Exact Inference

Marián Képesi
Signal Processing and Speech Communication Lab.

9. May. 2005

- **Outline:**
-
- Notation conventions
- Belief networks
- Secondary structures
- Integration of evidence
- Optimization opportunities
- Conclusions

Notation conventions

- Variables .. (A, B, C)
- values .. (a, b, c)
  - a is an instantination of A

- Sets of variables: .. $(\boldsymbol{A}, \boldsymbol{B}, \boldsymbol{C})$
  - their instantinations $(\mathbf{a}, \mathbf{b}, \mathbf{c})$
  - $\mathbf{x}$ is an instattination of $\mathbf{X}$

- Probability distributions $$\sum_{\mathbf{x}} P(\mathbf{x}) = 1$$

- Conditional probabilities $$\sum_{\mathbf{x}} P(\mathbf{x} \mid \mathbf{y}) = 1$$

· Notation conventions

- Potentials .. defined over a set of variables **X,**
  - a function that maps each instantiation **x** into a nonnegative real number

  1. Operation on potentials: Marginalization

The **marginalization** of $\phi_{\mathbf{Y}}$ into $\mathbf{X}$ is a potential $\phi_{\mathbf{X}}$, where each $\phi_{\mathbf{X}}(\mathbf{x})$ is computed as follows:

1. Identify the instantiations $\mathbf{y_1}, \mathbf{y_2}, \ldots$ that are consistent with $\mathbf{x}$.

2. Assign to $\phi_{\mathbf{X}}(\mathbf{x})$ the sum $\phi_{\mathbf{Y}}(\mathbf{y_1}) + \phi_{\mathbf{Y}}(\mathbf{y_2}) + \ldots$

This marginalization is denoted as follows:

$$\phi_{\mathbf{X}} = \sum_{\mathbf{Y} \setminus \mathbf{X}} \phi_{\mathbf{Y}}.$$

Belief networks

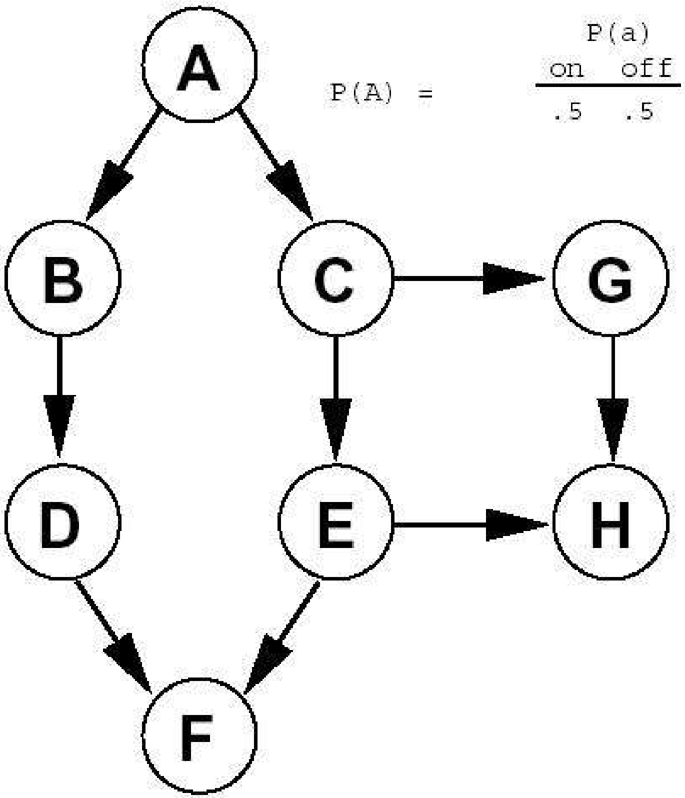
- 2. Operation on potentials: Multiplication

$$\phi_{\mathbf{Z}} = \phi_{\mathbf{X}} \phi_{\mathbf{Y}}$$

Example: U= $\{A, B, .., G, H\}$
each having values {on, off}

Example of probabilistic inference:
Compute the probability that A = on,
given the knowledge that C = on and E = off.

Belief networks



P(A) =

P(a) on	P(a) off
.5	.5

P(B|A) =

a	P(b|a) on	P(b|a) off
on	.5	.5
off	.4	.6

P(C|A) =

a	P(c|a) on	P(c|a) off
on	.7	.3
off	.2	.8

P(D|B) =

a	P(d|b) on	P(d|b) off
on	.9	.1
off	.5	.5

P(E|C) =

a	P(e|c) on	P(e|c) off
on	.3	.7
off	.6	.4

P(F|DE) =

d	e	P(f|de) on	P(f|de) off
on	on	.01	.99
on	off	.01	.99
off	on	.01	.99
off	off	.99	.01

P(G|C) =

c	P(g|c) on	P(g|c) off
on	.8	.2
off	.1	.9

P(H|EG) =

g	h	P(h|eg) on	P(h|eg) off
on	on	.05	.95
on	off	.95	.05
off	on	.95	.05
off	off	.95	.05

Example of a directed acyclic graph (DAG).

PPTC

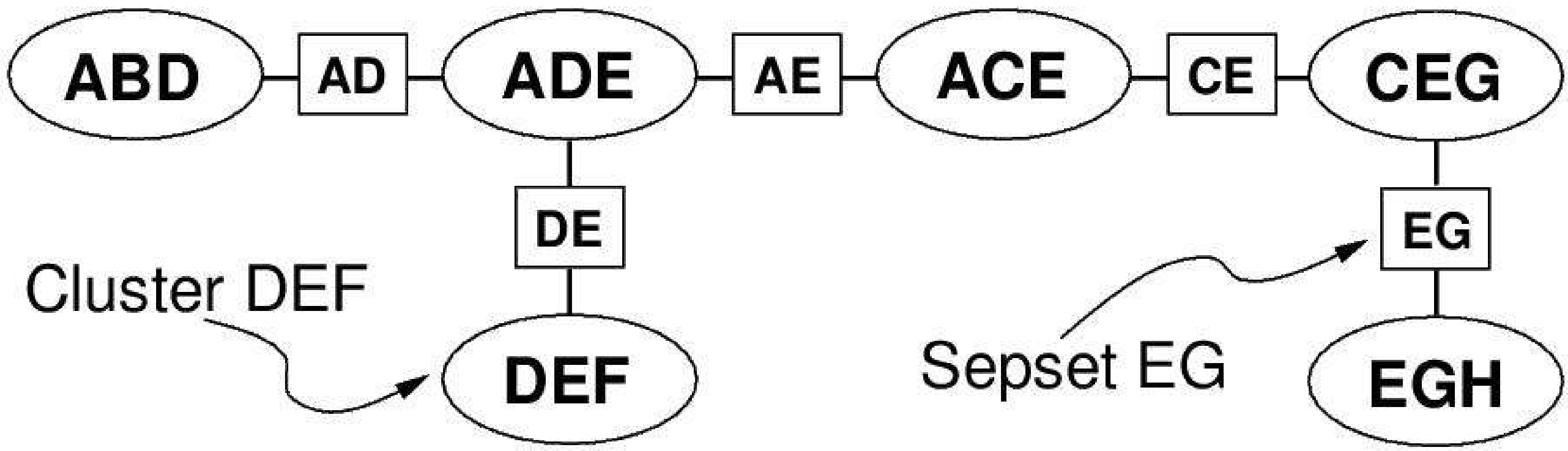
Probability Propagation in Trees of Clusters (PPTC)

(Lauritzen, Spiegelhatlter - 1988, Jensen-1990)

PPTC is a method for performing probabilistic inference on a belief network.

In general, probabilistic inference on a belief network is the process of computing $P(V = v \mid \mathbf{E} = \mathbf{e})$, or simply $P(v \mid \mathbf{e})$, where v is a value of a variable V and $\mathbf{e}$ is an assignment of values to a set of variables $\mathbf{E}$ in the belief network.

Secondary Structures



$\phi_{ABD} =$

a	b	d	$\phi_{ABD}(abd)$
on	on	on	.225
on	on	off	.025
on	off	on	.125
on	off	off	.125
off	on	on	.180
off	on	off	.020
off	off	on	.150
off	off	off	.150

$\phi_{AD} =$

a	d	$\phi_{AD}(ad)$
on	on	.35
on	off	.15
off	on	.33
off	off	.17

etc.

Example of a secondary struct.

Secondary Structures

Experts typically use belief networks to encode their domain, but PPTC performs probabilistic inference on a secondary structure

Secondary structure contains a graphical and a numerical component

Graphical:

- Each node in “Tau” is a **cluster.**
- Each edge in “Tau” is labeled with the intersection of the adjacent clusters; these labels are called separator sets, or **sepsets.**

Numerical:

- described using the notion of a **belief potential**. A belief potential is a function that maps each instantiation of a set of variables into a real number

Secondary structure

Each cluster **X** is associated with a belief potential fi_**x,**
Each sepset **S** is associated with a belief potential fi_**s.**

!! Belief potentials are not arbitrarily specified; they must satisfy the following constraints:

a) fi_**s** is consistent:

$$\sum_{\mathbf{X} \setminus \mathbf{S}} \phi_{\mathbf{X}} = \phi_{\mathbf{S}} \longrightarrow \phi_{AD} = \sum_{B} \phi_{ABD}$$

b)The belief potentials encode the joint distribution *P*(**U**) of the belief network

$$P(\mathbf{U}) = \frac{\prod_i \phi_{\mathbf{X_i}}}{\prod_j \phi_{\mathbf{S_j}}}, \longrightarrow P(\mathbf{U}) = \frac{\phi_{ABD}\,\phi_{ACE}\,\phi_{ADE}\,\phi_{CEG}\,\phi_{DEF}\,\phi_{EGH}}{\phi_{AD}\,\phi_{AE}\,\phi_{CE}\,\phi_{DE}\,\phi_{EG}}$$

A key step in PPTC is the construction of a secondary structure that satisfies the above constraints

Secondary structure

...then for each **cluster** holds: fi_**x** = *P* (**X**)
therefore we can compute the probab. distr. of any variable:

$$P(V) = \sum_{\mathbf{X}\setminus\{\mathbf{V}\}} \phi_{\mathbf{X}}$$

secondary structure in the **literature**:
join tree, junction tree, tree of belief universes, cluster tree, and clique tree.

we use the term **join tree** to refer to the graphical component, and the term **join tree potential** to refer generically to a cluster or sepset belief potential. We will also use the term join tree to refer to the entire secondary structure.

Building join trees

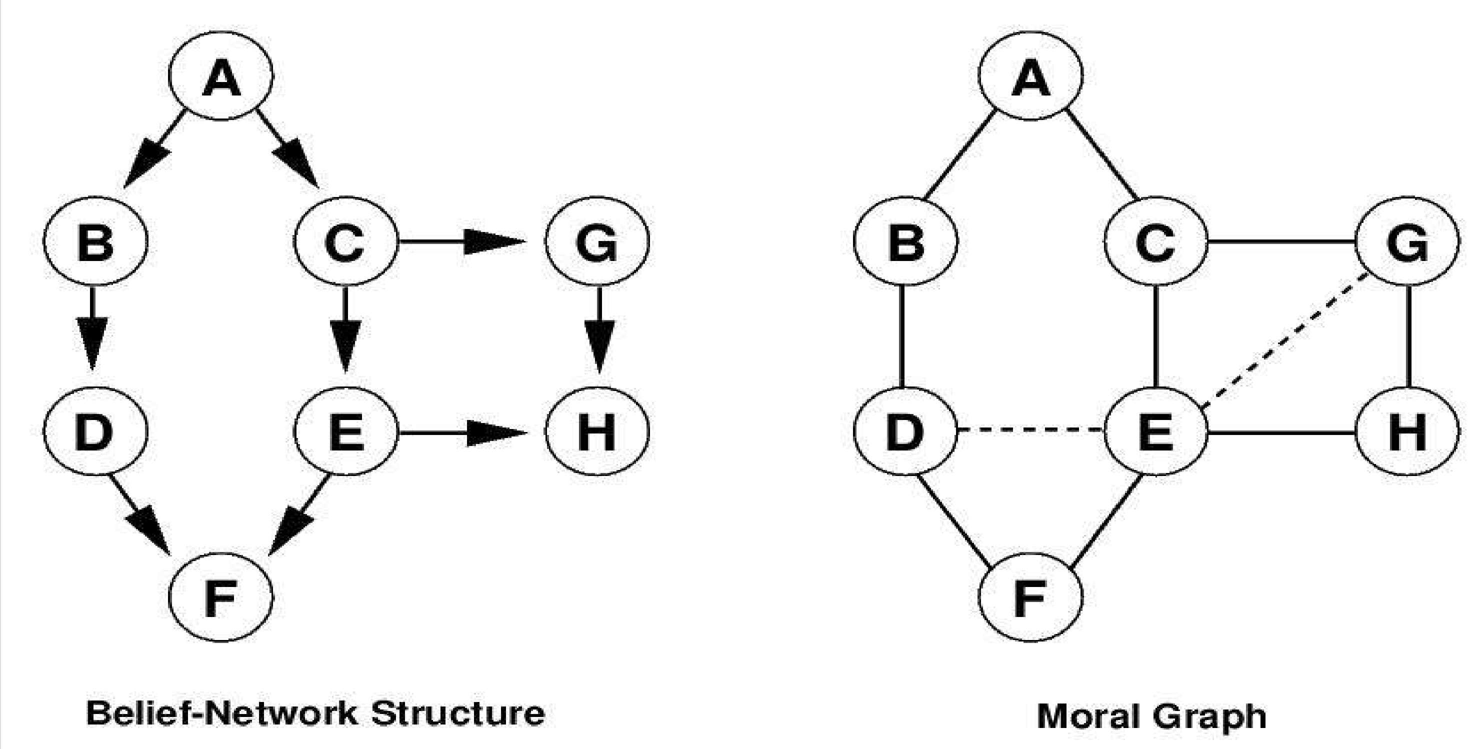
We begin with the DAG of a belief network, and apply a series of graphical transformations:

1. Construct an undirected graph, called a **moral graph**, from the DAG.
2. Selectively add arcs to the moral graph to form a **triangulated graph**.
3. From the triangulated graph, identify select subsets of nodes, called **cliques**.
4. Build a join tree, starting with the cliques as clusters: connect the clusters to form an undirected tree satisfying the join tree property, inserting the appropriate sepsets.

Steps 2 and 4 are nondeterministic; consequently, many different join trees can be built from the same DAG.

Building join trees

Moral Graph:

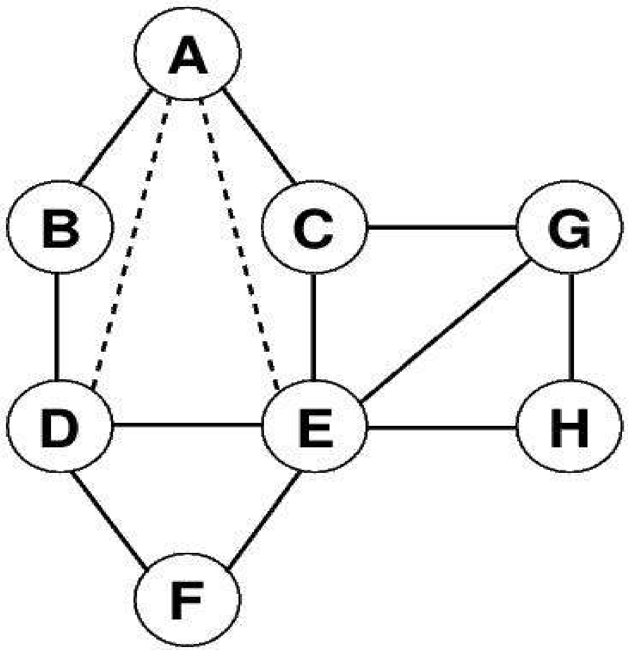


.

12

Building join trees

Triangulated Graph:



Triangulated Graph

Eliminated Vertex	Induced Cluster	Edges Added
H	EGH	none
G	CEG	none
F	DEF	none
C	ACE	(A, E)
B	ABD	(A, D)
D	ADE	none
E	AE	none
A	A	none

Elimination Ordering

- The criterion for selecting nodes to remove is now stated as follows: *Choose the node that causes the least number of edges to be added in Step 2b, breaking ties by choosing the node that induces the cluster with the smallest weight.*[7]

Building join trees

Identifying cliques:

A clique in an undirected graph G is a **subgraph of** G that is complete and maximal. **Complete** means that every pair of distinct nodes is connected by an edge. **Maximal** means that the clique is not properly contained in a larger, complete subgraph.

It can be done during the triangulation phase by saving **each induced cluster that is not a subset** of any previously saved cluster.

Revisiting our example:,
cliques of the triangulated graph: EGH, CEG, DEF, ACE, ABD, and ADE..

.

Building join trees

Building an **optimal tree**:

0. one single clique = **one tree**., *n trees*.
1. create candidate sepsets (*n*-1)
2. select them based on the criterion below..
3. Insert the sepset $\mathbf{S}_{XY}$ between the cliques **X** and **Y** only if **X** and **Y** are on different trees in the forest.

For the resulting clique tree *to satisfy the join tree property*, we must choose the candidate **sepset with the largest mass**.

When two or more sepsets of equal mass can be chosen, we can *optimize the inference time* on the resulting join tree by breaking the tie as follows: choose the candidate **sepset with the smallest cost**.

.

Building join trees

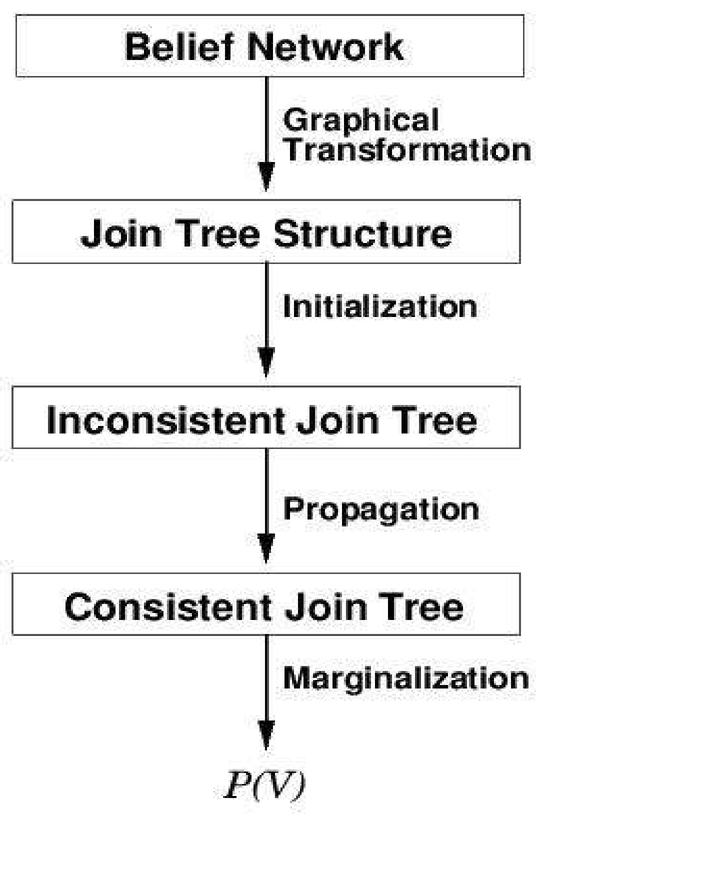
Building an **optimal tree**:

- The **mass** of a sepset $\mathbf{S_{XY}}$ is the number of variables it contains, or the number of variables in $\mathbf{X} \cap \mathbf{Y}$.
- The **cost** of a sepset $\mathbf{S_{XY}}$ is the weight of $\mathbf{X}$ plus the weight of $\mathbf{Y}$, where *weight* is defined as follows:
  - The **weight** of a variable V is the number of values of V.
  - The **weight** of a set of variables $\mathbf{X}$ is the product of the weights of the variables in $\mathbf{X}$.

If an **empty candidate sepset** is created, the search is terminated.

Principles of inference

We provide procedures for computing the join tree's numerical component, so that it satisfies the conditions



Block diagram of PPTC with no evidence.

Note:computing $P(V)$ corresponds to probabilistic inference in the context of **no evidence**.

We address later the more general problem of computing $P(V|\mathbf{e})$, in the context of evidence $\mathbf{e}$.

Principles of inference

I. Initialization:

1. For each cluster and sepset $\mathbf{X}$, set each $\phi_{\mathbf{X}}(\mathbf{x})$ to 1:

$$\phi_{\mathbf{X}} \longleftarrow 1.$$

2. For each variable V, perform the following: Assign to V a cluster $\mathbf{X}$ that contains $\mathbf{F_V}$;[11] call $\mathbf{X}$ the **parent cluster of $\mathbf{F_V}$**. Multiply $\phi_{\mathbf{X}}$ by $P(V \mid \mathbf{\Pi}_V)$:

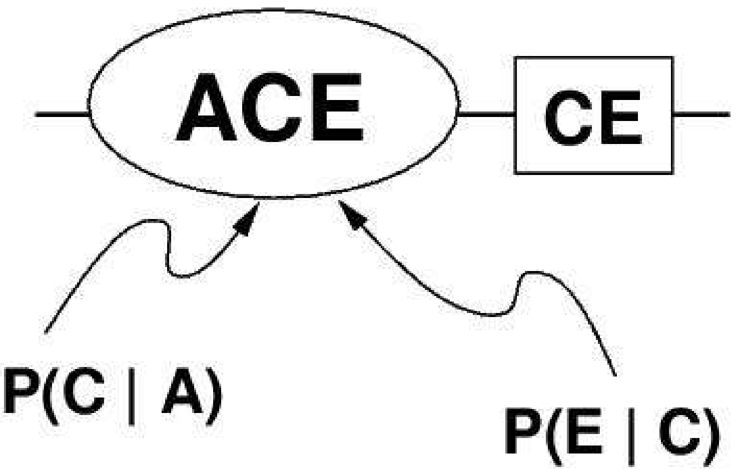
$$\phi_{\mathbf{X}} \longleftarrow \phi_{\mathbf{X}} P(V \mid \mathbf{\Pi}_V).$$

init. satisfies the joint distrib.: $$\frac{\prod_{i=1}^{N} \phi_{\mathbf{X_i}}}{\prod_{j=1}^{N-1} \phi_{\mathbf{S_j}}} = \frac{\prod_{k=1}^{Q} P(V_k \mid \mathbf{C}_{V_k})}{1} = P(\mathbf{U})$$

Principles of inference

I. Initialization:

..of cluster ACE and stepset CE:



P(C|A) =

a	P(c|a) on	P(c|a) off
on	.7	.3
off	.2	.8

P(D|B) =

a	P(d|b) on	P(d|b) off
on	.9	.1
off	.5	.5

- P(E|C) =

a	P(e|c) on	P(e|c) off
on	.3	.7
off	.6	.4

ϕ_{ACE}

a	c	e	Initial Values
on	on	on	1 × .7 × .3 = .21
on	on	off	1 × .7 × .7 = .49
on	off	on	1 × .3 × .6 = .18
on	off	off	1 × .3 × .4 = .12
off	on	on	1 × .2 × .3 = .06
off	on	off	1 × .2 × .7 = .14
off	off	on	1 × .8 × .6 = .48
off	off	off	1 × .8 × .4 = .32

ϕ_{CE}

c	e	Initial Values
on	on	1
on	off	1
off	on	1
off	off	1

etc.

Principles of inference

II. Global propagation:

After initializing the join tree potentials we perform **global propagation** in order to make them *locally consistent* (slide 10).

Global propagation consists of a series of **local manipulations**, called **message passes**, that occur between a cluster **X** and a neighboring cluster **Y**.

Global propagation causes each cluster to pass a message **to each of its neighbors**; these message passes are **ordered** so that each message pass will preserve the consistency introduced by previous message passes. When global propagation is completed, each cluster-sepset pair is *consistent*, and the join tree is *locally consistent*. (pp.9)

Principles of inference

II. Global propagation:

Single message pass:

Consider two adjacent clusters **X** and **Y** with sepset **R**:

1. Projection. Assign a new table to $\mathbf{R}$, saving the old table:

$$\phi_{\mathbf{R}}^{old} \longleftarrow \phi_{\mathbf{R}}.$$

$$\phi_{\mathbf{R}} \longleftarrow \sum_{\mathbf{X}\setminus\mathbf{R}} \phi_{\mathbf{X}}. \qquad (1)$$

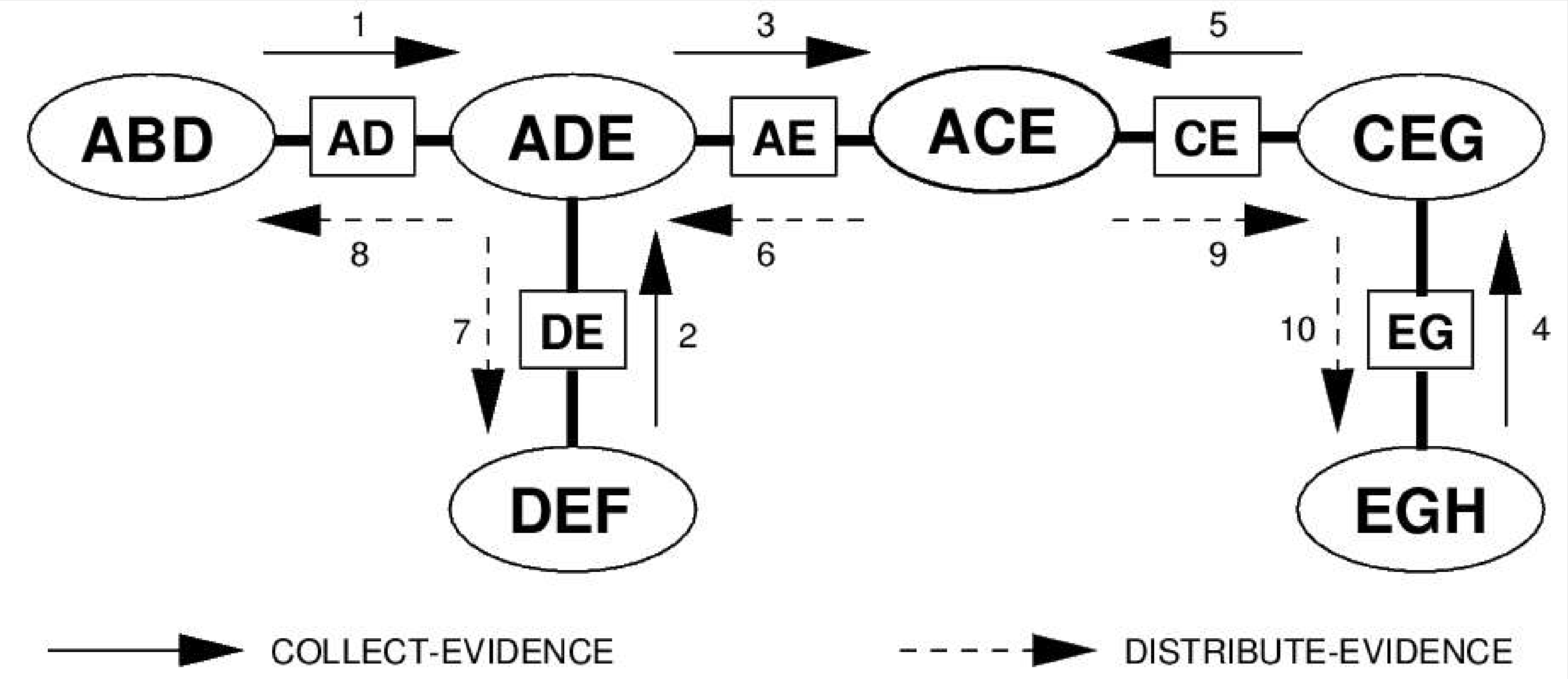
2. Absorption. Assign a new table to $\mathbf{Y}$, using both the old and the new tables of $\mathbf{R}$:

$$\phi_{\mathbf{Y}} \longleftarrow \phi_{\mathbf{Y}} \frac{\phi_{\mathbf{R}}}{\phi_{\mathbf{R}}^{old}}. \qquad (2)$$

.

Principles of inference

II. Global propagation - Multiple message pass
- choosing an arbitrary cluster **X**
- performing 2(n-1) message passes



! a cluster passes a message to a neighbor only after it has received messages from all of its other neighbors, assuring local consistency.

Principles of inference

III. Marginalization

once we have a consistent join tree, we can compute all $P(V)$:

1. Identify a cluster **X** that contains V
2. marginalize fi_**x**

$$P(V) = \sum_{\mathbf{X}\setminus\{\mathbf{V}\}} \phi_{\mathbf{X}}.$$

.

$\phi_{ABD} =$

a	b	d	$\phi_{ABD}(abd)$
on	on	on	.225
on	on	off	.025
on	off	on	.125
on	off	off	.125
off	on	on	.180
off	on	off	.020
off	off	on	.150
off	off	off	.150

$P(A) = \sum_{BD} \phi_{ABD} =$

a	P(a)
on	.225 + .025 + .125 + .125 = .500
off	.180 + .020 + .150 + .150 = .500

$P(D) = \sum_{AB} \phi_{ABD} =$

d	P(d)
on	.225 + .125 + .180 + .150 = .680
off	.025 + .125 + .020 + .150 = .320

Handling evidence

Observation – the simplest notion of evidence:

An observation is a statement of the form $V = v$. Collections of observations may be denoted by $\mathbf{E} = \mathbf{e}$, where e is the instantiation of the set of variables $\mathbf{E}$.

Observations are also referred to as **hard evidence**.

To encode observation for PPTC we define **Likelyhood** of V**:**

- If $V \in \mathbf{E}$—that is, if V is observed—then assign each $\Lambda_V(v)$ as follows:

$$\Lambda_V(v) = \begin{cases} 1, & \text{when } v \text{ is the observed value of } V \\ 0, & \text{otherwise} \end{cases}$$

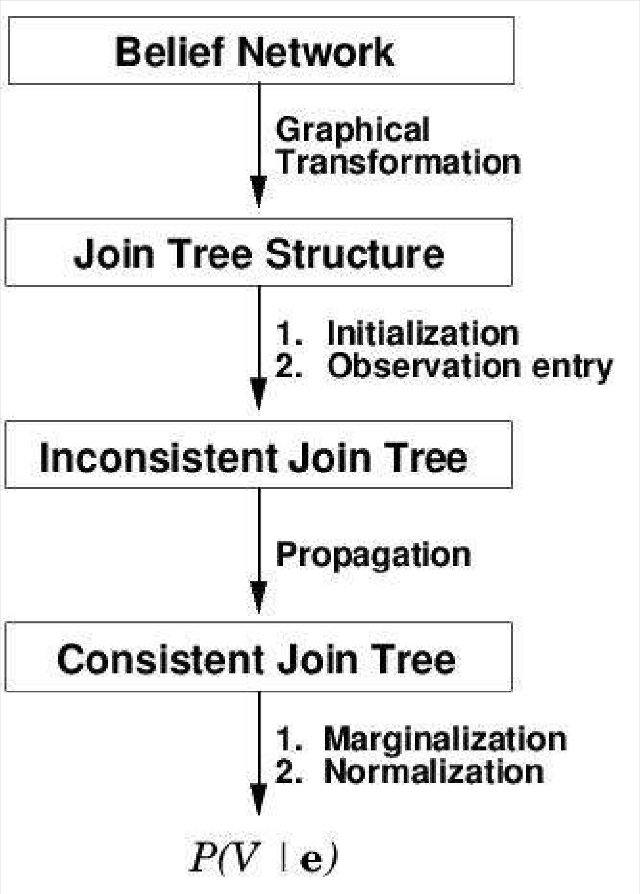
- If $V \notin \mathbf{E}$—that is, if the value of V is unknown—then assign $\Lambda_V(v) = 1$ for each value v.

.. when there are no observations, the likelihood of each variable consists of all 1's.

Handling evidence

We use likelihoods to encode the observations C = on and E = off (C and E are variables from the join tree):

Variable V	$\Lambda_V(v)$	
	$v = on$	$v = off$
A	1	1
B	1	1
C	1	0
D	1	1
E	0	1
F	1	1
G	1	1
H	1	1



.

Handling evidence

1. **Initialization with observation**: 1 extra step:

1. For each cluster and sepset $\mathbf{X}$, set each $\phi_{\mathbf{X}}(\mathbf{x})$ to 1:

$$\phi_{\mathbf{X}} \longleftarrow 1.$$

2. For each variable V:

(a) Assign to V a cluster $\mathbf{X}$ that contains $\mathbf{F_V}$; multiply $\phi_{\mathbf{X}}$ by $P(V \mid \mathbf{\Pi}_V)$:

$$\phi_{\mathbf{X}} \longleftarrow \phi_{\mathbf{X}} P(V \mid \mathbf{\Pi}_V).$$

(b) Set each likelihood element $\Lambda_V(v)$ to 1:

.

$$\Lambda_V \longleftarrow 1.$$

Handling evidence

2. **Observation Entry**:

Note: the likelihoods encode no observations.

We incorporate each observation $V = v$ by **encoding the observation as a likelihood**, and then incorporating this likelihood into the join tree, as follows:

1. Encode the observation $V = v$ as a likelihood Λ_V^{new}.

2. Identify a cluster $\mathbf{X}$ that contains V.[13]

3. Update $\phi_{\mathbf{X}}$ and Λ_V:

$$\phi_{\mathbf{X}} \longleftarrow \phi_{\mathbf{X}} \Lambda_V^{new}.$$
$$\Lambda_V \longleftarrow \Lambda_V^{new}.$$

Now instead of computing $P(\mathbf{X})$ and $P(V)$, we compute $P(\mathbf{X}; \mathbf{e})$ and $P(V; \mathbf{e})$, respectively.

.

Handling evidence

3. **Normalizations**:

After the join tree is made consistent through **global propagation**, we have, for each cluster (or sepset) **X**, fi**_x** = *P*(**X**; **e**).

Marginalizing a cluster potential fi**_x** into a variable *V:*

$$P(V, \mathbf{e}) = \sum_{\mathbf{X} \setminus \{\mathbf{V}\}} \phi_{\mathbf{X}}.$$

.. by normilizing this:

$$P(V \mid \mathbf{e}) = \frac{P(V, \mathbf{e})}{P(\mathbf{e})} = \frac{P(V, \mathbf{e})}{\sum_{V} P(V, \mathbf{e})}.$$

.

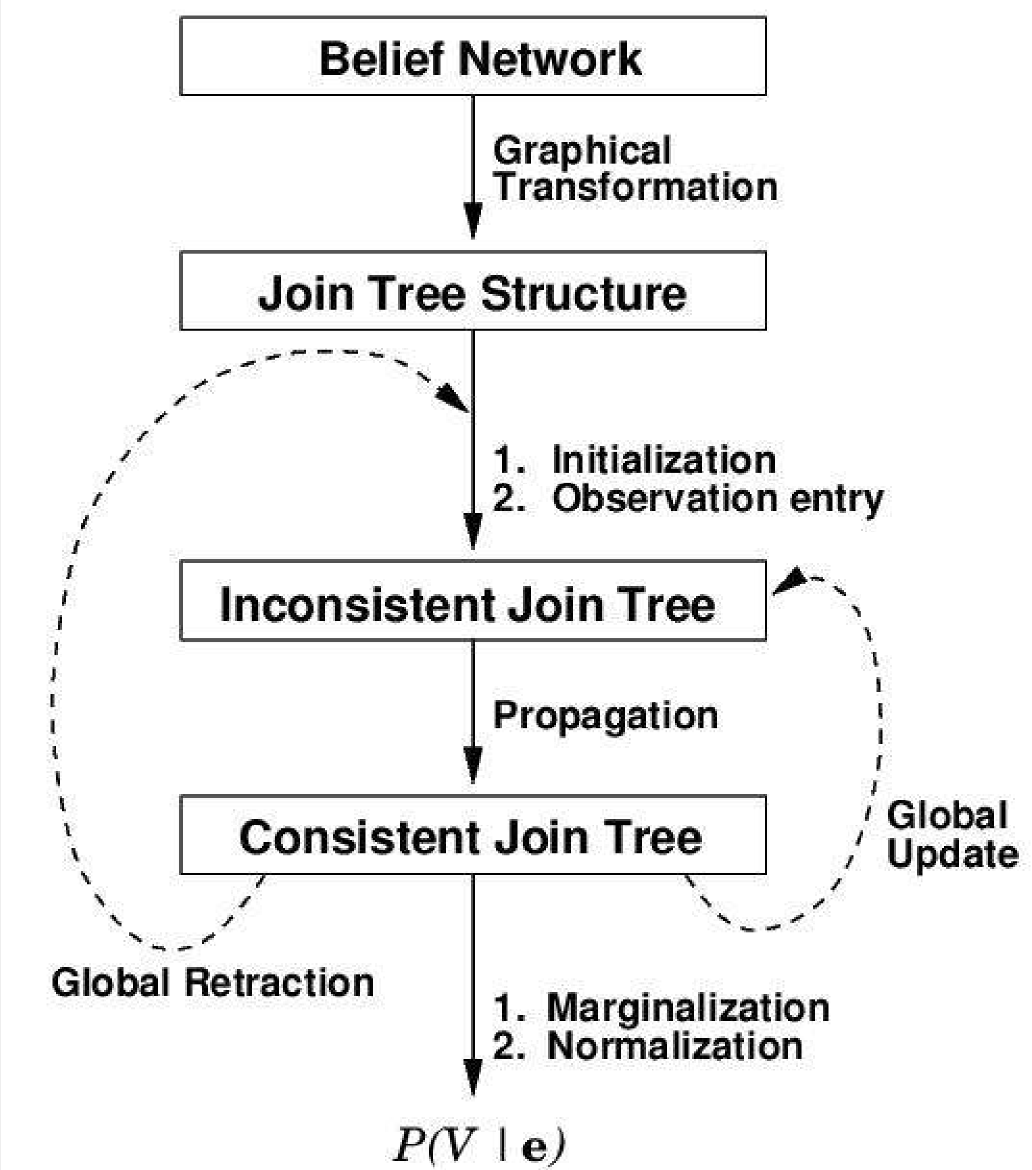
Handling evidence

Handling dynamic observations:

after computing $P(V \mid \mathbf{e}1)$,
we wish to compute $P(V\ \mathbf{j}\ \mathbf{e}2)$..

we can directly **modify the join tree potentials** in response to changes in the set of observations.

.



Fine

Thank you!

-

Identifying cliques:

-