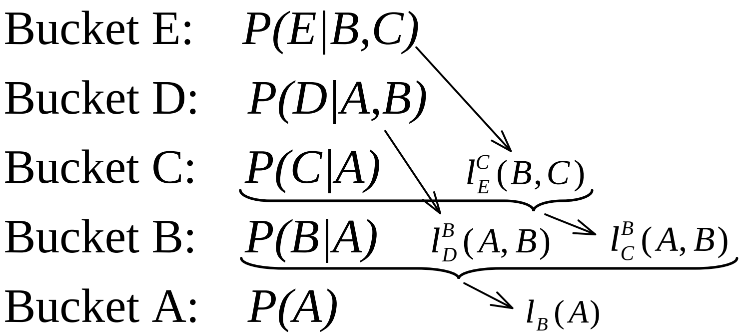
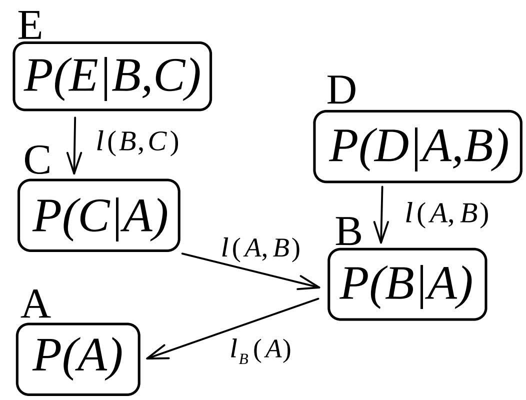
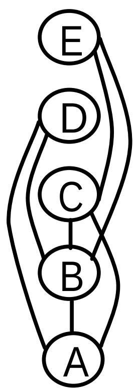
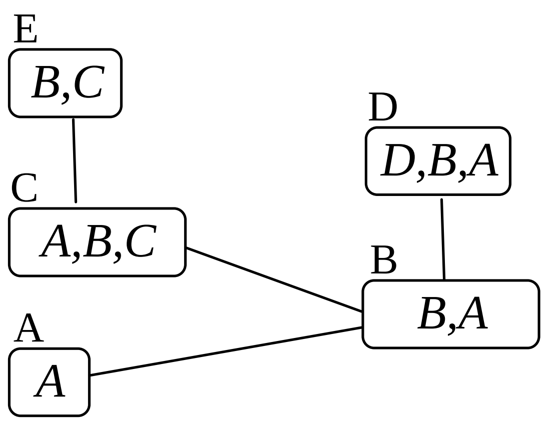
From Bucket-Elimination To Bucket Trees







T=

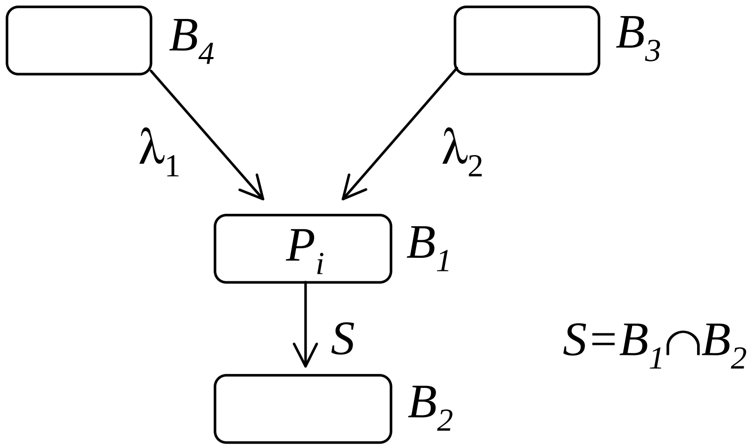


Definition: T is a bucket tree.

Theorem: T is an i-map of G.

- Variable-elimination can be viewed as message-passing (elimination) using a rooted bucket tree.
- Any variable (bucket) can be the root.

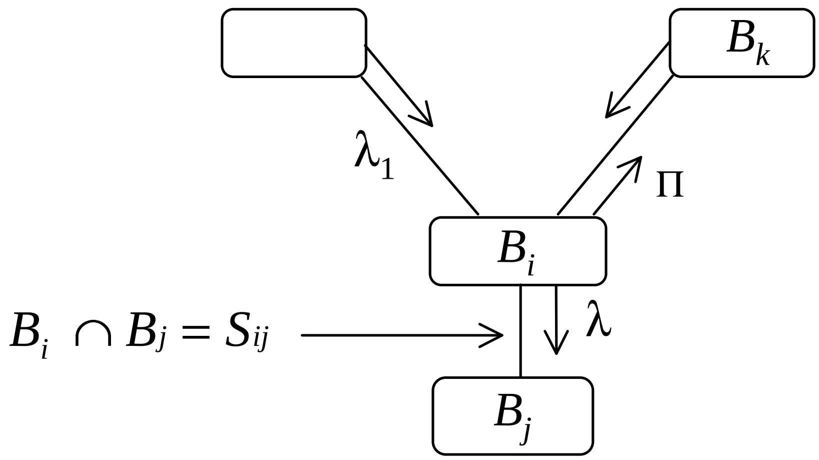
Generalization:Eliminate (sum over) Variables Not in Separators



- Multiply all incoming messages, and P_i*'s* in the bucket and sum over $B_1 \cap B_2$.
- $l_{B_1}^{B_2}(s) = \sum_{B_1 - s} (\Pi \ l_i) \bullet (\Pi \ P_i)$
- Given a rooted bucket tree, T, every node can be the "root" of the variables-elimination computation.
- If B_3 is the root, bucket B_2 and then Bucket B_1 should be processed; π-messages sent from B_2 to B_1 and from B_1 To B_3

Bucket Propagation Algorithm

- Input: A bucket tree $B_1 \ldots B_n$
- Output: For Each B_i and parent B_j, functions $\lambda_i^j(S_{ij})$ and $\pi_i^j(S_{ij})$ are exchanged.



Top Down:

- Let s $l_1 \ldots l_k$ messages from child nodes of B_i, $P_1 \ldots P_r$ in B_i original functions.
- $$l_i^j(S_{ij}) = \sum_{B_i - B_j} \prod_i l_i \bullet \prod_j P_j$$

Bottom Up:

- Let π^i_j be received from B_j.
- $$p_i^k(S_{ki}) = \sum_{B_k - B_i} (\prod_j P_j) \bullet p_j^i \bullet \prod_{i \neq k} l_i$$

- The belief of B_i
- $P(B_i) = \prod_i P_j \bullet \prod_i l_i \bullet p^i_j$
- if x index Bucket i
  get Bel(x) by summing out Bel(x) = $\alpha \sum_{S_{ij}} P(B_i)$

Propagation in a Bucket Tree

Definitions:

- Let G be a Bayesian network, d, an ordering and $B_1 \ldots B_n$ the final bucket created processing along $d = x_1 \ldots x_n$.
- Let B_i be the set of variables appearing in bucket i when it is processed.

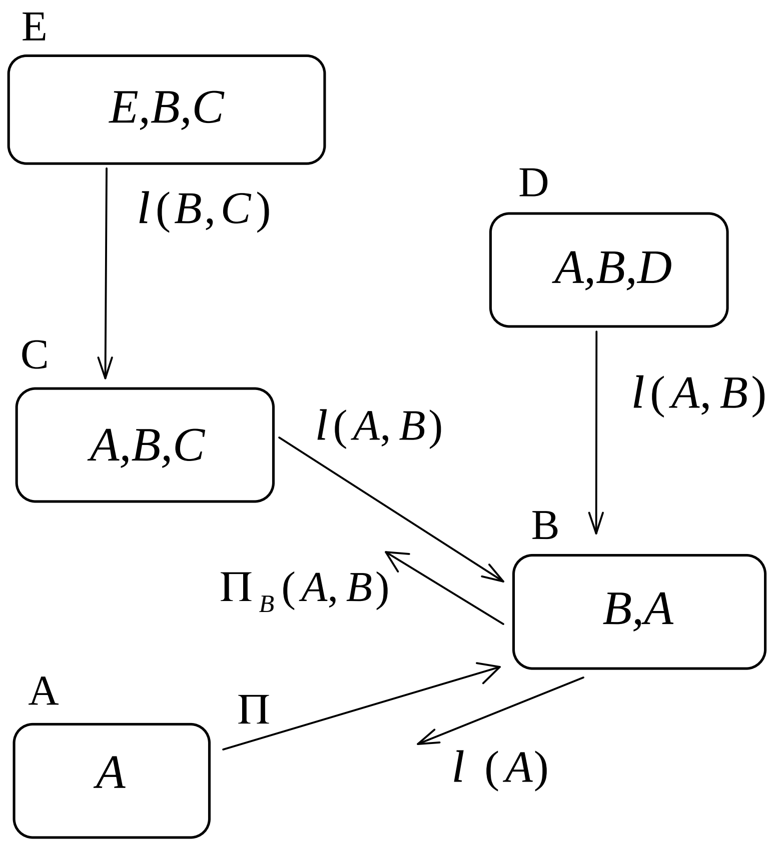
Bucket Tree:

- A bucket tree has each B_i cluster as a node and there is an arc from B_i to B_j if the function created at B_i was placed in B_j

Graph-Based Definition:

- Let G_d be the induced graph along d. Each variable x and it's earlier neighbors in a node, B_x. There is an arc from B_x to B_y if y is the closest parent of x.

Upwards Messages On The Bucket Tree



$$\Pi(A) = P(A)$$

$$\Pi_B^P(A,B) = P(B,A) \bullet l_C^B(A,B)$$

$$\Pi_B^C(A,B) = P(B,A) \bullet \Pi(A) \bullet l_D^B(A,B)$$

$$\Pi_C^E(B,C) = \sum_A P(C,A) \bullet \Pi_B^C(A,B)$$