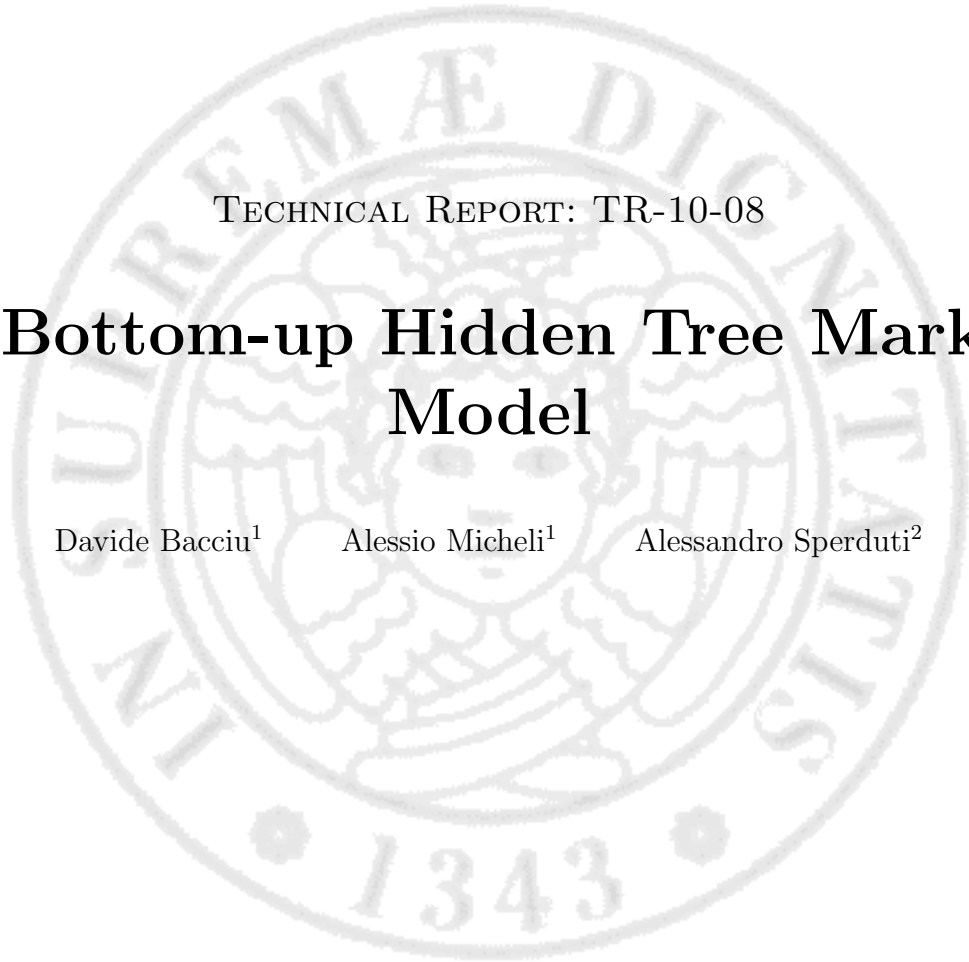
Università di Pisa
Dipartimento di Informatica



Technical Report: TR-10-08

A Bottom-up Hidden Tree Markov Model

Davide Bacciu[1] Alessio Micheli[1] Alessandro Sperduti[2]

April 30, 2010

address: Largo B. Pontecorvo 3, 56127 Pisa, Italy. tel: +39 050 2212700 fax: +39 050 2212726

A Bottom-up Hidden Tree Markov Model

Davide Bacciu[1], Alessio Micheli[1], and Alessandro Sperduti[2]

[1] Dipartimento di Informatica, Università di Pisa,
Largo B. Pontecorvo 3, Pisa, Italy,
{bacciu,micheli}@di.unipi.it
[2] Dipartimento di Matematica Pura e Applicata, Università di Padova,
Via Trieste 63, Padova, Italy,
sperduti@math.unipd.it

Abstract. Hidden Tree Markov Models describe probability distributions over tree-structured data by defining a top-down generative process from the root to the leaves of the tree. We provide a novel compositional hidden tree Markov model that inverts the generative process, allowing hidden states to better correlate and model the co-occurrence of substructures among the child subtrees of internal nodes. To this end, we introduce a mixed memory approximation that factorizes the joint children-to-parent state transition matrix as a mixture of pairwise transitions. This Technical Report provides and in-depth introduction to the Bottom-Up Hidden Tree Markov Model, including the details of the learning and inference procedures.

1 Introduction

The Hidden Tree Markov Model (HTMM) has been introduced as a general tool for modeling probability distributions over spaces of trees [1, 2]. Similarly to how an Hidden Markov Model (HMM) processes sequential data, HTMM defines a generative process for labeled trees that starts at the root node and ends with the leaves emission. Given the direction of the generative process, we refer to these models as *top-down* HTMMs.

Taken as a Bayesian network, a top-down tree generative process builds on a strong assumption, i.e. that child subtrees are independent provided that the parent state is observed. This entails that no hidden state of the Markovian model can capture information concerning the co-occurrence of particular substructures in its child subtrees. As for the root node, the top-down generative process results in its hidden state assignment depending solely on the prior distribution, likewise to the initial state of an HMM. Although this appears as a perfectly sensible choice when dealing with sequential data, its application to hierarchically structured information is quite counterintuitive, as commonsense suggests that the root is the best node where to convey information concerning the whole tree structure.

A natural way to consider the dependency between sibling subtrees while routing sufficient structural information to the root node is to take a bottom-up approach to tree generation, which entails inverting the parent-child causal dependencies in the HTMM model. The *Hidden Recursive Model* (HRM) [3] has long since postulated the opportunity of a bottom-up, or recursive, probabilistic approach, but this has been described only within the scope of a theoretical framework whose realization seemed to be limited to small trees with binary out-degree, due to the computational problem introduced by the inversion of the parent-children causal relationship. Since each node in a tree has at most one parent but a possibly large number of children, the introduction of a bottom-up state transition brings in an explosion of the parameters space whose size grows exponentially with the nodes' outdegree. This has, so far, prevented the development of *practical* bottom-up probabilistic models for trees, whereas in other areas of the machine learning community, such as neural networks [3, 4], the bottom-up approach is the prominent model

This TR is a technical annex for the in-press article: D. Bacciu, A. Micheli and A. Sperduti, "Compositional Generative Mapping of Structured Data", *Proceedings of the 2010 International Joint Conference on Neural Networks (IJCNN 2010)*, 2010, IEEE

for non-flat data. This is motivated by the inherently recursive nature of hierarchical information, which can be effectively captured only by taking a compositional approach that first discovers the less articulated substructures at the bottom of the tree, before tackling with the complexity of deep subtrees.

We introduce a novel *Bottom-Up Hidden Tree Markov Model* (BHTMM, in short) that defines a generative process propagating from the leaves to the root of the tree, where the joint state transition problem is efficiently tackled by resorting to an approach known as *mixed memory* approximation in coupled HMMs [5] or as *switching parent* model in dynamic Bayesian networks [6]. This approximation allows factorizing complex state spaces into a mixture of *simpler* distributions and, as such, can be used to break the joint children-to-parent transition matrix into a mixture of pairwise child-to-parent state transition probabilities. Here, we show novel and efficient procedures for parameter learning and inference that incorporate the mixed memory approximation and account for the inversion of the conditional independence relationships of the bottom-up model.

2 The Bottom-Up Hidden Tree Markov Model (BHTMM)

A rooted tree $\overline{\mathbf{y}}^n$ is an acyclic directed graph consisting of a set of nodes $\mathbf{U}_{\overline{\mathbf{y}}^n}$, each characterized by an observed label y_u, that are connected with each other through a set of directed edges defining a parent to child relationship. By definition, each node u has at maximum one parent, denoted as $pa(u)$. We also assume a finite maximum outdegree L, i.e. the number of children of a node. Likewise sequential data, an observed tree is modeled by a generative process defined by a set of hidden states variables $Q_u \in [1, \ldots, C]$ following the same indexing as the observed node u.

The Markovian assumption for a top-down HTMM dictates that the current state of a node u depends solely on that of its parent $pa(u)$ (see the corresponding Bayesian Network in Fig. 1). Given an observed tree $\overline{\mathbf{y}}^n$ and the hidden states assignment $Q_1 = x_{i_1}, \ldots, Q_{U_{\mathbf{y}^\mathbf{n}}} = x_{i_{U_{\mathbf{y}^\mathbf{n}}}}$, their joint distribution can be factorized by exploiting such conditional independence relations, obtaining

$$P(\overline{\mathbf{y}}^n, Q_1, \ldots, Q_{U_{\overline{\mathbf{y}}^n}}) = P(Q_1)p(y_1|Q_1) \prod_{u=2}^{U_{\overline{\mathbf{y}}^n}} P(y_u|Q_u)P(Q_u|Q_{pa(u)}). \tag{1}$$

A Bottom-up Hidden Tree Markov Model (BHTMM) reverses the parent-to-children dependency described by (1), hence assuming that state transitions are dependent on the hidden state of the children (confront the top-down model in Fig. 1 with the bottom-up in Fig. 2). Again, assume a known hidden states assignment $Q_1 = x_{i_1}, \ldots, Q_{U_{\mathbf{y}^\mathbf{n}}} = x_{i_{U_{\mathbf{y}^\mathbf{n}}}}$; the bottom-up generative process for tree $\overline{\mathbf{y}}^n$ factorizes as

$$\begin{aligned} P(\overline{\mathbf{y}}^n, Q_1 = x_{i_1}, \ldots, Q_{U_{\overline{\mathbf{y}}^n}} = x_{i_{U_{\overline{\mathbf{y}}^n}}}) = & \prod_{u' \in \text{leaf}(\overline{\mathbf{y}}^n)} \overbrace{P(Q_{u'} = x_{i_{u'}})}^{prior} \times \overbrace{P(y_{u'}|Q_{u'} = x_{i_{u'}})}^{leaves\ emission} \\ \times & \prod_{u \in \mathbf{U}_{\overline{\mathbf{y}}^n} \setminus \text{leaf}(\overline{\mathbf{y}}^n)} \underbrace{P(y_u|Q_u = x_{i_u})}_{emission} \times \underbrace{P(Q_u = x_{i_u}|Q_{ch_1(u)} = x_{i_{ch_1(u)}}, \ldots, Q_{ch_L(u)} = x_{i_{ch_L(u)}})}_{transition} \end{aligned} \tag{2}$$

where $\text{leaf}(\overline{\mathbf{y}}^n)$ denotes the set of leaves in tree $\overline{\mathbf{y}}^n$ and $ch_l(u)$ identifies the l-th child of node u. The transition probability in (2) states that the hidden state of an internal node is conditional on all its children state. The likelihood of the BHTMM model is obtained by marginalizing the unknown hidden state associations in (2)

$$\mathcal{L} = \prod_{n=1}^{N} \sum_{x_{i_1}, \ldots, x_{i_{i_{U_{\overline{\mathbf{y}}^n}}}}} P(\overline{\mathbf{y}}^n, Q_1 = x_{i_1}, \ldots, Q_{U_{\overline{\mathbf{y}}^n}} = x_{i_{U_{\overline{\mathbf{y}}^n}}}) \tag{3}$$

where N is the number of trees in the data set. In practice, the sum-marginalization in (3) is rewritten as a (more tractable) product over state assignments, when maximizing the log-likelihood *completed* by hidden indicator

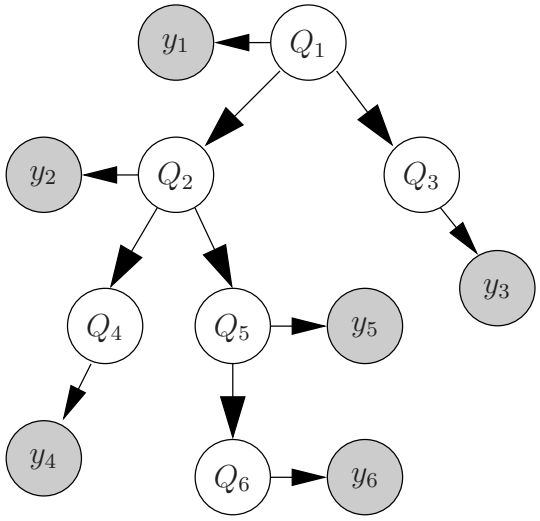


Fig. 1. Bayesian network for the top-down Hidden Tree Markov Model: shaded nodes denote observed labels while hidden state variables are depicted as empty nodes.

variables z_{ui}^n that denote which state x_i is responsible for the generation of node u (see [7] for details). The resulting complete log-likelihood is

$$\begin{aligned}\log \mathcal{L}_c = &\log \prod_{n=1}^{N} \prod_{u' \in \text{leaf}(\overline{\mathbf{y}}^n)} \prod_{i=1}^{C} \{P(Q_{u'} = x_i) \times P(y_{u'} | Q_{u'} = x_i)\}^{z_{u'i}^n} \\ &\times \prod_{u \in \mathbf{U}_{\overline{\mathbf{y}}^n} \setminus \text{leaf}(\overline{\mathbf{y}}^n)} \prod_{i=1}^{C} \prod_{j_1,\ldots,j_{L_u}=1}^{C} \{P(y_u | Q_u = x_i)\}^{z_{ui}^n} \times \left\{ P(Q_u = x_i | \bigcap_{l=1}^{L} Q_{ch_l(u)} = x_{j_l}) \right\}^{z_{ui}^n \prod_{l=1}^{L} z_{ch_l(u) j_l}^n} .\end{aligned} \tag{4}$$

where the Markovian assumption on the label emission has been used to take $P(y_u|Q_u = x_i)$ out of the $z_{ui}^n \prod_{l=1}^{L} z_{ch_l(u)j_l}^n$ exponentiation (i.e. emission y_u is conditionally independent from the rest of the variables given Q_u).

The term $P(Q_u = x_i | \bigcap_{l=1}^{L} Q_{ch_l(u)} = x_{j_l})$ is a short form for the children-dependent state transition probability in (2), while the innermost product in the second line of (4) marginalizes the hidden state assignment for the children nodes. Notice that L represents the maximum number of distinct children in the trees: for the purpose of this paper, we assume strict positional stationarity, hence we define a null state $x_\emptyset$ that is used to model the absence of a the l-th child, i.e. $Q_{ch_l(u)} = x_\emptyset$.

The problem with the formulation in (4) is that it becomes computationally impractical for trees other than binary due to the explosive size of the joint conditional transition distribution, that is order of C^{L+1}. From an alternative perspective, the model described in (4) can be interpreted as an higher order Hidden Markov model (HO-HMM), i.e. with a memory of length $L > 1$. Within this context, it has been proposed an approximation of the transition distribution as a mixture of *simpler* distributions. In practice, such re-parametrization, known as *Mixed Memory Markov Model* [5] or *Mixture Transition Distribution* [8], represents the joint transition matrix as a convex combination of L elementary transition matrices. In our scenario, we seek an approximation to the joint transition

$$P(Q_u = x_i | \bigcap_{l=1}^{L} Q_{ch_l(u)} = x_{j_l}) = P(Q_u = x_i | Q_{ch_1(u)} = x_{j_1}, Q_{ch_2(u)} = x_{j_2}, \ldots, Q_{ch_L(u)} = x_{j_L}). \tag{5}$$

This corresponds to a graphical model with L children nodes $Q_{ch_1(u)}, \ldots, Q_{ch_L(u)}$ with directed arrows entering Q_u. Suppose that we introduce a fictitious node (with an arrow entering Q_u, see Fig. 2) that defines a switching latent variable $S_u \in \{1, \ldots, L\}$ such that

$$P(Q_u = x_i | S_u = l, Q_{ch_1(u)} = x_{j_1}, \ldots, Q_{ch_l(u)} = x_{j_l}, \ldots, Q_{ch_L(u)} = x_{j_L}) = P(Q_u = x_i | Q_{ch_l(u)} = x_{j_l}). \tag{6}$$

Given that we interpret S_u as a latent variable, we can marginalize it out to recover the original expression in (5), that is

$$\begin{aligned} &P(Q_u = x_i | Q_{ch_1(u)} = x_{j_1}, \ldots, Q_{ch_l(u)} = x_{j_l}, \ldots, Q_{ch_L(u)} = x_{j_L}) \\ &= \sum_{l=1}^{L} P(Q_u = x_i, S_u = l | Q_{ch_1(u)} = x_{j_1}, Q_{ch_2(u)} = x_{j_2}, \ldots, Q_{ch_L(u)} = x_{j_L}) \\ &= \sum_{l=1}^{L} P(S_u = l) P(Q_u = x_i | Q_{ch_l(u)} = x_{j_l}) \end{aligned} \tag{7}$$

where we've used the assumption that S_u is independent of $Q_{ch_1(u)}, \ldots, Q_{ch_L(u)}$. This equation states that the joint transition distribution can be approximated as a mixture of L elementary distributions $P(Q_u = x_i | Q_{ch_l(u)} = x_{j_l})$ where the influence of the l-th children on state transition of node u is determined by the weight $P(S_u = l)$. Given that the switching variable is latent, we can learn its prior distribution $P(S_u = l)$ as part of the EM process.

The actual size of the mixture transition approximation depends on the stationarity assumptions that are taken. For instance, the original Mixed Memory model in [5] assumes *full stationarity*, given that the mixing weights (i.e. the prior distribution of the switching parent) is independent w.r.t. the node u, i.e. $\varphi_l \approx \sum_u P(S_u = l)$: this choice has a straightforward interpretation in terms of time series, since φ_l collects statistics on the influence of events that appeared l time steps before the current event. Hence, parameter φ_l determines the generic memory of the system and has a size equal to L, i.e. the maximum allowed memory in the model, resulting in an approximation requiring only $O(C^2 + L_{max})$ parameters (where L_{max} is the maximum out degree among the trees). On the other hand, we might be interested in learning the correlation between a node and its child chains l, hence retaining some form of *positional stationarity* which can be implemented, for instance, by modeling the following mixture transition approximation

$$\begin{aligned} &P(Q_u = x_i | Q_{ch_1(u)} = x_{j_1}, \ldots, Q_{ch_l(u)} = x_{j_l}, \ldots, Q_{ch_L(u)} = x_{j_L}) \\ &= \sum_{l=1}^{L} P(S_u = l) p^l(Q_u = x_i | Q_{ch_l(u)} = x_{j_l}) \end{aligned} \tag{8}$$

where the transition probability $p^l(Q_u = x_i | Q_{ch_l(u)} = x_{j_l})$ now explicitly depends on the position l of the child node, that is to say that there are L_{max} transition distributions for each couple of parent-child states, resulting in an approximation that requires $O(L_{max}C^2 + L_{max})$ independent parameters, which is still feasible if compared to the original $O(C^{L_{max}+1})$.

By substituting the results of (7) into the complete likelihood in (4), we obtain the following approximation

$$\begin{aligned} \log \mathcal{L}_c = \log &\prod_{n=1}^{N} \prod_{u' \in \text{leaf}(\overline{\mathbf{y}}^n)} \prod_{i=1}^{C} \{P(Q_{u'} = x_i) \times P(y_{u'} | Q_{u'} = x_i)\}^{z^n_{u'i}} \\ \times &\prod_{u \in \mathbf{U}_{\overline{\mathbf{y}}^n} \setminus \text{leaf}(\overline{\mathbf{y}}^n)} \prod_{i=1}^{C} \prod_{j_1, \ldots, j_{L_u}=1}^{C} \{P(y_u | Q_u = x_i)\}^{z^n_{ui}} \times \left\{ \sum_{l=1}^{L} P(S_u = l) P(Q_u = x_i | Q_{ch_l(u)} = x_{j_l}) \right\}^{z^n_{ui} \prod_{l=1}^{L} z^n_{ch_l(u) j_l}}, \end{aligned} \tag{9}$$

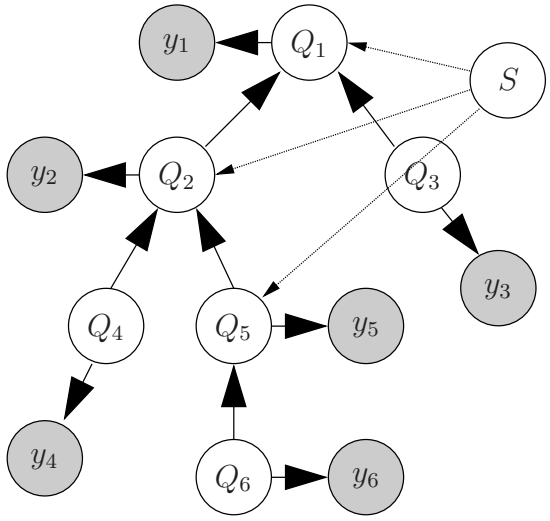


Fig. 2. Bayesian network for the Bottom-up Hidden Tree Markov Model with a shared (i.e. stationary) Switching Parent.

that, by introducing indicator variables t_{ul}^n for the switching parent, can be reformulated as follows

$$\log \mathcal{L}_c = \log \prod_{n=1}^{N} \prod_{u' \in \text{leaf}(\overline{\mathbf{y}}^n)} \prod_{i=1}^{C} \{P(Q_{u'} = x_i) \times P(y_{u'} | Q_{u'} = x_i)\}^{z_{u'i}^n} \times \prod_{u \in \mathbf{U}_{\overline{\mathbf{y}}^n} \setminus \text{leaf}(\overline{\mathbf{y}}^n)} \prod_{i=1}^{C} \prod_{j_1,\ldots,j_{L_u}=1}^{C} \{P(y_u | Q_u = x_i)\}^{z_{ui}^n} \left\{ \prod_{l=1}^{L} \{P(S_u = l) P(Q_u = x_i | Q_{ch_l(u)} = x_{j_l})\}^{t_{ul}^n} \right\}^{z_{ui}^n \prod_{l=1}^{L} z_{ch_l(u) j_l}^n}. \tag{10}$$

Finally, this rewrites as

$$\begin{aligned} \log \mathcal{L}_c = \log \prod_{n=1}^{N} \prod_{u' \in \text{leaf}(\overline{\mathbf{y}}^n)} \prod_{i=1}^{C} \{P(Q_{u'} = x_i) \times P(y_{u'} | Q_{u'} = x_i)\}^{z_{u'i}^n} \\ \times \prod_{u \in \mathbf{U}_{\overline{\mathbf{y}}^n} \setminus \text{leaf}(\overline{\mathbf{y}}^n)} \prod_{i=1}^{C} \prod_{j=1}^{C} \prod_{l=1}^{L} \{P(y_u | Q_u = x_i)\}^{z_{ui}^n} \times \{P(S_u = l) P(Q_u = x_i | Q_{ch_l(u)} = x_j)\}^{z_{ui}^n t_{ul}^n z_{ch_l(u)j}^n} \end{aligned} \tag{11}$$

whose graphical interpretation is given in Fig. 2. Likewise in the top-down HTMM, we can use EM to update the prior, emission and transition distributions as part of the M-Step; in addition, we will need to estimate the mixed memory distribution $P(S_u = l)$ and the related posterior (in the E-Step).

3 Parameter Fitting in the Bottom-Up Model

As discussed previously, the actual form of the model parameters depends on the stationarity assumptions: in the following we assume a mixed memory model with child-specific (i.e. positional) transition matrices. The complete log-likelihood of the model in (11) is rewritten to explicitly include the positional dependency on the child node l

as follows

$$\begin{aligned}\log \mathcal{L}_c = &\sum_{n=1}^{N} \sum_{u' \in \text{leaf}(\overline{\mathbf{y}}^n)} \sum_{i=1}^{C} z_{u'i}^n \log P^{pos(u')}(Q_{u'} = x_i) \\ &+ \sum_{n=1}^{N} \sum_{u \in \mathbf{U}_{\overline{\mathbf{y}}^n}} \sum_{i=1}^{C} z_{ui}^n \log P(y_u | Q_u = x_i) \\ &+ \sum_{n=1}^{N} \sum_{u \in \mathbf{U}_{\overline{\mathbf{y}}^n} \setminus \text{leaf}(\overline{\mathbf{y}}^n)} \sum_{l=1}^{L} t_{ul}^n \log P(S_u = l) \\ &+ \sum_{n=1}^{N} \sum_{u \in \mathbf{U}_{\overline{\mathbf{y}}^n} \setminus \text{leaf}(\overline{\mathbf{y}}^n)} \sum_{i,j=1}^{C} \sum_{l=1}^{L} \overline{z}_{uijl}^n \log P^l(Q_u = x_i | Q_{ch_l(u)} = x_j),\end{aligned} \tag{12}$$

where $pos(u)$ returns the position of node u in the subtree of $pa(u)$. The new indicator $\overline{z}_{uijl} = z_{ui}^n t_{ul}^n z_{ch_l(u)j}^n$ is a reformulation of the previous hidden variables stating that node u is in state x_i while its l-th child is in state x_j.

3.1 E-Step

Following the EM algorithm, we need to maximize the expectation of the complete log-likelihood in (12). In the E-Step, the expected value of the log likelihood function is computed with respect to the distribution the hidden variables Z, conditional on the observed data $\mathbf{Y}$ and the current estimate of the parameters $\theta^{(k)}$, that is

$$E[\log \mathcal{L}_c(\theta; \mathbf{Y}, Z) | \mathbf{Y}, \theta^{(k)}]$$

As discussed previously, the model parameters θ are the positional initial state probability $P^{pos(u')}(Q_{u'} = x_i)$, the positional stationary state transition probability $P^l(Q_u = x_i | Q_{ch_l(u)} = x_j)$, the mixed memory distribution $P(S_u = l)$ as well as the parameters of the emission distribution $P(y_u | Q_u = x_i)$. The hidden variables Z comprise both the hidden state variables z_{ui}^n, the switching parents variables t_{ul}^n, as well as the position-dependent indicators $\overline{z}_{uijl}$ for the mixed memory model. Given the complete log-likelihood in (12), their conditional expected values correspond to the following posterior probabilities

$$E[z_{ui}^n | \overline{\mathbf{y}}^n, \theta^{(k)}] = P(Q_u = x_i | \overline{\mathbf{y}}^n), \tag{13}$$

$$E[\overline{z}_{uijl} | \overline{\mathbf{y}}^n, \theta^{(k)}] = P(Q_u = x_i, Q_{ch_l(u)} = x_j, S_u = l | \overline{\mathbf{y}}^n) \tag{14}$$

$$E[t_{ul} | \overline{\mathbf{y}}^n, \theta^{(k)}] = P(S_u = l | \overline{\mathbf{y}}^n) \tag{15}$$

where $E[\overline{z}_{uijl} | \overline{\mathbf{y}}^n, \theta^{(k)}]$ is the joint posterior probability of node u being in state x_i while its l-th child is in state x_j. These posteriors can be estimated by message passing on the structure on the nodes dependency graph by applying the principles of the *upward-downward* algorithm. The BHTMM model reverses the conditional dependencies with respect to the standard HTMM model thus preventing the use of standard upward-downward rules. In the following, we present a procedure tailored to the bottom-up mixed memory HTMM model, i.e. *inverted upward-downward* algorithm, based on the smoothed probabilities model in [9] and that incorporates the estimation of the switching parents posterior distribution.

In the following, we will show the details of the inverted upward/downward passes; throughout the derivation of the E-step rules we use the following additional notation:

- $\overline{\mathbf{Y}}_u = \overline{\mathbf{y}}_u$ denotes the observed subtree rooted at node u (i.e. $\overline{\mathbf{y}}_1$ is the whole tree);
- $\overline{\mathbf{Y}}_{ch_l(u)} = \overline{\mathbf{y}}_{ch_l(u)}$ denotes the l-th child subtree of node u;
- $\overline{\mathbf{Y}}_{1 \setminus u} = \overline{\mathbf{y}}_{1 \setminus u}$ is the observed tree (i.e. rooted at 1) without the $\overline{\mathbf{y}}_u$ subtree.

The upward-downward algorithm allows computing the posteriors $P(Q_u = x_i|\mathbf{y})$ and $P(Q_u = x_i, Q_{ch_l(u)} = x_j, S_u = l|\mathbf{y})$ by exploiting a factorization into terms computed by a *downward pass*, i.e. from the root to the leaves of the tree, and by an *upward pass*, i.e. from the leaves to the root of the tree. The smoothed algorithm in [9] exploits the following decomposition

$$\begin{aligned}\epsilon_u(i) = P(Q_u = x_i|\overline{\mathbf{Y}}_1 = \overline{\mathbf{y}}_1) = P(Q_u = x_i|\overline{\mathbf{Y}}_{1\setminus u} = \overline{\mathbf{y}}_{1\setminus u}, \overline{\mathbf{Y}}_u = \overline{\mathbf{y}}_u) \overset{\text{bayes}}{=} \\ = \frac{P(\overline{\mathbf{Y}}_{1\setminus u} = \overline{\mathbf{y}}_{1\setminus u}|Q_u = x_i)}{P(\overline{\mathbf{Y}}_{1\setminus u} = \overline{\mathbf{y}}_{1\setminus u}|\overline{\mathbf{Y}}_u = \overline{\mathbf{y}}_u)} P(Q_u = x_i|\overline{\mathbf{Y}}_u = \overline{\mathbf{y}}_u)\end{aligned} \tag{16}$$

such that the factorization $\epsilon_u(i) = \alpha_u(i)\beta_u(i)$ can be computed using the *upward* parameters

$$\beta_u(i) = P(Q_u = x_i|\overline{\mathbf{Y}}_u = \overline{\mathbf{y}}_u) \tag{17}$$

as well as the *downward parameters*

$$\alpha_u(i) = \frac{P(\overline{\mathbf{Y}}_{1\setminus u} = \overline{\mathbf{y}}_{1\setminus u}|Q_u = x_i)}{P(\overline{\mathbf{Y}}_{1\setminus u} = \overline{\mathbf{y}}_{1\setminus u}|\overline{\mathbf{Y}}_u = \overline{\mathbf{y}}_u)}. \tag{18}$$

In order to compute the $\beta_u(i)$ parameters in the bottom-up pass, it is also needed to compute and propagate the *auxiliary upward parameters*

$$\beta_{u,ch_l(u)}(i) = \frac{P(\overline{\mathbf{Y}}_{ch_l(u)} = \overline{\mathbf{y}}_{ch_l(u)}|Q_u = x_i)}{P(\overline{\mathbf{Y}}_{ch_l(u)} = \overline{\mathbf{y}}_{ch_l(u)})}. \tag{19}$$

Upward Recursion. The upward recursion can be straightforwardly computed for the *leaf nodes* y_u as

$$\begin{aligned}\beta_u(i) = P(Q_u = x_i|Y_u = y_u) \overset{\text{bayes}}{=} \frac{P(Y_u = y_u|Q_u = x_i)P^{pos(u)}(Q_u = x_i)}{P(Y_u = y_u)} \\ = \frac{P(Y_u = y_u|Q_u = x_i)P^{pos(u)}(Q_u = x_i)}{N_u}\end{aligned} \tag{20}$$

where $P^{pos(u)}(Q_u = x_i)$ is the current estimate of the prior distribution on the leaves, while N_u is chosen to ensure $\sum_j \beta_u(i) = 1$, yielding

$$\beta_u(i) = \frac{P(Y_u = y_u|Q_u = x_i)P^{pos(u)}(Q_u = x_i)}{\sum_{j=1}^{C} P(Y_u = y_u|Q_u = x_j)P^{pos(u)}(Q_u = x_j)}\ ,\ \forall u \in \text{leaf}(\mathbf{y}). \tag{21}$$

The $\beta_u(j)$ values on the *internal* and *root* nodes u is computed by the following recursion

$$\begin{aligned}\beta_u(i) = P(Q_u = x_i|\overline{\mathbf{Y}}_u = \overline{\mathbf{y}}_u) \overset{\text{cond. prob.}}{=} \frac{P(Q_u = x_i, \overline{\mathbf{Y}}_u = \overline{\mathbf{y}}_u)}{P(\overline{\mathbf{Y}}_u = \overline{\mathbf{y}}_u)} \overset{\text{subtree decomp.}}{=} \\ = \frac{P(Q_u = x_i, Y_u = y_u, \overline{\mathbf{Y}}_{ch_1(u)} = \overline{\mathbf{y}}_{ch_1(u)}, \ldots, \overline{\mathbf{Y}}_{ch_L(u)} = \overline{\mathbf{y}}_{ch_L(u)})}{P(\overline{\mathbf{Y}}_u = \overline{\mathbf{y}}_u)} \overset{\text{cond. prob.}}{=} \\ = \frac{P(Y_u = y_u|Q_u = x_i, \overline{\mathbf{y}}_{ch_1(u)}, \ldots, \overline{\mathbf{y}}_{ch_L(u)})P(Q_u = x_i, \overline{\mathbf{y}}_{ch_1(u)}, \ldots, \overline{\mathbf{y}}_{ch_L(u)})}{P(\overline{\mathbf{y}}_u)} \\ = \frac{P(Y_u = y_u|Q_u = x_i)P(Q_u = x_i, \overline{\mathbf{y}}_{ch_1(u)}, \ldots, \overline{\mathbf{y}}_{ch_L(u)})}{P(\overline{\mathbf{y}}_u)}\end{aligned} \tag{22}$$

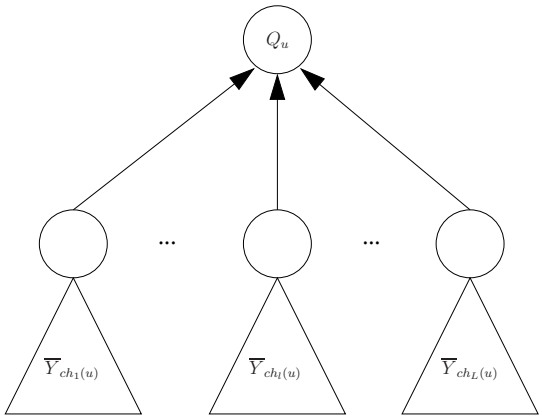


Fig. 3. Child subtrees meet head to head at Q_u: hence they are independent given that Q_u (nor its descendants) are not observed.

where notation has been contracted for the sake of conciseness. The last equality in (22) holds given the Hidden Markov model assumption where node emission is conditionally independent from other variables given the realization of node hidden state. The second term in the numerator needs to be further decomposed following the dependencies in the Bayesian network, that is

$$\begin{aligned} P(Q_u = x_i, \overline{\mathbf{y}}_{ch_1(u)}, \ldots, \overline{\mathbf{y}}_{ch_L(u)}) &= P(Q_u = x_i | \overline{\mathbf{y}}_{ch_1(u)}, \ldots, \overline{\mathbf{y}}_{ch_L(u)}) P(\overline{\mathbf{y}}_{ch_1(u)}, \ldots, \overline{\mathbf{y}}_{ch_L(u)}) \overset{\text{child indep.}}{=} \\ &= P(Q_u = x_i | \overline{\mathbf{y}}_{ch_1(u)}, \ldots, \overline{\mathbf{y}}_{ch_L(u)}) \prod_{l=1}^{L} P(\overline{\mathbf{y}}_{ch_l(u)}) \end{aligned} \tag{23}$$

where the last equality holds given the independency of child subtrees $\overline{\mathbf{y}}_{ch_l(u)}$ when the parent hidden node Q_u is not observed (see Fig. 3).

To factorize the distribution $P(Q_u = x_i | \overline{\mathbf{y}}_{ch_1(u)}, \ldots, \overline{\mathbf{y}}_{ch_L(u)})$ we need to look further into the structure of the BHTMM dependency graph. In the standard upward-downward algorithm, this distribution can be factorized in terms of the product of the children distributions $P(\overline{\mathbf{y}}_{ch_l(u)} | Q_u = x_i)$. Such an equality holds for a top-down HTMM model (i.e. with dependencies oriented from the root to the leaves) since children subtrees are d-separated by the parent hidden state variable Q_u (see Fig. 4.a). In the bottom-up HTMM model, on the other hand, the dependency relations are inverted and the children subtrees u, i.e. $\overline{\mathbf{y}}_{ch_1(u)}, \ldots, \overline{\mathbf{y}}_{ch_L(u)}$, share a common descendant in Bayesian terms, that is the hidden variable Q_u (see Fig. 4.b). Hence, the realization of the hidden variable Q_u introduces a pairwise dependency between all the children subtrees in the corresponding moral graph [10]. Therefore, the joint distribution $P(Q_u = x_i | \overline{\mathbf{y}}_{ch_1(u)}, \ldots, \overline{\mathbf{y}}_{ch_L(u)})$ cannot be straightforwardly factorized. However, if we introduce again

the switching parents approximation, we can rewrite it as follows

$$\begin{aligned}
P(Q_u = x_i|\overline{\mathbf{y}}_{ch_1(u)},\ldots,\overline{\mathbf{y}}_{ch_L(u)}) &\overset{\text{marginalize}}{=} \sum_{l=1}^{L} P(Q_u = x_i, S_u = l|\overline{\mathbf{y}}_{ch_1(u)},\ldots,\overline{\mathbf{y}}_{ch_L(u)}) \overset{\text{cond. prob.}}{=} \\
&= \sum_{l=1}^{L} P(S_u = l|\overline{\mathbf{y}}_{ch_1(u)},\ldots,\overline{\mathbf{y}}_{ch_L(u)}) P(Q_u = x_i|S_u = l, \overline{\mathbf{y}}_{ch_1(u)},\ldots,\overline{\mathbf{y}}_{ch_L(u)}) \overset{\text{switch par.}}{=} \\
&= \sum_{l=1}^{L} P(S_u = l) P(Q_u = x_i|\overline{\mathbf{y}}_{ch_l(u)}) \overset{\text{bayes}}{=} \\
&= \sum_{l=1}^{L} P(S_u = l) \frac{P(\overline{\mathbf{y}}_{ch_l(u)}|Q_u = x_i) P(Q_u = x_i)}{P(\overline{\mathbf{y}}_{ch_l(u)})}.
\end{aligned} \tag{24}$$

Inserting the results of (23) and (24) back into (22) yields

$$\begin{aligned}
\beta_u(i) &= \left(\sum_{l=1}^{L} P(S_u = l) \frac{P(\overline{\mathbf{y}}_{ch_l(u)}|Q_u = x_i)}{P(\overline{\mathbf{y}}_{ch_l(u)})} P(Q_u = x_i) \right) \frac{\prod_{l'=1}^{L} P(\overline{\mathbf{y}}_{ch_{l'}(u)})}{P(\overline{\mathbf{y}}_u)} P(Y_u = y_u|Q_u = x_i) \overset{\beta_{u,ch_l(u)} \text{ def.}}{=} \\
&= \left(\sum_{l=1}^{L} P(S_u = l) \beta_{u,ch_l(u)}(i) P(Q_u = x_i) \right) \frac{\prod_{l'=1}^{L} P(\overline{\mathbf{y}}_{ch_{l'}(u)})}{P(\overline{\mathbf{y}}_u)} P(Y_u = y_u|Q_u = x_i) \overset{N_u \text{ def.}}{=} \\
&= \frac{P(Y_u = y_u|Q_u = x_i) \sum_{l=1}^{L} P(S_u = l) \beta_{u,ch_l(u)}(i) P(Q_u = x_i)}{N_u},
\end{aligned} \tag{25}$$

where N_u is, again, a normalization factor ensuring $\sum_j \beta_u(i) = 1$, yielding

$$\beta_u(i) = \frac{P(Y_u = y_u|Q_u = x_i) \sum_{l=1}^{L} P(S_u = l) \beta_{u,ch_l(u)}(i) P(Q_u = x_i)}{\sum_{j=1}^{C} \sum_{l'=1}^{L} P(Y_u = y_u|Q_u = x_j) P(S_u = l') \beta_{u,ch_{l'}(u)}(j) P(Q_u = x_j)}. \tag{26}$$

In order to compute the expression in (26), we need to propagate the values for computing the *internal node prior* $P(Q_u = x_i)$, as well as the $\beta_{u,ch_l(u)}(i)$ from the each child l of node u. The standard upward-downward algorithm requires an initial downward recursion in order to propagate the prior $P(Q_1 = x_i)$ from the root node to the internal nodes and the leaves. In our bottom-up approach, priors are propagated from the leaves to the root as part of the upward recursion (hence there is no need of an additional downward pass): the corresponding update rule is

$$P(Q_u = x_i) = \sum_{l=1}^{L} \sum_{j=1}^{C} P(S_u = l) p^l(Q_u = x_i|Q_{ch_l(u)} = x_j) P(Q_{ch_l(u)} = x_j). \tag{27}$$

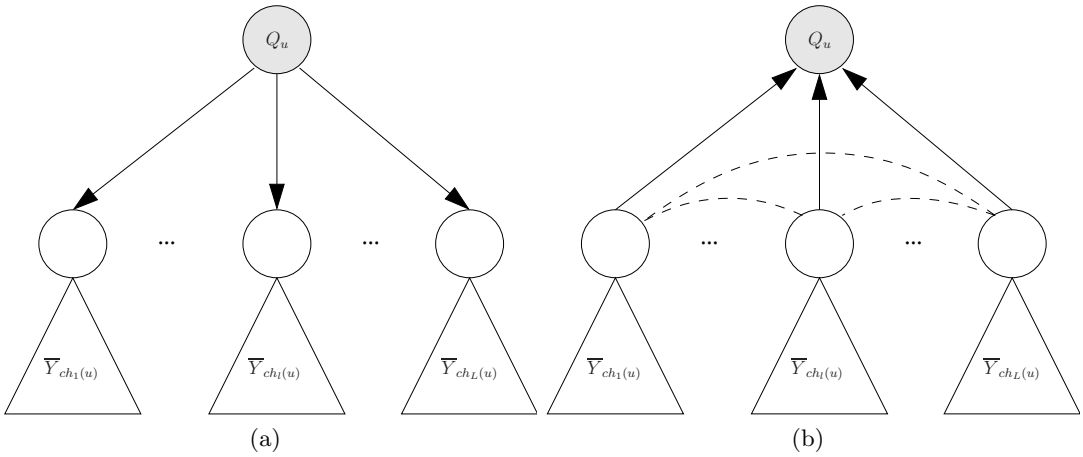


Fig. 4. Child subtrees d-separation: (a) in the *top-down model*, the subtrees meet tail to tail at Q_u and share no common ancestors, hence they are conditionally independent given the observation of Q_u. In the *bottom-up model* (b), the child subtrees meet head-to-head in Q_u; since Q_u is a common ancestor of the subtrees $\overline{\mathbf{Y}}_{ch_l(u)}$, then its realization introduces pairwise dependency links in the corresponding moral graph (the dashed lines in (b)) [10]. Therefore the subtrees are not conditionally independent give Q_u.

The computation of the $\beta_{u,ch_l(u)}(i)$ is performed as part of the upward pass following the factorization

$$
\begin{aligned}
\beta_{u,ch_l(u)}(i) &= \frac{P(\overline{\mathbf{Y}}_{ch_l(u)} = \overline{\mathbf{y}}_{ch_l u} | Q_u = x_i)}{P(\overline{\mathbf{Y}}_{ch_l(u)} = \overline{\mathbf{y}}_{ch_l(u)})} \overset{\text{marginal}}{=} \\
&= \frac{\sum_{j=1}^{C} P(\overline{\mathbf{Y}}_{ch_l(u)} = \overline{\mathbf{y}}_{ch_l(u)}, Q_{ch_l(u)} = x_j | Q_u = x_i)}{P(\overline{\mathbf{Y}}_{ch_l(u)} = \overline{\mathbf{y}}_{ch_l(u)})} \overset{\text{Bayes}}{=} \\
&= \frac{\sum_{j=1}^{C} P(Q_u = x_i | \overline{\mathbf{Y}}_{ch_l(u)} = \overline{\mathbf{y}}_{ch_l(u)}, Q_{ch_l(u)} = x_j) P(\overline{\mathbf{Y}}_{ch_l(u)} = \overline{\mathbf{y}}_{ch_l(u)}, Q_{ch_l(u)} = x_j)}{P(\overline{\mathbf{Y}}_{ch_l(u)} = \overline{\mathbf{y}}_{ch_l(u)}) P(Q_u = x_i)} \overset{\text{Bayes}}{=} \\
&= \frac{\sum_{j=1}^{C} p^l(Q_u = x_i | Q_{ch_l(u)} = x_j) P(Q_{ch_l(u)} = x_j | \overline{\mathbf{Y}}_{ch_l(u)} = \overline{\mathbf{y}}_{ch_l(u)}) P(\overline{\mathbf{Y}}_{ch_l(u)} = \overline{\mathbf{y}}_{ch_l(u)})}{P(\overline{\mathbf{Y}}_{ch_l(u)} = \overline{\mathbf{y}}_{ch_l(u)}) P(Q_u = x_i)} \overset{\text{simplif}}{=} \\
&= \frac{\sum_{j=1}^{C} p^l(Q_u = x_i | Q_{ch_l(u)} = x_j) P(Q_{ch_l(u)} = x_j | \overline{\mathbf{Y}}_{ch_l(u)} = \overline{\mathbf{y}}_{ch_l(u)})}{P(Q_u = x_i)} \overset{\beta_{ch_l(u)}\text{def}}{=} \\
&= \frac{\sum_{j=1}^{C} p^l(Q_u = x_i | Q_{ch_l(u)} = x_j) \beta_{ch_l(u)}(j)}{P(Q_u = x_i)},
\end{aligned}
\tag{28}
$$

which concludes the derivation of the upward pass.

Downward Recursion. Learning in the downward recursion is based on computing the factors $\alpha_u(i)$, or directly the smoothed probabilities $\epsilon_u(i)$. Given the definition in (16), the smoothed probability of the root node Q_1 is computed as

$$
\epsilon_1(i) = P(Q_1 = x_i | \overline{\mathbf{Y}}_1 = \overline{\mathbf{y}}_1) = \beta_1(i) \tag{29}
$$

where $\beta_1(i)$ has been computed during the upwards pass. Each internal and leaf node is, clearly, an l-th child of his parent node (where l defines its position among the children). Hence, to estimate the posterior in (14) we need to

decompose the pairwise smoothed probability as

$$\begin{aligned}\epsilon^{l}_{u,ch_l(u)}(i,j) &= P(Q_u = x_i, Q_{ch_l(u)} = x_j, S_u = l | \overline{\mathbf{Y}}_1 = \overline{\mathbf{y}}_1) \overset{\text{cond. prob.}}{=} \\ &= P(Q_u = x_i, Q_{ch_l(u)} = x_j, S_u = l, \overline{\mathbf{Y}}_1 = \overline{\mathbf{y}}_1) P(\overline{\mathbf{Y}}_1 = \overline{\mathbf{y}}_1).\end{aligned} \tag{30}$$

From Bayes rule, we know that, for every node u', $P(Q_{u'} = x_{j'} | \overline{\mathbf{Y}}_1 = \overline{\mathbf{y}}_1) = P(Q_{u'} = x_{j'}, \overline{\mathbf{Y}}_1 = \overline{\mathbf{y}}_1) P(\overline{\mathbf{Y}}_1 = \overline{\mathbf{y}}_1)$: choosing $u' = u$ and inserting this result into (30), it can be rewritten as

$$\epsilon^{l}_{u,ch_l(u)}(i,j) = \frac{P(Q_u = x_i, Q_{ch_l(u)} = x_j, S_u = l, \overline{\mathbf{Y}}_1 = \overline{\mathbf{y}}_1) P(Q_u = x_i | \overline{\mathbf{Y}}_1 = \overline{\mathbf{y}}_1)}{P(Q_u = x_i, \overline{\mathbf{Y}}_1 = \overline{\mathbf{y}}_1)}. \tag{31}$$

The denominator in (31) can be factorized, for any child $ch_l(u)$ of node u, by the following Bayes expansion

$$\begin{aligned}P(Q_u = x_i, \overline{\mathbf{Y}}_1 = \overline{\mathbf{y}}_1) &= P(Q_u = x_i, \overline{\mathbf{Y}}_{ch_l(u)} = \overline{\mathbf{y}}_{ch_l(u)}, \overline{\mathbf{Y}}_{1 \setminus ch_l(u)} = \overline{\mathbf{y}}_{1 \setminus ch_l(u)}) \overset{\text{cond. prob.}}{=} \\ &= P(Q_u = x_i, \overline{\mathbf{Y}}_{1 \setminus ch_l(u)}) P(\overline{\mathbf{Y}}_{ch_l(u)} | Q_u = x_i, \overline{\mathbf{Y}}_{1 \setminus ch_l(u)}) \overset{\text{Bayes}}{=} \\ &= P(Q_u = x_i, \overline{\mathbf{Y}}_{1 \setminus ch_l(u)}) \\ &\quad \times \frac{P(\overline{\mathbf{Y}}_{ch_l(u)}) P(Q_u = x_i | \overline{\mathbf{Y}}_{ch_l(u)}) P(\overline{\mathbf{Y}}_{1 \setminus ch_l(u)} | Q_u = x_i, \overline{\mathbf{Y}}_{ch_l(u)})}{P(\overline{\mathbf{Y}}_{1 \setminus ch_l(u)} | Q_u = x_i) P(Q_u = x_i)}.\end{aligned} \tag{32}$$

where, for the sake of conciseness, we have been using $\overline{\mathbf{Y}}_{u'}$ as a short form for $\overline{\mathbf{Y}}_{u'} = \overline{\mathbf{y}}_{u'}$. Within a top-down HTMM model we could have used conditional independency to rewrite $P(\overline{\mathbf{Y}}_{1 \setminus ch_l(u)} | Q_u = x_i, \overline{\mathbf{Y}}_{ch_l(u)}) = P(\overline{\mathbf{Y}}_{1 \setminus ch_l(u)} | Q_u = x_i)$ and simplify the expression in (32) accordingly. However, in a bottom-up model, such independence assumption does not hold due to the coupling between the child subtrees of u, that are enclosed in $\overline{\mathbf{Y}}_{1 \setminus ch_l(u)}$, and the l-th child subtree in the conditioning part (see the graphical interpretation in Fig. 5). Hence, we seek for an alternative factorization: first, we rewrite

$$\begin{aligned}&P(\overline{\mathbf{Y}}_{1 \setminus ch_l(u)} | Q_u = x_i) \overset{\text{tree decomp}}{=} \\ &= P(y_u, \overline{\mathbf{Y}}_{1 \setminus u}, \overline{\mathbf{Y}}_{ch_1(u)}, \ldots, \overline{\mathbf{Y}}_{ch_{l-1}(u)}, \overline{\mathbf{Y}}_{ch_{l+1}(u)}, \ldots, \overline{\mathbf{Y}}_{ch_L(u)} | Q_u = x_i) \overset{\text{cond. prob.}}{=} \\ &= P(y_u | \overline{\mathbf{Y}}_{1 \setminus u}, \overline{\mathbf{Y}}_{ch_1(u)}, \ldots, \overline{\mathbf{Y}}_{ch_{l-1}(u)}, \overline{\mathbf{Y}}_{ch_{l+1}(u)}, \ldots, \overline{\mathbf{Y}}_{ch_L(u)}, Q_u = x_i) \\ &\quad P(\overline{\mathbf{Y}}_{1 \setminus u}, \overline{\mathbf{Y}}_{ch_1(u)}, \ldots, \overline{\mathbf{Y}}_{ch_{l-1}(u)}, \overline{\mathbf{Y}}_{ch_{l+1}(u)}, \ldots, \overline{\mathbf{Y}}_{ch_L(u)} | Q_u = x_i) \overset{\text{HTMM emission}}{=} \\ &= P(y_u | Q_u = x_i) P(\overline{\mathbf{Y}}_{1 \setminus u}, \overline{\mathbf{Y}}_{ch_1(u)}, \ldots, \overline{\mathbf{Y}}_{ch_{l-1}(u)}, \overline{\mathbf{Y}}_{ch_{l+1}(u)}, \ldots, \overline{\mathbf{Y}}_{ch_L(u)} | Q_u = x_i) \overset{\text{cond. prob.}}{=} \\ &= P(y_u | Q_u = x_i) P(\overline{\mathbf{Y}}_{1 \setminus u} | \overline{\mathbf{Y}}_{ch_1(u)}, \ldots, \overline{\mathbf{Y}}_{ch_{l-1}(u)}, \overline{\mathbf{Y}}_{ch_{l+1}(u)}, \ldots, \overline{\mathbf{Y}}_{ch_L(u)}, Q_u = x_i) \\ &\quad P(\overline{\mathbf{Y}}_{ch_1(u)}, \ldots, \overline{\mathbf{Y}}_{ch_{l-1}(u)}, \overline{\mathbf{Y}}_{ch_{l+1}(u)}, \ldots, \overline{\mathbf{Y}}_{ch_L(u)} | Q_u = x_i) \overset{\text{cond ind}}{=} \\ &= P(y_u | Q_u = x_i) P(\overline{\mathbf{Y}}_{1 \setminus u} | Q_u = x_i) \\ &\quad P(\overline{\mathbf{Y}}_{ch_1(u)}, \ldots, \overline{\mathbf{Y}}_{ch_{l-1}(u)}, \overline{\mathbf{Y}}_{ch_{l+1}(u)}, \ldots, \overline{\mathbf{Y}}_{ch_L(u)} | Q_u = x_i) \overset{\text{Bayes}}{=} \\ &= P(y_u | Q_u = x_i) P(\overline{\mathbf{Y}}_{1 \setminus u} | Q_u = x_i) P(\overline{\mathbf{Y}}_{ch_1(u)}, \ldots, \overline{\mathbf{Y}}_{ch_{l-1}(u)}, \overline{\mathbf{Y}}_{ch_{l+1}(u)}, \ldots, \overline{\mathbf{Y}}_{ch_L(u)}) \\ &\quad \frac{P(Q_u = x_i | \overline{\mathbf{Y}}_{ch_1(u)}, \ldots, \overline{\mathbf{Y}}_{ch_{l-1}(u)}, \overline{\mathbf{Y}}_{ch_{l+1}(u)}, \ldots, \overline{\mathbf{Y}}_{ch_L(u)})}{P(Q_u = x_i)} \overset{\text{cond ind}}{=} \\ &= P(y_u | Q_u = x_i) P(\overline{\mathbf{Y}}_{1 \setminus u} | Q_u = x_i) \prod_{l' \neq l} P(\overline{\mathbf{Y}}_{ch_{l'}(u)}) \frac{P(Q_u = x_i | \overline{\mathbf{Y}}_{ch_1(u)}, \ldots, \overline{\mathbf{Y}}_{ch_{l-1}(u)}, \overline{\mathbf{Y}}_{ch_{l+1}(u)}, \ldots, \overline{\mathbf{Y}}_{ch_L(u)})}{P(Q_u = x_i)}.\end{aligned} \tag{33}$$

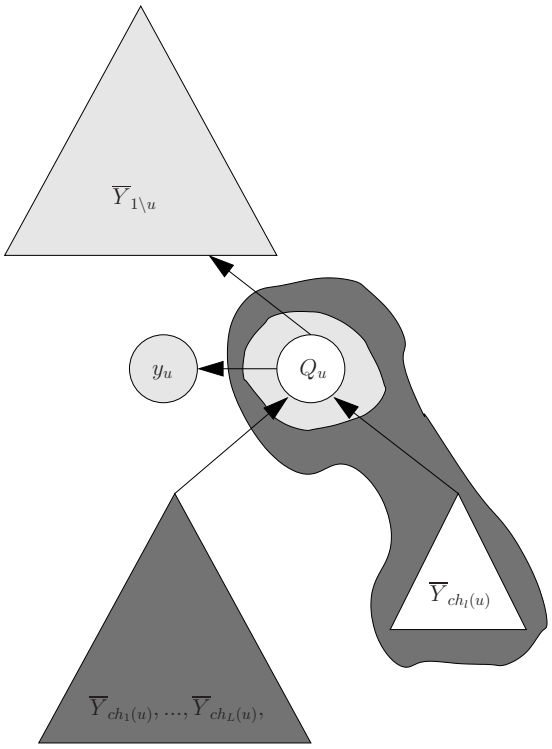


Fig. 5. Markov Blankets for the components of the $\overline{\mathbf{Y}}_{1\setminus ch_l(u)}$ tree with respect to hidden state Q_u and subtree $\overline{\mathbf{Y}}_{ch_l(u)}$: the tree of ancestors of node u, i.e. $\overline{\mathbf{Y}}_{1\setminus u}$, as well as the emission y_u are d-separated from the rest of the tree by Q_u, which is their Markov blanket (in light gray). On the other hand, the Markov blanket for the child subtrees $\overline{\mathbf{Y}}_{ch_1(u)}, \ldots, \overline{\mathbf{Y}}_{ch_{l-1}(u)}, \overline{\mathbf{Y}}_{ch_{l+1}(u)}, \ldots \overline{\mathbf{Y}}_{ch_L(u)}$ (in dark gray) has to include also the subtree $\overline{\mathbf{Y}}_{ch_l(u)}$ given that it is the parent of a common child (i.e. Q_u).

Similarly, we factorize the term in the numerator of (32) as follows,

$$
\begin{aligned}
&P(\overline{\mathbf{Y}}_{1\setminus ch_l(u)}|Q_u = x_i, \overline{\mathbf{Y}}_{ch_l(u)}) \overset{\text{Bayes+ind}}{=} \\
&\quad = P(y_u|Q_u = x_i)P(\overline{\mathbf{Y}}_{1\setminus u}|Q_u = x_i) \\
&\qquad P(\overline{\mathbf{Y}}_{ch_1(u)}, \ldots, \overline{\mathbf{Y}}_{ch_l(u)}, \overline{\mathbf{Y}}_{ch_{l+1}(u)}, \ldots, \overline{\mathbf{Y}}_{ch_L(u)}|Q_u = x_i, \overline{\mathbf{Y}}_{ch_l(u)}) \overset{\text{Bayes}}{=} \\
&\quad = P(y_u|Q_u = x_i)P(\overline{\mathbf{Y}}_{1\setminus u}|Q_u = x_i) \\
&\qquad \frac{P(\overline{\mathbf{Y}}_{ch_1(u)}, \ldots, \overline{\mathbf{Y}}_{ch_l(u)}, \ldots, \overline{\mathbf{Y}}_{ch_L(u)}|Q_u = x_i)}{P(\overline{\mathbf{Y}}_{ch_l(u)})|Q_u = x_i)} \overset{\text{Bayes+ind}}{=} \\
&\quad = P(y_u|Q_u = x_i)P(\overline{\mathbf{Y}}_{1\setminus u}|Q_u = x_i) \prod_{l'=1}^{L} P(\overline{\mathbf{Y}}_{ch_{l'}(u)}) \frac{P(Q_u = x_i|\overline{\mathbf{Y}}_{ch_1(u)}, \ldots, \overline{\mathbf{Y}}_{ch_l(u)}, \ldots, \overline{\mathbf{Y}}_{ch_L(u)})P(Q_u = x_i)}{P(Q_u = x_i)P(\overline{\mathbf{Y}}_{ch_l(u)}|Q_u = x_i)}
\end{aligned}
\tag{34}
$$

Inserting the results of (33) and (34) back into (32), yields

$$
\begin{aligned}
P(Q_u = x_i, \overline{\mathbf{Y}}_1 = \overline{\mathbf{y}}_1) &= P(Q_u = x_i, \overline{\mathbf{Y}}_{1\setminus ch_l(u)}) \frac{P(\overline{\mathbf{Y}}_{ch_l(u)}) P(Q_u = x_i | \overline{\mathbf{Y}}_{ch_l(u)})}{P(Q_u = x_i)} \\
&\quad \times \frac{P(y_u | Q_u = x_i) P(\overline{\mathbf{Y}}_{1\setminus u} | Q_u = x_i) \prod_{l'=1}^{L} P(\overline{\mathbf{Y}}_{ch_{l'}(u)}) P(Q_u = x_i)}{P(\overline{\mathbf{Y}}_{ch_l(u)} | Q_u = x_i) P(y_u | Q_u = x_i) P(\overline{\mathbf{Y}}_{1\setminus u} | Q_u = x_i) \prod_{l' \neq l} P(\overline{\mathbf{Y}}_{ch_{l'}(u)})} \\
&\quad \times \frac{P(Q_u = x_i | \overline{\mathbf{Y}}_{ch_1(u)}, \ldots, \overline{\mathbf{Y}}_{ch_l(u)}, \ldots, \overline{\mathbf{Y}}_{ch_L(u)})}{P(Q_u = x_i | \overline{\mathbf{Y}}_{ch_1(u)}, \ldots, \overline{\mathbf{Y}}_{ch_{l-1}(u)}, \overline{\mathbf{Y}}_{ch_{l+1}(u)}, \ldots, \overline{\mathbf{Y}}_{ch_L(u)})} \overset{\text{simplif}}{=} \\
&= P(Q_u = x_i, \overline{\mathbf{Y}}_{1\setminus ch_l(u)}) \frac{P(\overline{\mathbf{Y}}_{ch_l(u)}) P(Q_u = x_i | \overline{\mathbf{Y}}_{ch_l(u)})}{P(Q_u = x_i)} \\
&\quad \times \frac{P(\overline{\mathbf{Y}}_{ch_l(u)}) P(Q_u = x_i | \overline{\mathbf{Y}}_{ch_1(u)}, \ldots, \overline{\mathbf{Y}}_{ch_l(u)}, \ldots, \overline{\mathbf{Y}}_{ch_L(u)}) P(Q_u = x_i)}{P(\overline{\mathbf{Y}}_{ch_l(u)} | Q_u = x_i) P(Q_u = x_i | \overline{\mathbf{Y}}_{ch_1(u)}, \ldots, \overline{\mathbf{Y}}_{ch_{l-1}(u)}, \overline{\mathbf{Y}}_{ch_{l+1}(u)}, \ldots, \overline{\mathbf{Y}}_{ch_L(u)})} \overset{\text{Bayes}}{=} \\
&= P(Q_u = x_i, \overline{\mathbf{Y}}_{1\setminus ch_l(u)}) P(\overline{\mathbf{Y}}_{ch_l(u)} | Q_u = x_i) \\
&\quad \times \frac{P(\overline{\mathbf{Y}}_{ch_l(u)}) P(Q_u = x_i | \overline{\mathbf{Y}}_{ch_1(u)}, \ldots, \overline{\mathbf{Y}}_{ch_l(u)}, \ldots, \overline{\mathbf{Y}}_{ch_L(u)}) P(Q_u = x_i)}{P(\overline{\mathbf{Y}}_{ch_l(u)} | Q_u = x_i) P(Q_u = x_i | \overline{\mathbf{Y}}_{ch_1(u)}, \ldots, \overline{\mathbf{Y}}_{ch_{l-1}(u)}, \overline{\mathbf{Y}}_{ch_{l+1}(u)}, \ldots, \overline{\mathbf{Y}}_{ch_L(u)})} \overset{\text{simplif}}{=} \\
&= P(Q_u = x_i, \overline{\mathbf{Y}}_{1\setminus ch_l(u)}) \frac{P(\overline{\mathbf{Y}}_{ch_l(u)}) P(Q_u = x_i | \overline{\mathbf{Y}}_{ch_1(u)}, \ldots, \overline{\mathbf{Y}}_{ch_l(u)}, \ldots, \overline{\mathbf{Y}}_{ch_L(u)}) P(Q_u = x_i)}{P(Q_u = x_i | \overline{\mathbf{Y}}_{ch_1(u)}, \ldots, \overline{\mathbf{Y}}_{ch_{l-1}(u)}, \overline{\mathbf{Y}}_{ch_{l+1}(u)}, \ldots, \overline{\mathbf{Y}}_{ch_L(u)})},
\end{aligned}
\tag{35}
$$

which is the final expression for the denominator of (31).

On the other hand, the first term in the numerator of (31) factorizes as follows

$$
\begin{aligned}
P(Q_u = x_i, Q_{ch_l(u)} = x_j, S_u = l, \overline{\mathbf{Y}}_1) &= P(Q_u = x_i, Q_{ch_l(u)} = x_j, S_u = l, \overline{\mathbf{Y}}_{ch_l(u)}, \overline{\mathbf{Y}}_{1\setminus ch_l(u)}) \overset{\text{cond. prob.}}{=} \\
&= P(\overline{\mathbf{Y}}_{ch_l(u)} | Q_u = x_i, Q_{ch_l(u)} = x_j, S_u = l, \overline{\mathbf{Y}}_{1\setminus ch_l(u)}) P(Q_u = x_i, Q_{ch_l(u)} = x_j, S_u = l, \overline{\mathbf{Y}}_{1\setminus ch_l(u)}) \overset{\text{cond.ind.}}{=} \\
&= P(\overline{\mathbf{Y}}_{ch_l(u)} | Q_{ch_l(u)} = x_j) P(Q_u = x_i, Q_{ch_l(u)} = x_j, S_u = l, \overline{\mathbf{Y}}_{1\setminus ch_l(u)}) \overset{\text{Bayes}}{=} \\
&= P(\overline{\mathbf{Y}}_{ch_l(u)} | Q_{ch_l(u)} = x_j) P(Q_u = x_i, \overline{\mathbf{Y}}_{1\setminus ch_l(u)}) P(Q_{ch_l(u)} = x_j, S_u = l | Q_u = x_i, \overline{\mathbf{Y}}_{1\setminus ch_l(u)}) \overset{\text{marginal}}{=} \\
&= P(\overline{\mathbf{Y}}_{ch_l(u)} | Q_{ch_l(u)} = x_j) P(Q_u = x_i, \overline{\mathbf{Y}}_{1\setminus ch_l(u)}) \\
&\quad \times \sum_{\overline{J}_{1\setminus l}=1}^{C} P(Q_{ch_1(u)} = x_{j_1}, \ldots, Q_{ch_L(u)} = x_{j_L}, Q_{ch_l(u)} = x_j, S_u = l | Q_u = x_i, \overline{\mathbf{Y}}_{1\setminus ch_l(u)}) = \\
&= P(\overline{\mathbf{Y}}_{ch_l(u)} | Q_{ch_l(u)} = x_j) P(Q_u = x_i, \overline{\mathbf{Y}}_{1\setminus ch_l(u)}) \\
&\quad \times \sum_{\overline{J}_{1\setminus l}=1}^{C} P(\overline{\mathbf{CH}}_{1\setminus l}(u) = \overline{J}_{1\setminus l}, Q_{ch_l(u)} = x_j, S_u = l | Q_u = x_i, \overline{\mathbf{Y}}_{1\setminus ch_l(u)})
\end{aligned}
\tag{36}
$$

where the sum marginalizes over the hidden states $\overline{J}_{1\setminus l} = \{j_1, \ldots, j_{l-1}, , j_{l+1}, \ldots, j_L\}$ of the children $\overline{\mathbf{CH}}_{1\setminus l}(u) = \{Q_{ch_1(u)}, \ldots, Q_{ch_{l-1}(u)}, \ldots Q_{ch_{l+1}(u)}, \ldots, Q_{ch_L(u)}\}$ of node u except the l-th child node.

Applying Bayes formula to the argument of the summation yields

$$P(\overline{\mathbf{CH}}_{1\setminus l}(u) = \overline{J}_{1\setminus l}, Q_{ch_l(u)} = x_j, S_u = l | Q_u = x_i, \overline{\mathbf{Y}}_{1\setminus ch_l(u)}) =$$
$$= \frac{\overbrace{P(\overline{\mathbf{Y}}_{1\setminus ch_l(u)} | \overline{\mathbf{CH}}_{1\setminus l}(u) = \overline{J}_{1\setminus l}, Q_{ch_l(u)} = x_j, Q_u = x_i, S_u = l)}^{A1} \overbrace{P(\overline{\mathbf{CH}}_{1\setminus l}(u) = \overline{J}_{1\setminus l}, Q_{ch_l(u)} = x_j, Q_u = x_i, S_u = l)}^{A2}}{P(Q_u = x_i) \underbrace{P(\overline{\mathbf{Y}}_{1\setminus ch_l(u)} | Q_u = x_i)}_{A3}}. \tag{37}$$

The term A2 in (37) can be expanded as follows

$$\begin{aligned}
&P(\overline{\mathbf{CH}}_{1\setminus l}(u) = \overline{J}_{1\setminus l}, Q_{ch_l(u)} = x_j, Q_u = x_i, S_u = l) \overset{\text{cond. prob.}}{=} \\
&= P(Q_u = x_i, S_u = l | \overline{\mathbf{CH}}_{1\setminus l}(u) = \overline{J}_{1\setminus l}, Q_{ch_l(u)} = x_j) P(\overline{\mathbf{CH}}_{1\setminus l}(u) = \overline{J}_{1\setminus l}, Q_{ch_l(u)} = x_j) \overset{\text{switch par}}{=} \\
&= p(S_u = l) p^l(Q_u = x_i | Q_{ch_l(u)} = x_j) P(\overline{\mathbf{CH}}_{1\setminus l}(u) = \overline{J}_{1\setminus l}, Q_{ch_l(u)} = x_j) \overset{\text{cond ind}}{=} \\
&= p(S_u = l) p^l(Q_u = x_i | Q_{ch_l(u)} = x_j) \prod_{l'=1}^{L} P(Q_{ch_{l'}(u)} = x_{j_{l'}}),
\end{aligned} \tag{38}$$

where the latter equality stems from the fact that the hidden states of the children are independent when the parent state is not observed. The term A1, on the other hand, can be factorizes as follows

$$\begin{aligned}
&P(\overline{\mathbf{Y}}_{1\setminus ch_l(u)} | \overline{\mathbf{CH}}_{1\setminus l}(u) = \overline{J}_{1\setminus l}, Q_{ch_l(u)} = x_j, Q_u = x_i, S_u = l) \overset{\text{tree decomp}}{=} \\
&= P(y_u, \overline{\mathbf{Y}}_{1\setminus u}, \overline{\mathbf{Y}}_{ch_1(u)}, \ldots, \overline{\mathbf{Y}}_{ch_{l-1}(u)}, \overline{\mathbf{Y}}_{ch_{l+1}(u)}, \ldots, \overline{\mathbf{Y}}_{ch_L(u)} | \overline{\mathbf{CH}}_{1\setminus l}(u) = \overline{J}_{1\setminus l}, Q_{ch_l(u)} = x_j, Q_u = x_i, S_u = l) \overset{\text{c.pr.}}{=} \\
&= P(y_u | \overline{\mathbf{Y}}_{1\setminus u}, \overline{\mathbf{Y}}_{ch_1(u)}, \ldots, \overline{\mathbf{Y}}_{ch_L(u)}, \overline{\mathbf{CH}}_{1\setminus l}(u) = \overline{J}_{1\setminus l}, Q_{ch_l(u)} = x_j, Q_u = x_i, S_u = l) \\
&\quad \times P(\overline{\mathbf{Y}}_{1\setminus u}, \overline{\mathbf{Y}}_{ch_1(u)}, \ldots, \overline{\mathbf{Y}}_{ch_L(u)}, \overline{\mathbf{CH}}_{1\setminus l}(u) = \overline{J}_{1\setminus l}, Q_{ch_l(u)} = x_j, Q_u = x_i, S_u = l) \overset{\text{HTMM emission}}{=} \\
&= P(y_u | Q_u = x_i) \\
&\quad \times P(\overline{\mathbf{Y}}_{1\setminus u}, \overline{\mathbf{Y}}_{ch_1(u)}, \ldots, \overline{\mathbf{Y}}_{ch_L(u)}, \overline{\mathbf{CH}}_{1\setminus l}(u) = \overline{J}_{1\setminus l}, Q_{ch_l(u)} = x_j, Q_u = x_i, S_u = l) \overset{\text{cond. prob.}}{=} \\
&= P(y_u | Q_u = x_i) P(\overline{\mathbf{Y}}_{1\setminus u} | \overline{\mathbf{Y}}_{ch_1(u)}, \ldots, \overline{\mathbf{Y}}_{ch_L(u)}, \overline{\mathbf{CH}}_{1\setminus l}(u) = \overline{J}_{1\setminus l}, Q_{ch_l(u)} = x_j, Q_u = x_i, S_u = l) \\
&\quad \times P(\overline{\mathbf{Y}}_{ch_1(u)}, \ldots, \overline{\mathbf{Y}}_{ch_L(u)}, \overline{\mathbf{CH}}_{1\setminus l}(u) = \overline{J}_{1\setminus l}, Q_{ch_l(u)} = x_j, Q_u = x_i, S_u = l) \overset{\text{cond ind}}{=} \\
&= P(y_u | Q_u = x_i) P(\overline{\mathbf{Y}}_{1\setminus u} | Q_u = x_i) \\
&\quad \times P(\overline{\mathbf{Y}}_{ch_1(u)}, \ldots, \overline{\mathbf{Y}}_{ch_L(u)}, \overline{\mathbf{CH}}_{1\setminus l}(u) = \overline{J}_{1\setminus l}, Q_{ch_l(u)} = x_j, Q_u = x_i, S_u = l) \overset{\text{cond. prob.}}{=} \\
&= P(y_u | Q_u = x_i) P(\overline{\mathbf{Y}}_{1\setminus u} | Q_u = x_i) P(\overline{\mathbf{Y}}_{ch_1(u)} | \overline{\mathbf{Y}}_{ch_2(u)}, \ldots, \overline{\mathbf{Y}}_{ch_L(u)}, \overline{\mathbf{CH}}_{1\setminus l}(u) = \overline{J}_{1\setminus l}, Q_{ch_l(u)} = x_j, Q_u = x_i, S_u = l) \\
&\quad \times P(\overline{\mathbf{Y}}_{ch_2(u)}, \ldots, \overline{\mathbf{Y}}_{ch_L(u)}, \overline{\mathbf{CH}}_{1\setminus l}(u) = \overline{J}_{1\setminus l}, Q_{ch_l(u)} = x_j, Q_u = x_i, S_u = l) \overset{\text{cond ind}}{=} \\
&= P(y_u | Q_u = x_i) P(\overline{\mathbf{Y}}_{1\setminus u} | Q_u = x_i) P(\overline{\mathbf{Y}}_{ch_1(u)} | Q_{ch_1(u)} = x_j) \\
&\quad \times P(\overline{\mathbf{Y}}_{ch_2(u)}, \ldots, \overline{\mathbf{Y}}_{ch_L(u)}, \overline{\mathbf{CH}}_{1\setminus l}(u) = \overline{J}_{1\setminus l}, Q_{ch_l(u)} = x_j, Q_u = x_i, S_u = l) = \\
&= \cdots = P(y_u | Q_u = x_i) P(\overline{\mathbf{Y}}_{1\setminus u} | Q_u = x_i) \prod_{l' \neq l} P(\overline{\mathbf{Y}}_{ch_{l'}(u)} | Q_{ch_{l'}(u)} = x_{j_{l'}})
\end{aligned} \tag{39}$$

where the product runs over all the children l' of node u except l, given that $\overline{\mathbf{Y}}_{1\setminus ch_l(u)}$ includes all the tree except the subtree rooted in $ch_l(u)$. Finally, for the term A3 we can use the expression in (33). Introducing the expressions

for terms A1, A2 and A3 back into (37) yields

$$
\begin{aligned}
&P(\overline{\mathbf{CH}}_{1\setminus l}(u) = \overline{J}_{1\setminus l}, Q_{ch_l(u)} = x_j, S_u = l | Q_u = x_i, \overline{\mathbf{Y}}_{1\setminus ch_l(u)}) = \\
&= \overbrace{p(S_u = l) P(^l Q_u = x_i | Q_{ch_l(u)} = x_j) \prod_{l'=1}^{L} P(Q_{ch_{l'}(u)} = x_{j_{l'}})}^{A2} \\
&\times \underbrace{\frac{\overbrace{P(y_u | Q_u = x_i) P(\overline{\mathbf{Y}}_{1\setminus u} | Q_u = x_i) \prod_{l' \neq l} P(\overline{\mathbf{Y}}_{ch_{l'}(u)} | Q_{ch_{l'}(u)} = x_{j_{l'}})}^{A1}}{\frac{P(y_u | Q_u = x_i) P(\overline{\mathbf{Y}}_{1\setminus u} | Q_u = x_i)}{P(Q_u = x_i)} \prod_{l' \neq l} P(\overline{\mathbf{Y}}_{ch_{l'}(u)}) P(Q_u = x_i | \overline{\mathbf{Y}}_{ch_1(u)}, \ldots, \overline{\mathbf{Y}}_{ch_{l-1}(u)}, \overline{\mathbf{Y}}_{ch_{l+1}(u)}, \ldots, \overline{\mathbf{Y}}_{ch_L(u)})}}_{A3}
\end{aligned}
\tag{40}
$$

By applying simplifications and Bayesian transformation we obtain the final expression

$$
\begin{aligned}
&P(\overline{\mathbf{CH}}_{1\setminus l}(u) = \overline{J}_{1\setminus l}, Q_{ch_l(u)} = x_j, S_u = l | Q_u = x_i, \overline{\mathbf{Y}}_{1\setminus ch_l(u)}) = \\
&= \frac{p(S_u = l) p^l(Q_u = x_i | Q_{ch_l(u)} = x_j) P(Q_{ch_l(u)} = x_{j_l}) P(Q_u = x_i)}{P(Q_u = x_i | \overline{\mathbf{Y}}_{ch_1(u)}, \ldots, \overline{\mathbf{Y}}_{ch_{l-1}(u)}, \overline{\mathbf{Y}}_{ch_{l+1}(u)}, \ldots, \overline{\mathbf{Y}}_{ch_L(u)})} \prod_{l' \neq l} \frac{P(Q_{ch_{l'}(u)} = x_{j_{l'}}) P(\overline{\mathbf{Y}}_{ch_{l'}(u)} | Q_{ch_{l'}(u)} = x_{j_{l'}})}{P(\overline{\mathbf{Y}}_{ch_{l'}(u)})} \overset{\text{Bayes}}{=} \\
&= \frac{p(S_u = l) p^l(Q_u = x_i | Q_{ch_l(u)} = x_j) P(Q_{ch_l(u)} = x_{j_l}) P(Q_u = x_i)}{P(Q_u = x_i | \overline{\mathbf{Y}}_{ch_1(u)}, \ldots, \overline{\mathbf{Y}}_{ch_{l-1}(u)}, \overline{\mathbf{Y}}_{ch_{l+1}(u)}, \ldots, \overline{\mathbf{Y}}_{ch_L(u)})} \prod_{l' \neq l} P(Q_{ch_{l'}(u)} = x_{j_{l'}} | \overline{\mathbf{Y}}_{ch_{l'}(u)}) \overset{\text{Def } \beta_{ch_{l'}(u)}(j_{l'})}{=} \\
&= \frac{p(S_u = l) p^l(Q_u = x_i | Q_{ch_l(u)} = x_j) P(Q_{ch_l(u)} = x_{j_l}) P(Q_u = x_i)}{P(Q_u = x_i | \overline{\mathbf{Y}}_{ch_1(u)}, \ldots, \overline{\mathbf{Y}}_{ch_{l-1}(u)}, \overline{\mathbf{Y}}_{ch_{l+1}(u)}, \ldots, \overline{\mathbf{Y}}_{ch_L(u)})} \prod_{l' \neq l} \beta_{ch_{l'}(u)}(j_{l'})
\end{aligned}
\tag{41}
$$

By plugging the results of (41) into the pairwise transition probability in (36), we obtain

$$
\begin{aligned}
P(Q_u = x_i, Q_{ch_l(u)} = x_j, S_u = l, \overline{\mathbf{Y}}_1) = & \\
&= p(S_u = l) p^l(Q_u = x_i | Q_{ch_l(u)} = x_j) P(Q_{ch_l(u)} = x_{j_l}) P(Q_u = x_i) \\
&\times \frac{P(\overline{\mathbf{Y}}_{ch_l(u)} | Q_{ch_l(u)} = x_j) P(Q_u = x_i, \overline{\mathbf{Y}}_{1\setminus ch_l(u)})}{P(Q_u = x_i | \overline{\mathbf{Y}}_{ch_1(u)}, \ldots, \overline{\mathbf{Y}}_{ch_{l-1}(u)}, \overline{\mathbf{Y}}_{ch_{l+1}(u)}, \ldots, \overline{\mathbf{Y}}_{ch_L(u)})} \\
&\times \sum_{\overline{J}_{1\setminus l}=1}^{C} \prod_{l' \neq l} \beta_{ch_{l'}(u)}(j_{l'}).
\end{aligned}
\tag{42}
$$

Notice that the term $\sum_{\overline{J}_{1\setminus l}=1}^{C} \prod_{l' \neq l} \beta_{ch_{l'}(u)}(j_{l'}) = 1$ given that it marginalizes out all the hidden state independently for each child subtree. Hence, it can be omitted in the final formulation. Introducing the results of (35) and (42)

into the smoothed probability in (31) yields

$$\begin{aligned}\epsilon^l_{u,ch_l(u)}(i,j) &= P(Q_u = x_i, Q_{ch_l(u)} = x_j, S_u = l|\overline{\mathbf{Y}}_1 = \overline{\mathbf{y}}_1)\\ &= P(Q_u = x_i|\overline{\mathbf{Y}}_1 = \overline{\mathbf{y}}_1)p(S_u = l)p^l(Q_u = x_i|Q_{ch_l(u)} = x_j)P(Q_{ch_l(u)} = x_{j_l})P(Q_u = x_i)\\ &\quad \times \frac{P(\overline{\mathbf{Y}}_{ch_l(u)}|Q_{ch_l(u)} = x_j)P(Q_u = x_i, \overline{\mathbf{Y}}_{1\setminus ch_l(u)})}{P(Q_u = x_i, \overline{\mathbf{Y}}_{1\setminus ch_l(u)})P(\overline{\mathbf{Y}}_{ch_l(u)})P(Q_u = x_i|\overline{\mathbf{Y}}_{ch_1(u)}, \ldots, \overline{\mathbf{Y}}_{ch_l(u)}, \ldots, \overline{\mathbf{Y}}_{ch_L(u)})}\\ &\quad \times \frac{P(Q_u = x_i|\overline{\mathbf{Y}}_{ch_1(u)}, \ldots, \overline{\mathbf{Y}}_{ch_{l-1}(u)}, \overline{\mathbf{Y}}_{ch_{l+1}(u)}, \ldots, \overline{\mathbf{Y}}_{ch_L(u)})}{P(Q_u = x_i|\overline{\mathbf{Y}}_{ch_1(u)}, \ldots, \overline{\mathbf{Y}}_{ch_{l-1}(u)}, \overline{\mathbf{Y}}_{ch_{l+1}(u)}, \ldots, \overline{\mathbf{Y}}_{ch_L(u)})P(Q_u = x_i)}.\end{aligned} \tag{43}$$

By simplifying and restructuring the order of some terms, this rewrites as

$$\begin{aligned}&\epsilon^l_{u,ch_l(u)}(i,j) =\\ &\quad = \frac{P(Q_u = x_i|\overline{\mathbf{Y}}_1 = \overline{\mathbf{y}}_1)P(\overline{\mathbf{Y}}_{ch_l(u)}|Q_{ch_l(u)} = x_j)P(Q_{ch_l(u)} = x_j)P(S_u = l)p^l(Q_u = x_i|Q_{ch_l(u)} = x_j)}{P(\overline{\mathbf{Y}}_{ch_l(u)})P(Q_u = x_i|\overline{\mathbf{Y}}_{ch_1(u)}, \ldots, \overline{\mathbf{Y}}_{ch_l(u)}, \ldots, \overline{\mathbf{Y}}_{ch_L(u)})}.\end{aligned} \tag{44}$$

Given the definition of $\beta_{ch_l(u)}(j)$, we can rewrite it as

$$\beta_{ch_l(u)}(j) = P(Q_{ch_l(u)} = x_j|\overline{\mathbf{Y}}_{ch_l(u)}) = \frac{P(\overline{\mathbf{Y}}_{ch_l(u)}|Q_{ch_l(u)} = x_j)P(Q_{ch_l(u)} = x_j)}{P(\overline{\mathbf{Y}}_{ch_l(u)})}. \tag{45}$$

and by introducing this result in (44) it yields to

$$\epsilon^l_{u,ch_l(u)}(i,j) = \frac{P(Q_u = x_i|\overline{\mathbf{Y}}_1 = \overline{\mathbf{y}}_1)\beta_{ch_l(u)}(j)P(S_u = l)p^l(Q_u = x_i|Q_{ch_l(u)} = x_j)}{P(Q_u = x_i|\overline{\mathbf{Y}}_{ch_1(u)}, \ldots, \overline{\mathbf{Y}}_{ch_l(u)}, \ldots, \overline{\mathbf{Y}}_{ch_L(u)})}, \tag{46}$$

where we still need to determine an expression for the term in the denominator. Again, by resorting to the switching parents approximation, we can rewrite such term as follows

$$\begin{aligned}P(Q_u = x_i|\overline{\mathbf{Y}}_{ch_1(u)}, \ldots, \overline{\mathbf{Y}}_{ch_l(u)}, \ldots, \overline{\mathbf{Y}}_{ch_L(u)}) &\overset{\text{marginal } S_u^{l'}}{=}\\ = \sum_{l'=1}^{L} P(Q_u = x_i S_u = l'|\overline{\mathbf{Y}}_{ch_1(u)}, \ldots, \overline{\mathbf{Y}}_{ch_l(u)}, , \ldots, \overline{\mathbf{Y}}_{ch_L(u)},) &\overset{\text{switch par}}{=}\\ = \sum_{l'=1}^{L} P(S_u = l')P(Q_u = x_i|\overline{\mathbf{Y}}_{ch_{l'}(u)}) &\overset{\text{Bayes}}{=}\\ = \sum_{l'=1}^{L} \frac{P(S_u = l')P(\overline{\mathbf{Y}}_{ch_{l'}(u)}|Q_u = x_i)P(Q_u = x_i)}{P(\overline{\mathbf{Y}}_{ch_{l'}(u)})} &\overset{\text{def } \beta_{u,ch_{l''}(u)}(i)}{=}\\ = P(Q_u = x_i)\sum_{l'=1}^{L} P(S_u = l')\beta_{u,ch_{l'}(u)}(i).\end{aligned} \tag{47}$$

By inserting the results of (47) back into (46), this finally yields to

$$\begin{aligned}\epsilon^l_{u,ch_l(u)}(i,j) &= \frac{P(Q_u = x_i|\overline{\mathbf{Y}}_1 = \overline{\mathbf{y}}_1)\beta_{ch_l(u)}(j)P(S_u = l)p^l(Q_u = x_i|Q_{ch_l(u)} = x_j)}{P(Q_u = x_i)\sum_{l''=1}^{L} P(S_u = l'')\beta_{u,ch_{l''}(u)}(i)} &\overset{\text{def } \epsilon_u(i)}{=}\\ &= \frac{\epsilon_u(i)\beta_{ch_l(u)}(j)P(S_u = l)p^l(Q_u = x_i|Q_{ch_l(u)} = x_j)}{P(Q_u = x_i)\sum_{l''=1}^{L} P(S_u = l'')\beta_{u,ch_{l''}(u)}(i)}.\end{aligned} \tag{48}$$

From the definition of the pairwise transition probability in (48), we can easily obtain an expression for $\epsilon_{ch_l(u)}(j)$ by marginalizing the hidden states of the parent node, that is

$$\begin{aligned}\epsilon_{ch_l(u)}(j) &= \sum_{i=1}^{C} \epsilon^l_{u,ch_l(u)}(i,j) = \\ &= \beta_{ch_l(u)}(j) \sum_{i=1}^{C} \frac{\epsilon_u(i) P(S_u = l) p^l(Q_u = x_i | Q_{ch_l(u)} = x_j)}{P(Q_u = x_i) \sum_{l''=1}^{L} P(S_u = l'') \beta_{u,ch_{l''}(u)}(i)}.\end{aligned} \tag{49}$$

The downward recursion is based on the $\alpha_u(i)$ term in the factorization $\epsilon_u(i) = \alpha_u(i)\beta_u(i)$. The basis of the recursion is at the root node: it is straightforward that $\alpha_1(i) = 1$ for all hidden state assignments x_i. The recursive step is computed for each node $u' = ch_l(u)$, that is the l-th child of parent u, as

$$\alpha_{ch_l u}(j) = \frac{\epsilon_{ch_l u}(j)}{\beta_{ch_l u}(j)} = \sum_{i=1}^{C} \frac{\epsilon_u(i) P(S_u = l) p^l(Q_u = x_i | Q_{ch_l(u)} = x_j)}{P(Q_u = x_i) \sum_{l''=1}^{L} P(S_u = l'') \beta_{u,ch_{l''}(u)}(i)}. \tag{50}$$

Since $\epsilon_u(i) = \alpha_u(i)\beta_u(i)$, this yields to

$$\alpha_{ch_l u}(j) = \sum_{i=1}^{C} \frac{\alpha_u(i)\beta_u(i) P(S_u = l) p^l(Q_u = x_i | Q_{ch_l(u)} = x_j)}{P(Q_u = x_i) \sum_{l''=1}^{L} P(S_u = l'') \beta_{u,ch_{l''}(u)}(i)}, \tag{51}$$

that is the final update equation for the downward recursion. Notice that the posterior distribution of the switching parents indicator variables in (15) can be straightforwardly obtained by marginalization of the pairwise transition probability in (48), that is

$$P(S_u = l | \overline{\mathbf{Y}}_1 = \overline{\mathbf{y}}_1) = \sum_{i,j=1}^{C} = P(Q_u = x_i, Q_{ch_l(u)} = x_j, S_u = l | \overline{\mathbf{Y}}_1 = \overline{\mathbf{y}}_1) = \sum_{i,j=1}^{C} \epsilon^l_{u,ch_l(u)}(i,j). \tag{52}$$

3.2 M-Step.

The posterior probabilities obtained at the previous step are sufficient statistics to compute the expectation of the complete log-likelihood

$$\begin{aligned}E[\log \mathcal{L}_c] = &\sum_{n=1}^{N} \sum_{u' \in \text{leaf}(\overline{\mathbf{y}}^n)} \sum_{i=1}^{C} P(Q_{u'} = x_i | \overline{\mathbf{y}}^n) \log P^{pos(u)}(Q_{u'} = x_i) \\ &+ \sum_{n=1}^{N} \sum_{u \in \mathbf{U}_{\overline{\mathbf{y}}^n}} \sum_{i=1}^{C} P(Q_u = x_i | \overline{\mathbf{y}}^n) \log P(y_u | Q_u = x_i) \\ &+ \sum_{n=1}^{N} \sum_{u \in \mathbf{U}_{\overline{\mathbf{y}}^n} \setminus \text{leaf}(\overline{\mathbf{y}}^n)} \sum_{l=1}^{L} P(S_u = l | \overline{\mathbf{y}}^n) \log P(S_u = l) \\ &+ \sum_{n=1}^{N} \sum_{u \in \mathbf{U}_{\overline{\mathbf{y}}^n} \setminus \text{leaf}(\overline{\mathbf{y}}^n)} \sum_{i,j=1}^{C} \sum_{l=1}^{L} P(Q_u = x_i, Q_{ch_l(u)} = x_j, S_u = l | \overline{\mathbf{y}}^n) \log p^l(Q_u = x_i | Q_{ch_l(u)} = x_j)\end{aligned} \tag{53}$$

whose maximization with respect to the model parameters θ leads to the update equations for the M-Step. Lets consider the following parametrization for the bottom-up HTMM model

– the positional *initial state probability* matrix $\pi = \{\pi_i^l\}_{i=1,\dots,C}^{l=1,\dots,L}$ such that

$$\pi_i^l = P^{pos(u)}(Q_u = x_i), \text{ subject to } \pi_i^l \geq 0 \text{ and} \sum_{i=1}^{C} \pi_i^l = 1 \text{ for } 1 \leq l \leq L$$

– the *mixed memory probability* matrix $\varphi = \{\varphi_l\}_{l=1,\dots,L}$ such that

$$\varphi_l = P(S_u = l), \text{ subject to } \varphi_l \geq 0 \text{ and} \sum_{l=1}^{L} \varphi_l = 1$$

– the position-dependent *state transition probability* matrices $A = \{A_1, \dots, A_l, \dots, A_L\}$ where $A_l = \{a_{ij}^l\}_{i,j=1,\dots,C}^{l=1,\dots,L}$ such that

$$a_{ij}^l = P^l(Q_u = x_i | Q_{ch_l(u)} = x_j), \text{ subject to } a_{ij}^l \geq 0 \text{ and} \sum_{i=1}^{C} a_{ij}^l = 1, \text{ for } 1 \leq l \leq L$$

– the *emission probability* matrix $B = \{b_i(m)\}_{i=1,\dots,C}$ such that

$$b_i(m) = P(y_u = o_m | Q_u = x_i), \text{ subject to } b_i(m) \geq 0 \text{ and} \sum_{m=1}^{M} b_i(m) = 1$$

where $o_1, \dots, o_M$ is, for instance, a finite output alphabet.

Following such parametrization, lets denote the model parameters as $\theta = (\pi, \varphi, A, B)$ and their current estimate as $\theta^{(k)}$. The sum-to-one constraints for the model parameters can be incorporated in the maximization of the likelihood expectation in (53) using Lagrange multipliers, yielding the following EM auxiliary function

$$\begin{aligned} \mathcal{Q}(\theta|\theta^{(k)}) = E[\log \mathcal{L}_c|\theta^{(k)}] + \sum_l \omega_l \left(\sum_{i=1}^{C} \pi_i^l - 1 \right) + \eta \left(\sum_{l=1}^{L} \varphi_l - 1 \right) + \sum_{j,l} \mu_i^l \left(\sum_{i=1}^{C} a_{ij}^l - 1 \right) \\ + \sum_i \nu_i \left(\sum_{m=1}^{M} b_i(m) - 1 \right) \end{aligned} \tag{54}$$

where $E[\log \mathcal{L}_c|\theta^{(k)}]$ is the expectation of the complete log-likelihood in (53) computed using the current estimate $\theta^{(k)}$ of the model parameters. Given such parametrization, we can rewrite $\mathcal{Q}(\theta|\theta^{(k)})$ as a sum of functions that can be optimized separately, that is

$$\mathcal{Q}(\theta|\theta^{(k)}) = \mathcal{Q}_\pi + \mathcal{Q}_\varphi + \mathcal{Q}_A + \mathcal{Q}_B \tag{55}$$

where each $\mathcal{Q}_x$ term incorporates both the likelihood term and the Lagrange multiplier that are dependent from $x \in \theta$. The iterative update rules are obtained as

$$\theta^{(k+1)}) = \arg\max_\theta \mathcal{Q}(\theta|\theta^{(k)}) \tag{56}$$

by taking the first derivative of (56) with respect to the model parameters θ and solving it with respect to the unknowns.

Prior Probability. Given the i-th hidden state and the l-th position in the child subtree, the prior probability update can be obtained by solving $\frac{\partial \mathcal{Q}_\pi}{\partial \pi_i^l} = 0$, that yields

$$\pi_i^{l\,(k+1)} = \frac{\sum_{n=1}^{N} \sum_{u \in \text{leaf}(\overline{\mathbf{y}}^n)} \delta(pos(u), l) P(Q_u = x_i | \overline{\mathbf{y}}^n, \theta^{(k)})}{\sum_{n=1}^{N} N L_n^l} \tag{57}$$

where NL_n^l is the total number of leaves of the n-th tree that are the l-th child of their parent. The indicator function $\delta(pos(u), l$ returns 1 when the $pos(u) = l$ and 0 otherwise. The posterior $P(Q_u = x_i|\overline{\mathbf{y}}^n, \theta^{(k)})$ corresponds to the *state occupancy probability* $\epsilon_u^n(i)$ at node u and can be straightforwardly obtained by marginalization of the state transition posterior, i.e.

$$\epsilon_u^n(i) = P(Q_u = x_i|\overline{\mathbf{y}}^n, \theta^{(k)}) = \sum_{j=1}^{C}\sum_{l=1}^{L} P(Q_{pa(u)} = x_j, Q_u = x_i, S_{pa(u)} = l|\overline{\mathbf{y}}_n, \theta^{(k)}) = \sum_{j=1}^{C}\sum_{l=1}^{L} \epsilon_{pa(u),u}^{l,n}(j,i). \tag{58}$$

The final update rule for prior probability is

$$\pi_i^{l\,(k+1)} = \frac{\sum_{n=1}^{N}\sum_{u\in \text{leaf}(\overline{\mathbf{y}}^n)}\sum_{j=1}^{C}\epsilon_{pa(u),u}^{l,n}(j,i)}{\sum_{n=1}^{N} NL_n^l}. \tag{59}$$

In the remainder of the section, we will omit the $\theta^{(k)}$ term in the distribution estimates for enhancing readability.

Mixed-Memory Probability. Differentiating with respect to φ_l yields

$$\frac{\partial \mathcal{Q}_\varphi}{\partial \varphi_l} = \sum_{n=1}^{N}\sum_{u\in \mathbf{U}_{\overline{\mathbf{y}}^n}\setminus \text{leaf}(\overline{\mathbf{y}}^n)} \frac{1}{\varphi_l} P(S_u = l|\overline{\mathbf{y}}^n) + \eta = 0. \tag{60}$$

By solving the Lagrange multipliers and inserting the posterior definition in (52), we obtain the following update rule

$$\begin{aligned}
\varphi_l^{(k+1)} &= \frac{\sum_{n=1}^{N}\sum_{u\in \mathbf{U}_{\overline{\mathbf{y}}^n}\setminus \text{leaf}(\overline{\mathbf{y}}^n)} P(S_u = l|\overline{\mathbf{y}}^n)}{\sum_{n=1}^{N}\sum_{u\in \mathbf{U}_{\overline{\mathbf{y}}^n}\setminus \text{leaf}(\overline{\mathbf{y}}^n)}\sum_{l=1}^{L} P(S_u = l|\overline{\mathbf{y}}^n)} \\
&= \frac{\sum_{n=1}^{N}\sum_{u\in \mathbf{U}_{\overline{\mathbf{y}}^n}\setminus \text{leaf}(\overline{\mathbf{y}}^n)}\sum_{i,j=1}^{C}\epsilon_{u,ch_l(u)}^{l,n}(i,j)}{\sum_{n=1}^{N}\sum_{u\in \mathbf{U}_{\overline{\mathbf{y}}^n}\setminus \text{leaf}(\overline{\mathbf{y}}^n)}\sum_{l=1}^{L}\sum_{i,j=1}^{C}\epsilon_{u,ch_l(u)}^{l,n}(i,j)} \\
&= \frac{\sum_{n=1}^{N}\sum_{u\in \mathbf{U}_{\overline{\mathbf{y}}^n}\setminus \text{leaf}(\overline{\mathbf{y}}^n)}\sum_{i,j=1}^{C}\epsilon_{u,ch_l(u)}^{l,n}(i,j)}{L\sum_{n=1}^{N} NI_n}
\end{aligned} \tag{61}$$

where NI_n is the number of internal nodes (non leaves) of the n-th tree.

Positional State-Transition Probability. Following the process described above, differentiating with respect to a_{ij}^l yields

$$a_{ij}^l = \frac{\sum_{n=1}^{N}\sum_{u\in \mathbf{U}_{\overline{\mathbf{y}}^n}\setminus \text{leaf}(\overline{\mathbf{y}}^n)}\epsilon_{u,ch_l(u)}^{l,n}(i,j)}{\sum_{n=1}^{N}\sum_{u\in \mathbf{U}_{\overline{\mathbf{y}}^n}\setminus \text{leaf}(\overline{\mathbf{y}}^n)}\sum_{i=1}^{C}\epsilon_{u,ch_l(u)}^{l,n}(i,j)}. \tag{62}$$

Label Emission Probability. The exact parametrization of the emission distribution depends on form of the observed labels, but the update equations are no different from those of a *standard* hidden Markov model. For instance, for discrete labels the emission distribution is multinomial and the update equation is, trivially, as follows

$$b_i(m) = \frac{\sum_{n=1}^{N}\sum_{u\in \mathbf{U}_{\overline{\mathbf{y}}^n}} P(Q_u = x_i|\overline{\mathbf{y}}^n)\delta(y_u, o_m)}{\sum_{n=1}^{N}\sum_{u\in \mathbf{U}_{\overline{\mathbf{y}}^n}} P(Q_u = x_i|\overline{\mathbf{y}}^n)} = \frac{\sum_{n=1}^{N}\sum_{u\in \mathbf{U}_{\overline{\mathbf{y}}^n}} \epsilon_u^n(i)\delta(y_u, o_m)}{\sum_{n=1}^{N}\sum_{u\in \mathbf{U}_{\overline{\mathbf{y}}^n}} \epsilon_u^n(i)} \tag{63}$$

where the delta function $\delta(y_u, o_m)$ ensures that only the observations equal to o_m contribute to the m-th emission probability.

In case of d-dimensional continuous observations $o \in \mathbb{R}^d$, the emission corresponding to the i-th hidden state can by modeled by a Normal distribution

$$b_i(o) = \mathcal{N}(o; \mu_i, \Sigma_i) = \frac{1}{(2\pi)^{\frac{d}{2}} |\Sigma_i|^{\frac{1}{2}}} \exp\left(-\frac{1}{2}(o-\mu_i)^T \Sigma_i^{-1}(o-\mu_i)\right).$$

Update rules for the emission parameters are obtained by maximization of the following likelihood component

$$\begin{aligned}\mathcal{Q}_B &= \sum_{n=1}^{N} \sum_{u \in \mathbf{U}_{\overline{\mathbf{y}}^n}} \sum_{i=1}^{C} P(Q_u = x_i | \overline{\mathbf{y}}^n) \log \mathcal{N}(y_u; \mu_i, \sigma_i) \\ &= \sum_{n=1}^{N} \sum_{u \in \mathbf{U}_{\overline{\mathbf{y}}^n}} \sum_{i=1}^{C} P(Q_u = x_i | \overline{\mathbf{y}}^n) \left\{ -\frac{d}{2}\log(2\pi) - \frac{1}{2}\log(|\Sigma_i|) - \frac{1}{2}(y_u-\mu_i)^T \Sigma_i^{-1}(y_u-\mu_i) \right\}\end{aligned} \tag{64}$$

with respect to Σ_i and μ_i, yielding

$$\mu_i^{(k+1)} = \frac{\sum_{n=1}^{N} \sum_{u \in \mathbf{U}_{\overline{\mathbf{y}}^n}} y_u \epsilon_u^n(i)}{\sum_{n=1}^{N} \sum_{u \in \mathbf{U}_{\overline{\mathbf{y}}^n}} \epsilon_u^n(i)}, \tag{65}$$

and

$$\Sigma_i^{(k+1)} = \frac{\sum_{n=1}^{N} \sum_{u \in \mathbf{U}_{\overline{\mathbf{y}}^n}} (y_u-\mu_i)(y_u-\mu_i)^T \epsilon_u^n(i)}{\sum_{n=1}^{N} \sum_{u \in \mathbf{U}_{\overline{\mathbf{y}}^n}} \epsilon_u^n(i)}. \tag{66}$$

4 Bottom-up Viterbi Algorithm

The Viterbi algorithm determines the most likely hidden states assignment $\overline{\mathbf{Q}}_1 = \overline{\mathbf{x}}$ for a given observed tree $\overline{\mathbf{Y}}_1 = \overline{\mathbf{y}}_1$ or, equivalently,

$$\max_{\overline{\mathbf{x}}} P(\overline{\mathbf{Y}}_1 = \overline{\mathbf{y}}_1, \overline{\mathbf{Q}}_1 = \overline{\mathbf{x}}). \tag{67}$$

The Viterbi algorithm for a bottom-up HTMM model entails an upward recursion from the leaves to the root of the tree, which follows from a factorization of (67). In particular, given a node u in the tree, we can rewrite it as

$$\max_{\overline{\mathbf{x}}} P(\overline{\mathbf{Y}}_1 = \overline{\mathbf{y}}_1, \overline{\mathbf{Q}}_1 = \overline{\mathbf{x}}) = \max_{\overline{\mathbf{x}}} P(\overline{\mathbf{Y}}_{\overline{\mathbf{CH}}(u)}, \overline{\mathbf{Y}}_{1\setminus\overline{\mathbf{CH}}(u)}, \overline{\mathbf{Q}}_u = \overline{\mathbf{x}}_u, \overline{\mathbf{Q}}_{1\setminus u} = \overline{\mathbf{x}}_{1\setminus u}), \tag{68}$$

where $\overline{\mathbf{Y}}_u$ is used as short form for $\overline{\mathbf{Y}}_u = \overline{\mathbf{y}}_u$ and $\overline{\mathbf{CH}}(u)$ indicates the set of children of node u (hence $\overline{\mathbf{Y}}_{\overline{\mathbf{CH}}(u)}$ stands for the observed subtrees rooted at each of the child of u, while $\overline{\mathbf{Y}}_{1\setminus\overline{\mathbf{CH}}(u)}$ is the original observed tree without the child subtrees of node u). By straightforward application of Bayes formula, equation (68) rewrites as

$$\begin{aligned}&\max_{\overline{\mathbf{x}}} P(\overline{\mathbf{Y}}_1 = \overline{\mathbf{y}}_1, \overline{\mathbf{Q}}_1 = \overline{\mathbf{x}}) \overset{\text{cond. prob.}}{=} \\ &= \max_{\overline{\mathbf{x}}} \left\{ P(\overline{\mathbf{Y}}_{1\setminus\overline{\mathbf{CH}}(u)}, \overline{\mathbf{Q}}_{1\setminus u} = \overline{\mathbf{x}}_{1\setminus u} | \overline{\mathbf{Q}}_u = \overline{\mathbf{x}}_u, \overline{\mathbf{Y}}_{\overline{\mathbf{CH}}(u)}) P(\overline{\mathbf{Q}}_u = \overline{\mathbf{x}}_u, \overline{\mathbf{Y}}_{\overline{\mathbf{CH}}(u)}) \right\} \overset{\text{cond ind}}{=} \\ &= \max_{\overline{\mathbf{x}}} \left\{ P(\overline{\mathbf{Y}}_{1\setminus\overline{\mathbf{CH}}(u)}, \overline{\mathbf{Q}}_{1\setminus u} = \overline{\mathbf{x}}_{1\setminus u} | Q_u = x_u) P(\overline{\mathbf{Q}}_u = \overline{\mathbf{x}}_u, \overline{\mathbf{Y}}_{\overline{\mathbf{CH}}(u)}) \right\} \overset{\text{max distr}}{=} \\ &= \max_{x_{i_u}} \left\{ \max_{\overline{\mathbf{x}}_{1\setminus u}} P(\overline{\mathbf{Y}}_{1\setminus\overline{\mathbf{CH}}(u)}, \overline{\mathbf{Q}}_{1\setminus u} = \overline{\mathbf{x}}_{1\setminus u} | Q_u = x_{i_u}) \max_{\overline{\mathbf{x}}_{\overline{\mathbf{CH}}(u)}} P(Q_u = x_{i_u}, \overline{\mathbf{Q}}_{\overline{\mathbf{CH}}(u)} = \overline{\mathbf{x}}_{\overline{\mathbf{CH}}(u)}, \overline{\mathbf{Y}}_{\overline{\mathbf{CH}}(u)}) \right\}\end{aligned} \tag{69}$$

which is the basis for the upward recursion. Let us define

$$\delta_u(i) = \max_{\overline{\mathbf{x}}_{\overline{\mathbf{CH}}(u)}} P(Q_u = x_i, \overline{\mathbf{Q}}_{\overline{\mathbf{CH}}(u)} = \overline{\mathbf{x}}_{\overline{\mathbf{CH}}(u)}, \overline{\mathbf{Y}}_{\overline{\mathbf{CH}}(u)}), \tag{70}$$

then, for each leaf node u' we initialize the recursion as

$$\delta_{u'}(i) = \max_{\overline{\mathbf{x}}_{\overline{\mathbf{CH}}(u')}} P(Q'_u = x_i) = P^{pos(u')}(Q'_u = x_i), \tag{71}$$

that is equivalent to the prior distribution of the bottom-up HTMM model. For each internal node taken upwards, we have the following factorization

$$\begin{aligned}
\delta_u(i) &= \max_{\overline{\mathbf{x}}_{\overline{\mathbf{CH}}(u)}} P(Q_u = x_i, \overline{\mathbf{Q}}_{\overline{\mathbf{CH}}(u)} = \overline{\mathbf{x}}_{\overline{\mathbf{CH}}(u)}, \overline{\mathbf{Y}}_{\overline{\mathbf{CH}}(u)}) \overset{\text{cond. prob.}}{=} \\
&= \max_{\overline{\mathbf{x}}_{\overline{\mathbf{CH}}(u)}} \left\{ P(Q_u = x_i | \overline{\mathbf{Q}}_{\overline{\mathbf{CH}}(u)} = \overline{\mathbf{x}}_{\overline{\mathbf{CH}}(u)}, \overline{\mathbf{Y}}_{\overline{\mathbf{CH}}(u)}) P(\overline{\mathbf{Q}}_{\overline{\mathbf{CH}}(u)} = \overline{\mathbf{x}}_{\overline{\mathbf{CH}}(u)}, \overline{\mathbf{Y}}_{\overline{\mathbf{CH}}(u)}) \right\} \overset{\text{cond ind}}{=} \\
&= \max_{\overline{\mathbf{x}}_{\overline{\mathbf{CH}}(u)}} \left\{ P(Q_u = x_i | Q_{ch_1(u)} = x_{i_1}, \ldots, Q_{ch_L(u)} = x_{i_L}) P(\overline{\mathbf{Q}}_{\overline{\mathbf{CH}}(u)} = \overline{\mathbf{x}}_{\overline{\mathbf{CH}}(u)}, \overline{\mathbf{Y}}_{\overline{\mathbf{CH}}(u)}) \right\} \overset{\text{switch par}}{=} \\
&= \max_{\overline{\mathbf{x}}_{\overline{\mathbf{CH}}(u)}} \left\{ \sum_{l=1}^{L} P(S_u = l) P^l(Q_u = x_i | Q_{ch_l(u)} = x_{i_l}) P(\overline{\mathbf{Q}}_{\overline{\mathbf{CH}}(u)} = \overline{\mathbf{x}}_{\overline{\mathbf{CH}}(u)}, \overline{\mathbf{Y}}_{\overline{\mathbf{CH}}(u)}) \right\} \overset{\text{cond ind}}{=} \\
&= \max_{\overline{\mathbf{x}}_{\overline{\mathbf{CH}}(u)}} \left\{ \sum_{l=1}^{L} P(S_u = l) P^l(Q_u = x_i | Q_{ch_l(u)} = x_{i_l}) \prod_{v \in \overline{\mathbf{CH}}(u)} P(\overline{\mathbf{Q}}_v = \overline{\mathbf{x}}_v, \overline{\mathbf{Y}}_v) \right\},
\end{aligned} \tag{72}$$

where the latter equality holds since the couples of observed and hidden subtrees are independent from each other when the parent (y_u, Q_u) is not given (formal proof can be obtained by repeated Bayesian decomposition, showing that the Markov blanket of an observed subtree $\overline{\mathbf{Y}}_v$ includes only the corresponding hidden states $\overline{\mathbf{Q}}_v$). Since $\overline{\mathbf{Y}}_v = y_v \cup \overline{\mathbf{Y}}_{\overline{\mathbf{CH}}(v)}$, we can rewrite (72) as follows

$$\begin{aligned}
\delta_u(i) &= \max_{\overline{\mathbf{x}}_{\overline{\mathbf{CH}}(u)}} \left\{ \sum_{l=1}^{L} P(S_u = l) P^l(Q_u = x_i | Q_{ch_l(u)} = x_{i_l}) \prod_{v \in \overline{\mathbf{CH}}(u)} P(\overline{\mathbf{Q}}_v = \overline{\mathbf{x}}_v, y_v, \overline{\mathbf{Y}}_{\overline{\mathbf{CH}}(v)}) \right\} \overset{\text{Bayes+emission}}{=} \\
&= \max_{\overline{\mathbf{x}}_{\overline{\mathbf{CH}}(u)}} \left\{ \sum_{l=1}^{L} P(S_u = l) P^l(Q_u = x_i | Q_{ch_l(u)} = x_{i_l}) \prod_{v \in \overline{\mathbf{CH}}(u)} P(y_v | Q_v = x_{i_v}) P(\overline{\mathbf{Q}}_v = \overline{\mathbf{x}}_v, \overline{\mathbf{Y}}_{\overline{\mathbf{CH}}(v)}) \right\} \overset{\text{max distr}}{=} \\
&= \max_{x_{i_1}, \ldots, x_{i_l}, \ldots, x_{i_L}} \left\{ \sum_{l=1}^{L} P(S_u = l) P^l(Q_u = x_i | Q_{ch_l(u)} = x_{i_l}) \right. \\
&\quad \left. \times \prod_{v=1}^{L} P(y_v | Q_v = x_{i_v}) \prod_{v' \in \overline{\mathbf{CH}}(u)} \max_{\overline{\mathbf{x}}_{\overline{\mathbf{CH}}(v)}} P(\overline{\mathbf{Q}}_{v'} = \overline{\mathbf{x}}_{v'}, \overline{\mathbf{Y}}_{\overline{\mathbf{CH}}(v')}) \right\} \overset{\text{def } \delta_{v'}}{=} \\
&= \max_{x_{i_1}, \ldots, x_{i_l}, \ldots, x_{i_L}} \left\{ \sum_{l=1}^{L} P(S_u = l) P^l(Q_u = x_i | Q_{ch_l(u)} = x_{i_l}) \prod_{v=1}^{L} P(y_v | Q_v = x_{i_v}) \prod_{v' \in \overline{\mathbf{CH}}(u)} \delta_{v'}(i'_v) \right\}.
\end{aligned} \tag{73}$$

Equation (73) states that an intermediate node u receives, from each child subtree v', a $\delta_{v'}$ value and an emission probability given an hidden state assignment for the child node. Then, node u determines the most likely hidden states assignments for its direct child nodes, given its hidden state assignment x_i. Finally, it forwards upwards the generated $\delta_u(i)$ values to its parent node.

Such a recursion ends at the root node, where the first term in (69) evaluates to $P(\overline{\mathbf{Y}}_{1 \setminus \overline{\mathbf{CH}}(1)}, \overline{\mathbf{Q}}_{1 \setminus 1} = \overline{\mathbf{x}}_{1 \setminus 1} | Q_1 = x_{i_1}) = P(y_1 | Q_1 = x_{i_1})$: at this point, the root node can determine the hidden state assignment $x^*_{i_1}$ that maximizes

the joint probability $P(\overline{\mathbf{Y}}_1 = \overline{\mathbf{y}}_1, \overline{\mathbf{Q}}_1 = \overline{\mathbf{x}})$. Such an hidden state can be used to backtrack the most likely hidden state assignments for the rest of the nodes in the tree, hence obtaining the generating latent points for each of the subtrees in $\overline{\mathbf{y}}_1$.

References

1. Crouse, M., Nowak, R., Baraniuk, R.: Wavelet-based statistical signal-processing using hidden markov-models. IEEE Trans. Signal Process. **46**(4) (April 1998) 886–902
2. Diligenti, M., Frasconi, P., Gori, M.: Hidden tree markov models for document image classification. IEEE Trans. Pattern Anal. Mach. Intell. **25**(4) (2003) 519–523
3. Frasconi, P., Gori, M., Sperduti, A.: A general framework for adaptive processing of data structures. IEEE Trans. Neural Netw. **9**(5) (1998) 768–786
4. Hammer, B., Micheli, A., Sperduti, A., Strickert, M.: A general framework for unsupervised processing of structured data. Neurocomputing **57** (2004) 3–35
5. Saul, L.K., Jordan, M.I.: Mixed memory markov models: Decomposing complex stochastic processes as mixtures of simpler ones. Mach. Learn. **37**(1) (1999) 75–87
6. Bilmes, J.: Dynamic Bayesian Multinets. In: Proc. of the 16th Conf. on Uncert. in Artif. Intell. (2000) 38–45
7. Jordan, M.I., Jacobs, R.A.: Hierarchical mixtures of experts and the EM algorithm. Neural Computation **6**(2) (1994) 181–214
8. Raftery, A.E.: A model for high-order markov chains. Journal of the Royal Statistical Society. Series B (Methodological) **47**(3) (1985) 528–539
9. Durand, J., Goncalves, P., Guedon, Y., Rhone-Alpes, I., Montbonnot, F.: Computational methods for hidden Markov tree models-an application to wavelet trees. IEEE Transactions on Signal Processing **52**(9) (2004) 2551–2560
10. Lauritzen, S.: Graphical models. Oxford University Press, USA (1996)