16.1 Message passing

Cliques \( v \) and \( w \) are neighbour cliques and \( s \) is the separator. Initially, assume that we don’t have local consistancy, i.e. we do not have:

\[
p(H, E = e) = \frac{\psi_v \psi_w}{\phi_s}
\]

If we let message passing from left to right, i.e. from clique \( v \) through \( s \) to \( w \),

\[
\phi_s^* = \sum_{v \setminus s} \psi_v, \text{ this is to update separator. } \psi_w^* = \frac{\partial}{\partial \psi_w} \psi_w \text{ to update } w. \text{ Now we have passed messages from } \psi_v \rightarrow \phi_s^* \rightarrow \psi_w^*.
\]

Next, let message pass backward:

\[
\phi_s^{**} = \sum_{w \setminus s} \psi_w^*
\]

\[
\psi_c^{**} = \frac{\phi_s^{**}}{\phi_s^*} \psi_c^*
\]
\[ \psi_w^{**} = \psi_w^* \]

Now looking at the following, and we have consistency between \( V \) and \( W \) since:

\[
\sum_{v \neq w} \psi_v^{**} = \sum_{v \neq w} \frac{\phi_v^{**}}{\phi_w^*} \psi_v^* = \phi_w^{**} \sum_{v \neq w} \psi_v^* = \frac{\phi_w^{**}}{\phi_w^*} \phi_v^* = \phi_v^{**}
\]

As an example, look at the following graph:

In the graph above, the cliques are: \( AB; BC \).

\[ \psi_{AB} = P(A, B), \psi_{BC} = P(C|B), \text{ and } \phi_B = 1 \]

Message passing in the forward direction:

\[
\psi_{(AB)^*} = \sum_{\{A, B\} \setminus B} P(A, B) = \sum_{A} P(A, B) = P(B)
\]
Lecture 16: May 27, 2000

\[ \psi_{BC}^* = \frac{P(B)}{\phi_B} P(C|B) = \frac{P(B)}{1} P(C|B) = P(C, B) \]

Passing backward:

\[ \phi_B^{**} = \sum_C \psi_{BC}^* = \sum_C P(C, B) = P(B) \]

\[ \psi_{AB}^{**} = \frac{\phi_B^{**}}{\phi_B^*} \psi_{AB}^* = \frac{P(B)}{P(B)} p(A, B) = P(A, B) \]

⇒ Localconsistency

### 16.2 What happens when we introduce evidence?

As an example, \( A = 1 \).

Solution: Force the \( A = 0 \) portion of \( \psi_{AB} \) to be zero,

\[ \phi_B^* = \sum_A \psi_{AB} = P(A=1, B) \]

\[ \psi_{AB} \Rightarrow A=1 \psi_{AB} = \]

\[ \psi_B^*[0.240.56] \]

\[ \psi_{BC}^* = P(A=1, B) \frac{P(C|B)}{\phi_B^*} P(A=1, B) \] \[ P(C|B) = P(A=1, B, C) \] \[ \Rightarrow \psi_{AB}^* = P(A=1, B) \] \[ \phi_B^* = P(A=1, B) \]

\[ \psi_{BC}^* = P(A=1, B, C) \]

### 16.3 What happens when multiple cliques exists?

Ordering of message passing becomes important.

Looking at the following tree of cliques:
\( v \) and \( w \) are consistent, \( w \) and \( D_1 \) are consistent, but are \( v \) and \( D_1 \) consistent? 

The key issue is: ordering of message passing.

### 16.4 Message Passing Protocol

**Definition 16.1.** A clique can send a message to a neighbouring cliques only when it has received messages from all its other neighbours.

**Theorem 16.2.** The Message Passing Protocol renders the cliques locally consistent to all pairs of connected cliques in the tree.

**Proof.** 

To prove this, we consider two cases: Case 1: \( V \) already sent a message to \( W \) \( \Rightarrow \) \( V \) already received messages from all its neighbours.

Then \( V \) and \( W \) receive no more messages Therefore, consistent
Case 2: V had not yet sent a message to W, so W → V, and waits. (Message has to be back from V sometime in future.) Later, V will send $\phi_s^* \psi_w$?

$\Rightarrow$ inconsistency

### 16.5 Hugin Algorithm

![Figure 16.6: A Figure](image)

First, we will use function $\text{update}(w, v)$ to update clique $w$ with $v$, i.e. the following operations is done:

$$\phi_s^* = \sum_{v \in \phi_e} \psi_v$$

$$\psi_w^* = \frac{\phi_s^*}{\phi_e} \psi_w$$

The algorithm uses two routines:

- **CollectEvidence(node)**
  - for each child of node
    - update(node, CollectEvidence(child));
  - return node;

- **DistributeEvidence(node)**
  - for every child of node
    - update(child, code);
    - DistributeEvidence(child);
  - return

**Algorithm (root)**

CollectEvidence(root);
DistributeEvidence(root);

**Theorem 16.3.** CollectEvidence and DistributeEvidence obeys Message Passing Protocol.

**Proof.**
16.6 Marginals