BIOINFORMATICS

Vol. 20 Suppl. 1 2004, pages i161–i168
DOI: 10.1093/bioinformatics/bth917

Efficient approximations for learning phylogenetic HMM models from data

Vladimir Jojic[1,], Nebojsa Jojic[1], Chris Meek[1], Dan Geiger[2], Adam Siepel[3], David Haussler[3,4] and D. Heckerman[1]*

[1]Microsoft Research, Redmond, WA 98052, USA, [2]Technion—Israel Institute of Technology Computer Science Department, Haifa 32000, Israel, [3]Center for Biomolecular Science and Engineering and [4]Howard Hughes Medical Institute, University of California Santa Cruz, CA 95064, USA

Received on January 15, 2004; accepted on March 1, 2004

ABSTRACT

Motivation: We consider models useful for learning an evolutionary or phylogenetic tree from data consisting of DNA sequences corresponding to the leaves of the tree. In particular, we consider a general probabilistic model described in Siepel and Haussler that we call the phylogenetic-HMM model which generalizes the classical probabilistic models of Neyman and Felsenstein. Unfortunately, computing the likelihood of phylogenetic-HMM models is intractable. We consider several approximations for computing the likelihood of such models including an approximation introduced in Siepel and Haussler, loopy belief propagation and several variational methods.

Results: We demonstrate that, unlike the other approximations, variational methods are accurate and are guaranteed to lower bound the likelihood. In addition, we identify a particular variational approximation to be best—one in which the posterior distribution is variationally approximated using the classic Neyman–Felsenstein model. The application of our best approximation to data from the cystic fibrosis transmembrane conductance regulator gene region across nine eutherian mammals reveals a CpG effect.

Contact: vjojic@psi.toronto.edu

1 INTRODUCTION

We consider the problem of learning an evolutionary or phylogenetic tree from data consisting of DNA sequences corresponding to the leaves of the tree. We concentrate on a standard probabilistic model-selection approach wherein models are scored by some criterion (e.g. maximum likelihood, Bayesian information criterion) and some heuristic search method is used to find a tree or set of trees with high scores (e.g. Felsenstein, 1981; Durbin *et al*., 1998). We further concentrate on methods for computing the likelihood of a given model.

The classic probabilistic model used in this approach is described by (e.g.) Neyman (1971) and Felsenstein (1981). The model incorporates several strong assumptions including (1) evolution takes place independently at each base-pair site, (2) the base-pair substitution rate is uniform over sites, (3) there is no recombination, (4) there is no lateral gene transfer and (5) the sequences are aligned. There have been numerous efforts to relax these assumptions (e.g. Siepel and Haussler, 2003; Yang, 1995; Felsenstein and Churchill, 1996; Strimmer and Moulton, 2000; Nakhleh *et al*., 2003). In this paper, we address the relaxation of the first assumption by considering the combined tree-HMM model described in Siepel and Haussler (2003) in which base-pair substitutions are dependent on neighboring bases. We call this hybrid model a phylogenetic-HMM model. We do not address the relaxation of the other assumptions so as to isolate the effects of the first assumption and to avoid substantial added complexity.

One important drawback of phylogenetic-HMM models is that evaluating the likelihood of such a model (and hence finding parameters that maximize this likelihood) is intractable. Recently, Siepel and Haussler (2003) introduced an efficient approximation for computing the likelihood of phylogenetic-HMM models. Unfortunately, as we shall see, this approximation has no accuracy guarantees and, thus, may be inappropriate for use in model selection. In this paper, we describe phylogenetic-HMM models in terms of Bayesian networks, also known as directed acyclic graphical (DAG) models (e.g. Pearl, 1988). We introduce several approximations developed for graphical models based on variational techniques (e.g. Jordan *et al*., 1999) that efficiently yield a lower-bound on the likelihood of a phylogenetic-HMM model. In experiments on real data, we show that these lower-bounds are tight. We also describe another approximation for computing likelihood in graphical models known as loopy belief propagation (e.g. Pearl, 1988). This approximation has no accuracy guarantees and, as we show experimentally, yields poor likelihood estimates.

*To whom correspondence should be addressed.

In Section 2, we describe phylogenetic-HMM models in terms of Bayesian networks or DAG models. In Section 3, we describe the approximation for evaluating the likelihood of phylogenetic-HMM models presented in Siepel and Haussler (2003). In Section 4, we discuss the theoretical basis for the variational approximation; and in Section 5, we introduce structured variational techniques tailored to our model. Structured variational approximations go beyond the standard mean-field approximation, and our tailored approximations (to our knowledge) have not been described previously. In Section 6, we discuss experimental results on real data and find that, among the approximations, the one that performs best is the structured approximation in which the posterior distribution is variationally approximated using the classic Neyman–Felsenstein model. In Section 7, we apply this approximation to data from the cystic fibrosis transmembrane conductance regulator (CFTR) gene region across nine eutherian mammals and, in doing so, identify a 'CpG effect' (a high mutation rate of CG to TG, due to methylation and spontaneous deamination).

2 THE MODEL

The phylogenetic-HMM models that we consider are identical to those described in Siepel and Haussler (2003). As we shall see, it will be convenient to describe these models as DAG models. Given a domain of interest having a set of finite variables $\boldsymbol{s} = (s_1, \ldots, s_n)$ with a positive joint distribution $p(\boldsymbol{s})$, a DAG model for $\boldsymbol{s}$ is a pair $(\mathcal{G}, \mathcal{P})$ where $\mathcal{G}$ is a directed acyclic graph and $\mathcal{P}$ is a set of conditional probability distributions. Each node in $\mathcal{G}$ corresponds to a variable in $\boldsymbol{s}$. We use S_i to refer to both the variable and its corresponding node. Arcs in the graph correspond to probabilistic dependencies or, more precisely, the absence of arcs correspond to probabilistic independencies. These independencies are precisely those that allow us to write the joint distribution as follows: $p(\boldsymbol{s}) = \prod_{i=1}^{N} p(s_i \mid \mathbf{pa}(s_i))$, where $\mathbf{pa}(s_i)$ are the parents of s_i in the graph. The distributions $p(s_i \mid \mathbf{pa}(s_i))$ are called local probability distributions.

The DAG model structure corresponding to the tree model of (e.g.) Felsenstein (1981) is shown in Figure 1a. The variable h_j^i corresponds to the unknown nucleotide in ancestor species i at nucleotide site j. The variable x_j^i corresponds to the observed nucleotide in existing species i at site j. The strong assumption that evolution takes place independently at each nucleotide side is reflected in the lack of arcs among the sites. The DAG model corresponding to a simple phylogenetic-HMM model is shown in Figure 1b. In this dinucleotide model (Siepel and Haussler, 2003) the identity of nucleotide at site j is dependent on the ancestor nucleotide at site j (as in the Neyman–Felsenstein model), as well as the corresponding child and parent nucleotides at site $j-1$.

Additional complex models are defined in Siepel and Haussler (2003), wherein two or more previous slices influence the identity of the base as a given position. To discuss all such models, we introduce the following notation. First, we often drop the explicit indication of whether a variable is observed or not, using s_j^i to refer to the variable for species i at site position j. Further, we use the regularity of the Bayes net to define the connectivity by two sets of parent indices for each variable:

- $\mathcal{T}_i$ is a set of species indices that are parents of the i-th species. This parent information is the same for each site j and defines the phylogenetic tree. In Figure 1, e.g. $\mathcal{T}_3 = \{2\}$
- $\mathcal{C}_j$ is a set of site indices that are parents of the j-th site. This parent information is the same in each sequence i, and defines the Markov-chain model. In Figure 1b, e.g. $\mathcal{C}_2 = 1$. In fact, in this paper, it is always true that $\mathcal{C}_j = \{j-1\}$, but the derivations presented here can be carried out in an analogous way for more general situations, e.g. when $\mathcal{C}_j = \{j-1, j-2, \ldots, j-k\}$.

The indices of all parents of a variable s_j^i can then be written as

$$\mathcal{P}(s_j^i) = \mathcal{T}_i \times \mathcal{C}_j \cup \{i\} \times \mathcal{C}_j \cup \mathcal{T}_i \times \{j\}, \qquad (1)$$

and the parent variables are $\mathbf{pa}(s_j^i) = \{s_l^k\}_{(k,l)\in\mathcal{P}(s_j^i)}$. Or, to use a different notation, $\mathbf{pa}(s_j^i) = s_{\mathcal{P}(s_j^i)}$. We will also use $\mathbf{s}_j^{\mathcal{T}_i}$ to denote the parents of s_j^i that share its site index j, and $\mathbf{s}_{\mathcal{C}_j}^i$ to denote the parents that share its species index i. The joint probability distribution is $p(\boldsymbol{s}) = \prod_{i,j} p(s_j^i \mid s_{\mathcal{P}(s_j^i)})$. For instance, if $\mathcal{C}$ and $\mathcal{T}$ define a chain each, i.e. $\mathcal{T}_i = \{i-1\}$, $\mathcal{C}_j = \{j-1\}$, then $\mathbf{pa}(s_j^i) = \{s_{j-1}^{i-1}, s_{j-1}^i, s_j^{i-1}\}$, and the resulting grid probability model is defined as $p(\boldsymbol{s}) = \prod_{i,j} p(s_j^i \mid s_{j-1}^{i-1}, s_{j-1}^i, s_j^{i-1})$.

Our experiments are restricted to the dinucleotide model wherein $\mathcal{C}_j = \{j-1\}$ (although the tree is not reduced to a chain and $\mathcal{T}_i \neq \{i-1\}$). As in a regular phylogenetic tree, each node has a single parent species, and thus $\mathcal{T}_i$ has a single element, unless i is the root, in which case it is empty. Thus, $\mathbf{pa}(s_j^i)$ still has at most three variables as in the case of a grid model. However, we derive the methods in a general form so as to apply to other neighborhood relations, including the case when each site j in a sequence is connected to several previous sites, rather than just to site $j-1$, and the case of the horizontal gene transfer, where $\mathcal{T}_i$ could have more than one variable.

In the remainder of this section, we examine the local probability distributions $p(s \mid \mathbf{pa}(s))$ in our models. These distributions correspond to well-known models of DNA substitution.

First, consider the Neyman–Felsenstein model wherein evolution at each slice is independent. Here, we need $p(s_j^i \mid s_j^k)$, where species k is the parent of species i. Models of DNA substitution for this case are generally based on a

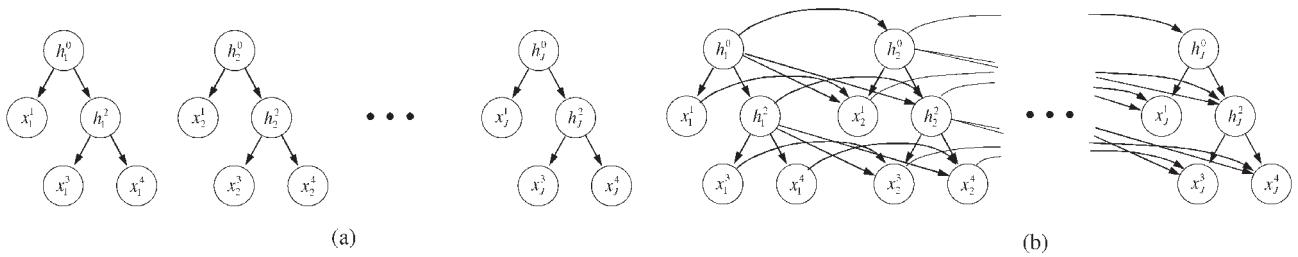

Fig. 1. Probabilistic phylogenetic models expressed as DAG models. (**a**) The Neyman–Felsenstein tree model. (**b**) The dinucleotide phylogenetic-HMM model.

continuous-time Markov model of base substitution, with the instantaneous rate of replacement of each base for each other defined by a rate matrix Q (Yang *et al.*, 1994; Whelan *et al.*, 2001). As a continuous-time Markov matrix, $Q = \{q_{i,j}\}$ $(1 \leq i, j \leq 4)$ is constrained to have each of its rows sum to zero. In an "unrestricted" model, Q has $4^2 - 4 - 1 = 11$ free parameters. To determine the local distribution, we assume that the Markov process defined by Q runs for a given (evolutionary) time t. Let $P(t)$ be the matrix of substitution probabilities—i.e. the local distribution—for length t (note that $P(t)$ is a discrete Markov matrix, with rows summing to 1, while Q is a continuous Markov matrix, with rows summing to 0). $P(t)$ is thus given by the solution to the differential equation $(d/dt)P(t) = P(t)Q$ with initial conditions $P(0) = I$, which is $\boldsymbol{P(t) = e^{Qt}}$. Q is generally diagonalizable as $Q = S\Lambda S^{-1}$, allowing $P(t)$ to be computed as $\boldsymbol{P(t) = Se^{\Lambda t}S^{-1}}$, where $e^{\Lambda t}$ is the diagonal matrix obtained by exponentiating each element on the main diagonal of Λt.

Now, consider the local probability distributions for the dinucleotide model used in our experiments. The local distribution is built from a continuous-time Markov model for dinucleotide pairs associated with a 16×16 Q matrix. Given time t, we obtain the conditional distribution of a dinucleotide pair given its parent species—say $p(s^i_j, s^i_{j-1} \mid s^k_j, s^k_{j-1})$—in the same manner as described for the Neyman–Felsenstein model. This distribution then determines $p(s^i_j \mid s^i_{j-1}, s^k_j, s^k_{j-1})$. To reduce the number of possible free parameters in Q, we assume that substitutions are strand symmetric. For example, we assume that the substitution rate for CG to TG is equal to that for CG to CA. In addition, simultaneous substitutions involving more than one base are disallowed [despite biological evidence for such changes (Averof *et al.*, 2000)]. This model is called the U2S model in Siepel and Haussler (2003).

We note that this model for substitutions is inconsistent in spirit with our dinucleotide model. In particular, the dinucleotide model assumes that substitutions are dependent on the current and previous base-pair sites. If such dependence were allowed to propagate across the many generations implicit in a single edge of an evolutionary tree model, then the substitution at any given site would be a function of the base pairs at all other sites. The same inconsistency is present and noted in Siepel and Haussler (2003). Methods for removing this inconsistency for the simple (two-sequence) case have been discussed (Jensen and Pedersen, 2000; Pedersen and Jensen, 2001; Arndt *et al.*, 2002), but these methods are difficult to extend to the general case.

3 A SIMPLE MARKOV-CHAIN APPROXIMATION

As discussed in the introduction, a key task in learning evolutionary trees from data is the evaluation of a given model's score. Here, we shall restrict our attention to the likelihood or log-likelihood score $\log p(\mathbf{x} \mid \theta) = \log \sum_{\mathbf{h}} p(\mathbf{x}, \mathbf{h} \mid \theta)$, where $\mathbf{x}$ and $\mathbf{h}$ are the set of all x^i_j and h^i_j, respectively, and θ are the parameters that specify the local probability distributions. The computation of this likelihood is also important, because it corresponds to the E step of expectation–maximization (EM) and EM-like algorithms that are used to identify the maximum likelihood parameters for a model. Felsenstein (1981) showed how to perform inference exactly and efficiently for his model. Pearl (1988) discovered the same algorithm—essentially a tree version of dynamic programming in which independence relations are used to rearrange sums of products as products of sums.

Unfortunately, no efficient inference method for the phylogenetic-HMM models are known. The phylogenetic-HMM models contain numerous undirected cycles, making it extremely unlikely that the log likelihood score can be computed efficiently.

In this section, we examine a simple Markov-chain approximation introduced in Siepel and Haussler (2003) for performing this inference. For simplicity, we describe this approximation for the dinucleotide model only. To understand this approximation, let $\mathbf{x}_j$ denote the set of observed nucleotides at site j. In the approximation, we model the observed data as a Markov chain: $p(\mathbf{x} \mid \theta) \cong p(\mathbf{x}_1 \mid \theta) \prod_{j=2}^{J} p(\mathbf{x}_j \mid \mathbf{x}_{j-1}, \theta)$. We further approximate each term by imposing additional (and mutually inconsistent) independence assumptions on the dinucleotide model in Figure 1b. For example, to compute the first term in this approximation, we assume that $\{s_1\}$ and $\{s_2, \ldots, s_J\}$ are independent. To compute the second term, we assume that $\{s_1, s_2\}$ and $\{s_3, \ldots, s_J\}$

are independent. To compute the third term, we assume that $\{s_1\}$, $\{s_2, s_3\}$ and $\{s_4, \ldots, s_J\}$ are mutually independent.

As may be expected, this approximation has no accuracy guarantees and consequently may be inappropriate as a criterion for model selection. To understand this observation, consider a tree structure that has a root node and two leaves. For simplicity, suppose that our alphabet has only two letters and the two sequences at the leaves are identical: 212. Consider two parameterized models for this structure having the following dinucleotide Q matrices (columns and rows are indexed by dinucleotides in lexicographic order):

$$\begin{pmatrix} -2 & 1 & 1 & 0 \\ 1 & -2 & 1 & 0 \\ 0 & 0 & 0 & 0 \\ 1 & 1 & 1 & -3 \end{pmatrix} \quad \begin{pmatrix} -3 & 1 & 1 & 1 \\ 0 & 0 & 0 & 0 \\ 0 & 1 & -2 & 1 \\ 0 & 1 & 1 & -2 \end{pmatrix}$$

Then, for the first model, the true and approximate likelihood of the data are 0.5361 and 0.0175, respectively. The approximation underestimates the true likelihood. For the second model, the true and approximate likelihood of the data are 0.01609 and 0.4602, respectively. In contrast to the first situation, the approximate overestimates the true likelihood.

In what follows, we consider an approximation that comes with a guarantee.

4 FREE ENERGY AND LOG LIKELIHOOD

As noted in the introduction, a standard criterion to optimize in graphical models is the likelihood or the log likelihood of the observed data, obtained by summing or integrating over the hidden variables for a given set of parameters $\boldsymbol{\theta}$, $\log p(\mathbf{x} \mid \theta) = \log \sum_{\mathbf{h}} p(\mathbf{x}, \mathbf{h} \mid \theta)$. However, for many models, including the model we study in this paper, finding the maximum-likelihood parameters and even just the computation of the log likelihood is intractable. Instead, we optimize the free energy of the model (Neal and Hinton, 1998; Jordan *et al.*, 1999)

$$\begin{aligned} F &= \sum_{\mathbf{h}} q(\mathbf{h}) \log \frac{q(\mathbf{h})}{p(\mathbf{x}, \mathbf{h} \mid \theta)} \\ &= \sum_{\mathbf{h}} q(\mathbf{h}) \log q(\mathbf{h}) - \sum_{\mathbf{h}} q(\mathbf{h}) \log p(\mathbf{x}, \mathbf{h} \mid \theta) \end{aligned}$$

where $q(\mathbf{h})$ is an arbitrary distribution. Making the substitution $q(\mathbf{h}) = p(\mathbf{h} \mid \mathbf{x}, \theta))$ yields $F = -\log p(\mathbf{x} \mid \theta)$. In addition, using Jensen's inequality, it can be shown that $F \geq -\log p(\mathbf{x} \mid \theta)$ for any probability distribution $q(\mathbf{h})$—i.e. for any function $q(\mathbf{h})$ such that $\sum_{\mathbf{h}} q(\mathbf{h}) = 1$. Thus, q is seen as an approximate posterior distribution, that can be used to compute a lower-bound on the log likelihood of the data.

Estimating the posterior distribution can be achieved by minimizing the free energy. If q is unrestricted, then F is minimized at $q = p(\mathbf{h} \mid \mathbf{x})$. For example, if in our model, $\mathcal{C}_j$ are all empty, the model decomposes into a collection of independent graphs (except that they share parameters), and for each of them, the free energy has the form:

$$\begin{aligned} F_{\text{tree}} = &\sum_{\{h^i\}_{i=1}^I} q(\{h^i\}_{i=1}^I) \log q(\{h^i\}_{i=1}^I) \\ &- \sum_{\{h^i\}_{i=1}^I} q(\{h^i\}_{i=1}^I) \sum_i \log p(s^i \mid \mathbf{s}^{\mathcal{T}_i}), \end{aligned}$$

where s denotes both hidden ($\mathbf{h}$) and observed ($\mathbf{x}$) variables, as described earlier. Minimizing this energy with respect to $q(\{h^i\}_{i=1}^I)$ leads to $q(\{h^i\}_{i=1}^I) = p(\{h^i\}_{i=1}^I \mid \mathbf{x})$ and $F = -\log p(\mathbf{x})$. We will later use this observation in reverse—i.e. we iteratively will update various costs of this form, but instead of optimizing the cost using a numerical procedure, we will use belief propagation (described in Section 2), to compute optimal $p(\{h^i\}_{i=1}^I \mid \mathbf{x})$ and $\log p(\mathbf{x})$ efficiently.

For the more complex case, when each variable is connected both vertically and horizontally, belief propagation cannot be used (except as an approximation), and (as we shall see) we are better off optimizing F by varying the function q with the aim of lowering it to be as close to $-\log p(\mathbf{x})$ as possible. Such approaches are known as variational techniques in the machine-learning community.

Because the inequality $F \geq -\log p(\mathbf{x} \mid \theta)$ is satisfied for all probability distributions q, it is possible to limit the search space to approximate forms of the function q that lead to more tractable optimization. As the bound is tighter the closer we get to the true posterior, one should naturally attempt to limit the severity of the approximations. Usual ways of approximating the posterior are to either choose a particular functional form (e.g. exponential, even if the true posterior does not follow this form), and/or to decouple hidden variables that are in the true posterior correlated. These approximations are typically less severe when the posterior has a small number of pronounced modes, although the optimization of the free energy could get stuck in a local minima.

One efficient form of q is $q = \prod_{i,j} q(h^i_j)$. This form leads to a standard variational approximation known as mean field. The derivation of the approximation resulting from this form is given in Jojic *et al.* (2003). In Section 5, we introduce forms with less independence leading to novel structured variational approximations that produce considerably better bounds on the log likelihood. An alternative standard method is loopy belief propagation, also described in Jojic *et al.* (2003).

5 STRUCTURED VARIATIONAL APPROXIMATIONS

In order to capture more correlations among the variables in the posterior, and at the same time avoid worrying about whether the marginal distributions agree, we can model the distribution q as a product of distributions defined on disjunct subsets of the model's hidden variables. In this section, we

develop two such approximations. In the first approximation, the variables are grouped according to the index j—i.e. each factor in q is a distribution over all variables in a single tree at site j (the classic Neyman–Felsenstein model). In the second approximation, we group variables according to the sequence index i.

5.1 Product of trees

Under the assumption that the posterior can be factored into J different individual probability distributions, each defined over nucleotides in different sequences but at the same site j, $q = \prod_j q_j(\{h_j^i\}_{i=1}^I)$, the free energy assumes the following form:

$$F = \sum_j \sum_{\mathbf{h}_j} q_j \log q_j - \sum_{i,j} \sum_{\mathbf{h}_j, \mathbf{h}_{k\in\mathcal{C}_j}} q_j \left[\prod_{k\in\mathcal{C}_j} q_k\right] \log p[s_j^i \mid \mathbf{pa}(s_j^i)],$$

where we use bold notation $\mathbf{h}_j = \{h_j^i\}_{i=1}^I$ to denote the set of all variables in the j-th tree. This expression is easily arrived at using the fact that $\sum_{\mathbf{h}_\ell} q_\ell = 1$. Consequently, thus in $\sum_{\mathbf{h}} q \log p(s_j^i \mid \mathbf{pa}(s_j^i))$, the distributions that do not use s_j^i, or $\mathbf{pa}(s_j^i)$ drop out. Note again that for simplicity in notation s denotes both hidden and observed variables, but the posteriors q are defined only over hidden variables.

Each individual distribution q_j, defined on variables $\mathbf{h}_j = \{h_j^i\}_{i=1}^I$, is used in multiple terms in the above summation, as variables from $\mathbf{h}_j$ are sometimes used as parents and sometimes as children in the conditionals $\log p(s_j^i \mid \mathbf{pa}(s_j^i))$. As opposed to deriving the estimation algorithm by setting the derivatives of the free energy to zero, we first isolate the part of the free energy that depends on q_j,

$$F_{q_j} = \sum_{\mathbf{h}_j} q_j \log q_j - \sum_{\mathbf{h}_j} q_j \sum_{\mathbf{h}_{k\in\mathcal{C}_j}} \left[\prod_{k\in\mathcal{C}_j} q_k\right] \sum_i \log p[s_j^i \mid \mathbf{pa}(s_j^i)] - \sum_{\mathbf{h}_j} q_j \sum_{k \mid j\in\mathcal{C}_k} \sum_{\substack{\mathbf{h}_k,\\ \mathbf{h}_{\mathcal{C}_k\setminus j}}} q_k \left[\prod_{\ell\in\mathcal{C}_k\setminus j} q_\ell\right] \sum_i \log p[s_k^i \mid \mathbf{pa}(s_k^i)].$$

Then, we rewrite it as

$$F_{q_j} = \sum_{\mathbf{h}_j} q_j \log q_j - \sum_{\mathbf{h}_j} q_j \sum_i \log \phi(s_j^i, \mathbf{s}_j^{\mathcal{T}_i}), \qquad (2)$$

with

$$\log \phi(s_j^i, \mathbf{s}_j^{\mathcal{T}_i}) = \sum_{\mathbf{h}_{k\in\mathcal{C}_j}} \left[\prod_{k\in\mathcal{C}_j} q_k\right] \log p[s_j^i \mid \mathbf{pa}(s_j^i)] + \sum_{k \mid j\in\mathcal{C}_k} \sum_{\substack{\mathbf{h}_k,\\ \mathbf{h}_{\mathcal{C}_k\setminus j}}} q_k \left[\prod_{\ell\in\mathcal{C}_k\setminus j} q_\ell\right] \log p[s_k^i \mid \mathbf{pa}(s_k^i)].$$

Finally, we recognize that if all relevant (neighboring) distributions but q_j are fixed, then (2) has exactly the form of the standard tree model (4). Thus, given the log potentials $\log \phi(s_j^i, \mathbf{s}_j^{\mathcal{T}_i})$, we can use the forward–backward algorithm (belief propagation on trees) to find q_j that optimizes F_{q_j} exactly, as discussed in Section 4. On the other hand, having computed the optimal q_j, it can be used to perform the necessary expectations in computation of the potentials (3) for the neighboring trees. Thus the optimization of the free energy can be performed by initializing all distributions q_j to be uniform, and then iterating over each distribution while keep the others fixed so as to minimize F. Iterations continue until the change in F is negligible.

Like the mean field method, and unlike the loopy belief propagation or the simple Markov-chain approximation (Siepel and Haussler, 2003), this algorithm comes with the guarantee that it will converge to the value of the free energy that bounds the negative log likelihood of the data. Unlike both the mean field and the loopy belief propagation technique, this algorithm captures in the posterior the entire marginal distribution on all tree variables for each site j, and thus, as we show later, provides a significantly better bound than other techniques.

5.2 Product of chains

Another similar way to approximate the posterior is to factor it into I different individual probability distributions, each defined over all nucleotides in one sequence, $q = \prod_i q^i(\{h_j^i\}_{j=1}^J)$. The variational optimization technique that uses this approximation is derived in the same way, essentially just by switching i and j and $\mathcal{T}_i$ and $\mathcal{C}_j$ in the equations of the previous section.

The advantage of this technique is that it groups many more variables into each distribution, as the length of the sequences can be of the order of hundred of thousands, while the number of different species is much smaller. However, the product-of-trees approximation focuses on the combinations of variables that are much more correlated, and (as we shall see in our experiments) is more accurate.

6 EXPERIMENTS AND DISCUSSION

In this section, we compare the performance of four different posterior approximations on the task of computing the log likelihood of the data as a model score: (1) variational inference using the product-of-trees approximation,

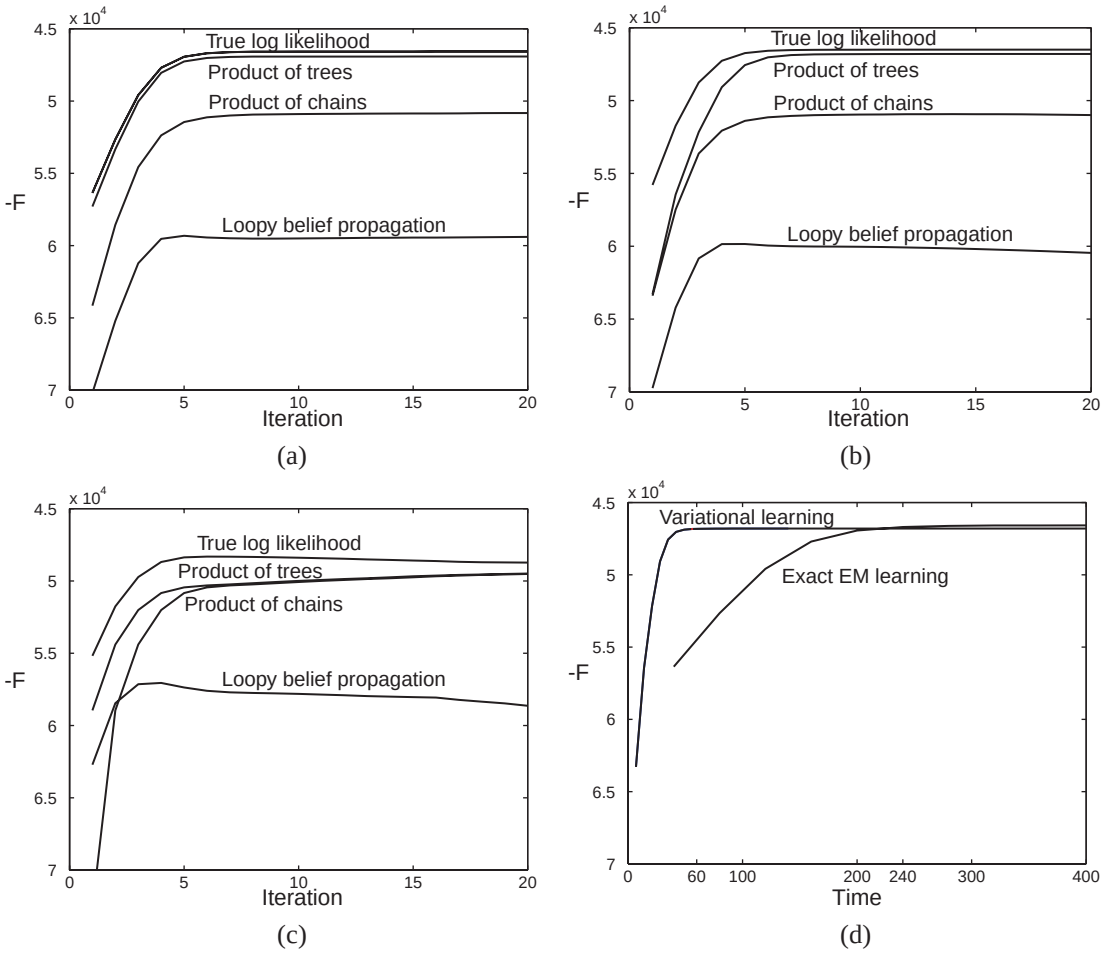

Fig. 2. Quality of the bounds (**a**)–(**c**); and the computational cost comparison (**d**).

(2) variational inference using the product-of-chains approximation (3) loopy belief propagation and (4) the simple Markov-chain approximation. The mean field and ICM approximations under perform these techniques.

We learn the parameters using a generalized EM algorithm (Neal and Hinton, 1998) in which the generalized E step is a computation of the approximate posterior as described in Section 5, and the M step is performed using BFGS quasi-Newton optimization.

As our main goal is to evaluate how close different approximations are to the exact inference, we also used an (expensive) technique to compute the exact log likelihood and posterior $p(\mathbf{h} \mid \mathbf{x})$. In this exact approach, we clique all observed nodes $\mathbf{x}_j$ and unobserved nodes $\mathbf{h}_j$, and consider them as new variables with a much larger configuration space. For example, when there are five hidden nodes in slice j in the model, the total number of possible configurations for $\mathbf{h}_j$ is 4^5. The modified model assumes a form of a single HMM; and we can estimate the true posterior in the form $\prod_j p(\mathbf{h}_j \mid \mathbf{h}_{j-1}, \mathbf{x})$ using the forward–backward algorithm, thus avoiding the brute force search over all 4^{I*J} configurations. Even when done in this manner, exact inference is still extremely slow, and that limited us to models with only three species at the leaves, hence two hidden sequences in the inner nodes of tree.

We ran tests on two subsets of data used in Siepel and Haussler (2003). Both datasets correspond to trees with three species at the leaves. Set A consisted of sequences from cow, mouse and human; and set B had sequences from cow, pig and dog. Sequences were of length 133 and 99 Kb, respectively, and were taken from the region of the CFTR gene. The alignment used in Siepel and Haussler (2003) was used here as well.

Figure 2 illustrates the tightness (or looseness) of the various bounds. In Figure 2a, we show exact log likelihood as well as the free energy of the variational and loopy belief propagation approximations during exact EM learning—i.e. during parameter learning that uses an exact-inference E step. These curves illustrate how tight the various approximations are for a range of model parameters. In Figure 2b, we show a similar graph, but for parameters obtained during generalized EM learning based on the product of trees approximation. In Figure 2c, the plots are for the parameters obtained during

	AA	AC	AG	AT	CA	CC	CG	CT	GA	GC	GG	GT	TA	TC	TG	TT
	Equilibrium frequencies															
	0.09	0.05	0.07	0.08	0.07	0.05	0.006	0.07	0.06	0.04	0.05	0.05	0.07	0.06	0.07	0.1
	Rate matrix															
AA	-1.29	0.14	0.44	0.12	0.16	0.00	0.00	0.00	0.32	0.00	0.00	0.00	0.11	0.00	0.00	0.00
AC	0.17	-2.00	0.17	0.90	0.00	0.17	0.00	0.00	0.00	0.45	0.00	0.00	0.00	0.14	0.00	0.00
AG	0.74	0.14	-1.67	0.16	0.00	0.00	0.15	0.00	0.00	0.00	0.37	0.00	0.00	0.00	0.10	0.00
AT	0.12	0.55	0.17	-1.68	0.00	0.00	0.00	0.17	0.00	0.00	0.00	0.55	0.00	0.00	0.00	0.12
CA	0.17	0.00	0.00	0.00	-2.01	0.18	0.56	0.14	0.12	0.00	0.00	0.00	0.85	0.00	0.00	0.00
CC	0.00	0.18	0.00	0.00	0.11	-2.19	0.14	0.80	0.00	0.14	0.00	0.00	0.00	0.82	0.00	0.00
CG	0.00	0.00	0.52	0.00	**7.81**	0.42	-17.50	0.52	0.00	0.00	0.42	0.00	0.00	0.00	**7.81**	0.00
CT	0.00	0.00	0.00	0.16	0.10	0.37	0.15	-1.67	0.00	0.00	0.00	0.14	0.00	0.00	0.00	0.74
GA	0.72	0.00	0.00	0.00	0.15	0.00	0.00	0.00	-1.68	0.12	0.44	0.11	0.15	0.00	0.00	0.00
GC	0.00	0.76	0.00	0.00	0.00	0.18	0.00	0.00	0.17	-2.21	0.18	0.76	0.00	0.17	0.00	0.00
GG	0.00	0.00	0.80	0.00	0.00	0.00	0.14	0.00	0.82	0.14	-2.19	0.18	0.00	0.00	0.11	0.00
GT	0.00	0.00	0.00	0.90	0.00	0.00	0.00	0.17	0.14	0.45	0.17	-2.00	0.00	0.00	0.00	0.17
TA	0.16	0.00	0.00	0.00	0.54	0.00	0.00	0.00	0.16	0.00	0.00	0.00	-1.75	0.16	0.54	0.16
TC	0.00	0.11	0.00	0.00	0.00	0.44	0.00	0.00	0.00	0.12	0.00	0.00	0.15	-1.68	0.15	0.72
TG	0.00	0.00	0.14	0.00	0.00	0.00	0.56	0.00	0.00	0.00	0.18	0.00	0.85	0.12	-2.01	0.17
TT	0.00	0.00	0.00	0.12	0.00	0.00	0.00	0.44	0.00	0.00	0.00	0.14	0.11	0.32	0.16	-1.29

Fig. 3. Equilibrium distribution of nucleotides and rate matrix estimated using product-of-trees method on sequence from region of the CFTR gene in human genome and homologous sequences from eight eutherian mammals. The matrix is scaled so that at the equilibrium the expected number of substitutions per site is one on a branch of length one. The rates corresponding to CpG effect are shown in boldface.

generalized EM learning based on the product of chains approximation. In order to lower the total amount of computation, these experiments were performed on shorter 20 Kb sequences from dataset A.

When tracking behavior of various bounds during exact EM iterations, we see that, for the product-of-trees approximation, the bound is much tighter than for the other two approximations. The difference between the bound and the true likelihood is due to the dependencies that are absent from the approximation but are present in the true posterior. The product-of-chains approximation, which is not capturing the correlations arising from evolution, is performing worse. Finally, the loopy belief propagation technique that captures only short-range dependencies performs worst of all.

Also interesting to note, the approximation errors accumulate during EM learning, resulting in a much worse final result in the product of chains approximation than one would expect looking at Figure 2a. This illustrates the importance of using an approximation that bounds the log likelihood as tightly as possible.

Figure 2d illustrates the computational gain that we obtain by using the product-of-trees approximation on this relatively small task. In this graph, we show the log likelihood estimate as the function of time during EM and variational EM learning. The computational gains are even more dramatic for a larger number of longer sequences. The complexity of the variational E step is linear in the number of hidden variables, which in turn is linear in number of data points; the complexity of the exact EM is exponential in the number of nodes in the tree and linear in length of the aligned sequences.

Finally we compared the simple Markov-chain approximation to the other bounds available on the full sets A and B. Using U2S parameters that are at a local maximum for the simple Markov-chain approximation, we computed the following values:

Method	Likelihood Set A	Set B
Exact EM	−296 413	−217 843
Variational	−299 756	−219 427
Siepel–Haussler	−300 139	−220 408

We see that the simple Markov-chain approximation method of likelihood computation underestimates the likelihood of data, compared to the product-of-trees variational bound.

7 APPLICATION

We applied our best approximation, the product of trees approximation, to the CFTR gene in human genome and homologous sequences from eight eutherian mammals: chimp, baboon, cow, pig, cat, dog, mouse and rat. These sequences have been selected from non-coding regions. Length of an aligned sequence for each of the species was 162 743 nucleotides. We ran our approximation with a near uniform Q matrix and obtained rate matrix estimates

presented in Figure 3. We found that the estimated rate matrix **Q** had mutation rates from CG to TG (and from CG to CA, due to the strand symmetry of the U2S model), indicative of the presence of a CpG effect. Note that, with a single-site evolutionary model as opposed to a dinucleotide model, a CpG effect cannot be captured by the rate matrix.

The substitution rates we estimated are within 6.7% of the estimates from Siepel and Haussler (2003). The rates in Siepel and Haussler (2003) were estimated using shorter 20 Kb sequences from the same region. We also found that the branch length estimates are on average 12.2% shorter than those estimated using PAML (Yang, 1997) with single-site models HKY (Hasegawa *et al*., 1985) and UNR (Yang, 1994) on the whole dataset. The unrooted phylogenetic trees for the nine species based on dinucleotide and mononucleotide models are shown in Jojic *et al*. (2003).

REFERENCES

Arndt,P.F., Burge,C.B. and Hwa,T. (2002) DNA sequence evolution with neighbor-dependent mutation, In Myers,G., Hannenhalli,S., Istrail,S., Pevzner,P. and Waterman,M. (eds), *Proceedings of the Sixth Annual International Conference on Computational Biology*. ACM, New York, pp. 32–38.

Averof,M., Rokas,A., Wolfe,K.H. and Sharp,P.M. (2000) Evidence for a high frequency of simultaneous double-nucleotide substitutions. *Science*, **287**, 1283–1286.

Durbin,R., Eddy,S., Krogh,A. and Mitchison,G. (1998) *Biological Sequence Analysis: Probabilistic Models of Proteins and Nucleic Acids*. Cambridge University Press, Cambridge.

Felsenstein,J. (1981) Evolutionary trees from DNA sequences. *J. Mol. Evol.*, **17**, 368–376.

Felsenstein,J. and Churchill,G.A. (1996) A hidden Markov model approach to variation among sites in rate of evolution. *Mol. Biol. Evol.*, **13**, 93–104.

Hasegawa,M., Kishino,H. and Yano,T. (1985) Dating the human–ape splitting by a molecular clock of mitochondrial DNA. *J. Mol. Evol.*, **22**, 160–174.

Jensen,J.L. and Pedersen,A.M.K. (2000) Probabilistic models of DNA sequence evolution with context dependent rates of substitution. *Adv. Appl. Prob.*, **32**, 499–517.

Jojic,V., Jojic,N., Meek,C., Geiger,D., Siepel,A., Haussler,D. and Heckerman,D. (2003) Efficient approximations for learning phylogenetic hmm models from data. *Technical Report 2003-62*, Microsoft Research.

Jordan,M., Ghahramani,Z., Jaakkola,T. and Saul,L. (1999), An introduction to variational methods for graphical models. *Learning in Graphical Models*. Michael & Jordan, MIT Press, Cambridge.

Nakhleh,L., Sun,J., Warnow,T., Linder,C., Moret,B. and Tholse,A. (2003) Towards the development of computational tools for evaluating phylogenetic network reconstruction methods. *Proceedings of the Eighth Pacific Symposium on Biocomputing*, **8**, 315–326.

Neal,R. and Hinton,G. (1998) A view of the EM algorithm that justifies incremental, sparse, and other variants. *Learning in Graphical Models*. Michael & Jordan, MIT Press, Cambridge, Kluwer, pp. 355–368.

Neyman,J. (1971) Molecular studies of evolution: A source of novel statistical problems. In Gupta,S. and Yackel,J. (eds), *Statistical Decision Theory and Related Topics*. Academic Press, New York, pp. 1–27.

Pearl,J. (1988) *Probabilistic Reasoning in Intelligent Systems: Networks of Plausible Inference*. Morgan Kaufmann, San Mateo, CA.

Pedersen,A.M.K. and Jensen,J.L. (2001) A dependent rates model and MCMC based methodology for the maximum likelihood analysis of sequences with overlapping reading frames. *Mol. Biol. Evol.*, **18**, 763–776.

Siepel,A. and Haussler,D. (2003) Combining phylogenetic and hidden Markov models in biosequence analysis. *Proceedings of the Seventh Annual International Conference on Computational Molecular Biology (RECOMB 2003)*, pp. 277–286.

Strimmer,K. and Moulton,V. (2000) Likelihood analysis of phylogenetic networks using directed graphical models. *Mol. Biol. and Evol.*, **17**, 875–881.

Whelan,S., Liò,P. and Goldman,N. (2001) Molecular phylogenetics: state-of-the-art methods for looking into the past. *Trends Genet.*, **17**, 262–272.

Yang,Z. (1994) Maximum likelihood phylogenetic estimation from DNA sequences with variable rates over sites: approximate methods. *J. Mol. Evol.*, **39**, 306–314.

Yang,Z. (1995) A space-time process model for the evolution of DNA sequences. *Genetics*, **139**, 993–1005.

Yang,Z. (1997) PAML: a program package for phylogenetic analysis by maximum likelihood. *CABIOS*, **13**, 555–556.

Yang,Z., Goldman,N. and Friday,A. (1994) Comparison of models for nucleotide substitution used in maximum likelihood phylogenetic estimation. *Mol. Biol. Evol.*, **11**, 316–224.