# CS 583: Probabilistic Graphical Models

# Topic: Junction Tree Algorithm

**Mustafa Bilgic**

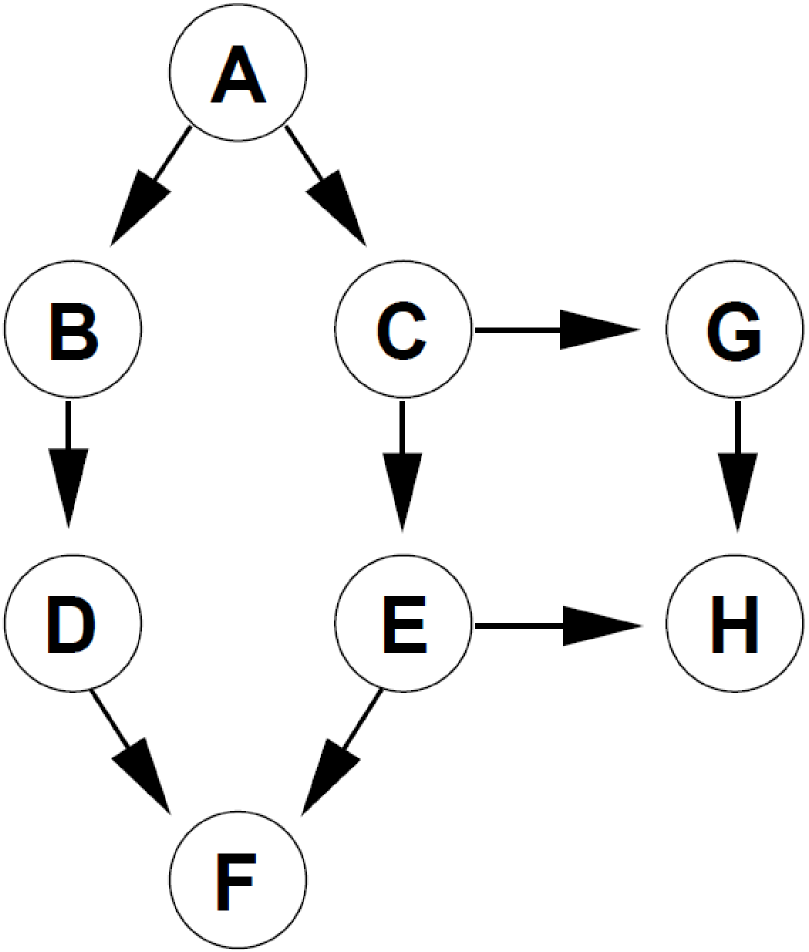
http://www.cs.iit.edu/~mbilgic

https://twitter.com/bilgicm

# MOTIVATION

- We are interested in multiple marginal/conditional probabilities
- In variable elimination, we define our target upfront and then eliminate the others
- If we need probabilities for other variables, there is no apparent way of reusing shared computations
- In the student example, assume that I'm interested in P(G) and P(L). What are some of the shared computations?

# Example



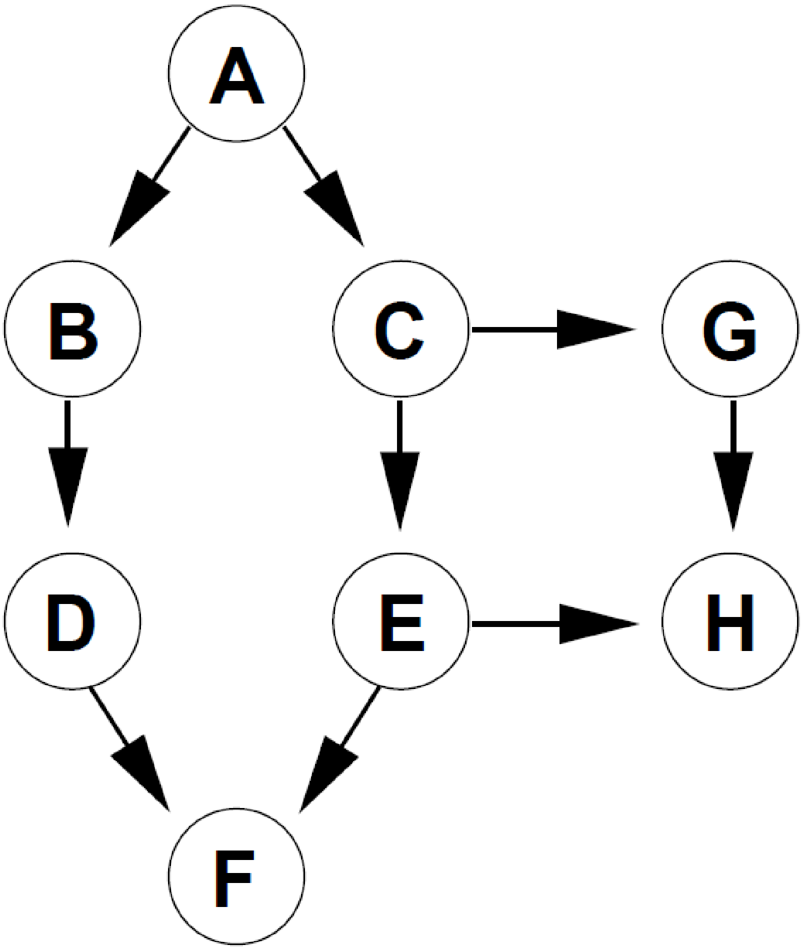

Calculate P(H) using variable elimination

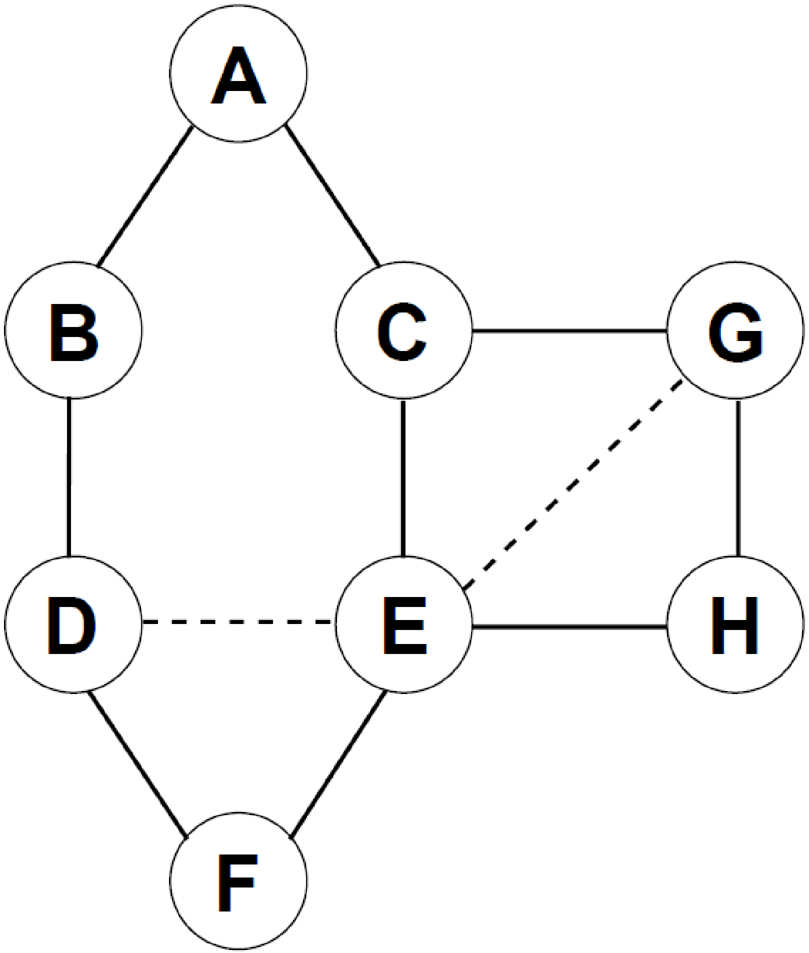
Now, calculate P(G) using variable elimination

# Variable elimination as graph transformation

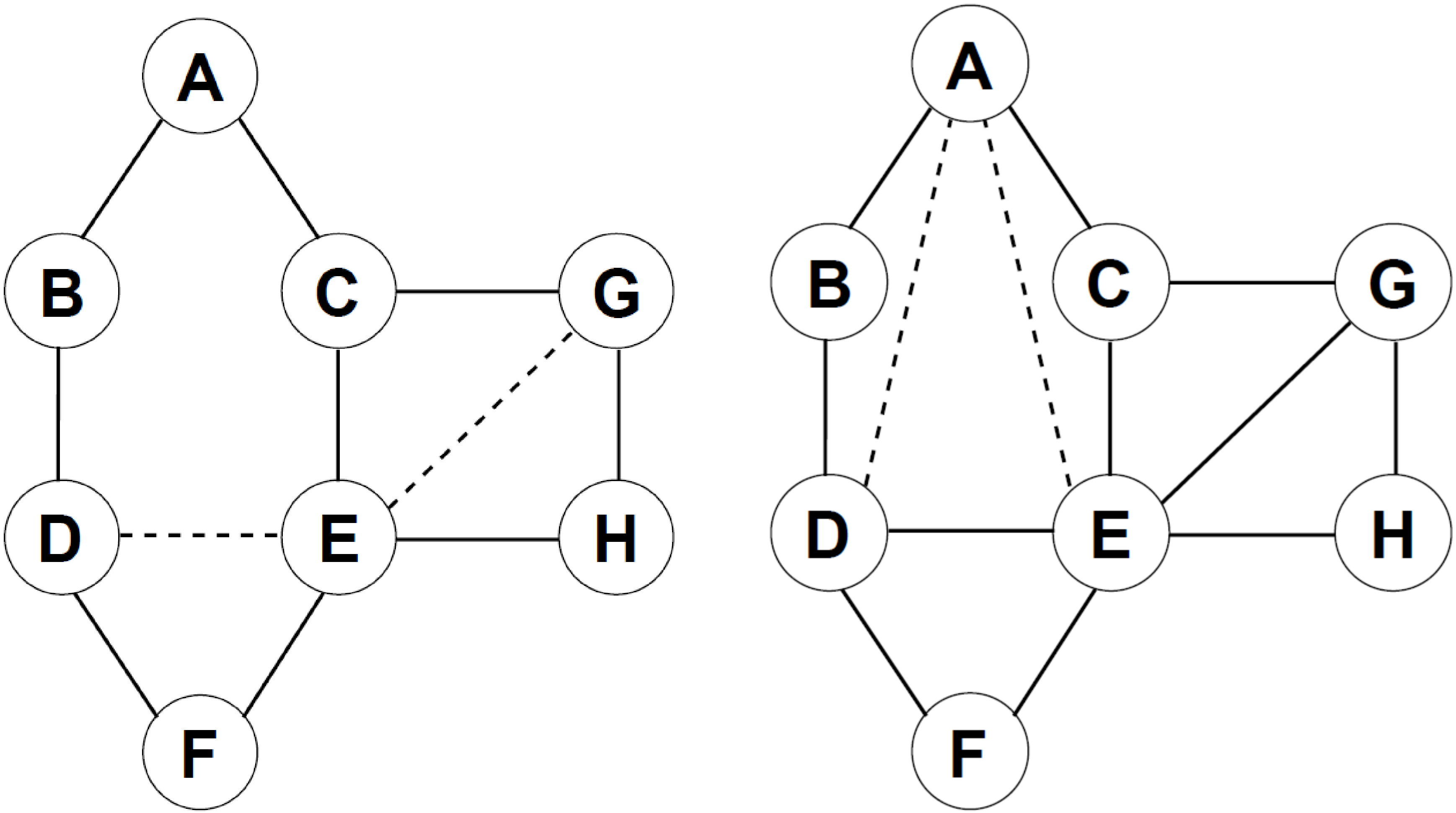
- First, construct the moral graph
- Then, eliminate variables so that each elimination introduces the fewest number of edges
- Take note of the factors

# Example - moralize



# Elimination order: H, G, F, C, B, D, E, A



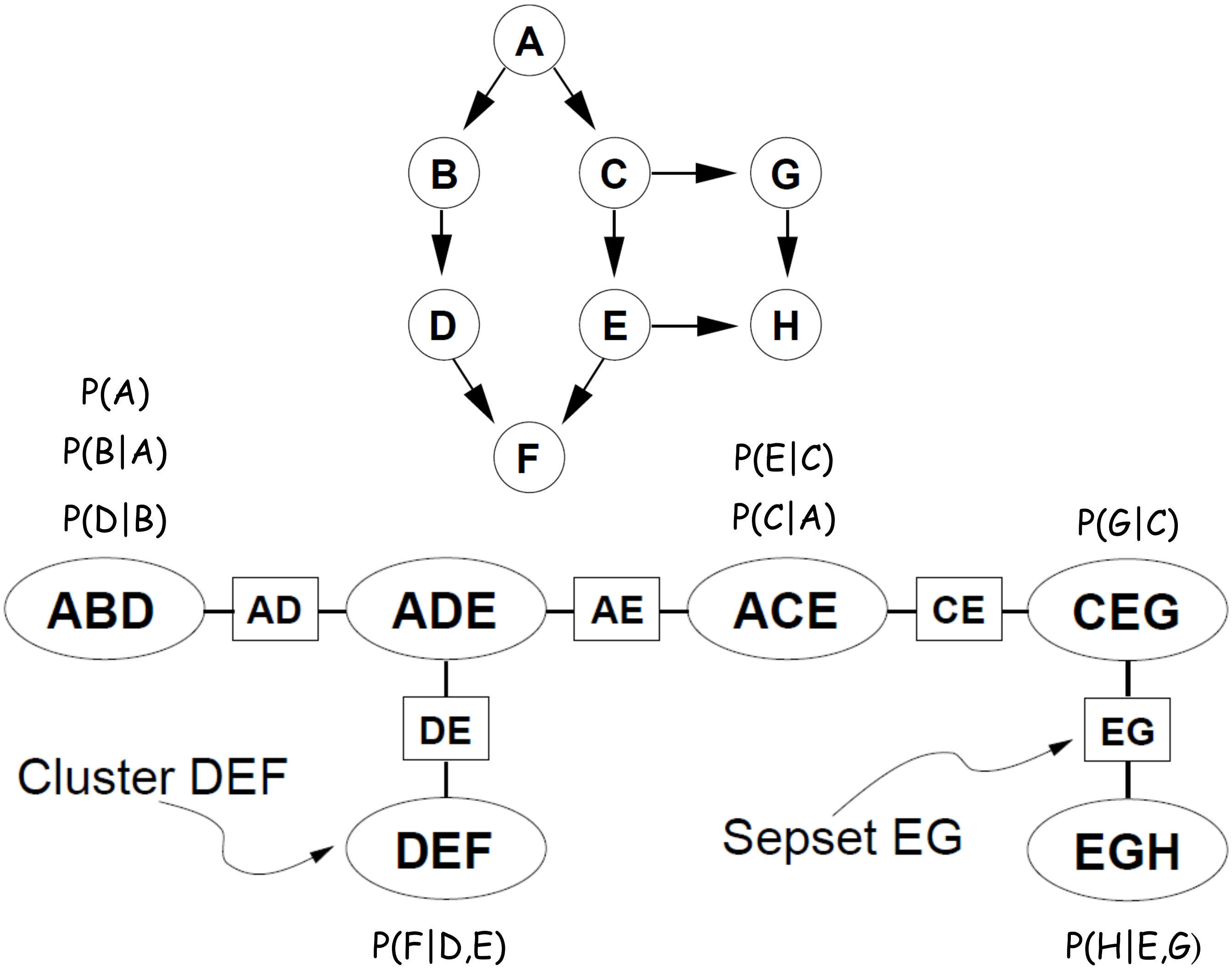
# CLUSTER GRAPH

- A *cluster graph* $\mathcal{U}$ for a set of factors $\Phi$ over $\mathcal{X}$ is an undirected graph, each of whose nodes are associated with a cluster $\boldsymbol{C}_i \subseteq \mathcal{X}$.
- A cluster graph must be *family preserving* – each factor $\phi \in \Phi$ must be associated with a cluster $\boldsymbol{C}_i$, denoted as $\alpha(\phi)$, such that Scope$[\phi] \subseteq \boldsymbol{C}_i$.
- Each edge between a pair of clusters $\boldsymbol{C}_i$ and $\boldsymbol{C}_j$ is associated with a *sepset* $\boldsymbol{S}_{ij} \subseteq \boldsymbol{C}_i \cap \boldsymbol{C}_j$

# Running intersection property

- Let $\mathcal{T}$ be a cluster tree. $\mathcal{T}$ has *running intersection property* if, whenever there is a variable $X$ such that $X \in \boldsymbol{C}_i$ and $X \in \boldsymbol{C}_j$, then $X$ is also in every cluster in the unique path in $\mathcal{T}$ between $\boldsymbol{C}_i$ and $\boldsymbol{C}_j$.
- A cluster tree that satisfies the running intersection property is also called the *join/clique/junction tree*.
- **Theorem**: A cluster tree obtained through a run of variable elimination satisfies the running intersection property; that is, it is a clique tree.

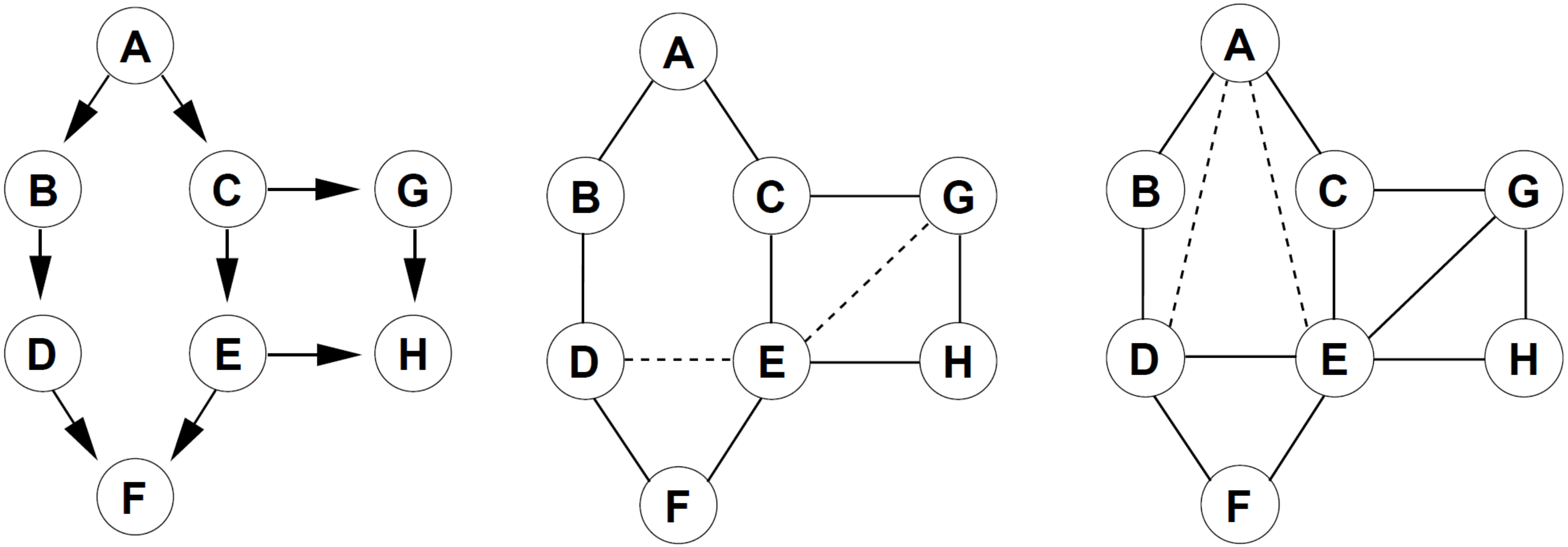
# EXAMPLE CLIQUE TREE

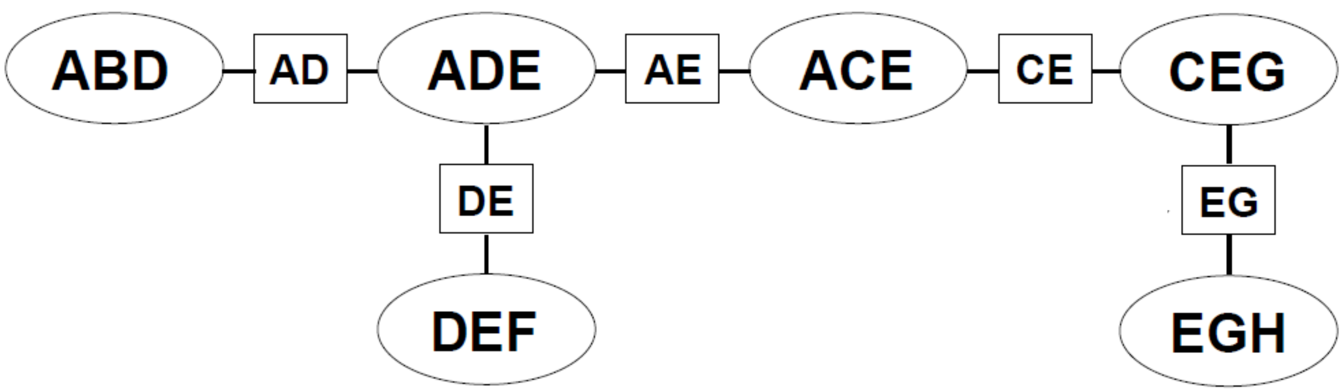


# CONSTRUCT A CLIQUE TREE

1. Moralize the graph
2. Pick a variable elimination order
3. Eliminate the variables, noting the maximal cliques
4. The cliques are the nodes of the tree
5. Until a tree is formed (i.e., n-1 edges are added)
    a) Connect two disconnected components by a maximal size sepset

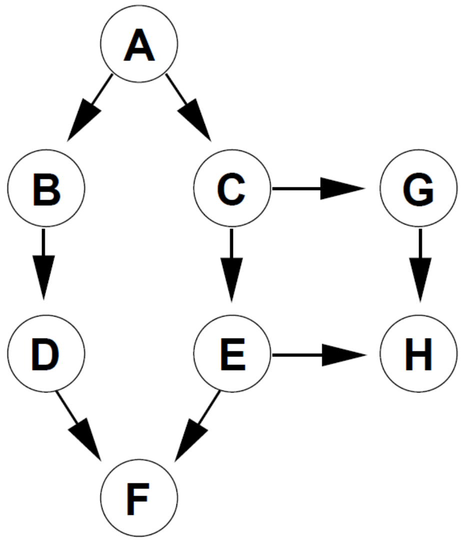
# Elimination order: H, G, F, C, B, D, E, A

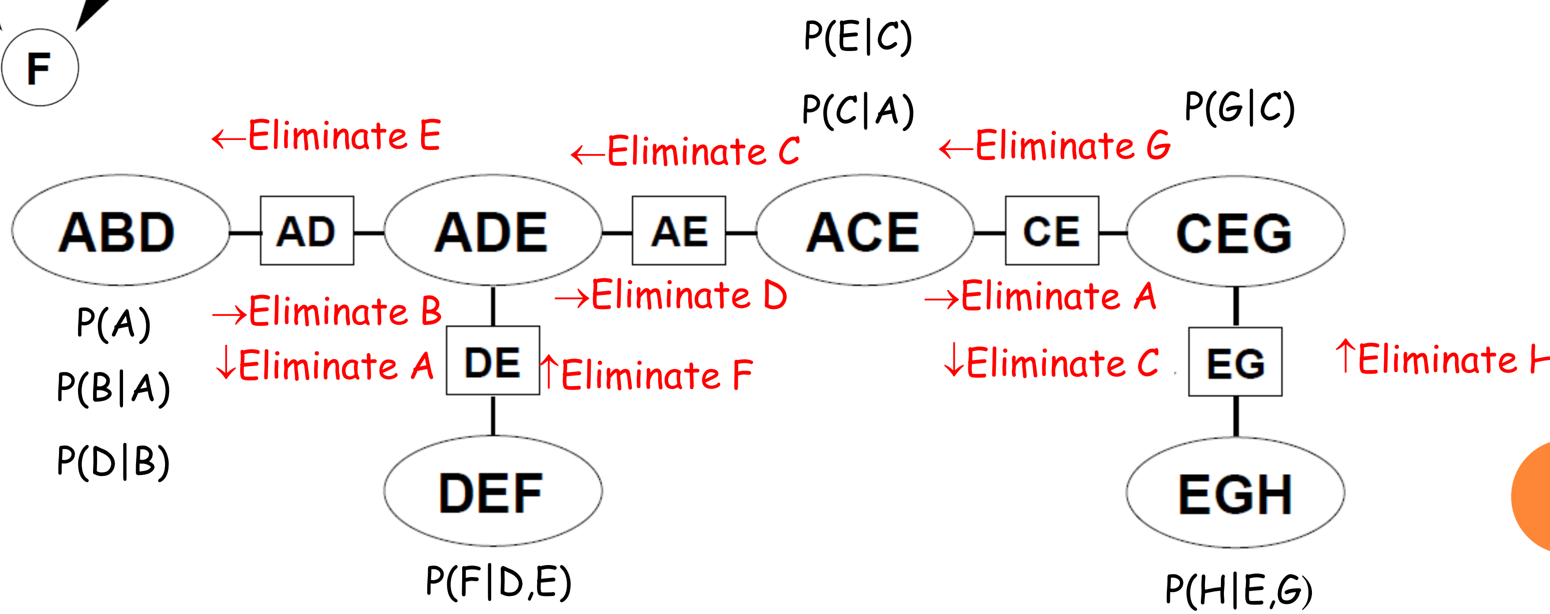



11

# VARIABLE ELIMINATION ON JUNCTION TREE



# MESSAGE PASSING ON JUNCTION TREE

- Clusters receive from and send messages to its neighbors
- Each message pass consists of elimination of one or more variables
- A cluster $\boldsymbol{C}_i$ is ready to send a message to its neighbor $\boldsymbol{C}_j$, when it receives messages from its *all other* neighbors
- A message from $\boldsymbol{C}_i$ to $\boldsymbol{C}_j$ is computed as follows
  - $\boldsymbol{C}_i$ multiples all the factors assigned to it, and all the messages it received from its *other* neighbors
  - It sums out $\boldsymbol{C}_i \setminus \boldsymbol{S}_{ij}$

# A MESSAGE

$$\delta_{i\to j} = \sum_{C_i \setminus S_{ij}} \left( \left( \prod_{\phi:\alpha(\phi)=i} \phi \right) \times \left( \prod_{k\in(Nb_i - \{j\})} \delta_{k\to i} \right) \right)$$

# Belief

$$\beta_i = \left( \prod_{\phi:\alpha(\phi)=i} \phi \right) \times \left( \prod_{k \in Nb_i} \delta_{k \to i} \right)$$