ECE 590.10: Graphical Models and Inference

Lecture: Belief Propagation

Henry D. Pfister

Department of Electrical and Computer Engineering
Duke University

Marginalization via Message Passing

- Tree factor graphs: Efficient marginalization via variable-leaf removal
  - Each step removes variable-leaf node and all child factor nodes
  - Parent factor node modified to summarize missing information
  - Can be seen as a sequential message passing algorithm
- Belief Propagation (BP)
  - Parallel message-passing algorithm similar to leaf removal
  - For a tree, computes exact marginal for each variable
  - Algorithm defined for a general graphs but not exact
- History
  - Hans Bethe used infinite tree idea to analyze alloys in 1935
  - Bob Gallager used special case to decode LDPC codes in 1961
  - Judea Pearl defined BP for general tree models in 1986

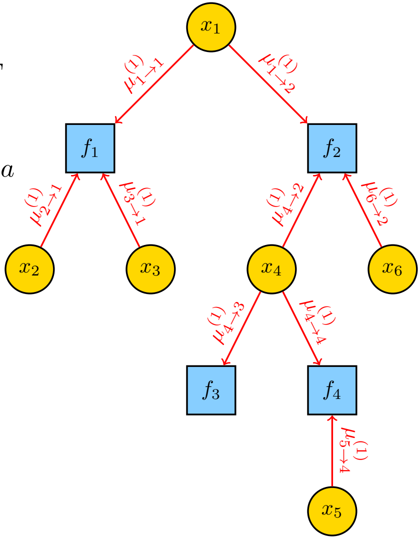
Belief Propagation

Factor Graph (FG):

- Variable nodes V, Factor nodes F
- Edges: $(i, a) \in E \subseteq V \times F$
- $F(i)/V(a) =$ neighbors of node-i/a

Belief Propagation (BP):

- Init: $\mu_{i \to a}^{(1)}(x_i) = 1 \ \forall \ (i, a) \in E$



Belief Propagation

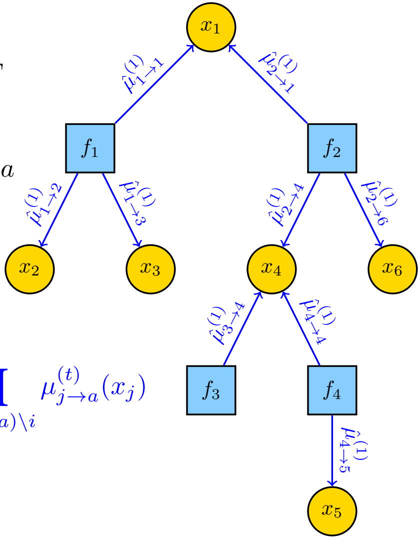
Factor Graph (FG):

- Variable nodes V, Factor nodes F
- Edges: $(i,a) \in E \subseteq V \times F$
- $F(i)/V(a) =$ neighbors of node-i/a

Belief Propagation (BP):

- Init: $\mu^{(1)}_{i\to a}(x_i) = 1 \ \forall\ (i,a) \in E$
- factor-a to variable-i message:

$$\hat{\mu}^{(t)}_{a\to i}(x_i) = \sum_{x_{V(a)\setminus i}} f_a(x_{V(a)}) \prod_{j\in V(a)\setminus i} \mu^{(t)}_{j\to a}(x_j)$$



Belief Propagation

Factor Graph (FG):

- Variable nodes V, Factor nodes F
- Edges: $(i,a) \in E \subseteq V \times F$
- $F(i)/V(a)$ = neighbors of node-i/a

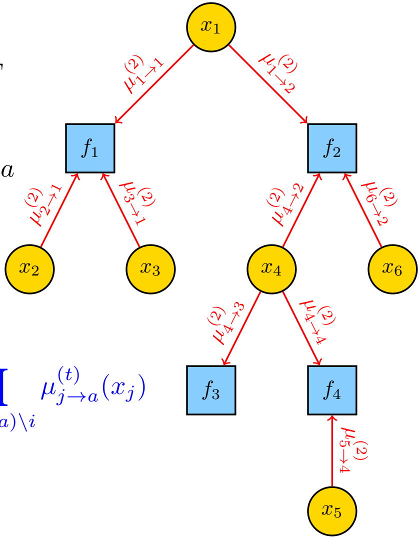
Belief Propagation (BP):

- Init: $\mu^{(1)}_{i \to a}(x_i) = 1 \ \forall \ (i,a) \in E$
- factor-a to variable-i message:

$$\hat{\mu}^{(t)}_{a \to i}(x_i) = \sum_{x_{V(a)\setminus i}} f_a(x_{V(a)}) \prod_{j \in V(a)\setminus i} \mu^{(t)}_{j \to a}(x_j)$$

- variable-i to factor-a message:

$$\mu^{(t+1)}_{i \to a}(x_i) = \prod_{b \in F(i)\setminus a} \hat{\mu}^{(t)}_{b \to i}(x_i)$$



Belief Propagation

Factor Graph (FG):

- Variable nodes V, Factor nodes F
- Edges: $(i,a) \in E \subseteq V \times F$
- $F(i)/V(a) =$ neighbors of node-i/a

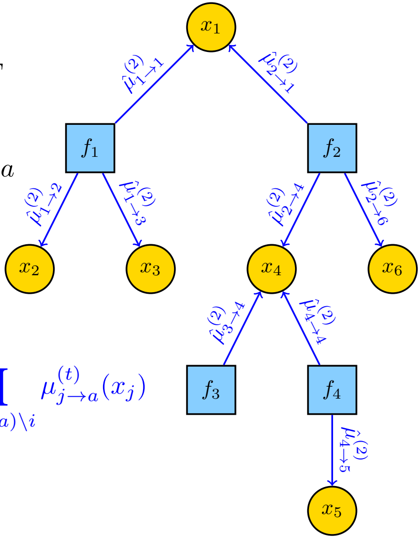
Belief Propagation (BP):

- Init: $\mu^{(1)}_{i\to a}(x_i) = 1 \ \forall\ (i,a) \in E$
- factor-a to variable-i message:

$$\hat{\mu}^{(t)}_{a\to i}(x_i) = \sum_{x_{V(a)\setminus i}} f_a(x_{V(a)}) \prod_{j\in V(a)\setminus i} \mu^{(t)}_{j\to a}(x_j)$$

- variable-i to factor-a message:

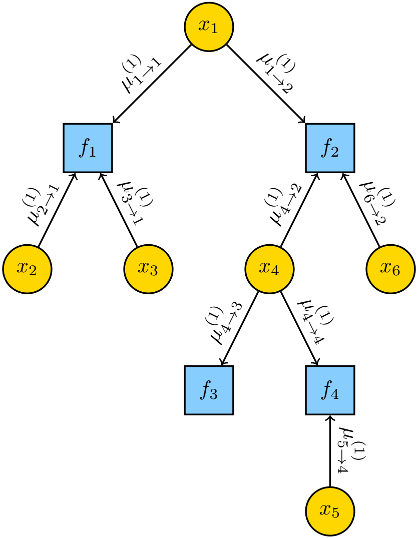
$$\mu^{(t+1)}_{i\to a}(x_i) = \prod_{b\in F(i)\setminus a} \hat{\mu}^{(t)}_{b\to i}(x_i)$$



Belief Propagation: Example

iteration 1: variable to factor

$$\mu_{i \to a}^{(1)}(x_i) = 1$$



Belief Propagation: Example

iteration 1: variable to factor

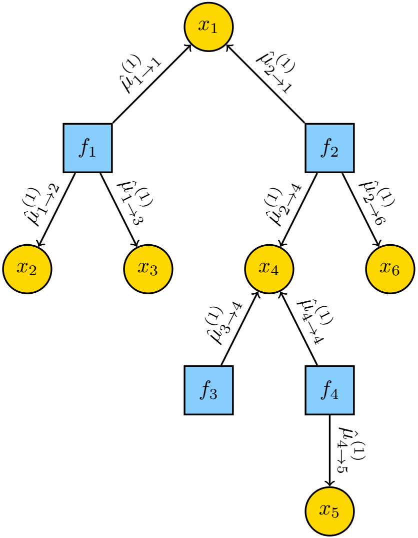
$$\mu_{i\to a}^{(1)}(x_i) = 1$$

iteration 1: factor to variable

$$\hat{\mu}_{4\to 4}^{(1)}(x_4) = \sum_{x_5} f_4(x_4, x_5)\mu_{5\to 4}^{(1)}(x_5)$$

$$= \sum_{x_5} f_4(x_4, x_5)$$

$$\hat{\mu}_{3\to 4}^{(1)}(x_4) = f_3(x_4)$$



Belief Propagation: Example

iteration 1: factor to variable

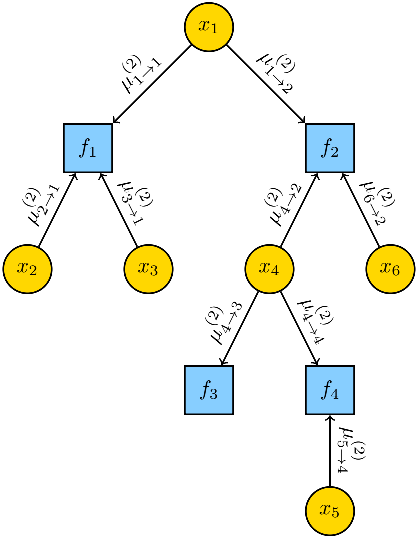
$$\hat{\mu}^{(1)}_{4\to 4}(x_4) = \sum_{x_5} f_4(x_4, x_5)\mu^{(1)}_{5\to 4}(x_5)$$
$$= \sum_{x_5} f_4(x_4, x_5)$$

$$\hat{\mu}^{(1)}_{3\to 4}(x_4) = f_3(x_4)$$

iteration 2: variable to factor

$$\mu^{(2)}_{4\to 2}(x_4) = \hat{\mu}^{(1)}_{4\to 4}(x_4)\hat{\mu}^{(1)}_{3\to 4}(x_4)$$
$$= f_3(x_4)\sum_{x_5} f_4(x_4, x_5)$$

$$\mu^{(2)}_{6\to 2}(x_6) = 1$$



Belief Propagation: Example

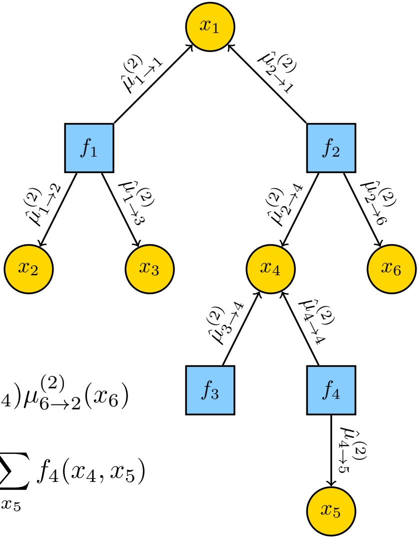
iteration 2: variable to factor

$$\mu^{(2)}_{4\to 2}(x_4) = \hat{\mu}^{(1)}_{4\to 4}(x_4)\hat{\mu}^{(1)}_{3\to 4}(x_4) = f_3(x_4)\sum_{x_5} f_4(x_4, x_5)$$

$$\mu^{(2)}_{6\to 2}(x_6) = 1$$

iteration 2: factor to variable

$$\hat{\mu}^{(1)}_{2\to 1}(x_1) = \sum_{x_4,x_6} f_2(x_1,x_4,x_6)\mu^{(2)}_{4\to 2}(x_4)\mu^{(2)}_{6\to 2}(x_6) = \sum_{x_4,x_6} f_2(x_1,x_4,x_6) f_3(x_4)\sum_{x_5} f_4(x_4,x_5) = f_2''(x_1)$$



Belief Propagation: Convergence

- For a tree factor graph, consider a FN a and adjacent VN i
  - Let $T(a \to i)$ be the subtree rooted at a given by cutting (i, a)
    - Ex: $T(2 \to 1)$ is subgraph induced by $x_4, x_5, x_6, f_2, f_3, f_4$
  - For all t, messages $\hat{\mu}_{a\to i}^{(t)}$ depend only on nodes in $T(a \to i)$
  - Let $h(a \to i)$ be the height of $T(a \to i)$ (i.e., max dist. to leaf)
  - Message $\hat{\mu}_{a\to i}^{(t)}$ converges (i.e., is constant) for $t \geq \left\lceil \frac{h(a\to i)+1}{2} \right\rceil$
  - All messages converge for $t \geq t^* = \left\lceil \frac{\mathrm{diam}(G)}{2} \right\rceil$

Belief Propagation: Output

- Let $F(a \to i)/V(a \to i)$ be the FNs/VNs in $T(a \to i)$
  - Then, $\hat{\mu}^{(t)}_{a\to i}(x_i)$ converges to the partial sum

$$\hat{\mu}^{(t^*)}_{a\to i}(x_i) = \sum_{x_{V(a\to i)}} \prod_{b\in F(a\to i)} f_b(x_{V(b)})$$

and the marginalization sum for x_i is given by

$$\begin{aligned} g_i(x_i) &\triangleq \sum_{x_1^n \backslash x_i} f(x_1,\ldots,x_n) = \sum_{x_1^n \backslash x_i} \prod_{b\in F} f_b(x_{V(b)}) \\ &= \sum_{x_1^n \backslash x_i} \prod_{a\in F(i)} \prod_{b\in F(a\to i)} f_b(x_{V(b)}) \\ &= \prod_{a\in F(i)} \sum_{x_{V(a\to i)\backslash i}} \prod_{b\in F(a\to i)} f_b(x_{V(b)}) \\ &= \prod_{a\in F(i)} \hat{\mu}^{(t^*)}_{b\to i}(x_i) \end{aligned}$$

Normalization

- Messages can also be normalized into distributions

$$\overline{\mu}^{(t)}_{i\to a}(x_i) = \frac{\mu^{(t)}_{i\to a}(x_i)}{\sum_{x_i'} \mu^{(t)}_{i\to a}(x_i')}$$

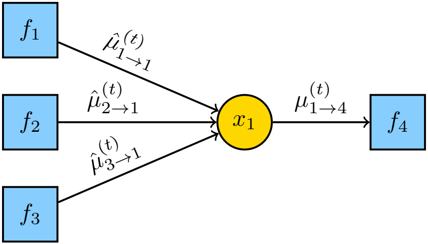
- Normalization doesn't affect message passing
  - Message passing: scaled input $\rightarrow$ scaled output
  - Just changes the overall constant

Log Likelihood-Ratio Messages for Binary Variables

- Normalized binary messages given by scalar: $\mu(1) = 1 - \mu(0)$
  - One can also use the likelihood ratio (LR): $\frac{\mu(0)}{\mu(1)}$
  - Or the log likelihood-ratio (LLR): $L = \ln \frac{\mu(0)}{\mu(1)}$
- For inference, LLR messages contain all the information:

$$L_{i \to a}^{(t)} = \ln \frac{\mu_{i \to a}^{(t)}(0)}{\mu_{i \to a}^{(t)}(1)} \qquad \hat{L}_{a \to i}^{(t)} = \ln \frac{\hat{\mu}_{a \to i}^{(t)}(0)}{\hat{\mu}_{a \to i}^{(t)}(1)}$$

VN Update for Binary Variables in LLR Domain



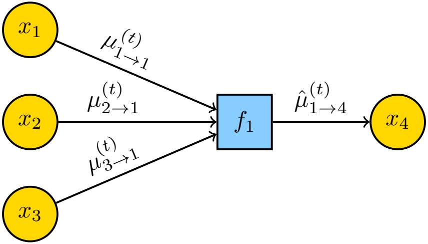
- Recall that the VN message-passing update is:

$$\mu^{(t+1)}_{i\to a}(x_i) = \prod_{b\in F(i)\backslash a} \hat{\mu}^{(t)}_{b\to i}(x_i)$$

- In the LLR domain, this simplifies to

$$L^{(t+1)}_{i\to a} = \ln \frac{\mu^{(t+1)}_{i\to a}(0)}{\mu^{(t+1)}_{i\to a}(1)} = \ln \frac{\prod_{b\in F(i)\backslash a} \hat{\mu}^{(t)}_{b\to i}(0)}{\prod_{b\in F(i)\backslash a} \hat{\mu}^{(t)}_{b\to i}(1)} = \sum_{b\in F(i)\backslash a} \hat{L}^{(t)}_{b\to i}$$

FN Update for Binary Variables in LLR Domain



- Recall that the FN message-passing update is:

$$\hat{\mu}^{(t)}_{a\to i}(x_i) = \sum_{x_{V(a)\setminus i}} f_1(x_{V(a)}) \prod_{j\in V(a)\setminus i} \mu^{(t)}_{j\to a}(x_j)$$

- In the LLR domain, this gives

$$\hat{L}^{(t)}_{a\to i} = \ln \frac{\hat{\mu}^{(t)}_{a\to i}(0)}{\hat{\mu}^{(t)}_{a\to i}(1)} = \ln \frac{\sum_{x_{V(a)}:x_i=0} f_1(x_{V(a)}) \prod_{j\in V(a)\setminus i} \mu^{(t)}_{j\to a}(x_j)}{\sum_{x_{V(a)}:x_i=1} f_1(x_{V(a)}) \prod_{j\in V(a)\setminus i} \mu^{(t)}_{j\to a}(x_j)}$$