration. Journal of Statistical Software, 40(8):1–18.

Eddelbuettel, D. and Sanderson, C. (2014). RcppArmadillo: Accelerating R with high-performance C++ linear algebra. Computational Statistics and Data Analysis, 71:1054–1063.

Felsenstein, J. (1981). Evolutionary trees from DNA sequences: A maximum likelihood approach. Journal of Molecular Evolution, 17(6):368–376.

Felsenstein, J. (2004). Inferring Phylogenies. Sinauer Associates, Sunderland, Massachusetts.

Gelman, A., Meng, X.-L., and Stern, H. (1996). Posterior predictive assessment of model fitness via realized discrepancies. Statistica Sinica, 6:733–807.

Guindon, S., Dufayard, J.-F., Lefort, V., Anisimova, M., Hordijk, W., and Gascuel, O. (2010). New algorithms and methods to estimate maximum-likelihood phylogenies: Assessing the performance of PhyML 3.0. Systematic Biology, 59(3):307–321.

Guindon, S., Rodrigo, A. G., Dyer, K. A., and Huelsenbeck, J. P. (2004). Modeling the site-specific variation of selection patterns along lineages. Proceedings of the National Academy of Sciences of the United States of America, 101(35):12957–12962.

Hobolth, A. and Jensen, J. L. (2005). Statistical inference in evolutionary models of DNA sequences via the EM algorithm. Statistical Applications in Genetics and Molecular Biology, 4(1):1–22.

Holmes, I. and Rubin, G. M. (2002). An expectation maximization algorithm for training hidden substitution models. Journal of Molecular Biology, 317(5):753–764.

Huelsenbeck, J. P., Nielsen, R., and Bollback, J. P. (2003). Stochastic mapping of morphological characters. Systematic Biology, 52(2):131–158.

Huelsenbeck, J. P. and Ronquist, F. (2001). MRBAYES: Bayesian inference of phylogenetic trees. Bioinformatics, 17(8):754–755.

Kellis, M., Patterson, N., Endrizzi, M., Birren, B., and Lander, E. S. (2003). Sequencing and comparison of yeast species to identify genes and regulatory elements. Nature, 423(6937):241–254.

Kenney, T. and Gu, H. (2012). Hessian calculation for phylogenetic likelihood based on the pruning algorithm and its applications. Statistical Applications in Genetics and Molecular Biology, 11(4).

Lemey, P., Minin, V. N., Bielejec, F., Pond, S. L. K., and Suchard, M. A. (2012). A counting renaissance: combining stochastic mapping and empirical Bayes to quickly detect amino acid sites under positive selection. Bioinformatics, 28(24):3248–3256.

Lystig, T. C. and Hughes, J. P. (2002). Exact computation of the observed information matrix for hidden Markov models. Journal of Computational and Graphical Statistics, 11(3):678–689.

Meng, X.-L. (1994). Posterior predictive p-values. The Annals of Statistics, 22(3):1142–1160.

Minin, V. N. and Suchard, M. A. (2008a). Counting labeled transitions in continuous-time Markov models of evolution. Journal of Mathematical Biology, 56(3):391–412.

Minin, V. N. and Suchard, M. A. (2008b). Fast, accurate and simulation-free stochastic mapping. Philosophical Transactions of the Royal Society of London B: Biological Sciences, 363(1512):3985–3995.

Mood, A. M., Graybill, F. A., and Boes, D. C. (1974). Introduction to the Theory of Statistics. International Student edition. McGraw-Hill.

Neuts, M. F. (1995). Algorithmic Probability: A Collection of Problems. Stochastic Modeling Series. Taylor & Francis.

Nielsen, R. (2002). Mapping mutations on phylogenies. Systematic Biology, 51(5):729–739.

Nielsen, R. and Yang, Z. (1998). Likelihood models for detecting positively selected amino acid sites and applications to the HIV-1 envelope gene. Genetics, 148(3):929–936.

Pagel, M. (1999). The maximum likelihood approach to reconstructing ancestral character states of discrete characters on phylogenies. Systematic Biology, 48(3):612–622.

Pollard, K. S., Hubisz, M. J., Rosenbloom, K. R., and Siepel, A. (2010). Detection of nonneutral substitution rates on mammalian phylogenies. Genome Research, 20(1):110–121.

Pollard, K. S., Salama, S. R., King, B., Kern, A. D., Dreszer, T., Katzman, S., Siepel, A., Pedersen, J. S., Bejerano, G., Baertsch, R., et al. (2006a). Forces shaping the fastest evolving regions in the human genome. PLoS Genetics, 2(10):e168.

Pollard, K. S., Salama, S. R., Lambert, N., Lambot, M.-A., Coppens, S., Pedersen, J. S., Katzman, S., King, B., Onodera, C., Siepel, A., et al. (2006b). An RNA gene expressed during cortical development evolved rapidly in humans. Nature, 443(7108):167–172.

Schadt, E. E., Sinsheimer, J. S., and Lange, K. (1998). Computational advances in maximum likelihood methods for molecular phylogeny. Genome Research, 8(3):222–233.

Siepel, A., Bejerano, G., Pedersen, J. S., Hinrichs, A. S., Hou, M., Rosenbloom, K., Clawson, H., Spieth, J., Hillier, L. W., Richards, S., et al. (2005). Evolutionarily conserved elements in vertebrate, insect, worm, and yeast genomes. Genome Research, 15(8):1034–1050.

Siepel, A., Pollard, K. S., and Haussler, D. (2006). New methods for detecting lineage-specific selection. In Research in Computational Molecular Biology: 10th Annual International Conference, RECOMB 2006, Venice, Italy, April 2-5, 2006, Proceedings, volume 3909, pages 190–205. Springer.

Tavaré, S. (1986). Some probabilistic and statistical problems in the analysis of DNA sequences. Lectures on Mathematics in the Life Sciences, 17:57–86.

Yang, Z. (1994). Maximum likelihood phylogenetic estimation from DNA sequences with variable rates over sites: Approximate methods. Journal of Molecular Evolution, 39(3):306–314.

Yang, Z. (1996). Among-site rate variation and its impact on phylogenetic analyses. Trends in Ecology & Evolution, 11(9):367–372.

Yang, Z. (2006). Computational Molecular Evolution, volume 284 of Oxford Series in Ecology and Evolution. Oxford University Press.

Zhai, W., Slatkin, M., and Nielsen, R. (2007). Exploring variation in the dN/dS ratio among sites and lineages using mutational mappings: applications to the influenza virus. Journal of Molecular Evolution, 65(3):340–348.

Zheng, Q. (2001). On the dispersion index of a Markovian molecular clock. Mathematical Biosciences, 172(2):115–128.

Appendix

Monte Carlo Summary Tables

	Standard errors (β-globin)				Standard errors (influenza)			
	$L = 50$	$L = 100$	$L = 200$	$L = 400$	$L = 50$	$L = 100$	$L = 200$	$L = 400$
$m = 100$	0.91	1.3	1.8	2.5	0.094	0.13	0.19	0.26
$m = 500$	0.41	0.57	0.82	1.2	0.042	0.060	0.086	0.12
$m = 1000$	0.29	0.41	0.58	0.82	0.030	0.042	0.061	0.085
$m = 10000$	0.092	0.13	0.18	0.26	0.0097	0.013	0.019	0.027

(a) Monte Carlo standard errors for the β-globin and influenza datasets

	$L = 50$	$L = 100$	$L = 200$	$L = 400$
β-globin	22	41	84	170
influenza	0.34	0.54	1.3	2.3

(b) Exact calculations of T_{var} for the β-globin and influenza datasets

Table 2: Monte Carlo summary tables for the β-globin and influenza datasets. (a) Monte Carlo standard errors associated with simulation-based T_{var} estimates. We compute these standard errors on randomly subsampled alignments of length L using m Monte Carlo replicates per site. Each table entry represents an averaged Monte Carlo standard error, where the averaging is done over 200 randomly subsampled posterior $\boldsymbol{\theta}$'s. (b) Exact computations of T_{var}. Each table entry denotes an average over exact values of T_{var}, where the averaging is done over the same 200 posterior samples of $\boldsymbol{\theta}$ mentioned above. All table entries in (a) and (b) are rounded to two significant digits.

Prior and Posterior Mapping Covariance Computation

In this subsection, we describe how to efficiently compute prior and posterior mapping covariances. We generalize the post-order tree traversal algorithm discussed in the main part of the paper and present the necessary formulas for calculating these covariances. Much of the intuition provided for our original tree traversal algorithm carries over to this generalized procedure.

Let $\mathrm{Cov}(H_{\Omega_1}, H_{\Omega_2})$ and $\mathrm{Cov}(H_{\Omega_1}, H_{\Omega_2}|\mathbf{D})$ denote the prior and posterior mapping covariances, respectively, where $\Omega_1, \Omega_2 \subseteq \Theta$ are predefined sets of branch indices. We consider first the calculation of the posterior mapping covariance $\mathrm{Cov}(H_{\Omega_1}, H_{\Omega_2}|\mathbf{D})$. The vectors $\mathbf{F}_u$ and $\mathbf{S}_b$ are defined and computed as in our original tree traversal procedure for all nodes $u \in \{1, ..., n-1, n, ..., 2n-1\}$ and branches $b \in \Theta$. We introduce the m-long vectors $\mathbf{V}_b^{(\Omega_1,[1])}$, $\mathbf{V}_b^{(\Omega_1,[2])}$, $\mathbf{V}_b^{(\Omega_2,[1])}$, $\mathbf{V}_b^{(\Omega_2,[2])}$, $\mathbf{V}_b^{(\Omega_1\cap\Omega_2,[1])}$, $\mathbf{V}_b^{(\Omega_1\cap\Omega_2,[2])}$, $\mathbf{W}_b^{(\Omega_1)}$, $\mathbf{W}_b^{(\Omega_2)}$, and $\mathbf{W}_b^{(\Omega_1,\Omega_2)}$ for all $b \in \Theta$. The ith entries in $\mathbf{V}_b^{(\Omega_1,[1])}$ and $\mathbf{V}_b^{(\Omega_1,[2])}$ are mathematically defined as:

$$\sum_{b^\dagger} \sum_{\mathbf{i}_b} e^{[1]}_{i^*_{p(b^\dagger)} i^*_{c(b^\dagger)}}(h, t_{b^\dagger}) \prod_{b^* \in \Theta_b \setminus \{b^\dagger\}} p_{i^*_{p(b^*)} i^*_{c(b^*)}}(t_{b^*}), \tag{41}$$

$$\sum_{b^\dagger} \sum_{\mathbf{i}_b} e^{[2]}_{i^*_{p(b^\dagger)} i^*_{c(b^\dagger)}}(h, t_{b^\dagger}) \prod_{b^* \in \Theta_b \setminus \{b^\dagger\}} p_{i^*_{p(b^*)} i^*_{c(b^*)}}(t_{b^*}), \tag{42}$$

respectively, where the state of parent node $p(b)$ is i and $b^\dagger \in \Omega_{1,b}$ for $\Omega_{1,b} = \Omega_1 \cap \Theta_b$. The entries in $\mathbf{V}_b^{(\Omega_2,[1])}$ and $\mathbf{V}_b^{(\Omega_2,[2])}$ and $\mathbf{V}_b^{(\Omega_1\cap\Omega_2,[1])}$ and $\mathbf{V}_b^{(\Omega_1\cap\Omega_2,[2])}$ are defined analogously by replacing Ω_1 with Ω_2 and $\Omega_1 \cap \Omega_2$, respectively, in the above definitions. The ith element of the vector $\mathbf{W}_b^{(\Omega_1)}$ is equal to:

$$\sum_{b^\dagger \neq b^{\dagger\dagger}} \sum_{\mathbf{i}_b} e^{[1]}_{i^*_{p(b^\dagger)} i^*_{c(b^\dagger)}}(h, t_{b^\dagger}) e^{[1]}_{i^*_{p(b^{\dagger\dagger})} i^*_{c(b^{\dagger\dagger})}}(h, t_{b^{\dagger\dagger}}) \prod_{b^* \in \Theta_b \setminus \{b^\dagger, b^{\dagger\dagger}\}} p_{i^*_{p(b^*)} i^*_{c(b^*)}}(t_{b^*}), \tag{43}$$

where the state of parent node $p(b)$ is i and $b^\dagger, b^{\dagger\dagger} \in \Omega_{1,b}$ for $\Omega_{1,b}$ defined as above. The elements in $\mathbf{W}_b^{(\Omega_2)}$ and $\mathbf{W}_b^{(\Omega_1,\Omega_2)}$ are similarly defined, except that $b^\dagger, b^{\dagger\dagger} \in \Omega_{2,b}$ and $b^\dagger \in \Omega_{1,b}, b^{\dagger\dagger} \in \Omega_{2,b}$, respectively, for the same $\Omega_{1,b}$ and $\Omega_{2,b} = \Omega_2 \cap \Theta_b$.

For all terminal branches $b \in \mathcal{E}$, we define:

$$\mathbf{V}_b^{(\Omega_1,[1])} = \mathbf{e}^{[1]}(h, t_b)\mathbf{F}_{c(b)} \mathbb{1}_{\{b \in \Omega_1\}}, \tag{44}$$

$$\mathbf{V}_b^{(\Omega_1,[2])} = \mathbf{e}^{[2]}(h,t_b)\mathbf{F}_{c(b)}\mathbb{1}_{\{b\in\Omega_1\}}, \tag{45}$$

$$\mathbf{V}_b^{(\Omega_2,[1])} = \mathbf{e}^{[1]}(h,t_b)\mathbf{F}_{c(b)}\mathbb{1}_{\{b\in\Omega_2\}}, \tag{46}$$

$$\mathbf{V}_b^{(\Omega_2,[2])} = \mathbf{e}^{[2]}(h,t_b)\mathbf{F}_{c(b)}\mathbb{1}_{\{b\in\Omega_2\}}, \tag{47}$$

$$\mathbf{V}_b^{(\Omega_1\cap\Omega_2,[1])} = \mathbf{e}^{[1]}(h,t_b)\mathbf{F}_{c(b)}\mathbb{1}_{\{b\in\Omega_1\cap\Omega_2\}}, \tag{48}$$

$$\mathbf{V}_b^{(\Omega_1\cap\Omega_2,[2])} = \mathbf{e}^{[2]}(h,t_b)\mathbf{F}_{c(b)}\mathbb{1}_{\{b\in\Omega_1\cap\Omega_2\}}. \tag{49}$$

All entries in $\mathbf{W}_b^{(\Omega_1)}$, $\mathbf{W}_b^{(\Omega_2)}$, and $\mathbf{W}_b^{(\Omega_1,\Omega_2)}$ for $b\in\mathcal{E}$ are set to 0. The recursive formulas for calculating these vectors at internal branches $b\in\mathcal{I}$ are:

$$\mathbf{V}_b^{(\Omega_1,[1])} = \mathbf{e}^{[1]}(h,t_b)\mathbf{F}_{c(b)}\mathbb{1}_{\{b\in\Omega_1\}} + \mathbf{P}(t_b)\Big(\mathbf{V}_{b_1}^{(\Omega_1,[1])}\circ\mathbf{S}_{b_2} + \mathbf{V}_{b_2}^{(\Omega_1,[1])}\circ\mathbf{S}_{b_1}\Big), \tag{50}$$

$$\mathbf{V}_b^{(\Omega_1,[2])} = \mathbf{e}^{[2]}(h,t_b)\mathbf{F}_{c(b)}\mathbb{1}_{\{b\in\Omega_1\}} + \mathbf{P}(t_b)\Big(\mathbf{V}_{b_1}^{(\Omega_1,[2])}\circ\mathbf{S}_{b_2} + \mathbf{V}_{b_2}^{(\Omega_1,[2])}\circ\mathbf{S}_{b_1}\Big), \tag{51}$$

$$\mathbf{V}_b^{(\Omega_2,[1])} = \mathbf{e}^{[1]}(h,t_b)\mathbf{F}_{c(b)}\mathbb{1}_{\{b\in\Omega_2\}} + \mathbf{P}(t_b)\Big(\mathbf{V}_{b_1}^{(\Omega_2,[1])}\circ\mathbf{S}_{b_2} + \mathbf{V}_{b_2}^{(\Omega_2,[1])}\circ\mathbf{S}_{b_1}\Big), \tag{52}$$

$$\mathbf{V}_b^{(\Omega_2,[2])} = \mathbf{e}^{[2]}(h,t_b)\mathbf{F}_{c(b)}\mathbb{1}_{\{b\in\Omega_2\}} + \mathbf{P}(t_b)\Big(\mathbf{V}_{b_1}^{(\Omega_2,[2])}\circ\mathbf{S}_{b_2} + \mathbf{V}_{b_2}^{(\Omega_2,[2])}\circ\mathbf{S}_{b_1}\Big), \tag{53}$$

$$\mathbf{V}_b^{(\Omega_1\cap\Omega_2,[1])} = \mathbf{e}^{[1]}(h,t_b)\mathbf{F}_{c(b)}\mathbb{1}_{\{b\in\Omega_1\cap\Omega_2\}} + \mathbf{P}(t_b)\Big(\mathbf{V}_{b_1}^{(\Omega_1\cap\Omega_2,[1])}\circ\mathbf{S}_{b_2} + \mathbf{V}_{b_2}^{(\Omega_1\cap\Omega_2,[1])}\circ\mathbf{S}_{b_1}\Big), \tag{54}$$

$$\mathbf{V}_b^{(\Omega_1\cap\Omega_2,[2])} = \mathbf{e}^{[2]}(h,t_b)\mathbf{F}_{c(b)}\mathbb{1}_{\{b\in\Omega_1\cap\Omega_2\}} + \mathbf{P}(t_b)\Big(\mathbf{V}_{b_1}^{(\Omega_1\cap\Omega_2,[2])}\circ\mathbf{S}_{b_2} + \mathbf{V}_{b_2}^{(\Omega_1\cap\Omega_2,[2])}\circ\mathbf{S}_{b_1}\Big), \tag{55}$$

$$\begin{aligned}\mathbf{W}_b^{(\Omega_1)} &= 2\times\mathbf{e}^{[1]}(h,t_b)\Big(\mathbf{V}_{b_1}^{(\Omega_1,[1])}\circ\mathbf{S}_{b_2} + \mathbf{V}_{b_2}^{(\Omega_1,[1])}\circ\mathbf{S}_{b_1}\Big)\mathbb{1}_{\{b\in\Omega_1\}}\\ &\quad+\mathbf{P}(t_b)\Big(2\times\mathbf{V}_{b_1}^{(\Omega_1,[1])}\circ\mathbf{V}_{b_2}^{(\Omega_1,[1])} + \mathbf{W}_{b_1}^{(\Omega_1)}\circ\mathbf{S}_{b_2} + \mathbf{W}_{b_2}^{(\Omega_1)}\circ\mathbf{S}_{b_1}\Big),\end{aligned} \tag{56}$$

$$\begin{aligned}\mathbf{W}_b^{(\Omega_2)} &= 2\times\mathbf{e}^{[1]}(h,t_b)\Big(\mathbf{V}_{b_1}^{(\Omega_2,[1])}\circ\mathbf{S}_{b_2} + \mathbf{V}_{b_2}^{(\Omega_2,[1])}\circ\mathbf{S}_{b_1}\Big)\mathbb{1}_{\{b\in\Omega_2\}}\\ &\quad+\mathbf{P}(t_b)\Big(2\times\mathbf{V}_{b_1}^{(\Omega_2,[1])}\circ\mathbf{V}_{b_2}^{(\Omega_2,[1])} + \mathbf{W}_{b_1}^{(\Omega_2)}\circ\mathbf{S}_{b_2} + \mathbf{W}_{b_2}^{(\Omega_2)}\circ\mathbf{S}_{b_1}\Big),\end{aligned} \tag{57}$$

$$\begin{aligned}\mathbf{W}_b^{(\Omega_1,\Omega_2)} &= \mathbf{e}^{[1]}(h,t_b)\Big(\mathbf{V}_{b_1}^{(\Omega_2,[1])}\circ\mathbf{S}_{b_2} + \mathbf{V}_{b_2}^{(\Omega_2,[1])}\circ\mathbf{S}_{b_1}\Big)\mathbb{1}_{\{b\in\Omega_1\}}\\ &\quad+\mathbf{e}^{[1]}(h,t_b)\Big(\mathbf{V}_{b_1}^{(\Omega_1,[1])}\circ\mathbf{S}_{b_2} + \mathbf{V}_{b_2}^{(\Omega_1,[1])}\circ\mathbf{S}_{b_1}\Big)\mathbb{1}_{\{b\in\Omega_2\}}\\ &\quad+\mathbf{P}(t_b)\Big(\mathbf{V}_{b_1}^{(\Omega_1,[1])}\circ\mathbf{V}_{b_2}^{(\Omega_2,[1])} + \mathbf{V}_{b_2}^{(\Omega_1,[1])}\circ\mathbf{V}_{b_1}^{(\Omega_2,[1])}\\ &\qquad+\mathbf{W}_{b_1}^{(\Omega_1,\Omega_2)}\circ\mathbf{S}_{b_2} + \mathbf{W}_{b_2}^{(\Omega_1,\Omega_2)}\circ\mathbf{S}_{b_1}\Big),\end{aligned} \tag{58}$$

where b_1 and b_2 represent the two branches that are “below” branch b. This generalized tree traversal algorithm terminates after computing $\mathbf{F}_u$, $\mathbf{S}_b$, $\mathbf{V}_b^{(\Omega_1,[1])}$, $\mathbf{V}_b^{(\Omega_1,[2])}$, $\mathbf{V}_b^{(\Omega_2,[1])}$, $\mathbf{V}_b^{(\Omega_2,[2])}$, $\mathbf{V}_b^{(\Omega_1\cap\Omega_2,[1])}$, $\mathbf{V}_b^{(\Omega_1\cap\Omega_2,[2])}$, $\mathbf{W}_b^{(\Omega_1)}$, $\mathbf{W}_b^{(\Omega_2)}$, and $\mathbf{W}_b^{(\Omega_1,\Omega_2)}$ for $u = root$ and $b\in\{root_1, root_2\}$, where $root$ denotes the root node label and $root_1$ and $root_2$ represent the two branches connecting the root node to its children. The restricted mapping moments of interest are calculated as follows:

$$\mathrm{E}(H_{\Omega_1}\mathbb{1}_{\mathbf{D}}) = \boldsymbol{\pi}^T\Big(\mathbf{V}_{root_1}^{(\Omega_1,[1])}\circ\mathbf{S}_{root_2} + \mathbf{V}_{root_2}^{(\Omega_1,[1])}\circ\mathbf{S}_{root_1}\Big), \tag{59}$$

$$\mathrm{E}(H_{\Omega_2}\mathbb{1}_{\mathbf{D}}) = \boldsymbol{\pi}^T\Big(\mathbf{V}_{root_1}^{(\Omega_2,[1])}\circ\mathbf{S}_{root_2} + \mathbf{V}_{root_2}^{(\Omega_2,[1])}\circ\mathbf{S}_{root_1}\Big), \tag{60}$$

$$\begin{aligned}\mathrm{E}(H_{\Omega_1}^2 \mathbb{1}_{\mathbf{D}}) = \boldsymbol{\pi}^T \Big[& 2 \times \mathbf{V}_{root_1}^{(\Omega_1,[1])} \circ \mathbf{V}_{root_2}^{(\Omega_1,[1])} + \mathbf{W}_{root_1}^{(\Omega_1)} \circ \mathbf{S}_{root_2} + \mathbf{W}_{root_2}^{(\Omega_1)} \circ \mathbf{S}_{root_1} \\ &+ \big(\mathbf{V}_{root_1}^{(\Omega_1,[1])} + \mathbf{V}_{root_1}^{(\Omega_1,[2])}\big) \circ \mathbf{S}_{root_2} + \big(\mathbf{V}_{root_2}^{(\Omega_1,[1])} + \mathbf{V}_{root_2}^{(\Omega_1,[2])}\big) \circ \mathbf{S}_{root_1}\Big],\end{aligned} \tag{61}$$

$$\begin{aligned}\mathrm{E}(H_{\Omega_2}^2 \mathbb{1}_{\mathbf{D}}) = \boldsymbol{\pi}^T \Big[& 2 \times \mathbf{V}_{root_1}^{(\Omega_2,[1])} \circ \mathbf{V}_{root_2}^{(\Omega_2,[1])} + \mathbf{W}_{root_1}^{(\Omega_2)} \circ \mathbf{S}_{root_2} + \mathbf{W}_{root_2}^{(\Omega_2)} \circ \mathbf{S}_{root_1} \\ &+ \big(\mathbf{V}_{root_1}^{(\Omega_2,[1])} + \mathbf{V}_{root_1}^{(\Omega_2,[2])}\big) \circ \mathbf{S}_{root_2} + \big(\mathbf{V}_{root_2}^{(\Omega_2,[1])} + \mathbf{V}_{root_2}^{(\Omega_2,[2])}\big) \circ \mathbf{S}_{root_1}\Big],\end{aligned} \tag{62}$$

$$\begin{aligned}\mathrm{E}(H_{\Omega_1} H_{\Omega_2} \mathbb{1}_{\mathbf{D}}) = \boldsymbol{\pi}^T \Big[& \mathbf{V}_{root_1}^{(\Omega_1,[1])} \circ \mathbf{V}_{root_2}^{(\Omega_2,[1])} + \mathbf{V}_{root_2}^{(\Omega_1,[1])} \circ \mathbf{V}_{root_1}^{(\Omega_2,[1])} \\ &+ \mathbf{W}_{root_1}^{(\Omega_1,\Omega_2)} \circ \mathbf{S}_{root_2} + \mathbf{W}_{root_2}^{(\Omega_1,\Omega_2)} \circ \mathbf{S}_{root_1} \\ &+ \big(\mathbf{V}_{root_1}^{(\Omega_1\cap\Omega_2,[1])} + \mathbf{V}_{root_1}^{(\Omega_1\cap\Omega_2,[2])}\big) \circ \mathbf{S}_{root_2} \\ &+ \big(\mathbf{V}_{root_2}^{(\Omega_1\cap\Omega_2,[1])} + \mathbf{V}_{root_2}^{(\Omega_1\cap\Omega_2,[2])}\big) \circ \mathbf{S}_{root_1}\Big].\end{aligned} \tag{63}$$

We also know that $\mathrm{P}(\mathbf{D}) = \boldsymbol{\pi}^T \mathbf{F}_{root}$ (Felsenstein, 1981). Our efficient computations of the above restricted mapping moments allow us to calculate the associated posterior mapping moments using the following equations:

$$\mathrm{E}(H_{\Omega_1}|\mathbf{D}) = \frac{\mathrm{E}(H_{\Omega_1}\mathbb{1}_{\mathbf{D}})}{\mathrm{P}(\mathbf{D})}, \tag{64}$$

$$\mathrm{E}(H_{\Omega_2}|\mathbf{D}) = \frac{\mathrm{E}(H_{\Omega_2}\mathbb{1}_{\mathbf{D}})}{\mathrm{P}(\mathbf{D})}, \tag{65}$$

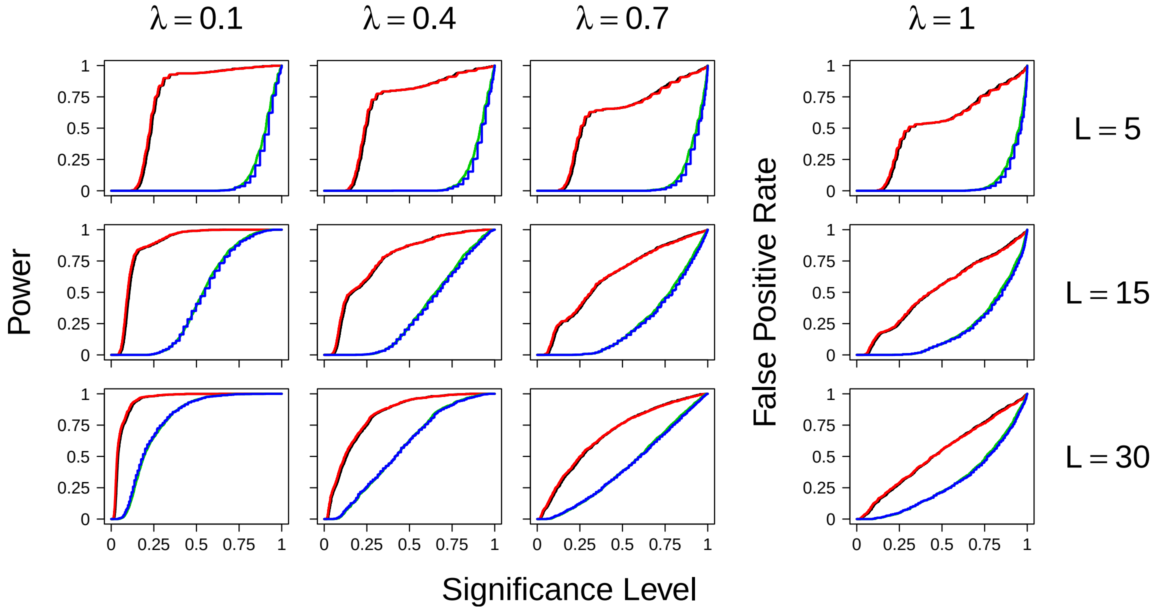
$$\mathrm{Var}(H_{\Omega_1}|\mathbf{D}) = \mathrm{E}(H_{\Omega_1}^2|\mathbf{D}) - \mathrm{E}(H_{\Omega_1}|\mathbf{D})^2 = \frac{\mathrm{E}(H_{\Omega_1}^2\mathbb{1}_{\mathbf{D}})}{\mathrm{P}(\mathbf{D})} - \left[\frac{\mathrm{E}(H_{\Omega_1}\mathbb{1}_{\mathbf{D}})}{\mathrm{P}(\mathbf{D})}\right]^2, \tag{66}$$

$$\mathrm{Var}(H_{\Omega_2}|\mathbf{D}) = \mathrm{E}(H_{\Omega_2}^2|\mathbf{D}) - \mathrm{E}(H_{\Omega_2}|\mathbf{D})^2 = \frac{\mathrm{E}(H_{\Omega_2}^2\mathbb{1}_{\mathbf{D}})}{\mathrm{P}(\mathbf{D})} - \left[\frac{\mathrm{E}(H_{\Omega_2}\mathbb{1}_{\mathbf{D}})}{\mathrm{P}(\mathbf{D})}\right]^2, \tag{67}$$

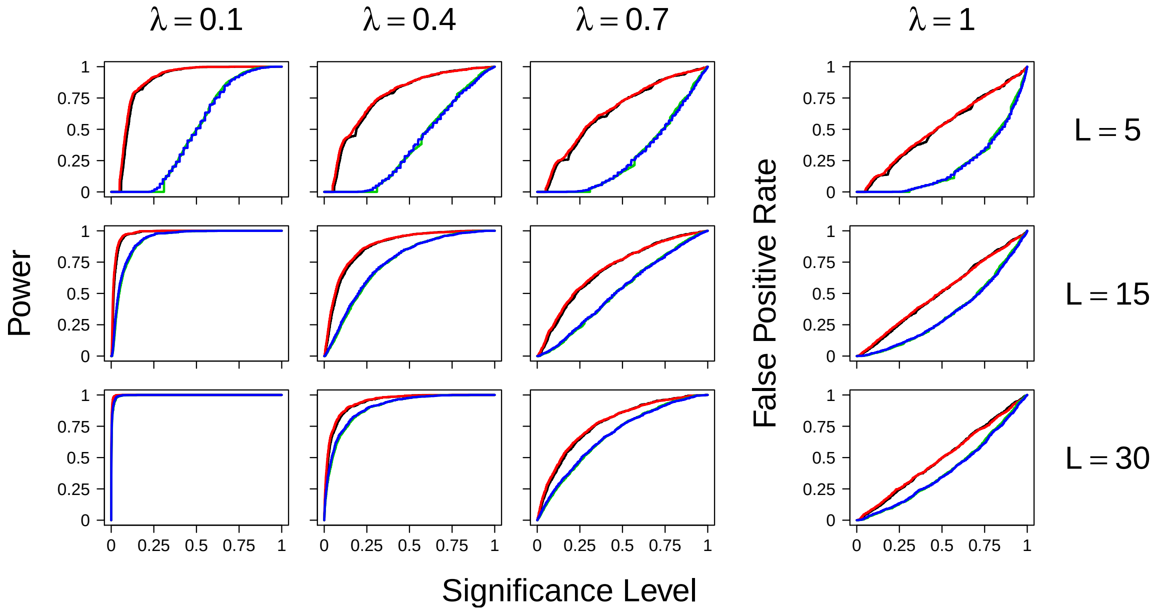
$$\begin{aligned}\mathrm{Cov}(H_{\Omega_1}, H_{\Omega_2}|\mathbf{D}) &= \mathrm{E}(H_{\Omega_1}H_{\Omega_2}|\mathbf{D}) - \mathrm{E}(H_{\Omega_1}|\mathbf{D})\mathrm{E}(H_{\Omega_2}|\mathbf{D}) \\ &= \frac{\mathrm{E}(H_{\Omega_1}H_{\Omega_2}\mathbb{1}_{\mathbf{D}})}{\mathrm{P}(\mathbf{D})} - \left[\frac{\mathrm{E}(H_{\Omega_1}\mathbb{1}_{\mathbf{D}})}{\mathrm{P}(\mathbf{D})}\right]\left[\frac{\mathrm{E}(H_{\Omega_2}\mathbb{1}_{\mathbf{D}})}{\mathrm{P}(\mathbf{D})}\right].\end{aligned} \tag{68}$$

The prior mapping moments can be obtained by modifying the tree traversal procedure discussed in this subsection. The only changes that need to be made are to the initializations of $\mathbf{F}_u$ at all terminal nodes u. If we set $F_{ui} = 1$ for all terminal nodes $u \in \{n, ..., 2n-1\}$ and $i = 1, ..., m$, then our algorithm will be able to compute the prior moments of interest. In this case, equations (64)-(68) are used to calculate the prior mapping moments. Thus, the prior and posterior mapping covariances are computed in a similar fashion, but according to different initializations of the $\mathbf{F}_u$ vectors at the tips of the phylogeny.

SPH Subtree Simulation Plots



(a) Power and false positive rate curves for $\rho = 0.25$



(b) Power and false positive rate curves for $\rho = 0.85$

Figure 6: Power and false positive rate plots from our subtree simulation experiments. The power and false positive rate curves for the original SPH marginal (conditional) subtree test are shown in green (blue), while the corresponding performance curves for the modified SPH marginal (conditional) subtree test are displayed in black (red). In this figure, we present performance plots for $L = 5, 15, 30$; $\rho = 0.25, 0.85$; and $\lambda = 0.1, 0.4, 0.7, 1$.