CS 1571 Introduction to AI
Lecture 20

Bayesian belief networks

Milos Hauskrecht
milos@cs.pitt.edu
5329 Sennott Square

Modeling uncertainty with probabilities

- Defining the **full joint distribution** makes it possible to represent and reason with uncertainty in a uniform way
- We are able to handle an arbitrary inference problem

Problems:

- **Space complexity.** To store a full joint distribution we need to remember $O(d^n)$ numbers.

  n – number of random variables, *d* – number of values
- **Inference (time) complexity.** To compute some queries requires $O(d^n)$ steps.
- **Acquisition problem.** Who is going to define all of the probability entries?

Bayesian belief networks (BBNs)

Bayesian belief networks.

- Represent the full joint distribution over the variables more compactly with a **smaller number of parameters**.
- Take advantage of **conditional and marginal independences** among random variables

- **A and B are independent**

$$P(A,B) = P(A)P(B)$$

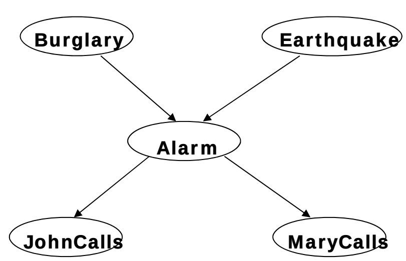
- **A and B are conditionally independent given C**

$$P(A,B \mid C) = P(A \mid C)P(B \mid C)$$
$$P(A \mid C,B) = P(A \mid C)$$

Alarm system example

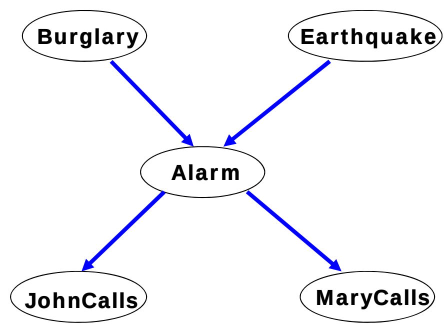
- Assume your house has an **alarm system** against **burglary**. You live in the seismically active area and the alarm system can get occasionally set off by an **earthquake**. You have two neighbors, **Mary** and **John**, who do not know each other. If they hear the alarm they call you, but this is not guaranteed.
- We want to represent the probability distribution of events:
  - Burglary, Earthquake, Alarm, Mary calls and John calls

Causal relations



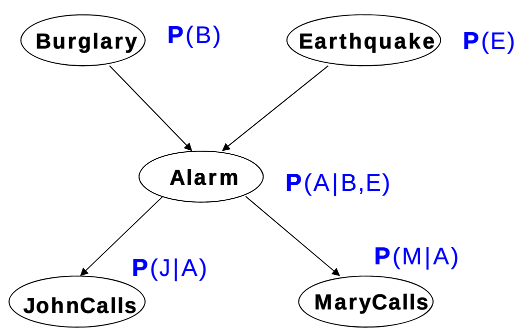
Bayesian belief network

1. **Directed acyclic graph**
   - **Nodes** = random variables
     Burglary, Earthquake, Alarm, Mary calls and John calls
   - **Links** = direct (causal) dependencies between variables.
     The chance of Alarm is influenced by Earthquake, The chance of John calling is affected by the Alarm

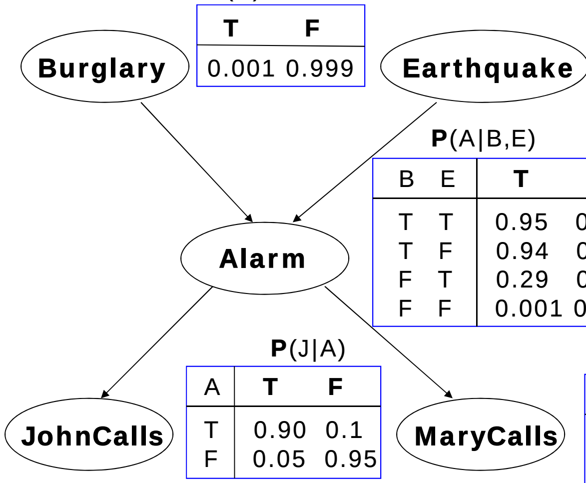


Bayesian belief network

2. **Local conditional distributions**
   - relate variables and their parents



Bayesian belief network



P(B)

T	F
0.001	0.999

P(E)

T	F
0.002	0.998

P(A|B,E)

B	E	T	F
T	T	0.95	0.05
T	F	0.94	0.06
F	T	0.29	0.71
F	F	0.001	0.999

P(J|A)

A	T	F
T	0.90	0.1
F	0.05	0.95

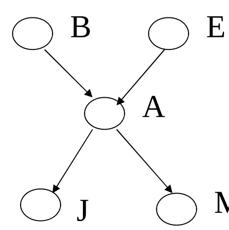
P(M|A)

A	T	F
T	0.7	0.3
F	0.01	0.99

Bayesian belief networks (general)

Two components: $B = (S, \Theta_S)$

- **Directed acyclic graph**
  - Nodes correspond to random variables
  - (Missing) links encode independences



- **Parameters**
  - Local conditional probability distributions for every variable-parent configuration

$$\mathbf{P}(X_i \mid pa(X_i))$$

Where:

$pa(X_i)$ - stand for parents of X_i

P(A|B,E)

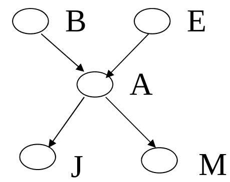
B	E	T	F
T	T	0.95	0.05
T	F	0.94	0.06
F	T	0.29	0.71
F	F	0.001	0.999

Full joint distribution in BBNs

Full joint distribution is defined in terms of local conditional distributions (obtained via the chain rule):

$$\mathbf{P}(X_1, X_2,.., X_n) = \prod_{i=1,..n} \mathbf{P}(X_i \mid pa(X_i))$$

Example:



Assume the following assignment of values to random variables
$B = T, E = T, A = T, J = T, M = F$

Then its probability is:
$P(B = T, E = T, A = T, J = T, M = F) =$

$$P(B = T)P(E = T)P(A = T \mid B = T, E = T)P(J = T \mid A = T)P(M = F \mid A = T)$$

Bayesian belief networks (BBNs)

Bayesian belief networks

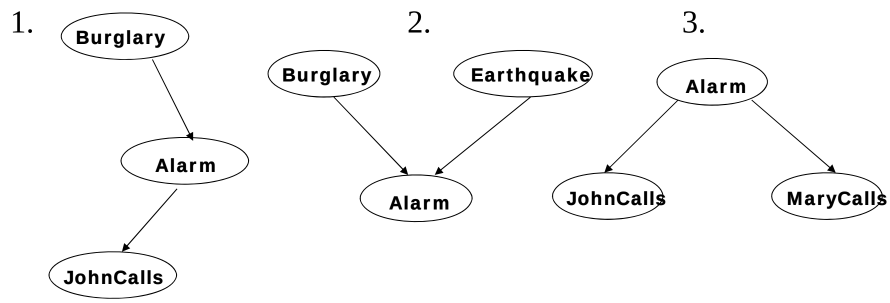
- Represent the full joint distribution over the variables more compactly using the product of local conditionals.
- **But how did we get to local parameterizations?**

Answer:

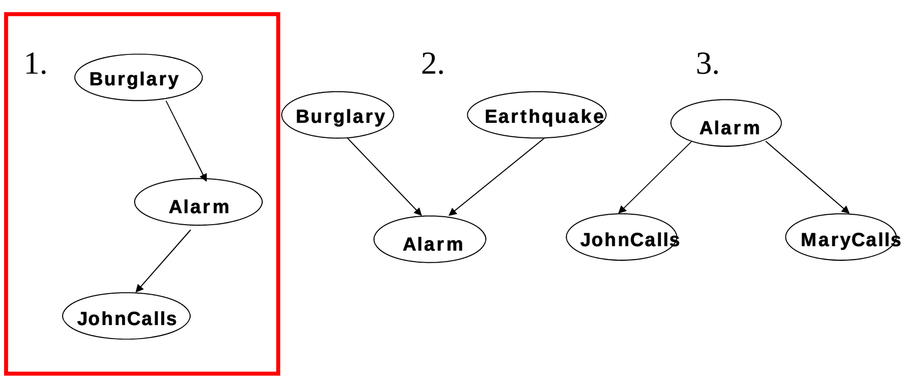
- **Chain rule +**
- **Graphical structure** encodes **conditional and marginal independences** among random variables
- **A and B are independent** $P(A, B) = P(A)P(B)$
- **A and B are conditionally independent given C**
  $P(A \mid C, B) = P(A \mid C)$ $P(A, B \mid C) = P(A \mid C)P(B \mid C)$
- **The graph structure implies the decomposition !!!**

Independences in BBNs

3 basic independence structures:



Independences in BBNs

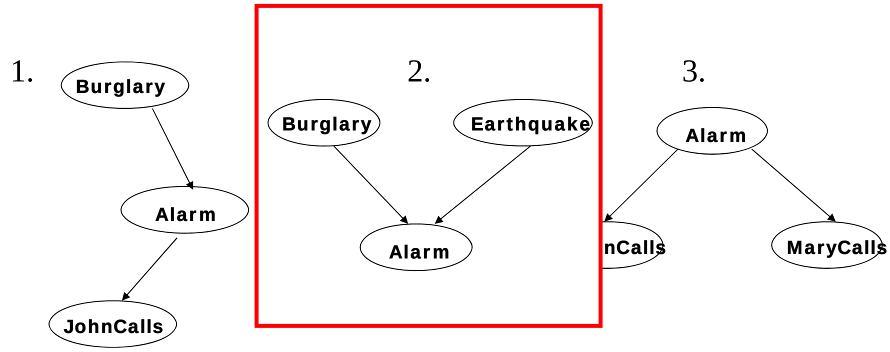


1. JohnCalls **is independent** of Burglary **given** Alarm

$$P(J \mid A, B) = P(J \mid A)$$

$$P(J, B \mid A) = P(J \mid A) P(B \mid A)$$

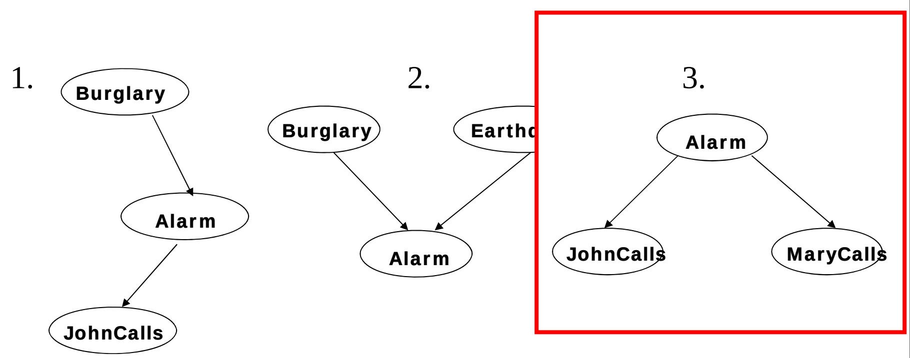
Independences in BBNs



2. Burglary **is independent** of Earthquake (not knowing Alarm)
   Burglary and Earthquake **become dependent** given Alarm !!

$$P(B,E) = P(B)P(E)$$

Independences in BBNs



3. MaryCalls **is independent** of JohnCalls **given** Alarm

$$P(J \mid A, M) = P(J \mid A)$$

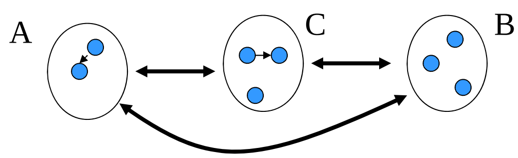
$$P(J, M \mid A) = P(J \mid A)P(M \mid A)$$

Independences in BBN

- BBN distribution models many conditional independence relations among distant variables and sets of variables
- These are defined in terms of the graphical criterion called d-separation
- **D-separation and independence**
  - Let X,Y and Z be three sets of nodes
  - If X and Y are d-separated by Z, then X and Y are conditionally independent given Z
- **D-separation :**
  - A is d-separated from B given C if every undirected path between them is **blocked with C**
- **Path blocking**
  - 3 cases that expand on three basic independence structures

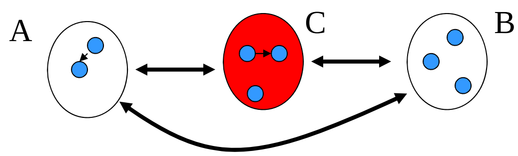
Undirected path blocking

A is d-separated from B given C if every undirected path between them is **blocked**



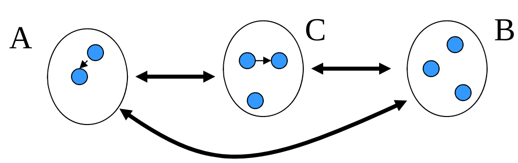
Undirected path blocking

A is d-separated from B given C if every undirected path between them is **blocked**

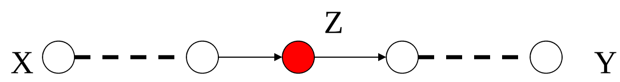


Undirected path blocking

A is d-separated from B given C if every undirected path between them is **blocked**



- **1. Path blocking with a linear substructure**



X in A

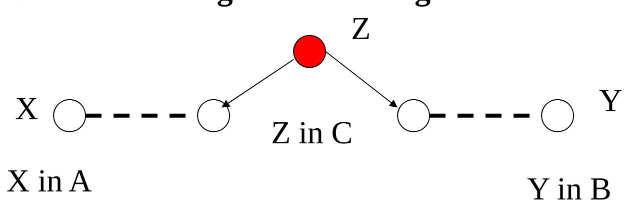
Z in C

Y in B

Undirected path blocking

A is d-separated from B given C if every undirected path between them is **blocked**

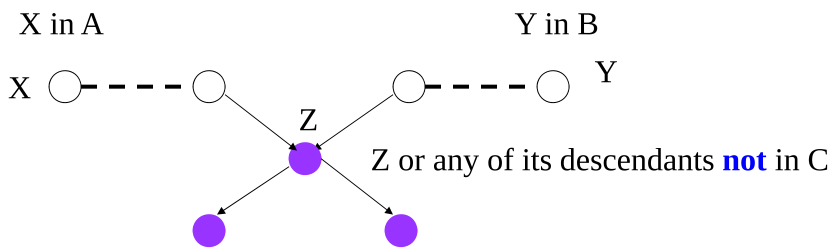
- **2. Path blocking with the wedge substructure**



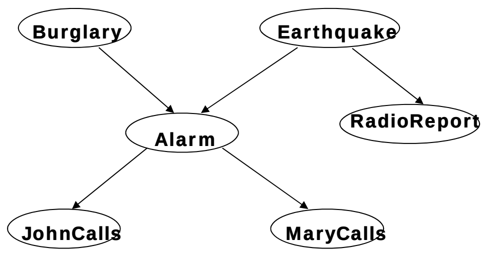
Undirected path blocking

A is d-separated from B given C if every undirected path between them is **blocked**

- **3. Path blocking with the vee substructure**



Independences in BBNs



- Earthquake and Burglary are independent given MaryCalls **F**
- Burglary and MaryCalls are independent (not knowing Alarm) **F**
- Burglary and RadioReport are independent given Earthquake **T**
- Burglary and RadioReport are independent given MaryCalls **F**

Bayesian belief networks (BBNs)

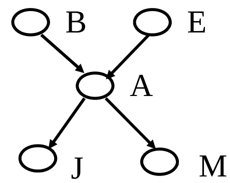
Bayesian belief networks

- Represents the full joint distribution over the variables more compactly using the product of local conditionals.
- **So how did we get to local parameterizations?**

$$\mathbf{P}(X_1, X_2,.., X_n) = \prod_{i=1,..n} \mathbf{P}(X_i \mid pa(X_i))$$

- **The decomposition is implied by the set of independences encoded in the belief network.**

Full joint distribution in BBNs



Rewrite the full joint probability using the product rule:

$P(B=T,E=T,A=T,J=T,M=F)=$

$=P(J=T \mid B=T,E=T,A=T,M=F)P(B=T,E=T,A=T,M=F)$

$=P(J=T \mid A=T)P(B=T,E=T,A=T,M=F)$

$P(M=F \mid B=T,E=T,A=T)P(B=T,E=T,A=T)$

$P(M=F \mid A=T)P(B=T,E=T,A=T)$

$P(A=T \mid B=T,E=T)P(B=T,E=T)$

$P(B=T)P(E=T)$

$=P(J=T \mid A=T)P(M=F \mid A=T)P(A=T \mid B=T,E=T)P(B=T)P(E=T)$

Parameter complexity problem

- In the BBN the **full joint distribution** is defined as:

$$\mathbf{P}(X_1, X_2, \ldots, X_n) = \prod_{i=1,..n} \mathbf{P}(X_i \mid pa(X_i))$$

- **What did we save?**

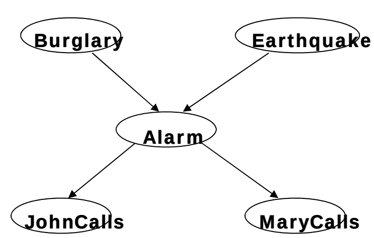
Alarm example: 5 binary (True, False) variables

# of parameters of the full joint:

$2^5 = 32$

One parameter is for free:

$2^5 - 1 = 31$



Parameter complexity problem

- In the BBN the **full joint distribution** is defined as:

$$\mathbf{P}(X_1, X_2, \ldots, X_n) = \prod_{i=1,..n} \mathbf{P}(X_i \mid pa(X_i))$$

- **What did we save?**

Alarm example: 5 binary (True, False) variables

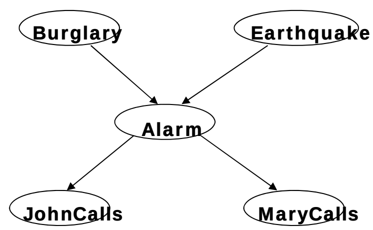
# of parameters of the full joint:

$$2^5 = 32$$

One parameter is for free:

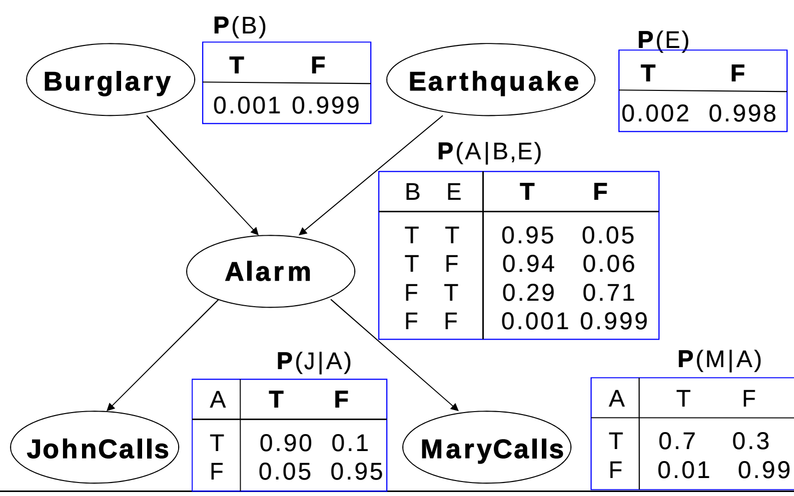
$$2^5 - 1 = 31$$

# of parameters of the BBN: ?



Bayesian belief network.

- In the BBN the **full joint distribution** is expressed using a set of local conditional distributions



P(B)

T	F
0.001	0.999

P(E)

T	F
0.002	0.998

P(A|B,E)

B	E	T	F
T	T	0.95	0.05
T	F	0.94	0.06
F	T	0.29	0.71
F	F	0.001	0.999

P(J|A)

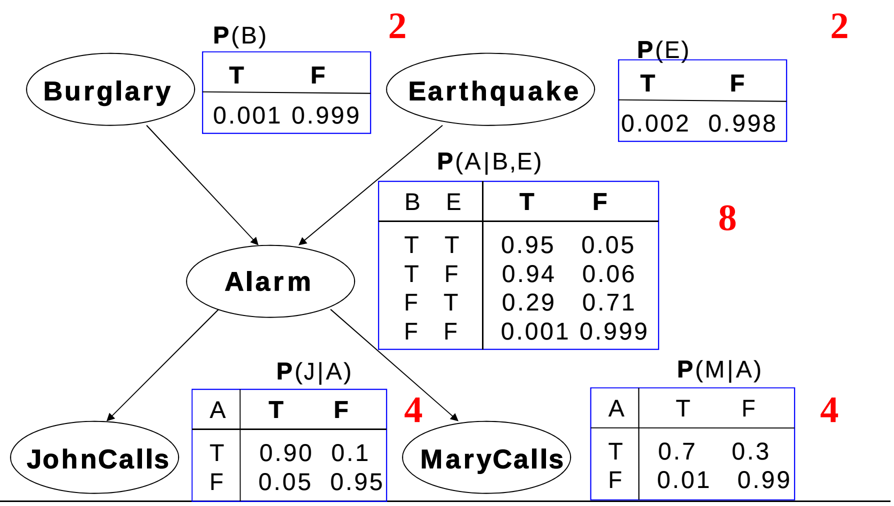
A	T	F
T	0.90	0.1
F	0.05	0.95

P(M|A)

A	T	F
T	0.7	0.3
F	0.01	0.99

Bayesian belief network.

- In the BBN the **full joint distribution** is expressed using a set of local conditional distributions



P(B)

T	F
0.001	0.999

P(E)

T	F
0.002	0.998

P(A|B,E)

B	E	T	F
T	T	0.95	0.05
T	F	0.94	0.06
F	T	0.29	0.71
F	F	0.001	0.999

P(J|A)

A	T	F
T	0.90	0.1
F	0.05	0.95

P(M|A)

A	T	F
T	0.7	0.3
F	0.01	0.99

Parameter complexity problem

- In the BBN the **full joint distribution** is defined as:

$$\mathbf{P}(X_1, X_2, \ldots, X_n) = \prod_{i=1,..n} \mathbf{P}(X_i \mid pa(X_i))$$

- **What did we save?**

Alarm example: 5 binary (True, False) variables

# of parameters of the full joint:

$$2^5 = 32$$

One parameter is for free:

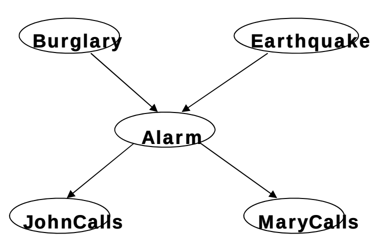
$$2^5 - 1 = 31$$

# of parameters of the BBN:

$$2^3 + 2(2^2) + 2(2) = 20$$

One parameter in every conditional is for free:

$$2^2 + 2(2) + 2(1) = 10$$



Model acquisition problem

The structure of the BBN

- typically reflects causal relations
  (BBNs are also sometime referred to as **causal networks)**
- Causal structure is intuitive in many applications domain and it is relatively easy to define to the domain expert

Probability parameters of BBN

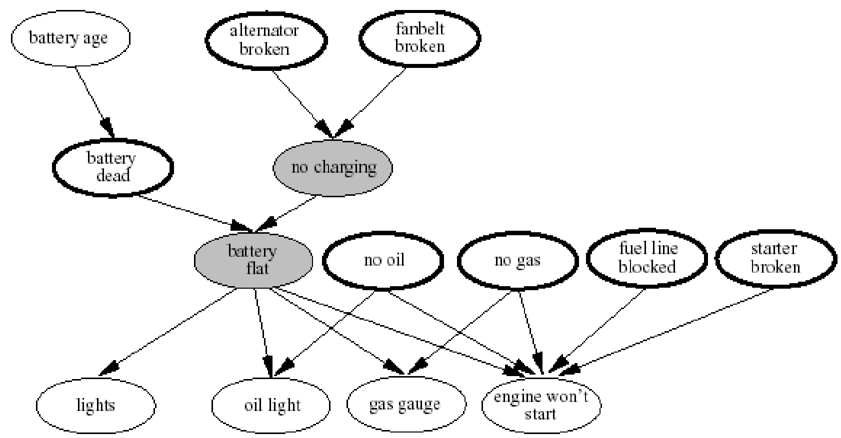
- are conditional distributions relating random variables and their parents
- Complexity is much smaller than the full joint
- It is much easier to obtain such probabilities from the expert or learn them automatically from data

BBNs built in practice

- **In various areas:**
  - Intelligent user interfaces (Microsoft)
  - Troubleshooting, diagnosis of a technical device
  - Medical diagnosis:
    - Pathfinder (Intellipath)
    - CPSC
    - Munin
    - QMR-DT
  - Collaborative filtering
  - Military applications
  - Business and finance
    - Insurance, credit applications

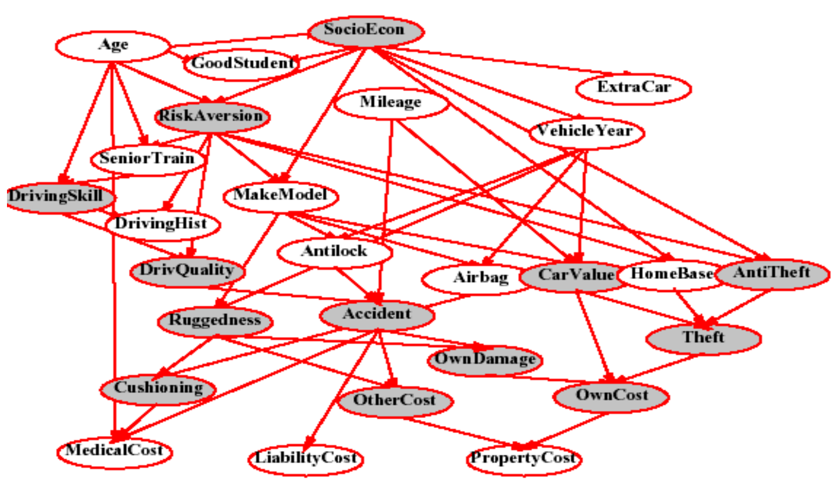
Diagnosis of car engine

- Diagnose the engine start problem

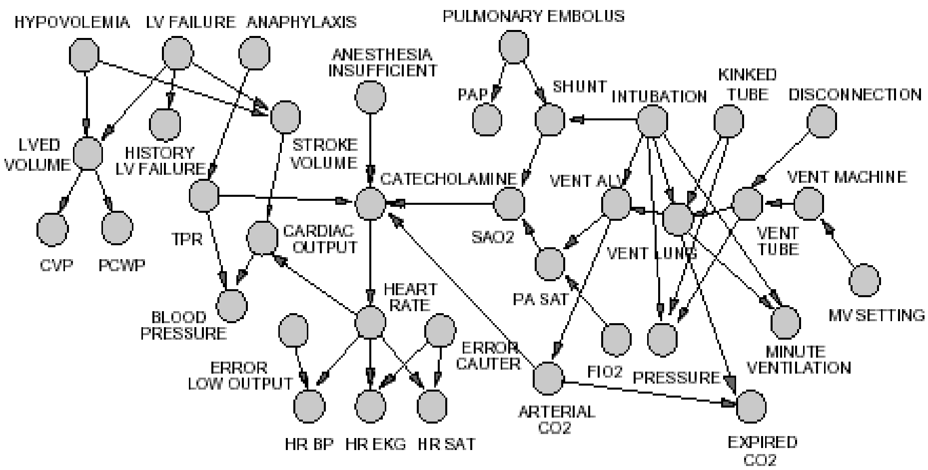


Car insurance example

- Predict claim costs (medical, liability) based on application data

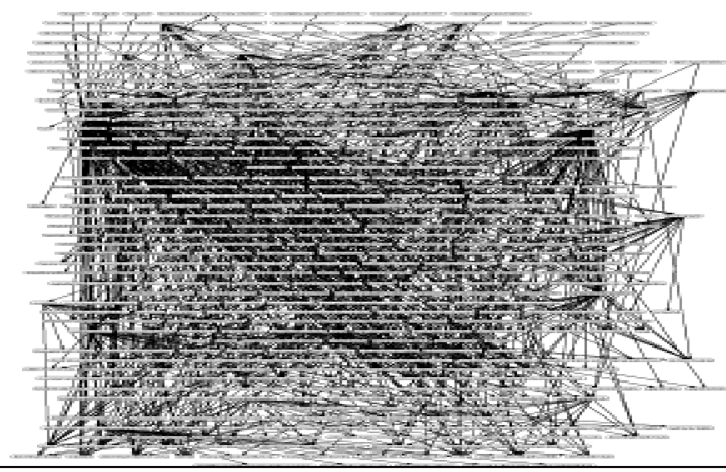


(ICU) Alarm network



CPCS

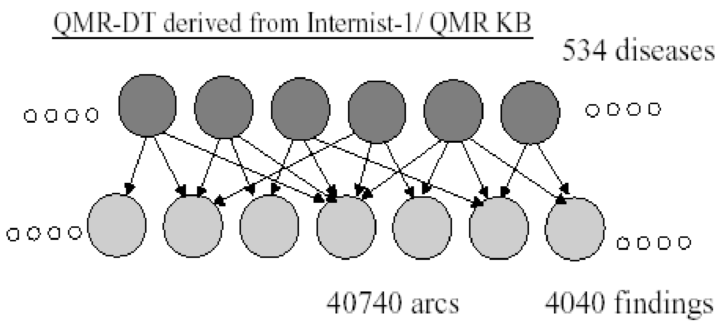
- **C**omputer-based **P**atient **C**ase **S**imulation system (CPCS-PM) developed by Parker and Miller (University of Pittsburgh)
- 422 nodes and 867 arcs



QMR-DT

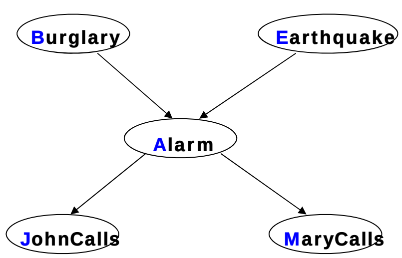
- **Medical diagnosis in internal medicine**
- **Based on QMR system built at U Pittsburgh**

Bipartite network of disease/findings relations



Inference in Bayesian network

- **Bad news:**
  - Exact inference problem in BBNs is NP-hard (Cooper)
  - Approximate inference is NP-hard (Dagum, Luby)
- **But** very often we can achieve significant improvements
- Assume our Alarm network



- Assume we want to compute: $P(J = T)$

Inference in Bayesian networks

Computing: $P(J=T)$

Approach 1. Blind approach.

- Sum out all un-instantiated variables from the full joint,
- express the joint distribution as a product of conditionals

$$P(J=T) =$$

$$= \sum_{b\in T,F}\sum_{e\in T,F}\sum_{a\in T,F}\sum_{m\in T,F} P(B=b,E=e,A=a,J=T,M=m)$$

$$= \sum_{b\in T,F}\sum_{e\in T,F}\sum_{a\in T,F}\sum_{m\in T,F} P(J=T\mid A=a)P(M=m\mid A=a)P(A=a\mid B=b,E=e)P(B=b)P(E=e)$$

Computational cost:

Number of additions: 15

Number of products: 16*4=64

Inference in Bayesian networks

Approach 2. Interleave sums and products

- Combines sums and product in a smart way (multiplications by constants can be taken out of the sum)

$$\sum_{\{x\}} af(x) = a\sum_{\{x\}} f(x)$$

$$P(J=T) =$$

$$= \sum_{b\in T,F}\sum_{e\in T,F}\sum_{a\in T,F}\sum_{m\in T,F} P(J=T\mid A=a)P(M=m\mid A=a)P(A=a\mid B=b,E=e)P(B=b)P(E=e)$$

$$= \sum_{b\in T,F}\sum_{a\in T.F}\sum_{m\in T,F} P(J=T\mid A=a)P(M=m\mid A=a)P(B=b)\left[\sum_{e\in T,F} P(A=a\mid B=b,E=e)P(E=e)\right]$$

$$= \sum_{a\in T,F} P(J=T\mid A=a)\left[\sum_{m\in T,F} P(M=m\mid A=a)\right]\left[\sum_{b\in T,F} P(B=b)\left[\sum_{e\in T,F} P(A=a\mid B=b,E=e)P(E=e)\right]\right]$$

Computational cost: ?

Inference in Bayesian networks

Approach 2. Interleave sums and products

- Combines sums and product in a smart way (multiplications by constants can be taken out of the sum)

$$= \sum_{a \in T,F} P(J=T \mid A=a) \left[\sum_{m \in T,F} P(M=m \mid A=a) \right] \left[\sum_{b \in T,F} P(B=b) \left[\sum_{e \in T,F} P(A=a \mid B=b, E=e) P(E=e) \right] \right]$$

1

Computational cost:

Number of additions: ?

CS 1571 Intro to AI M. Hauskrecht

Inference in Bayesian networks

Approach 2. Interleave sums and products

- Combines sums and product in a smart way (multiplications by constants can be taken out of the sum)

$$= \sum_{a \in T,F} P(J=T \mid A=a) \left[\sum_{m \in T,F} P(M=m \mid A=a) \right] \left[\sum_{b \in T,F} P(B=b) \left[\sum_{e \in T,F} P(A=a \mid B=b, E=e) P(E=e) \right] \right]$$

2*1

Computational cost:

Number of additions: ?

CS 1571 Intro to AI M. Hauskrecht

Inference in Bayesian networks

Approach 2. Interleave sums and products

- Combines sums and product in a smart way (multiplications by constants can be taken out of the sum)

$$= \sum_{a \in T,F} P(J=T \mid A=a) \left[\sum_{m \in T,F} P(M=m \mid A=a) \right] \left[\sum_{b \in T,F} P(B=b) \left[\sum_{e \in T,F} P(A=a \mid B=b, E=e) P(E=e) \right] \right]$$

2*2*1

Computational cost:

Number of additions: ?

Inference in Bayesian networks

Approach 2. Interleave sums and products

- Combines sums and product in a smart way (multiplications by constants can be taken out of the sum)

$$= \sum_{a \in T,F} P(J=T \mid A=a) \left[\sum_{m \in T,F} P(M=m \mid A=a) \right] \left[\sum_{b \in T,F} P(B=b) \left[\sum_{e \in T,F} P(A=a \mid B=b, E=e) P(E=e) \right] \right]$$

2*2*1

2*1

Computational cost:

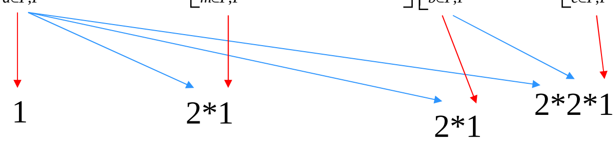
Number of additions: ?

Inference in Bayesian networks

Approach 2. Interleave sums and products

- Combines sums and product in a smart way (multiplications by constants can be taken out of the sum)

$$= \sum_{a\in T,F} P(J=T\mid A=a)\left[\sum_{m\in T,F} P(M=m\mid A=a)\right]\left[\sum_{b\in T,F} P(B=b)\left[\sum_{e\in T,F} P(A=a\mid B=b,E=e)P(E=e)\right]\right]$$



Computational cost:

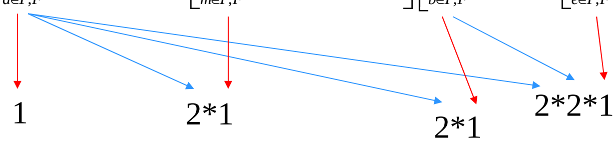
Number of additions: **?**

Inference in Bayesian networks

Approach 2. Interleave sums and products

- Combines sums and product in a smart way (multiplications by constants can be taken out of the sum)

$$= \sum_{a\in T,F} P(J=T\mid A=a)\left[\sum_{m\in T,F} P(M=m\mid A=a)\right]\left[\sum_{b\in T,F} P(B=b)\left[\sum_{e\in T,F} P(A=a\mid B=b,E=e)P(E=e)\right]\right]$$



Computational cost:

Number of additions: 1+2*[1+1+2*1]=**9**

Inference in Bayesian networks

Approach 2. Interleave sums and products

- Combines sums and product in a smart way (multiplications by constants can be taken out of the sum)

$$= \sum_{a \in T,F} P(J=T \mid A=a) \left[\sum_{m \in T,F} P(M=m \mid A=a) \right] \left[\sum_{b \in T,F} P(B=b) \left[\sum_{e \in T,F} P(A=a \mid B=b, E=e) P(E=e) \right] \right]$$

1

Computational cost:

Number of products: ?

Inference in Bayesian networks

Approach 2. Interleave sums and products

- Combines sums and product in a smart way (multiplications by constants can be taken out of the sum)

$$= \sum_{a \in T,F} P(J=T \mid A=a) \left[\sum_{m \in T,F} P(M=m \mid A=a) \right] \left[\sum_{b \in T,F} P(B=b) \left[\sum_{e \in T,F} P(A=a \mid B=b, E=e) P(E=e) \right] \right]$$

2*2 *2*1

Computational cost:

Number of products: ?

Inference in Bayesian networks

Approach 2. Interleave sums and products

- Combines sums and product in a smart way (multiplications by constants can be taken out of the sum)

$$= \sum_{a\in T,F} P(J=T\,|\,A=a)\left[\sum_{m\in T,F} P(M=m\,|\,A=a)\right]\left[\sum_{b\in T,F} P(B=b)\left[\sum_{e\in T,F} P(A=a\,|\,B=b,E=e)P(E=e)\right]\right]$$

2*2

2*2*1

2*2 *2*1

Computational cost:

Number of products: 2*[2+2*(1+2*1)]=**16**

Inference in Bayesian networks

Approach 2. Interleave sums and products

- Combines sums and product in a smart way (multiplications by constants can be taken out of the sum)

$$P(J=T)=$$

$$= \sum_{b\in T,F}\sum_{e\in T,F}\sum_{a\in T,F}\sum_{m\in T,F} P(J=T\,|\,A=a)P(M=m\,|\,A=a)P(A=a\,|\,B=b,E=e)P(B=b)P(E=e)$$

$$= \sum_{b\in T,F}\sum_{a\in T.F}\sum_{m\in T,F} P(J=T\,|\,A=a)P(M=m\,|\,A=a)P(B=b)\left[\sum_{e\in T,F} P(A=a\,|\,B=b,E=e)P(E=e)\right]$$

$$= \sum_{a\in T,F} P(J=T\,|\,A=a)\left[\sum_{m\in T,F} P(M=m\,|\,A=a)\right]\left[\sum_{b\in T,F} P(B=b)\left[\sum_{e\in T,F} P(A=a\,|\,B=b,E=e)P(E=e)\right]\right]$$

Computational cost:

Number of additions: 1+2*[1+1+2*1]=**9**

Number of products: 2*[2+2*(1+2*1)]=**16**

Variable elimination

- **Variable elimination:**
  - Similar idea but interleave sum and products one variable at the time during inference
  - E.g. Query $P(J=T)$ requires to eliminate A,B,E,M and this can be done in different order

$$P(J=T)=$$

$$=\sum_{b\in T,F}\sum_{e\in T,F}\sum_{a\in T,F}\sum_{m\in T,F} P(J=T\mid A=a)P(M=m\mid A=a)P(A=a\mid B=b,E=e)P(B=b)P(E=e)$$

Variable elimination

Assume order: M, E, B,A to calculate $P(J=T)$

$$=\sum_{b\in T,F}\sum_{e\in T,F}\sum_{a\in T,F}\sum_{m\in T,F} P(J=T\mid A=a)P(M=m\mid A=a)P(A=a\mid B=b,E=e)P(B=b)P(E=e)$$

$$=\sum_{b\in T,F}\sum_{e\in T,F}\sum_{a\in T,F} P(J=T\mid A=a)P(A=a\mid B=b,E=e)P(B=b)P(E=e)\left[\sum_{m\in T,F} P(M=m\mid A=a)\right]$$

$$=\sum_{b\in T,F}\sum_{e\in T,F}\sum_{a\in T,F} P(J=T\mid A=a)P(A=a\mid B=b,E=e)P(B=b)P(E=e) \quad 1$$

$$=\sum_{a\in T,F}\sum_{b\in T,F} P(J=T\mid A=a)P(B=b)\left[\sum_{e\in T,F} P(A=a\mid B=b,E=e)P(E=e)\right]$$

$$=\sum_{a\in T,F}\sum_{b\in T,F} P(J=T\mid A=a)P(B=b)\,\tau_1(A=a,B=b)$$

$$=\sum_{a\in T,F} P(J=T\mid A=a)\left[\sum_{b\in T,F} P(B=b)\,\tau_1(A=a,B=b)\right]$$

$$=\sum_{a\in T,F} P(J=T\mid A=a)\quad \tau_2(A=a) \quad = \boxed{P(J=T)}$$

Inference in Bayesian network

- **Exact inference algorithms:**
  - **Variable elimination** (Book)
  - **Recursive decomposition** (Cooper, Darwiche)
  - Symbolic inference (D'Ambrosio)
  - Belief propagation algorithm (Pearl)
  - **Clustering and joint tree approach (Lauritzen, Spiegelhalter)** (Book)
  - Arc reversal (Olmsted, Schachter)

- **Approximate inference algorithms:**
  - **Monte Carlo methods:** (Book)
    - Forward sampling, Likelihood sampling
  - Variational methods