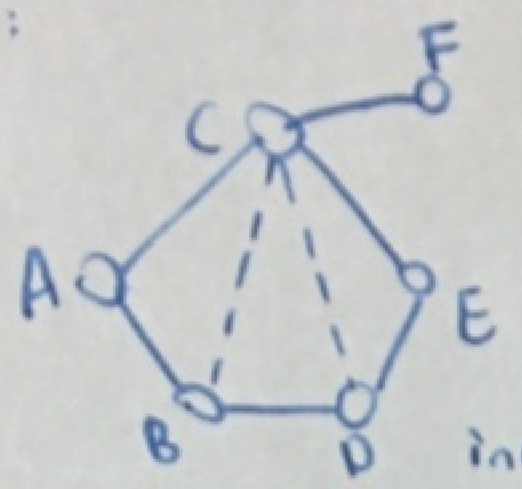
## Junction tree algorithm.

We start from <u>Variable Elimination</u>.

Example :



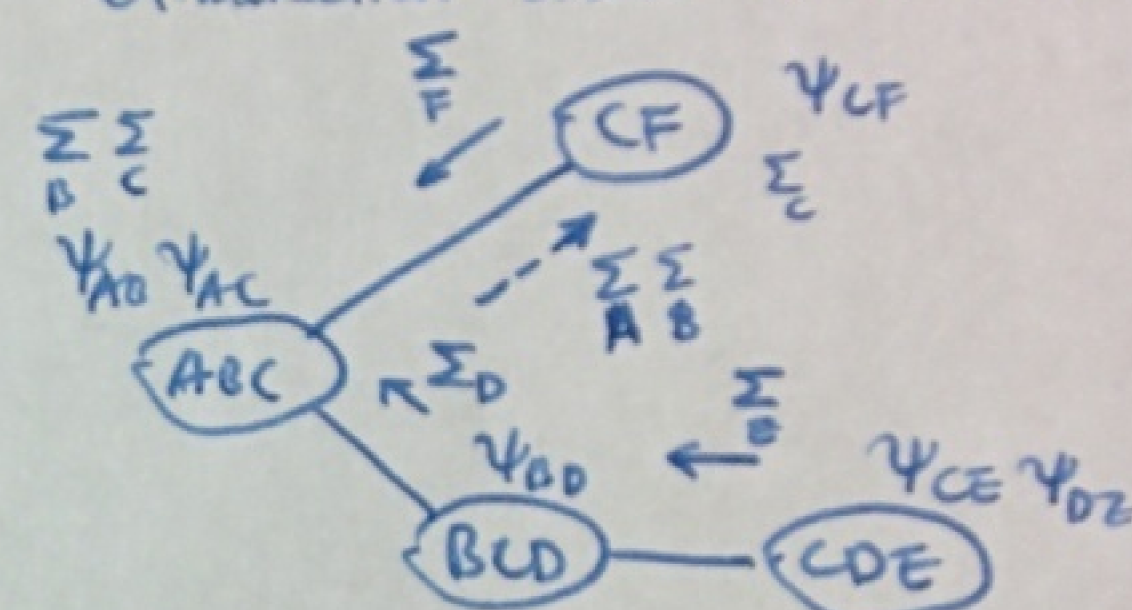

induced graph.

$$P(x) = \frac{1}{Z} \psi_{AB} \psi_{AC} \psi_{BD} \psi_{DE} \psi_{CE} \psi_{CF}.$$

If we want marginal distribution $P(A)$ elimination order F-E-D-C-B

$$P(A) \propto \sum_B \psi_{AB} \sum_C \psi_{AC} \underbrace{\sum_F \psi_{CF}}_{\psi'_C} \underbrace{\sum_D \psi_{BD} \underbrace{\sum_E \psi_{CE} \psi_{DE}}_{\psi'_{CD}}}_{\psi'_{BC}}$$

$\rightarrow$




$P(F)$ : E - D - B - A - C.

reuse partial sums.

$$\sum_C \psi_{CF} \sum_A \psi_{AC} \sum_B \psi_{AB} \underbrace{\sum_D \psi_{BD} \underbrace{\sum_E \psi_{CE} \psi_{DE}}_{\psi'_{CD}}}_{\psi'_{BC}}$$

Junction tree : data structure that can store partial computations, which can be reused to make computation of multiple queries efficient.

$\rightarrow$ no need to redo elimination each time.

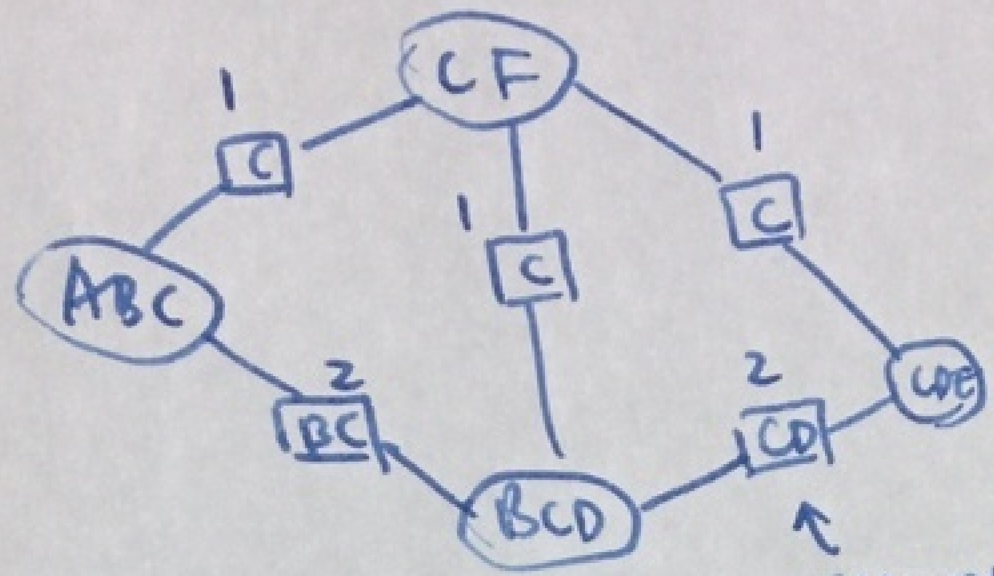
Once we get a clique tree, simply run sum-product on it.

$$[S = C \cap C',\ S'' = C \cap C''] \quad m_{C \to C'}(S) = \sum_{C \setminus S} \psi_C \prod_{C'' \in N(C) \setminus C'} m_{C'' \to C}(S'')$$

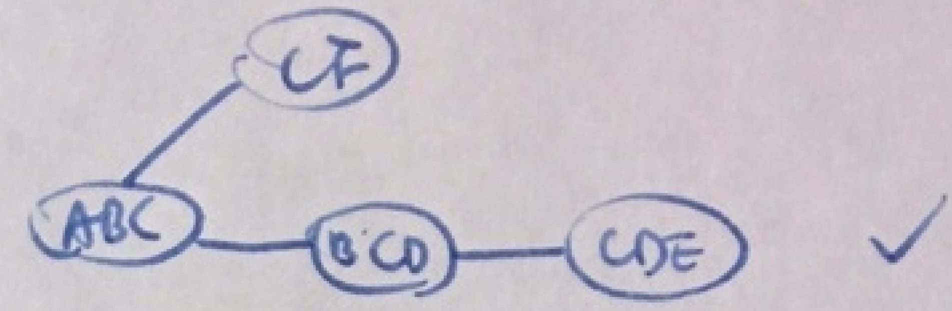
marginals :

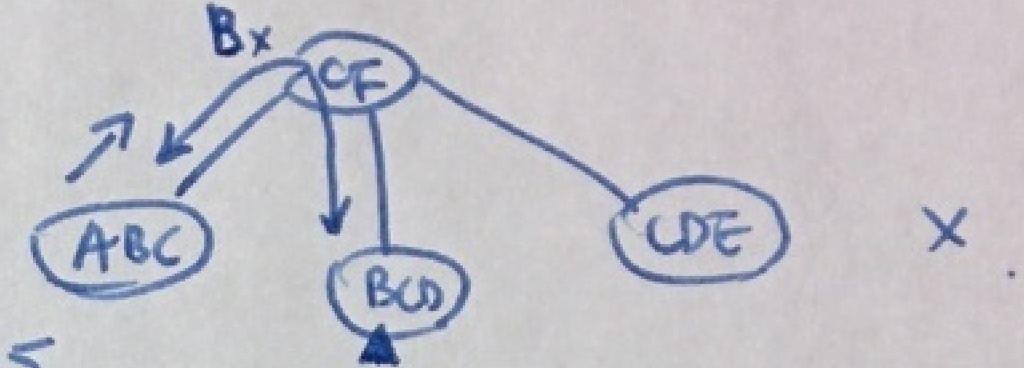
$$p(C) \propto \psi_C \prod_{C' \in N(C)} m_{C' \to C}(S).$$

There could be multiple clique trees, not all of them junction trees.




separator sets.




$\sum_A \sum_B$ ?

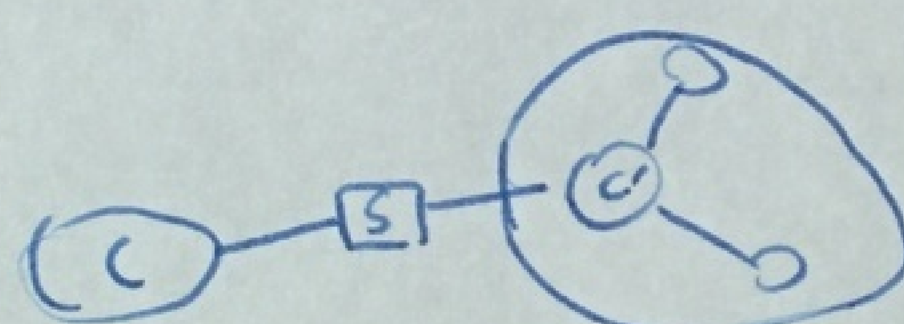
when passing messages cannot safely sum out variables.

☆ Junction tree property / Running intersection property.

For any clique nodes $C, C'$ in a clique tree, $S = C \cap C'$ must be contained in every node on the path connecting $C$ and $C'$.

**Def** Junction tree: clique trees that have this property.

<u>why useful?</u>




$R = C \setminus S$ only appears in $C$.

So guarantees safe sum over $R$.

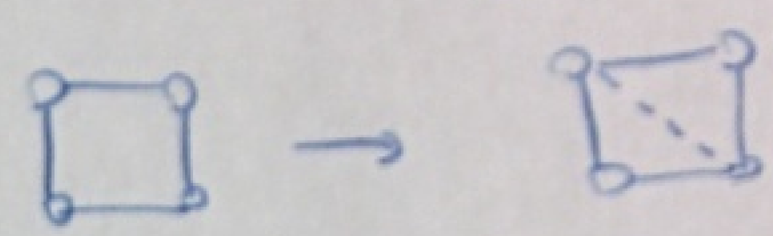
<u>How to get a junction tree?</u> – Maximum weight spanning tree.

Specifying an initial elimination order is also unnecessary.

More generally, the operation of adding fill edges to get the induced graph is called "triangulation"

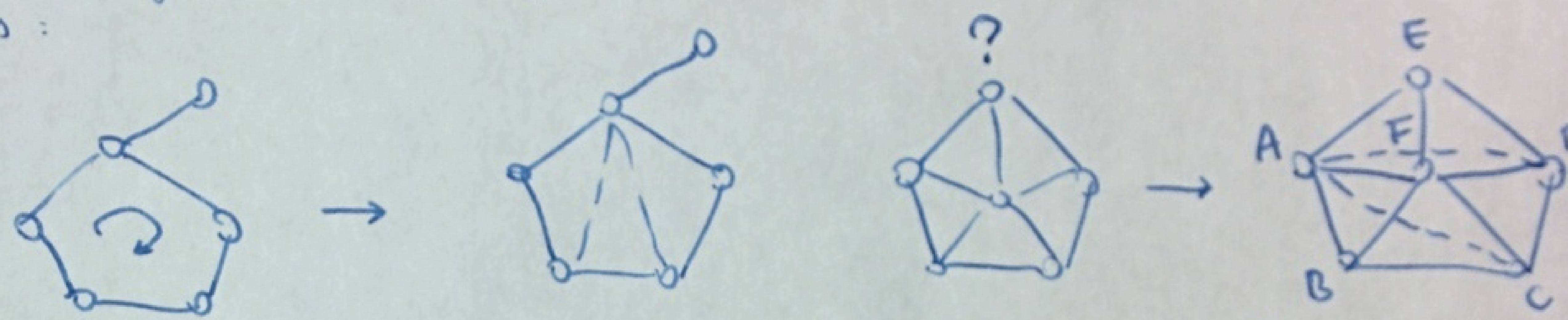
→ adding edges so that any cycle of length $\geq 4$ will have at least a chord.

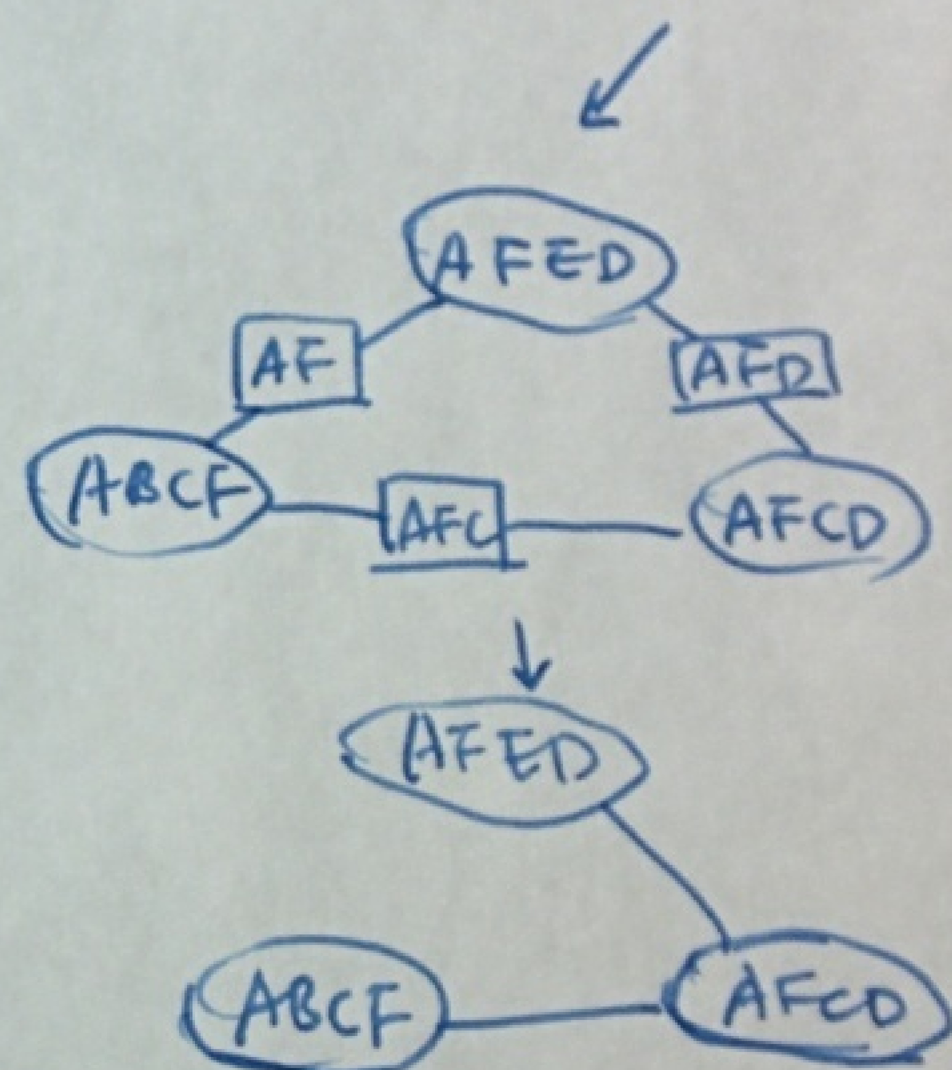
→ after this, the only possible chordless cycles are triangles



→ elimination guarantees a triangulated graph.

Examples:




Summary:

1. triangulation, find maximal cliques.
2. build junction tree.
3. initialize clique potentials.
4. run sum-product on junction tree.

Complexity:

1. 2. 3. Only need to do once for a given graph, can be reused for any distribution that factorizes according to this graph.

4. complexity at most $O(NK^{w+1})$.

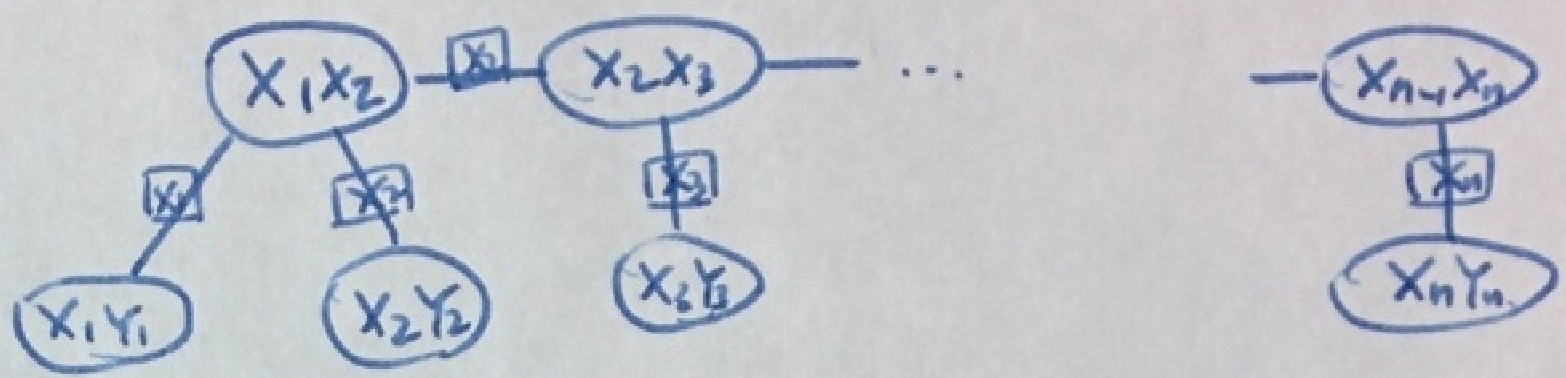
N: # cliques.
K: # states for each variable
w: tree-width.

Example: HMM: $P(X, Y) = P(x_1) \prod_{i=2}^{n} P(x_i \mid x_{i-1}) \prod_{i=1}^{n} P(y_i \mid x_i)$

$$\Downarrow$$

$$\prod_{i=2}^{n} \psi_{i,i-1}(x_i, x_{i-1}) \prod_{i=1}^{n} \psi_i(x_i, y_i)$$

already a junction tree. ←




$\psi_{12}(x_1, x_2) = P(x_1) P(x_2 \mid x_1)$ $\qquad$ $\psi_{i,i-1}(x_i, x_{i-1}) = P(x_i \mid x_{i-1})$, $i > 2$.

$\psi_i(x_i, y_i) = P(y_i \mid x_i)$.

Assume condition on $Y$, we want marginals of $P(X \mid Y)$ → simply clamp $Y$ to the specified value and everything else the same.

"Forward pass"

$$m_{X_1Y_1 \to X_1X_2}(x_1) = \psi_1(x_1, y_1) = P(y_1 \mid x_1)$$

$$m_{X_iY_i \to X_{i-1}X_i}(x_i) = \psi_i(x_i, y_i) = P(y_i \mid x_i).$$

$$\begin{aligned} m_{X_1X_2 \to X_2X_3}(x_2) &= \sum_{x_1} \psi_{X_1X_2}(x_1, x_2)\, m_{X_1Y_1 \to X_1X_2}(x_1)\, m_{X_2Y_2 \to X_1X_2}(x_2) \\ &= \sum_{x_1} P(x_1) P(x_2 \mid x_1) \cdot P(y_1 \mid x_1)\, P(y_2 \mid x_2) \\ &= \sum_{x_1} P(x_1, x_2, y_1, y_2) \\ &= P(x_2, y_1, y_2) \propto P(x_2 \mid y_1, y_2) = P(x_2 \mid y_{1:2}). \end{aligned}$$

$$\cancel{m_{X_iX_i \to X_iX_{i+1}}(x_i) = \sum_{x_{i-1}} \psi_{X_{i-1}X_i}(x_{i-1}, x_i)\, m_{X_{i+1}Y_{i+1} \to X_iX_{i+1}}(x_{i+1})}$$
$$\cancel{m_{X_iX_i \to X_iX_{i+1}}(x_i)}$$

$$\begin{aligned} m_{X_{i-1}X_i \to X_iX_{i+1}}(x_i) &= \sum_{x_{i-1}} \psi_{X_{i-1}X_i}(x_{i-1}, x_i) \cdot m_{X_{i-2}X_{i-1} \to X_{i-1}X_i}(x_{i-1})\, m_{X_iY_i \to X_{i-1}X_i}(x_i) \\ &= \sum_{x_{i-1}} P(x_i \mid x_{i-1})\, P(x_{i-1} \mid y_{1:i-1})\, P(y_i \mid x_i) \\ &= \sum_{x_{i-1}} P(x_i, x_{i-1}, y_i \mid y_{1:i-1}) \\ &= P(x_i, y_i \mid y_{1:i-1}) \propto P(x_i \mid y_{1:i}). \end{aligned}$$

Other variants and more discussions on junction trees can be found in

D. Koller & N. Friedman. Probabilistic Graphical Models: principles & techniques.

M. Jordan & C. Bishop. A Introduction to Graphical Models (draft).