Bayesian probabilistic networks

Model Building and Algorithms

Y.Y. Bayraktarli
bayraktarli@ibk.baug.ethz.ch

Definition

- Complex systems are characterized through intricate dependences.
- Given the uncertainties prevailing in engineering systems, these dependences are often of probabilistic nature.

Main problem is reasoning under uncertainty.

Reasoning under uncertainty

- Different approaches:
  - Rule-based systems
    - Fuzzy sets
    - Dempster-Shafer belief functions
    - Probabilities
  - Neural networks
  - Bayesian networks
- Bayesian networks (BN) are based on probabilities.
  BN's represents in a most compact way the joint distribution of all relevant variables by exploiting known conditional independences.

Determining the probabilities

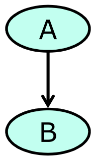
The basis for conditional probabilities in a Bayesian network can have a different epistemological status, ranging from well-founded theory over frequencies in a database to subjective estimates.

Causality

- A way of structuring a situation for reasoning under uncertainty is to construct a graph representing causal relations between events.

- A causal network consists of a set of variables and a set of directed links between variables.

A is a **parent** of B

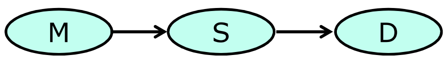


B is a **child** of A

- Variables represent events and may have any number of states, but are in exactly one of its states (mutually exclusiveness).

Causal networks

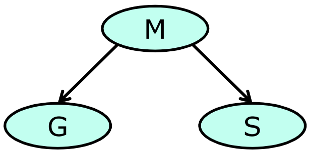
- Serial connections:



- D provide information about M, as long as the state of SD is not known with certainty.
  - ⟹ M and D are d-separated given SD
  - ⟹ Evidence may be transmitted through a serial connection unless the state of the variable in the connection is known.

Causal networks

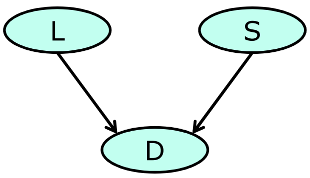
- Diverging connections:



- S provide information about G, as long as the state of M is not known with certainty.
  - ⟹ G and S are d-separated given M
  - ⟹ Evidence may be transmitted through a diverging connection unless the state of the variable in the connection is known.

Causal networks

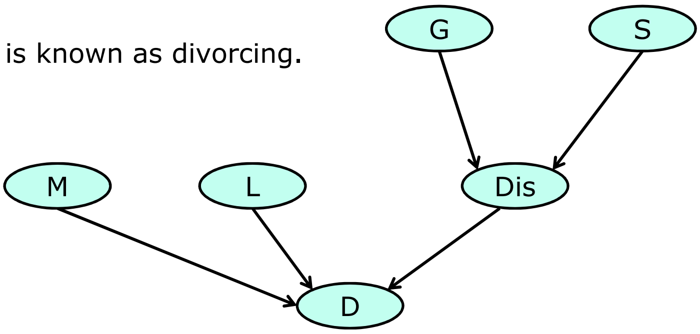
- Converging connections:



- S provide information about L, when the state of D is known with certainty.
  - ⟹ L and S are d-connected given D.
  - ⟹ Evidence may be transmitted through a converging connection if the state of the variable in the connection or one of its descendants receive evidence.

Divorcing

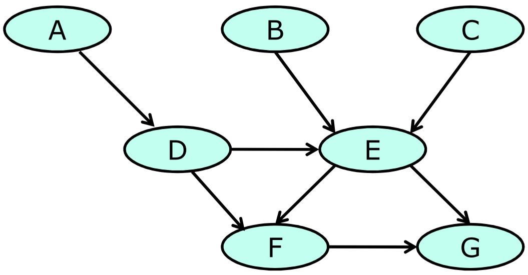
- Beneficial to have as few parents as possible.
- The number of parents can be reduced by intermediate variables.
- This is known as divorcing.



Eidgenossische Technische Hochschule Zürich
Swiss Federal Institute of Technology Zürich
Institute of Structural Engineering
Group Risk and Safety

d-Separation

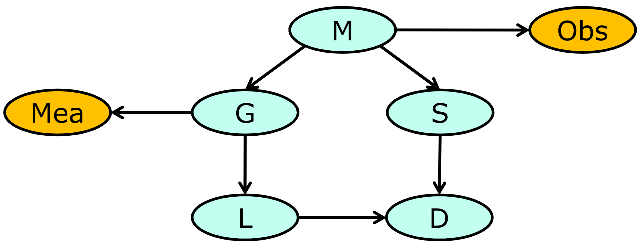
- Two distinct variables A and B in a causal network are d-separated if, for all paths between A and B, there is an intermediate variable V such that either
  - The connection is serial/diverging and V is instantiated
  - The connection is converging and neither V nor any of V's descendants have received evidence.



- D-separated parts can be computed separately.

Indicators

- Include all observations, i.e. indicators.
- Indicators are the input nodes used to update the network.



Eidgenossische Technische Hochschule Zürich
Swiss Federal Institute of Technology Zürich
Institute of Structural Engineering
Group Risk and Safety

Model Building

- Helpful to distinguish between:
  - Variables of interest (e.g. damage on structure, failure of a system). What we want to predict with the model?
  - Observable variables (e.g. PGA). What can we observe with certainty or quantifiable uncertainty?
  - Intermediate and unobservable variables (e.g. Spectral displacement). Parameters of a physical model that are computable.

Example for Model Building

- We are interested in possible damages on structures (D) due to earthquakes.

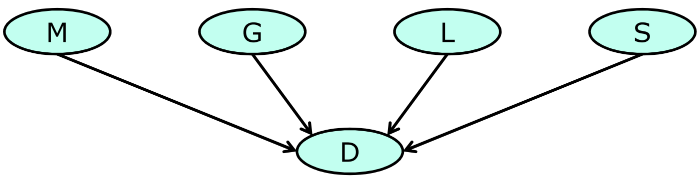
- First we identify, which parameters prevail this situation.

  - Magnitude of the earthquake (M)
  - Peak ground acceleration (PGA)
  - Spectral displacement (SD)
  - Soil liquefaction (LIQ)

Eidgenossische Technische Hochschule Zürich
Swiss Federal Institute of Technology Zürich
Institute of Structural Engineering
Group Risk and Safety

Example for Model Building

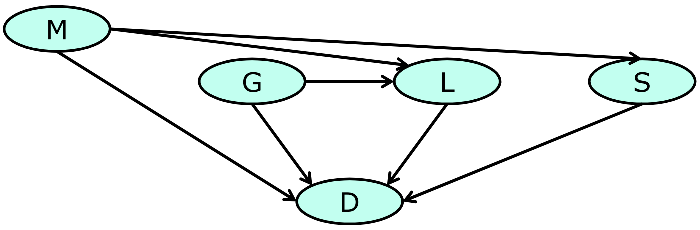
- In a first attempt we set up a network with only input and output variables.



M: Magnitude
G: Peak ground acceleration (PGA)
L: Liquefaction
S: Spectral displacement
D: Structural damage

Example for Model Building

- The model with non-causal relations (diagnostic model) is not correct:
  - We know that PGA and SD are not independent from M and LIQ is not independent from PGA.

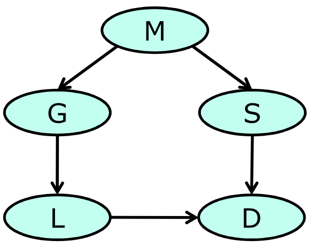


- A graphical model automatically reveals the analyst's intuitive and analytical understanding of the problem.

ETH Eidgenossische Technische Hochschule Zürich
Swiss Federal Institute of Technology Zürich
ibk Institute of Structural Engineering
Group Risk and Safety

Example for Model Building

- Most often humans can communicate sensibly about causal relations in a knowledge domain.
- Whenever possible the graphical model should represent causality, because this leads to minimal number of links.



Performance of Inference

- The performance of inference depends upon a number of factors. The most dominating factors are (assuming that all nodes are discrete):
  - the number of nodes,
  - the number of states of the nodes,
  - the number of parents of each node,
  - the density of the graph,
  - the lengths of any undirected cycle in the graph.

- In addition it is necessary to take other factors into account such as the structure of the evidence, the number of queries performed on the model, the algorithm used for inference, the triangulation of the graph, etc.

Bayesian networks

- A Bayesian network consists of:
  - A set of variables and a set of directed links between variables.
  - The variables are continuous or discretised into mutually exclusive states.
  - All variables and links form a DAG.
  - For each variable A with parents $B_1,\ldots,B_n$, the conditional probability tables $P(A \mid B_1,\ldots,B_n)$ are assigned.

- There is no requirement that the links represent causal relation.
- However, the d-separation property implied by the structure must hold.

Bayesian networks

- Let $U = (A_1, \ldots, A_n)$ be the universe of variables.
- If we have access to the joint probability table $P(U)$, then we can calculate any marginal distribution of any variable in U.
- $P(U)$ grows exponentially with number of variables.
  $\Longrightarrow$ Need for compact representation for $P(U)$.

- A Bayesian network over U is such a representation.

$$P(U) = \prod_i P(A_i \mid pa(A_i))$$

where $pa(A_i)$ is the parent set of A_i

Discretisation

- BN’s may have only discrete or only continuous variables. Hybrid BN’s are also allowed only when none of the continuous variables have discrete child nodes.

- Most variables are continuous. Therefore, their state space must be divided in discrete intervals.

  - Equidistant
  - Equal frequency
  - Supervised discretisation

Advantages of Bayesian networks

- BN's are powerful, when additional information becomes available due to Bayesian nature.
- Easy inference of the effect of the evidence on all variables in the model.
- The evaluation can be automated and embedded in a software, in contrast to simulation techniques or structural reliability analysis for computing probabilities.

But,

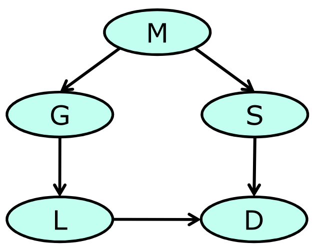
- Discretisation of the variables may be tedious and erroneous in particular when dealing with small probabilities.

Eidgenossische Technische Hochschule Zürich
Swiss Federal Institute of Technology Zürich

Institute of Structural Engineering
Group Risk and Safety

Example

- Let's use following BN for introducing the algorithms.

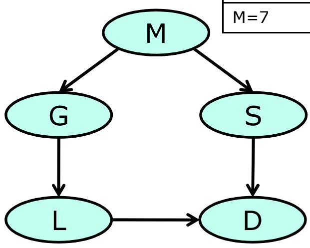


- The full probability distribution of the variables is:

$$P(M,G,S,L,D) = P(M)P(G\,|\,M)P(S\,|\,M)P(L\,|\,G)P(D\,|\,L,S)$$

Example

The conditional probability tables are:



M=0	0.9
M=7	0.1

	M=0	M=7
S=0	0.9	0.1
S=10cm	0.1	0.9

	L=yes		L=no	
	S=0	S=10	S=0	S=10
No	0.2	0.1	0.9	0.3
Collapse	0.8	0.9	0.1	0.7

	M=0	M=7
G=0	0.9	0.2
G=0.5g	0.1	0.8

	G=0	G=0.5g
L=yes	0.1	0.7
L=no	0.9	0.3

Algorithms

If a BN has only variables with discrete states, then it can be solved with exact inference, given that it is not too complex.

- **Bucket elimination** (variable elimination)
- Cluster algorithms (**junction tree**)

In case, when the space requirements cannot be met by the available hardware (e.g. too dense network, too many nodes and states) approximate methods may be used.

- Rejection sampling
- Likelihood weighting
- Self-Importance
- Adaptive importance
- Markov chain Monte-Carlo algorithm, e.g. **Gibbs sampling**

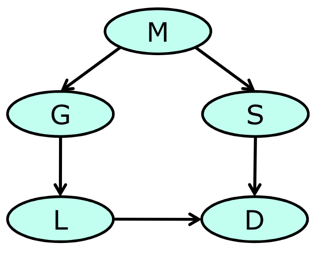
Bucket elimination - Terminology

- Tables used in BN's are generally called potentials (ϕ).
- A potential is a real-valued table over a domain of variables.
- Potentials can be multiplied and/or summed.
  - Unification $dom(\phi_1\phi_2) = dom(\phi_1) \bigcup dom(\phi_2)$
  - Commutative $\phi_1\phi_2 = \phi_2\phi_1$
  - Associative $(\phi_1\phi_2)\phi_3 = \phi_1(\phi_2\phi_3)$
  - Existence of unity $1 \cdot \phi = \phi$
- A potential can be marginalized.
  - Commutative $\sum_A \sum_B \phi = \sum_B \sum_A \phi$
  - Unit potential property $\sum_A P(A|V) = 1$
  - Distributive $\sum_A \phi_1\phi_2 = \phi_1 \sum_A \phi_2$ if $A \notin dom(\phi_1)$

Bucket elimination

The potentials are:

$$\phi_1 = P(M)$$
$$\phi_2 = P(G|M)$$
$$\phi_3 = P(S|M)$$
$$\phi_4 = P(L|G)$$
$$\phi_5 = P(D|S,L)$$



Calculating $P(D)$:

According to chain rule for BN's: $P(U) = \phi_1\phi_2\phi_3\phi_4\phi_5$

$$P(U) = \prod_i P(A_i \mid pa(A_i))$$

We need to marginalize out all variables except D:

$$P(D) = \sum_{M,G,S,L} \phi_1\phi_2\phi_3\phi_4\phi_5$$

Bucket elimination

Elimination order: M-P-S-L

M $$P(G,S,L,D)=\sum_M \phi_1(M)\phi_2(M,G)\phi_3(M,S)\phi_4(L,G)\phi_5(D,S,L)$$
$$=\phi_4(L,G)\phi_5(D,S,L)\sum_M \phi_1(M)\phi_2(M,G)\phi_3(M,S)$$

P $$P(S,L,D)=\sum_G \phi_4(L,G)\phi_5(D,S,L)\phi_1^{'}(G,S)$$
$$=\phi_5(D,S,L)\sum_G \phi_4(L,G)\phi_1^{'}(G,S)$$

S $$P(L,D)=\sum_S \phi_5(D,S,L)\phi_4^{'}(L,S)$$
$$=\sum_S \phi_5(D,S,L)\phi_4^{'}(L,S)$$

L $$P(D)=\sum_L \phi_5^{'}(D,L)$$

$$\Longrightarrow P(D)=\sum_L \sum_S \phi_5(D,S,L)\sum_G \phi_4(L,G)\sum_M \phi_1(M)\phi_2(M,G)\phi_3(M,S)$$

Eidgenossische Technische Hochschule Zürich
Swiss Federal Institute of Technology Zürich
Institute of Structural Engineering
Group Risk and Safety

Example – Bucket elimination

$$P(D)=\sum_L \sum_S \phi_5(D,S,L)\sum_G \phi_4(L,G)\sum_M \phi_1(M)\phi_2(M,G)\phi_3(M,S)$$

M=0	0.9
M=7	0.1

	M=0	M=7
G=0	0.9	0.2
G=0.5g	0.1	0.8

	M=0	M=7
S=0	0.9	0.1
S=10cm	0.1	0.9

Multiply:

	S=0		S=10	
	M=0	M=7	M=0	M=7
G=0	0.9*0.9*0.9= 0.729	0.2*0.1*0.1= 0.002	0.9*0.9*0.1= 0.081	0.2*0.1*0.9= 0.018
G=0.5g	0.1*0.9*0.9= 0.081	0.8*0.1*0.1= 0.008	0.1*0.9*0.1= 0.009	0.8*0.1*0.9= 0.072

Marginalize M:

$\phi_1^{'}(G,S)$

	S=0	S=10
G=0	0.729+0.002= 0.731	0.081+0.018= 0.099
G=0.5g	0.081+0.008= 0.089	0.009+0.072= 0.081

Eidgenossische Technische Hochschule Zürich
Swiss Federal Institute of Technology Zürich

Institute of Structural Engineering
Group Risk and Safety

Example – Bucket elimination

$$P(D)=\sum_L \sum_S \phi_5(D,S,L) \sum_G \phi_4(L,G)\phi_1'(G,S)$$

	G=0	G=0.5g
L=yes	0.1	0.7
L=no	0.9	0.3

	S=0	S=10
G=0	0.731	0.099
G=0.5g	0.089	0.081

Multiply:

	L=yes		L=no	
	G=0	G=0.5g	G=0	G=0.5g
S=0	0.731*0.1= 0.0731	0.089*0.7= 0.0623	0.731*0.9= 0.6579	0.089*0.3= 0.0267
S=10	0.099*0.1= 0.0099	0.081*0.7= 0.0567	0.099*0.9= 0.0891	0.081*0.3= 0.0243

Marginalize P:

$\phi_4'(L,S)$

	L=yes	L=no
S=0	0.0731+0.0623= 0.1354	0.6579+0.0267= 0.6846
S=10	0.0099+0.0567= 0.0666	0.0891+0.0243= 0.1134

Eidgenossische Technische Hochschule Zürich
Swiss Federal Institute of Technology Zürich

Institute of Structural Engineering
Group Risk and Safety

Example – Bucket elimination

$$P(D)=\sum_L \sum_S \phi_5(D,S,L)\phi_4'(L,S)$$

	L=yes		L=no	
	S=0	S=10	S=0	S=10
No	0.2	0.1	0.9	0.3
Collapse	0.8	0.9	0.1	0.7

	L=yes	L=no
S=0	0.1354	0.6846
S=10	0.0666	0.1134

Multiply:

	L=yes		L=no	
	S=0	S=10	S=0	S=10
No	0.2*0.1354= 0.02708	0.1*0.0666= 0.00666	0.9*0.6846= 0.61614	0.3*0.1134= 0.03402
Collapse	0.8*0.1354= 0.10832	0.9*0.0666= 0.05994	0.1*0.6846= 0.06846	0.7*0.1134= 0.07938

Marginalize S:

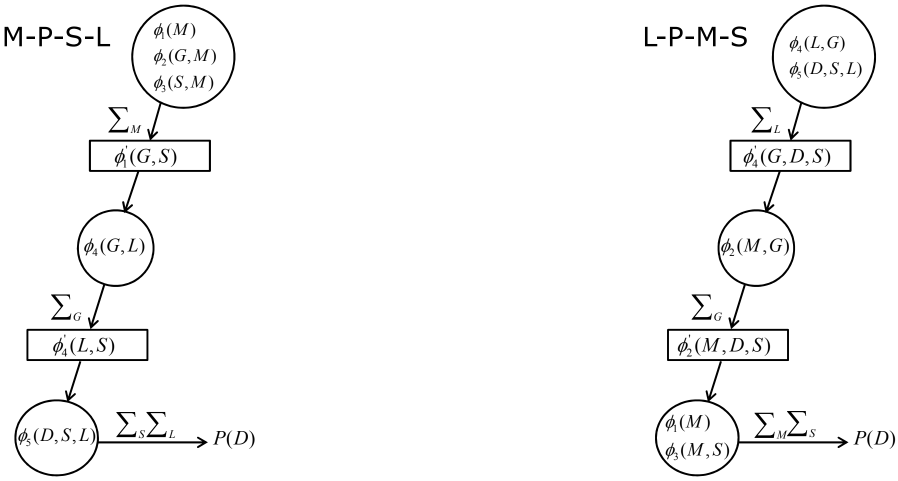
$\phi_5'(D,L)$

	L=yes	L=no
No	0.02708+0.00666=0.03374	0.61614+0.03402=0.65016
Collapse	0.10832+0.05994=0.16826	0.06846+0.07938=0.14784

Marginalize L: $P(D)$

No	0.03374+0.65016=0.6839
Collapse	0.16826+0.07938=0.3161

Bucket elimination – Elimination order



Task: finding an elimination order yielding the smallest domains to handle

Eidgenossische Technische Hochschule Zürich
Swiss Federal Institute of Technology Zürich

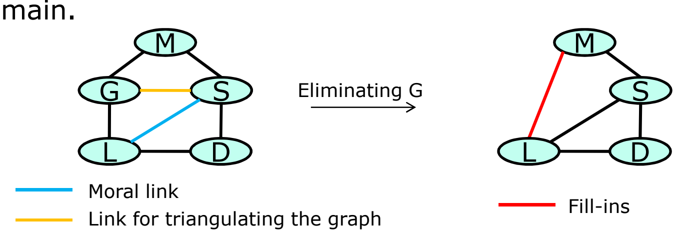
Institute of Structural Engineering
Group Risk and Safety

Graph Theory

Overview of the consequences of various elimination orders graphically.

Domain graphs

Undirected graph with variables of the universe as nodes and links between pairs of variables being members of the same domain.



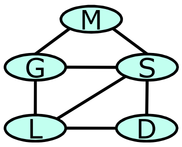
Fill-ins means having new potentials. Try to avoid fill-ins !!!

Graph Theory

An elimination sequence that does not introduce fill-ins requires less space than elimination sequences that introduces fill-ins.

In the graph theoretic framework, to calculate P(D) corresponds to constructing an elimination sequence ending with D.

For the domain graph it is possible to eliminate down to D without introducing fill-ins: M-G-L-S



Such a sequence is called a perfect elimination sequence.

Graph Theory

Which one is optimal, when there are several perfect elimination sequences?

The complexity of using a particular elimination sequence is characterized by the set of domains used.

for M-G-S-L $\{\{G,M,S\},\{G,L,S\},\{D,S,L\},\{L,D\}\}$

for M-G-L-S $\{\{G,M,S\},\{G,L,S\},\{D,S,L\},\{S,D\}\}$

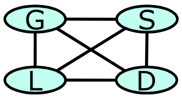
Domain set:

Set of domains of potentials produced during elimination.

Graph Theory

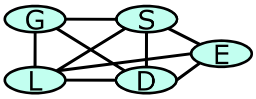
Complete set:

A set of nodes is complete if all nodes are pair wise linked



Clique set:

All perfect elimination sequences produce the same domain set, namely the set of cliques of the domain graph.

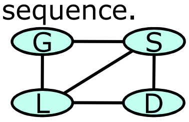


Any perfect elimination sequence ending with the variable D is optimal with respect to calculating P(D).

Graph Theory

Triangulated graph:

An undirected graph with a perfect elimination sequence.



Simplicial node:

Nodes with a complete neighbor set. (G,D)

Determining clique set in a graph:

Generally NP-hard. For triangulated graphs following easy procedure:

1. Eliminate simplicial node X. Set of neighbors of X plus X (Fx) is a clique candidate.
2. If Fx does not include all remaining nodes, go to 1.
3. Prune the set of cliques by removing sets that are subsets of other clique candidates.

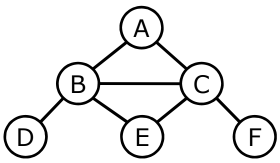
The resulting set is the set of cliques.

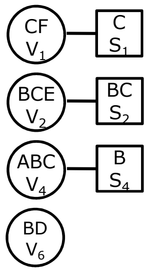
Join trees

- Definition: Let G be the set of cliques from an undirected graph, and let the cliques of G be organized in a tree. T is a join tree if for any pair of nodes V, W all nodes on the path between V and W contain the intersection $V \cap W$.

a. The cliques of all triangulated graphs can be organized in a join tree.

b. When the cliques of G can be organized in a join tree, then G is a triangulated graph.

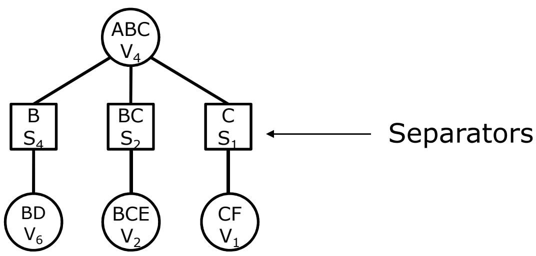




Elimination sequence: F-E-C-A-B-D

Join trees

- Join trees with separators: Separators are the variables which are common to several cliques.



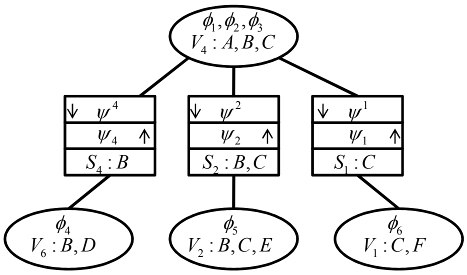
Eidgenossische Technische Hochschule Zürich
Swiss Federal Institute of Technology Zürich

Institute of Structural Engineering
Group Risk and Safety

Junction trees

- Junction trees are join trees where additionally
  - all potentials ϕ are assigned to a clique containing dom(ϕ).
  - each link has the appropriate separator assigned.
  - each separator contains two mailboxes.

- The messages in the mailboxes are sets of potentials.

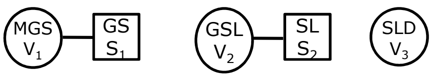


Eidgenossische Technische Hochschule Zürich
Swiss Federal Institute of Technology Zürich

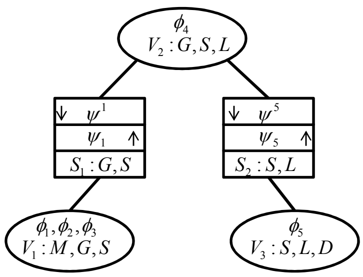
Institute of Structural Engineering
Group Risk and Safety

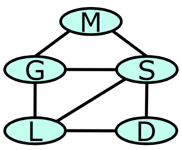
Junction trees – Example

- Cliques and separators:



- Construct the junction tree:





$$\psi_1 = \sum_M \phi_1 \phi_2 \phi_3$$
$$\psi^5 = \sum_G \phi_4 \sum_M \phi_1 \phi_2 \phi_3$$
$$P(D) = \sum_L \sum_S \phi_5 \sum_G \phi_4 \sum_M \phi_1 \phi_2 \phi_3$$

Elimination sequence: M-G-S-L

Triangulation of graphs

Definition:

Let V be a set of variables. For $X \notin V$, n(X) denotes the number of states of X. The size of V, sz(V) is the product $\Pi_V n(X)$. Let G be a triangulated graph extending the BN's moral graph and let $V_1 \ldots, V_n$ be the cliques of G. The size of G is the sum $size(G)=\sum sz(V_i)$

Heuristic:

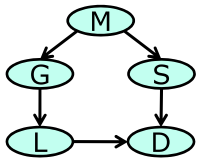
Eliminate repeatedly a simplicial node and if this not possible, eliminate a node X of minimal $sz(F_X)$

Stochastic Simulation

- For large problems the cliques in the triangulated graphs can be very large and space requirements can not be met.
- Approximate methods can then be applied. Stochastic Simulation is such an approximate method.
- The idea behind the simulation is that the causal model is used to simulate the flow of impact.

Example

M=0	0.9
M=7	0.1



	M=0	M=7
G=0	0.9	0.2
G=0.5g	0.1	0.8

	M=0	M=7
S=0	0.9	0.1
S=10cm	0.1	0.9

	P=0	P=0.5g
L=yes	0.1	0.7
L=no	0.9	0.3

	L=yes		L=no	
	S=0	S=10	S=0	S=10
No	0.2	0.1	0.9	0.3
Collapse	0.8	0.9	0.1	0.7

The idea is to draw a random configuration of the variables (M,G,S,L,D) and to do this a sufficient number of times.

Example – Forward sampling

First the state of M is sampled. According to the distribution the state is assigned, then another random number is drawn and the state assigned according to the conditional probability table, etc. So one set of configuration is sampled.

Repeating 100000 times and sorting yield following table:

	SLD							
MG	**111**	**112**	**121**	**122**	**211**	**212**	**221**	**222**
11	1505	5921	58790	6492	76	701	2124	5240
12	1116	4609	2214	247	53	584	83	198
21	5	18	154	13	16	154	472	1117
22	129	453	217	26	492	4639	642	1500

$$P(M) = \frac{89953}{100000} = 0.8995$$

$$P(G) = \frac{82798}{100000} = 0.8280$$

$$P(S) = \frac{81909}{100000} = 0.8191$$

$$P(L) = \frac{21171}{100000} = 0.2117$$

$$P(D) = \frac{68088}{100000} = 0.6809$$

The probability distributions for the variables are calculated by counting in the sample set.

Example – Gibbs Sampling

- In forward sampling too many simulations required for small probabilities.
- In Gibbs sampling we start with a "probable" configuration and change randomly the state of variables in causal order. In one sweep through the variables, a new configuration is determined. Then this is used for the further simulation.
- In this way a large sample consistent with the observation is produced.
- Problems:
  - The initial configuration may be improbable
  - We may stuck in certain areas of the configuration
  - It may be very hard to find a starting configuration. (NP-hard)

Summary

- Basic features of building graphical models are discussed.
- Two exact inference algorithms (bucket elimination and junction tree) were introduced.
- One approximate inference algorithm (Markov chain Monte-Carlo simulation) was introduced.
- A simple Bayesian network was calculated using the three algorithms.
- The solution of the same BN was demonstrated using the commercial software HUGIN.

References

- Jensen, F.V. (2001). **Bayesian networks and decision graphs**, Springer Verlag, New York, USA, ISBN 0-387-95259-4.

- Pearl, J. (1988). **Probabilistic Reasoning in Intelligent Systems: Networks of Plausible Inference**, Morgan Kaufmann Publishers, California, USA, ISBN 0-934613-73-7.

- Hussain, F., Liu, H., Tan, C.L., and Dash, M. (1999). **Discretisation: An enabling technique**, Technical report TRC6/99, National University of Singapore, School of Computing.

- Castillo, E., Gutierrez, J.M., Hadi, A.S. (1997). **Expert Systems and probabilistic network models,** Springer Verlag, New York, USA, ISBN 0-387-94858-9.