Algorithm 9.5 Conditioning algorithm

Procedure Sum-Product-Conditioning (
  Φ, // Set of factors, possibly reduced by evidence
  $\boldsymbol{Y}$, // Set of query variables
  $\boldsymbol{U}$ // Set of variables on which to condition
)
  for each $\boldsymbol{u} \in Val(\boldsymbol{U})$
    $\Phi_{\boldsymbol{u}} \leftarrow \{\phi[\boldsymbol{U} = \boldsymbol{u}] : \phi \in \Phi\}$
    Construct $\mathcal{H}_{\Phi_{\boldsymbol{u}}}$
    $(\alpha_{\boldsymbol{u}}, \phi_{\boldsymbol{u}}(\boldsymbol{Y})) \leftarrow$ Cond-Prob-VE($\mathcal{H}_{\Phi_{\boldsymbol{u}}}, \boldsymbol{Y}, \emptyset$)
  $\phi^*(\boldsymbol{Y}) \leftarrow \frac{\sum_{\boldsymbol{u}} \phi_{\boldsymbol{u}}(\boldsymbol{Y})}{\sum_{\boldsymbol{u}} \alpha_{\boldsymbol{u}}}$
  Return $\phi^*(\boldsymbol{Y})$

The conditioning algorithm is based on the following simple derivation. Let $\boldsymbol{U} \subseteq \boldsymbol{X}$ be any set of variables. Then we have that:

$$\tilde{P}_{\Phi}(\boldsymbol{Y}) = \sum_{\boldsymbol{u} \in Val(\boldsymbol{U})} \tilde{P}_{\Phi}(\boldsymbol{Y}, \boldsymbol{u}). \tag{9.11}$$

The key observation is that each term $\tilde{P}_{\Phi}(\boldsymbol{Y}, \boldsymbol{u})$ can be computed by marginalizing out the variables in $\boldsymbol{X} - \boldsymbol{U} - \boldsymbol{Y}$ *in the unnormalized measure* $\tilde{P}_{\Phi}[\boldsymbol{u}]$ *obtained by reducing* $\tilde{P}_{\Phi}$ *to the context* $\boldsymbol{u}$. As we have already discussed, the reduced measure is simply the measure defined by reducing each of the factors to the context $\boldsymbol{u}$. The reduction process generally produces a simpler structure, with a reduced inference cost.

We can use this formula to compute $P_{\Phi}(\boldsymbol{Y})$ as follows: We construct a network $\mathcal{H}_{\Phi}[\boldsymbol{u}]$ for each assignment $\boldsymbol{u}$; these networks have identical structures, but different parameters. We run sum-product inference in each of them, to obtain a factor over the desired query set $\boldsymbol{Y}$. We then simply add up these factors to obtain $\tilde{P}_{\Phi}(\boldsymbol{Y})$. We can also derive $P_{\Phi}(\boldsymbol{Y})$ by renormalizing this factor to obtain a distribution. As usual, the normalizing constant is the partition function for P_{Φ}. However, applying equation (9.11) to the case of $\boldsymbol{Y} = \emptyset$, we conclude that

$$Z_{\Phi} = \sum_{\boldsymbol{u}} Z_{\Phi[\boldsymbol{u}]}.$$

Thus, we can derive the overall partition function from the partition functions for the different subnetworks $\mathcal{H}_{\Phi[\boldsymbol{u}]}$. The final algorithm is shown in algorithm 9.5. (We note that Cond-Prob-VE was called without evidence, since we assumed for simplicity that our factors Φ have already been reduced with the evidence.)