

---

**Algorithm 14.1 Expectation propagation message passing for CLG networks**


---

```

Procedure CLG-M-Project-Distr (
     $Z$ , // Scope to remain following projection
     $\vec{\phi}$  // Set of canonical tables
)
1 // Compute overall measure using product of canonical tables
2  $\tilde{\beta} \leftarrow \prod_{\phi \in \vec{\phi}} \phi$ 
3 // Variables to be preserved
4  $A \leftarrow Z \cap \Delta$ 
5  $X \leftarrow Z \cap \Gamma$ 
6 // Variables to be eliminated
7  $B \leftarrow (\text{Scope}[\vec{\phi}] - A) \cap \Delta$ 
8  $Y \leftarrow (\text{Scope}[\vec{\phi}] - X) \cap \Gamma$ 
9 for each  $a, b \in \text{Val}(A, B)$ 
10  $\tau(a, b) \leftarrow \int \beta_i(a, b) dY$  using equation (14.5)
11  $\tilde{\sigma} \leftarrow \sum_B \tau$  using definition 14.3
12 return  $\tilde{\sigma}$ 

```

---

are eliminated using the marginalization operation of equation (14.5) over each entry in the canonical table separately. The discrete variables are then summed out. For this last step, there are two cases. If the factor  $\tau$  contains only discrete variables, then we use standard discrete marginalization. If not, then we use the weak marginalization operation of definition 14.3.

In principle, this application of the EP framework is fairly straightforward. There are, however, two important subtleties that arise in this setting.

### 14.3.3.1 Ordering Constraints

First, as we discussed, for weak marginalization to be well defined, the canonical form being marginalized needs to be a Gaussian distribution, and not merely a canonical form. In some cases, this requirement is satisfied simply because of the form of the potentials in the original network factorization. For example, in the Gaussian case, recall that our message passing process is well defined if our distribution is pairwise normalizable. In the conditional Gaussian case, we can guarantee normalizability if for each cluster over scope  $X$ , the initial factor in that cluster is a canonical table such that each canonical form entry  $C(X; K_d, h_d, g_d)$  is normalizable. Because normalizability is closed under factor product (because the sum of two PSD matrices is also PSD) and (both weak and strong) marginalization, this requirement guarantees us that all factors produced by a sum-product algorithm will be normalizable.

However, this requirement is not always easy to satisfy:

---

**Example 14.6**

Consider a CLG network structured as in figure 14.5a, and the clique tree shown in figure 14.5b. Note that the only clique where each canonical form is normalizable is  $C_1 = \{A, X\}$ ; in  $C_2 = \{B, X, Y\}$ , the canonical forms in the canonical table are all linear Gaussians whose integral is infinite, and hence cannot be collapsed in the operation of proposition 14.3.