where the last equality follows from calibration. Each of these probability expressions corresponds to a set of clique beliefs divided by a message. We can now perform variable elimination, using these factors in the usual way. ■

Algorithm 10.4 Out-of-clique inference in clique tree

Procedure CTree-Query (
  $\mathcal{T}$, // Clique tree over Φ
  $\{\beta_i\}, \{\mu_{i,j}\}$, // Calibrated clique and sepset beliefs for $\mathcal{T}$
  $\boldsymbol{Y}$ // A query
)
  Let $\mathcal{T}'$ be a subtree of $\mathcal{T}$ such that $\boldsymbol{Y} \subseteq$ *Scope*$[\mathcal{T}']$
  Select a clique $r \in \mathcal{V}_{\mathcal{T}'}$ to be the root
  $\Phi \leftarrow \beta_r$
  for each $i \in \mathcal{V}'_{\mathcal{T}}$
    $\phi \leftarrow \frac{\beta_i}{\mu_{i,p_r(i)}}$
    $\Phi \leftarrow \Phi \cup \{\phi\}$
  $\boldsymbol{Z} \leftarrow$ *Scope*$[\mathcal{T}'] - \boldsymbol{Y}$
  Let $\prec$ be some ordering over $\boldsymbol{Z}$
  return Sum-Product-VE$(\Phi, \boldsymbol{Z}, \prec)$

More generally, we can compute the joint probability $\tilde{P}_\Phi(\boldsymbol{Y})$ for an arbitrary subset $\boldsymbol{Y}$ by using the beliefs in a calibrated clique tree to define factors corresponding to conditional probabilities in $\tilde{P}_\Phi$, and then performing variable elimination over the resulting set of factors. The precise algorithm is shown in algorithm 10.4. The savings over simple variable elimination arise because we do not have to perform inference over the entire clique tree, but only over a portion of the tree that contains the variables $\boldsymbol{Y}$ that constitute our query. In cases where we have a very large clique tree, the savings can be significant.

10.3.3.3 Multiple Queries

Now, assume that we want to compute the probabilities of an entire set of queries where the variables are not together in a clique. For example, we might wish to compute $\tilde{P}_\Phi(X, Y)$ for every pair of variables $X, Y \in \mathcal{X} - \boldsymbol{E}$. Clearly, the approach of constructing a clique tree to ensure that our query variables are present in a single clique breaks down in this case: If every pair of variables is present in some clique, there must be some clique that contains all of the variables (see exercise 10.14).

A somewhat less naive approach is simply to run the variable elimination algorithm of algorithm 10.4 $\binom{n}{2}$ times, once for each pair of variables X, Y. However, because pairs of variables, on average, are fairly far from each other in the clique tree, this approach requires fairly substantial running time (see exercise 10.15). An even better approach can be obtained by using *dynamic programming*.

dynamic programming

Consider a calibrated clique tree $\mathcal{T}$ over Φ, and assume we want to compute the probability $\tilde{P}_\Phi(X, Y)$ for every pair of variables X, Y. We execute this process by gradually constructing a