Lecture 12: Uncertainty - 3

Victor Lesser

CMPSCI 683
Fall 2004

Outline

- Continuation of Inference in Belief Networks
- Automated Belief propagation in PolyTrees

d-separation: Direction-Dependent Separation

- **Network construction**
  - Conditional independence of a node and its predecessors, given its parents
  - The absence of a link between two variables does not guarantee their *independence*
- **Effective inference needs to exploit all available conditional independences**
  - Which set of nodes X are conditionally independent of another set Y, given a set of evidence nodes E
    - P(X,Y/E) = P(X/E) · P(Y/E)
  - Limits propagation of information
  - Comes directly from structure of network

d-separation

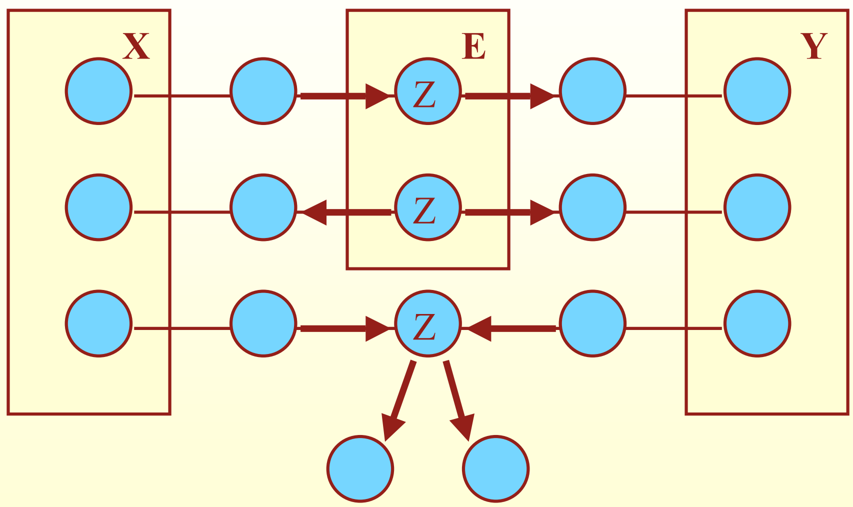
Definition: If X, Y and E are three disjoint subsets of nodes in a DAG, then E is said to d-separate X from Y if every undirected path from X to Y is blocked by E. A path is blocked if it contains a node Z such that:

(1) Z has one incoming and one outgoing arrow; or

(2) Z has two outgoing arrows; or

(3) Z has two incoming arrows and neither Z nor any of its descendants is in E.

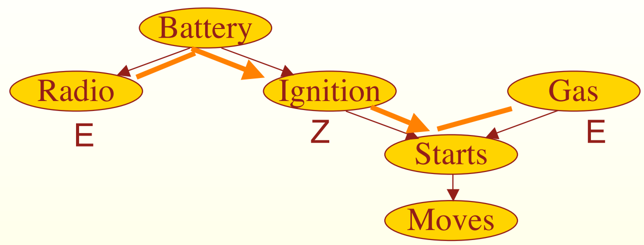
d-separation cont.



d-separation cont.

- **Property of belief networks**: if X and Y are d-separated by E, then X and Y are conditionally independent given E.
- An "if-and-only-if" relationship between the graph and the probabilistic model cannot always be achieved.

d-separation example- case 1

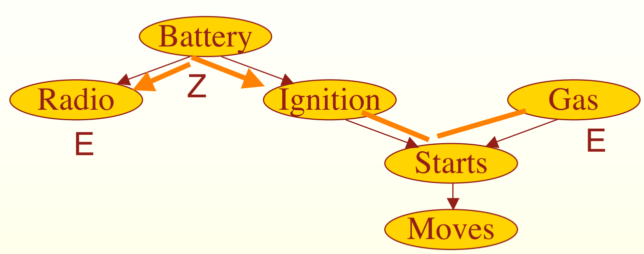


Whether there is *Gas* in the car and whether the car *Radio* plays are independent given evidence about whether the *SparkPlugs* fire [ignition] (***case 1***).

$$P(R,G/I) = P(R/I) \cdot P(G/I)$$

$$P(G/I,R) = P(G/I)$$

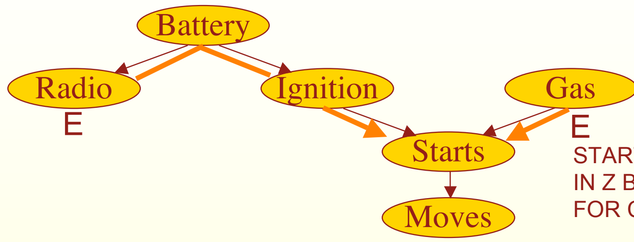
d-separation example- case 2



Gas and *Radio* are conditionally-independent if it is known if the battery works (***case2***).

$$P(R/B,G) = P(R/B); P(G/B,R)=P(G/B)$$

d-separation example - case 3



STARTS&MOVES IN Z BUT ALSO IN E FOR CASE 3

. *Gas* and *Radio* are independent given no evidence at all. But they are dependent given evidence about whether the car *Starts*. For example, if the car does not start, then the radio playing is increased evidence that we are out of gas. *Gas* and *Radio* are also dependent given evidence about whether the car *Moves*, because that is enabled by the car starting.

P(Gas/Radio)=P(Gas); P(Radio/Gas)=P(Radio)

P(Gas/ Radio,Start) not= P(Gas/Start)

Inference in Belief Networks

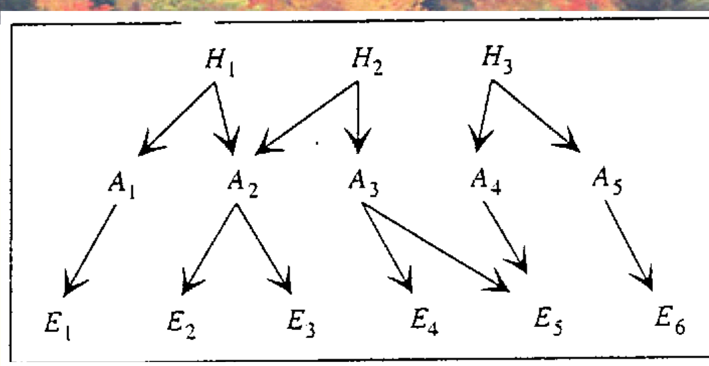
- BNs are fairly expressive and easily engineered representation for knowledge in probabilistic domains.
- They facilitate the development of inference algorithms.
- They are particularly suited for parallelization
- Current inference algorithms are efficient and can solve large real-world problems.

Network Features Affect Reasoning

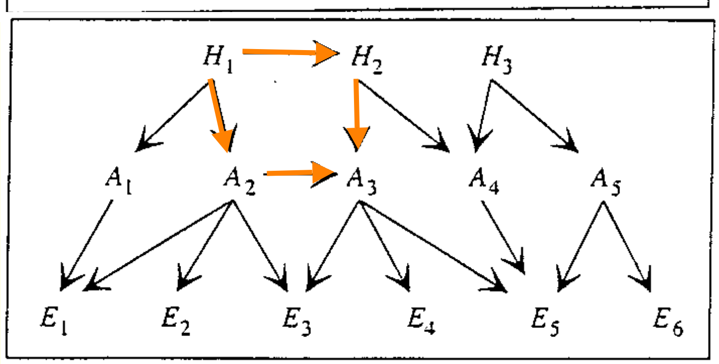
- Topology (trees, singly-connected, sparsely-connected, DAGs).
- Size (number of nodes).
- Type of variables (discrete, cont, functional, noisy-logical, mixed).
- Network dynamics (static, dynamic).

Belief Propagation in Polytrees

Polytree belief network, where nodes are singly connected
•Exact inference, Linear in size of network

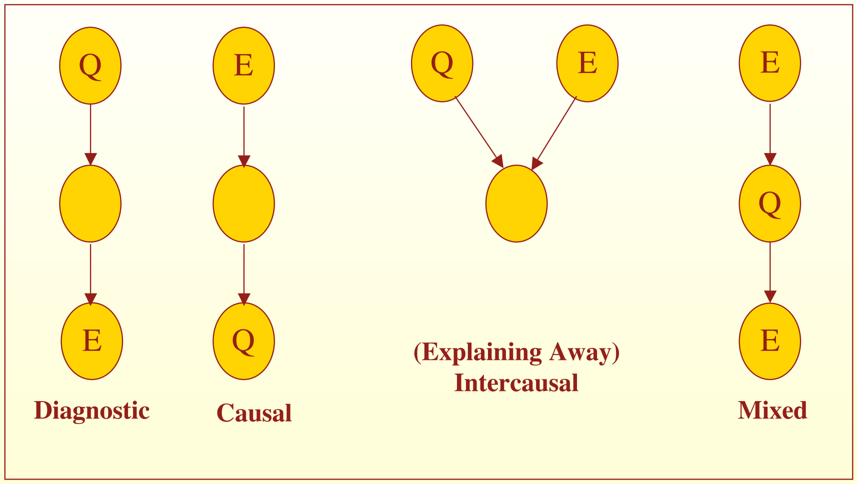


Multiconnected belief network. This is a DAG, but not a polytree.
•Exact inference, Worst case NP-hard



Reasoning in Belief Networks

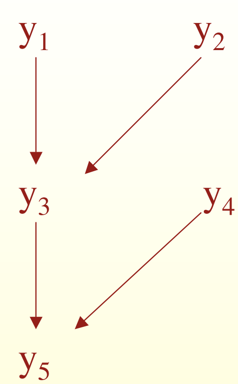
Simple examples of 4 patterns of reasoning that can be handled by belief networks. *E* represents an evidence variable; *Q* is a query variable.



P(Q/E) =?

Belief Network Calculation in Polytree: Evidence Above

- What is p(Y5|Y1,Y4)
  - **Define in terms of CPTs = p(Y5,Y4,Y3,Y2,Y1)**
  - **p(Y5|Y3,Y4p(Y4),p(Y3|Y1,Y2), p(Y2), p(Y1)**
  - **p(Y5|Y1,Y4)= p(Y5,Y1,Y4)/p(Y1,Y4)**
  - **Use cpt to sum over missing variables**
  - **p(Y5,Y1,Y4)= Sum(Y2,Y3) p(Y5,Y4,Y3,Y2,Y1)**
  - **assuming variables take on only truth or falsity.**

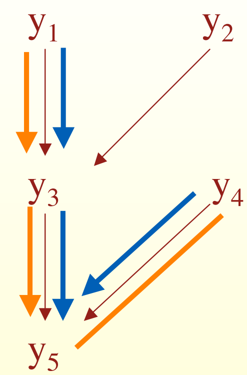


- p(Y5|Y1,Y4) = p(Y5,Y3|Y1, Y4) + P(Y5, not Y3|Y1,Y4)
  - **Connect to parents of Y5 not already part of expression, by marginalization**

- = SUM(Y3) p(Y5,Y3|Y1,Y4)

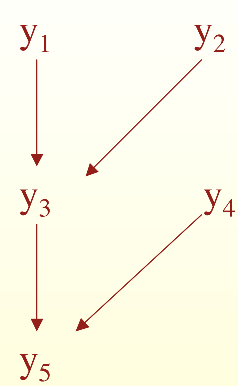
Continuation of Example Above

- = SUM(Y3)(p(Y5|Y3, Y1, Y4) * p(Y3| Y1, Y4))
  - $P(s_i,s_j|d) = P(s_i| s_j,d)\ P(s_j|d)$
- = SUM(Y3) p(Y5|Y3, Y4) * p(Y3| Y1, Y4)
  - Y1 conditionally independent of Y5 given Y3,
  - Y3 represents all the contributions of Y1 to Y5
  - Case 1: a node is conditionally independent of non-descendants given its parents
- = SUM(Y3) p(Y5|Y3, Y4) * p(Y3|Y1)
  - Y4 conditionally independent of Y3 given Y1
  - Case 3: Y3 not a descendant of Y5 which d-separates Y1 and Y4



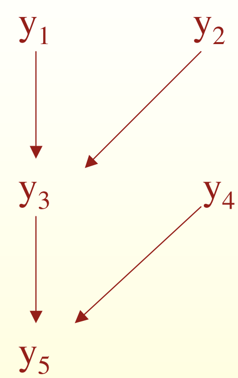
Continuation of Example Above

- = SUM(Y3) p(Y5|Y3, Y4) * (Sum (Y2)p(Y3, Y2 |Y1))
  - Connect to parents of Y3 not already part of expression
- = SUM(Y3) p(Y5|Y3, Y4) *(Sum (Y2) p(Y3|Y1,Y2) * p(Y2|Y1))
  - $p(s_i,s_j|d) = p(s_i| s_j,d)\ p(s_j|d)$; product rule
- = SUM(Y3) p(Y5|Y3, Y4) *(SUM(Y2) p(Y3|Y1,Y2)*p(Y2))
  - Y2 independent of Y1; p(Y2/Y1)=p(Y2)
  - Definition of Baysean network



Belief Network Calculation in Polytree: Evidence Below

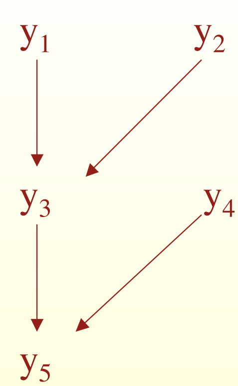
- What is p(Y1|Y5)
  - **p(Y1|Y5)=p(Y1,Y5)/p(Y5)**
  - **p(Y1,Y2,Y3,Y4,Y5) = in terms of cpt**
  - **p(Y5|Y3,Y4)p(Y3|Y1,Y2)p(Y1)p(Y2)p(Y4)**



- p(Y1|Y5) = p(Y5|Y1)p(Y1)/p(Y5)
  - **Bayes Rule**
- =K * p(Y5|Y1)p(Y1)

Continuation of Example Below

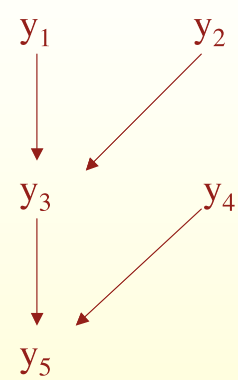
- =K * p(Y5|Y1)p(Y1)
- = K * (SUM(Y3) p(Y5|Y3)p(Y3|Y1)) p(Y1)
  - **Connect to Y3 parent of Y5 not already part of expression**
  - **$P(s_i | s_j)$ = SUM(d)$P(s_i | s_j, d)$ $P(d | s_j)$**
  - **Y1 conditionally independent of Y5 given Y3**
  - **p(Y5|Y3,Y1)= p(Y5|Y3)**



- = K * (SUM(Y3) (SUM(Y4)p(Y5|Y3,Y4)p(Y4|Y3))p(Y3|Y1)) p(Y1)
  - **Connect to Y4 parent of Y5 not already part of expression**
  - **$P(s_i | s_j)$ = SUM(d)$P(s_i | s_j, d)$ $P(d | s_j)$**
- = K * (SUM(Y3) (SUM(Y4)p(Y5|Y3,Y4)p(Y4))p(Y3|Y1)) p(Y1)
  - **Y4 independent of Y3; p(Y4|Y3)= p(Y4)**

Continuation of Example Below

- = K * (SUM(Y3) (SUM(Y4)p(Y5|Y3,Y4)p(Y4))p(Y3|Y1)) p(Y1)
- = K * (SUM(Y3) (SUM(Y4)p(Y5|Y3,Y4)p(Y4))(SUM(Y2)p(Y3|Y1,Y2)p(Y2|Y1))) p(Y1)
  - **Connect to Y2 parent of Y3 not already part of expression**
  - **$P(s_i \mid s_j) = SUM(d)P(s_i \mid s_j, d)\ P(d \mid s_j)$**
- = K * (SUM(Y3) (SUM(Y4)p(Y5|Y3,Y4)p(Y4))(SUM(Y2)p(Y3|Y1,Y2)p(Y2))) p(Y1)
  - **Y2 independent of Y1**
  - **Expression that can be calculated from cpt**



Variable Elimination

- **Can remove a lot of re-calculation/multiplications in expression**
- K * (SUM(Y3) (SUM(Y4)p(Y5|Y3,Y4)p(Y4))(SUM(Y2)p(Y3|Y1,Y2)p(Y2))) p(Y1)
- Summations over each variable are done only for those portions of the expression that depend on variable
- Save results of inner summing to avoid repeated calculation
  - **Create Intermediate Functions**
  - **F-Y2(Y3,Y1)= (SUM(Y2)p(Y3|Y1,Y2)p(Y2))**

Evidence Above and Below for Polytrees

If there is evidence both above and below P(Y3|Y5,Y2)

we separate the evidence into above, ε^+, and below, ε^-, portions and use a version of Bayes' rule to write

$$p(Q \mid \varepsilon^+, \varepsilon^-) = \frac{p(\varepsilon^- \mid Q, \varepsilon^+)\, p(Q \mid \varepsilon^+)}{p(\varepsilon^- \mid \varepsilon^+)}$$

we treat $\frac{1}{p(\varepsilon^- \mid \varepsilon^+)} = k_2$ **as a normalizing factor and write**

$$p(Q \mid \varepsilon^+, \varepsilon^-) = k_2\, p(\varepsilon^- \mid Q, \varepsilon^+)\, p(Q \mid \varepsilon^+)$$

Q d-separates ε^- **from** ε^+**, so**

$$p(Q \mid \varepsilon^+, \varepsilon^-) = k_2\, p(\varepsilon^- \mid Q)\, p(Q \mid \varepsilon^+)$$

We calculate the first probability in this product as part of the top-down procedure for calculating $p(Q \mid \varepsilon^-)$**. The second probability is calculated directly by the bottom-up procedure.**

Other types of queries

- Most probable explanation (MPE) or most likely hypothesis:

  The instantiation of *all* the remaining variables U with the highest probability given the evidence

  MPE(U | e) = argmax$_u$ P(u,e)
- Maximum a posteriori (MAP):

  The instantiation of *some* variables V with the highest probability given the evidence

  MAP(V | e) = argmax$_v$ P(v,e)

  Note that the assignment to A in MAP(A|e) might be completely different from the assignment to A in MAP({A,B} | e).
  - sum over values of B vs individual values of B
- **Other queries: probability of an arbitrary logical expression over query variables, decision policies, information value, *seeking evidence*, information gathering planning, etc.**

Incremental Updating of BN: Pearl's message passing algorithm

Notation:

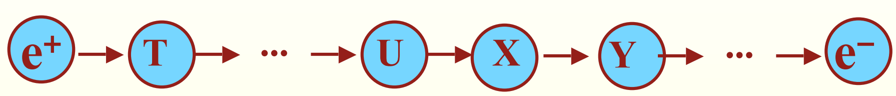
$M_{y|x}$ Conditional probability matix

e The evidence

$\text{Bel}(x) = P(x \mid e)$ Posterior distribution of x

$$f(x) \bullet M_{y|x} = \sum_x f(x) M_{y|x}$$

Simple chains



$e = \{e^+, e^-\}$

e^+ Represents the "causal" evidence

e^- Represents the "evidential" evidence

Need to compute Bel(x)

Simple Chains cont.

$$Bel(x) = P(x \mid e^{+}e^{-})$$

$$= \frac{P(e^{-} \mid x\, e^{+})P(x \mid e^{+})}{P(e^{-} \mid e^{+})} \qquad \text{Bayes rule}$$

$$= \alpha P(e^{-} \mid x\, e^{+})P(x \mid e^{+}) \qquad \text{Normalization}$$

$$= \alpha P(e^{-} \mid x)P(x \mid e^{+}) \qquad x \text{ d-sep } e^{+}e^{-}$$

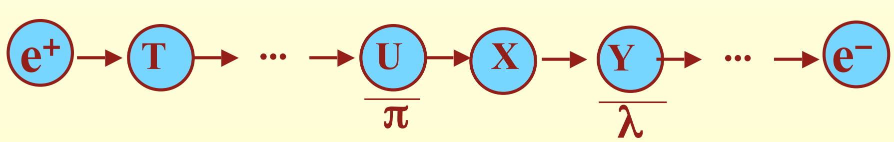
$$= \alpha \cdot \lambda(x) \cdot \pi(x)$$

The λ(x) and π(x) Messages

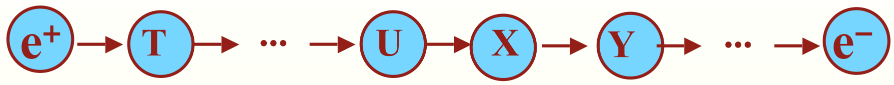
λ(x) **represents the degree to which x might explain the evidential support. -- $P(e^-/X)$**

π(x) **represents the direct causal support for x. -- $P(X/e^+)$**

Both λ(x) and π(x) can be calculated in terms of the λ and π values of the neighbors of x.



Computing λ(x) based on λ(y)



$$\lambda(x) = P(e^- \mid x)$$

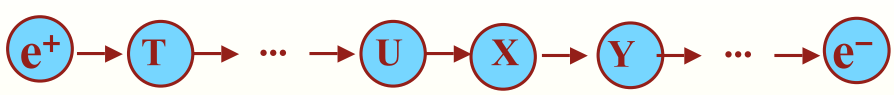
$$= \sum_y P(e^- \mid x, y) P(y \mid x)$$

$$= \sum_y P(e^- \mid y) P(y \mid x) \qquad y \text{ d-sep } x, e^-$$

$$= \sum_y \lambda(y) P(y \mid x)$$

$$= \lambda(y) \bullet M_{y|x}$$

Computing π(x) based on π(u)



$$\pi(x) = P(x \mid e^+)$$

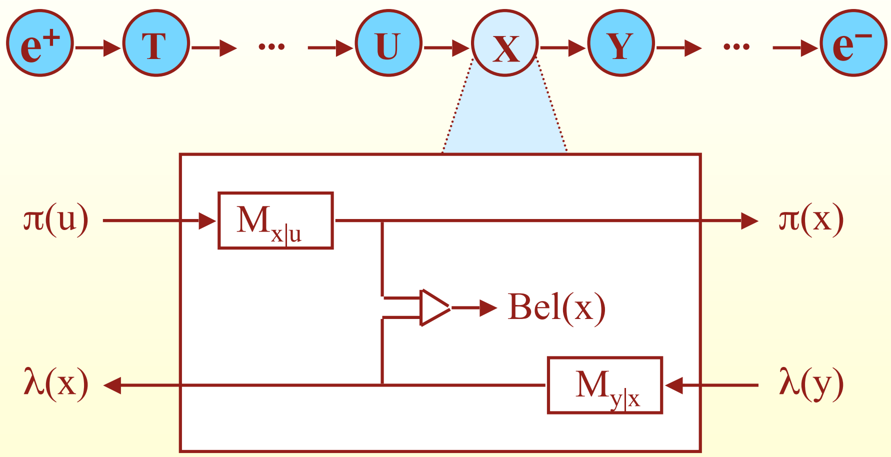
$$= \sum_u P(x \mid u\, e^+) P(u \mid e^+)$$

$$= \sum_u P(x \mid u) P(u \mid e^+) \qquad u \text{ d-sep } x, e^+$$

$$= \sum_u P(x \mid u) \pi(u)$$

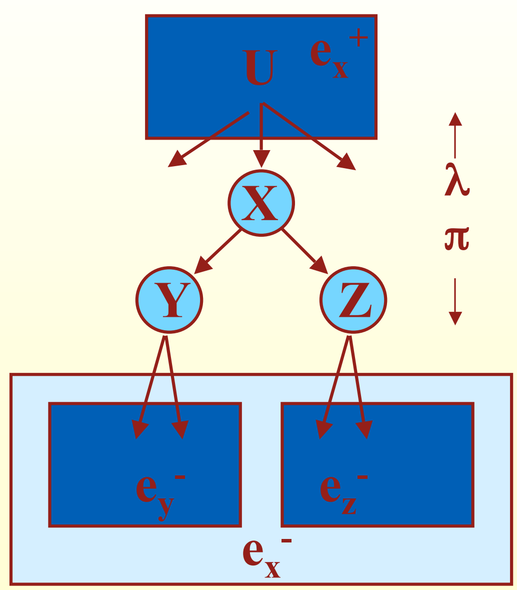
$$= \pi(u) \bullet M_{x \mid u}$$

Update scheme for chains

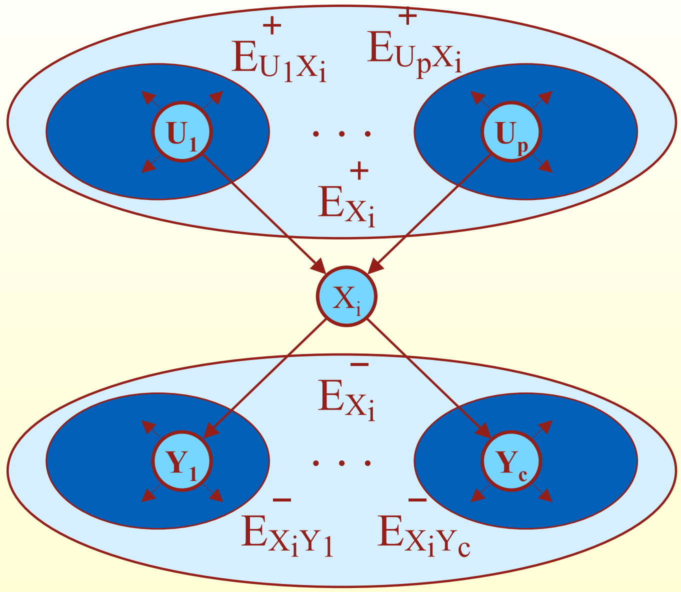


Belief Propagation in Trees

- Each node must combine the impact of λ-messages from several children.
- Each node must distribute a separate π-message to each child.



Propagation in Polytrees



Decomposing the evidence

The evidence E can be decomposed into two subsets:

- E_i^+, the subset of E that can be accessed from X_i through its parents.
- E_i^-, the subset of E that can be accessed from X_i through its children.

Parameters:

The current strength of the causal support, π, contributed by each incoming link $U_i \rightarrow x$:

$$\pi_x(U_i) = P(U_i \mid e^{+}_{uix})$$

The current strength of the diagnostic support, λ, contributed by each outgoing link $x \rightarrow Y_j$:

$$\lambda y_j(x) = P(e^{-}_{xyj} \mid x)$$

The fixed conditional probability matrix

$$P(x \mid u_1, \ldots, u_n))$$

Propagation Process

Step 1: Belief updating: Inspect msgs from parents & children and compute:

$$Bel(x) = \alpha \lambda(x) \pi(x) \; where:$$

$$\lambda(x) = \prod_j \lambda y_j(x)$$

$$\pi(x) = \sum_{u_1 \ldots u_n} P(x \mid u_1 \ldots u_n) \prod_i \pi_x(u_i)$$

Step 2: Bottom-up propagation: Compute λ msgs to send up.

$\lambda\, x(U_i)$ is the msg X sends to parent U_i.

$$\lambda_x = \beta \sum_x \lambda(x) \sum_{U_k = k \; \neq i} P(x \mid U_1 \ldots U_n) \prod_{k \neq i} \pi_x(U_k)$$

β is an arbitrary constant (factor out contributions to bel(x) from U_i)

Propagation, *cont'd:*

Step 3: Top-Down Propagation:

Compute π msgs to send down.

$\pi_{Y_j}(x)$ is sent from x to child Y_j

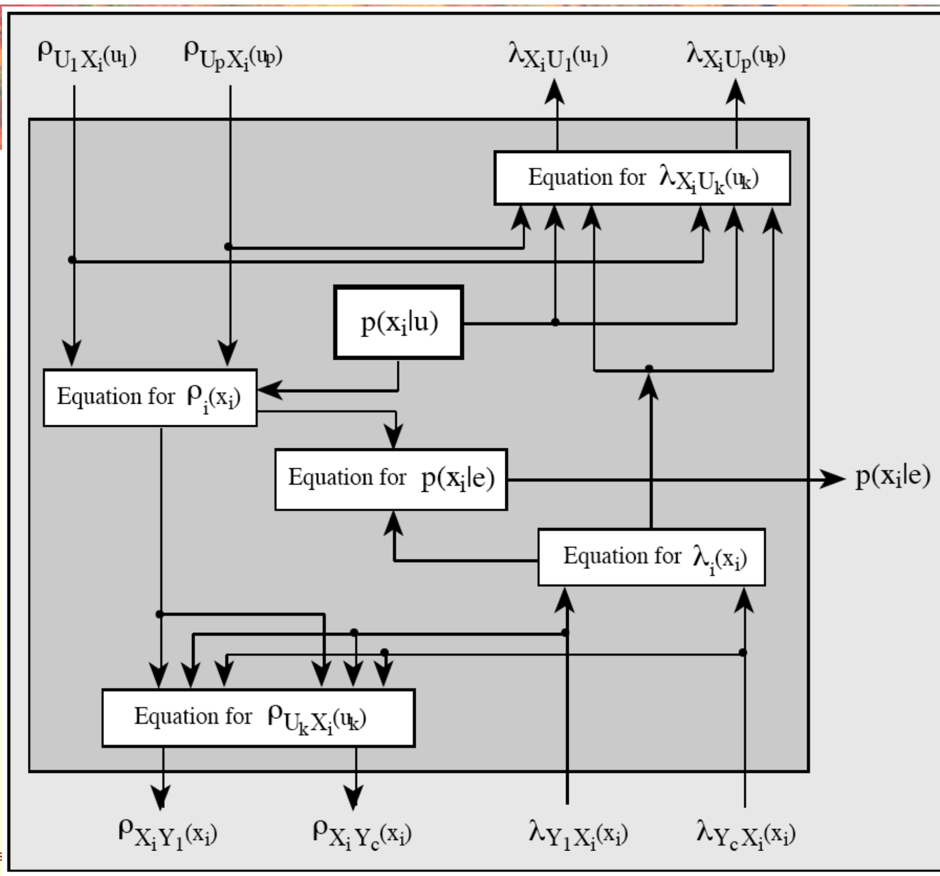
$$\pi_{Y_j}(x) = \alpha \left[\prod_{k \text{ not } j} \lambda_{Y_k}(x)\right] \sum_{U_1 \ldots U_n} P(x \mid U_1 \ldots U_n) \prod_i \pi_x(U_i)$$

$$= \alpha \bullet \frac{Bel(x)}{\lambda_{y_j}(x)}$$

(factor out contributions to bel(x) from Y_j)

Boundary Conditions:

1. **Root $\Rightarrow$ $\pi(x)$ is the prior prob. dist.**
2. **Childless node $\Rightarrow$ λ (x) = (1,....,1)**
3. **Evidence node $\Rightarrow$ λ(x) = (0,.....,1...,0)**



Next Lecture

- Approximate inference techniques
- Alternative approaches to uncertain reasoning