Algorithm 11.6 The message passing step in the expectation propagation algorithm. The algorithm performs approximate message propagation by projecting expected sufficient statistics.

Procedure M-Project-Distr (
  $\mathcal{Q}$, // target exponential family for projection
  $\vec{\phi}$ // Factor set
)
1 $\boldsymbol{X} \leftarrow$ *Scope*$[\vec{\phi}]$ // Variables in factor set
2 $\bar{\tau} \leftarrow \boldsymbol{E}_{\boldsymbol{x} \sim \prod_{\phi \in \vec{\phi}}}\left[\tau_{\mathcal{Q}_{i,j}}(\boldsymbol{x})\right]$
3 // Compute expectation of sufficient statistics relative to distribution defined by product of factors
4 $\theta \leftarrow$ M-project$(\bar{\tau})$
5 **return** (θ)

M-projection

we choose to approximate messages by distributions from (linear) exponential families. Specifically, assume that we restrict each sepset $\boldsymbol{S}_{i,j}$ to be represented within an exponential family $\mathcal{Q}_{i,j}$ defined by a sufficient statistics function $\tau_{i,j}$. When performing message passing from $\boldsymbol{C}_i$ to $\boldsymbol{C}_j$, we compute the marginal of $\tilde{\beta}_i$, usually represented as a factor set $\vec{\phi}_i$, and project it into $\mathcal{Q}_{i,j}$ using the *M-projection* operator M-project-distr$_{i,j}$. This computation is often done using inference procedure that takes into account the structure of $\tilde{\beta}_i$ as a factor set.

It turns out that the entire message passing operation can be formulated cleanly within this framework. If we are using an exponential family to represent our messages, then both the approximate clique marginal $\tilde{\sigma}_{i\to j}$ and the previous message $\tilde{\delta}_{j\to i}$ can be represented in the exponential form. Thus, if we ignore normalization factors, we have:

$$
\begin{aligned}
\tilde{\sigma}_{i\to j} &\propto \exp\left\{\langle\theta_{\tilde{\sigma}_{i\to j}}, \tau_{i,j}(\boldsymbol{s}_{i,j})\rangle\right\} \\
\tilde{\delta}_{j\to i} &\propto \exp\left\{\langle\theta_{\tilde{\delta}_{j\to i}}, \tau_{i,j}(\boldsymbol{s}_{i,j})\rangle\right\} \\
\tilde{\delta}_{i\to j} &= \frac{\tilde{\sigma}_{i\to j}}{\tilde{\delta}_{j\to i}} \propto \exp\left\{\langle(\theta_{\tilde{\sigma}_{i\to j}} - \theta_{\tilde{\delta}_{j\to i}}), \tau_{i,j}(\boldsymbol{s}_{i,j})\rangle\right\},
\end{aligned}
$$

where $\theta_{\tilde{\sigma}_{i\to j}}$ and $\theta_{\tilde{\delta}_{i\to j}}$ are the parameters of the messages $\tilde{\sigma}_{i\to j}$ and $\tilde{\delta}_{i\to j}$ respectively.

Since these messages are in an exponential family, it suffices to represent each of them by the parameters that describe them. We can then view propagation steps as updating these parameters. Specifically, we can rewrite the update step in line 4 of EP-Message as

$$
\theta_{\tilde{\delta}_{i\to j}} \leftarrow (\theta_{\tilde{\sigma}_{i\to j}} - \theta_{\tilde{\delta}_{j\to i}}). \tag{11.40}
$$

Note that, in the case of exact inference in discrete networks, the original update and the one using the exponential family representation are essentially identical, since the exponential family representation of factors is of the same size as the factor. Indeed, the standard update is often performed in a logarithmic representation (for reasons of numerical stability; see box 10.A), which gives rise precisely to the exponential family update.

The final issue we must address is the construction of the exponential-family representation of $\tilde{\sigma}_{i\to j}$ in line 1 of the algorithm. Recall that this process involves the M-projection of $\tilde{\beta}_i$