$E^-_{Y_i}$ is independent of X and $\mathbf{z}_i$ given y_i, and $E^+_{Y_i\backslash X}$ is independent of X and y_i. We can also pull a term with no $\mathbf{z}_i$ out of the $\mathbf{z}_i$ summation:

$$\mathbf{P}(E^-_X|X) = \prod_i \sum_{y_i} P(E^-_{Y_i}|y_i) \sum_{\mathbf{z}_i} \mathbf{P}(E^+_{Y_i\backslash X}|\mathbf{z}_i)\mathbf{P}(y_i, \mathbf{z}_i|X)$$

Apply Bayes' rule to $\mathbf{P}(E^+_{Y_i\backslash X}|\mathbf{z}_i)$:

$$\mathbf{P}(E^-_X|X) = \prod_i \sum_{y_i} P(E^-_{Y_i}|y_i) \sum_{\mathbf{z}_i} \frac{P(\mathbf{z}_i|E^+_{Y_i\backslash X})P(E^+_{Y_i\backslash X})}{P(\mathbf{z}_i)}\mathbf{P}(y_i, \mathbf{z}_i|X)$$

Rewriting the conjunction $Y_i, \mathbf{z}_i$:

$$\mathbf{P}(E^-_X|X) = \prod_i \sum_{y_i} P(E^-_{Y_i}|y_i) \sum_{\mathbf{z}_i} \frac{P(\mathbf{z}_i|E^+_{Y_i\backslash X})P(E^+_{Y_i\backslash X})}{P(\mathbf{z}_i)}\mathbf{P}(y_i|X, \mathbf{z}_i)\mathbf{P}(\mathbf{z}_i|X)$$

Now $\mathbf{P}(\mathbf{z}_i|X) = P(\mathbf{z}_i)$, because $\mathbf{Z}$ and X are d-separated, so we can cancel them out. We can also replace $P(E^+_{Y_i\backslash X})$ by a normalizing constant β_i:

$$\mathbf{P}(E^-_X|X) = \prod_i \sum_{y_i} P(E^-_{Y_i}|y_i) \sum_{\mathbf{z}_i} \beta_i P(\mathbf{z}_i|E^+_{Y_i\backslash X})\mathbf{P}(y_i|X, \mathbf{z}_i)$$

Finally, the parents of Y_i (the Z_{ij}) are independent of each other, so we can multiply them together, just as we did with the U_i parents previously. We also combine the β_i into one big normalizing constant β:

$$\mathbf{P}(E^-_X|X) = \beta \prod_i \sum_{y_i} P(E^-_{Y_i}|y_i) \sum_{\mathbf{z}_i} \mathbf{P}(y_i|X, \mathbf{z}_i) \prod_j P(z_{ij}|E_{Z_{ij}\backslash Y_i}) \tag{15.4}$$

Notice that each of the terms in the final expression is easily evaluated:

- $P(E^-_{Y_i}|y_i)$ is a recursive instance of $\mathbf{P}(E^-_X|X)$.
- $P(y_i|x\mathbf{z}_i)$ is a conditional probability table entry for Y_i.
- $P(z_{ij}|E_{Z_{ij}\backslash Y_i})$ is a recursive instance of the $\mathbf{P}(X|E)$ calculation—that is, $\mathbf{P}(X|E_{X\backslash})$

It is now a simple matter to turn all this into an algorithm. We will need two basic routines. SUPPORT-EXCEPT(X,V) computes $\mathbf{P}(X|E_{X\backslash V})$, using a slight generalization of Equation (15.3) to handle the "except" variable V. EVIDENCE-EXCEPT(X,V) computes $P(E^-_{X\backslash V}|X)$, using a generalization of Equation (15.4). The algorithms are shown in Figure 15.8.

The computation involves recursive calls that spread out from X along all paths in the network. The recursion terminates on evidence nodes, root nodes (which have no parents), and leaf nodes (which have no children). Each recursive call excludes the node from which it was called, so each node in the tree is covered only once. Hence the algorithm is linear in the number of nodes in the network. Remember that this only works because the network is a *polytree*. If there were more than one path between a pair of nodes, then either our recursions would count the same evidence more than once or they would fail to terminate.

We have chosen to present a "backward-chaining" algorithm because it is the simplest algorithm for polytrees. One drawback is that it computes the probability distribution for just one variable. If we wanted the posterior distributions for all the non-evidence variables, we would have to run the algorithm once for each. This would give a quadratic runtime, and