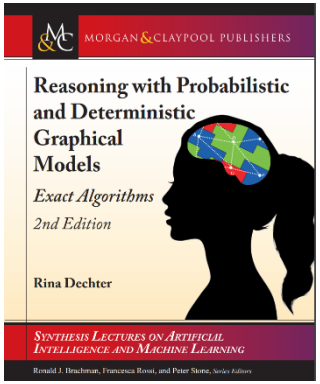
CS 295: Causal Reasoning

Rina Dechter

Exact Inference Algorithms
Bucket-elimination



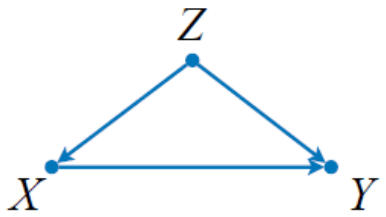
Chapter 4 Dechter's book

Factorizing Observational Distributions

Markovian Case

- The distribution $P(\mathbf{v})$ decomposes as:

$$P(\mathbf{v}) = \sum_{\mathbf{u}} P(\mathbf{u}) \prod_{V_i \in \mathbf{V}} P\left(v_i \mid v_1, \ldots, v_{i-1}, \mathbf{u}\right) = \sum_{\mathbf{u}} P(\mathbf{u}) \prod_{V_i \in \mathbf{V}} P\left(v_i \mid pa_i, u_i\right)$$



$$P(z,x,y) = \sum_{\mathbf{u}} P(\mathbf{u}) P(z \mid u_z) P(x \mid z, u_x) P(y \mid x, z, u_y)$$

$$= \left(\sum_{u_z} P(z \mid u_z) P(u_z) \right) \left(\sum_{u_x} P(x \mid z, u_x) P(u_x) \right) \left(\sum_{u_y} P(y \mid x, z, u_y) P(u_y) \right)$$

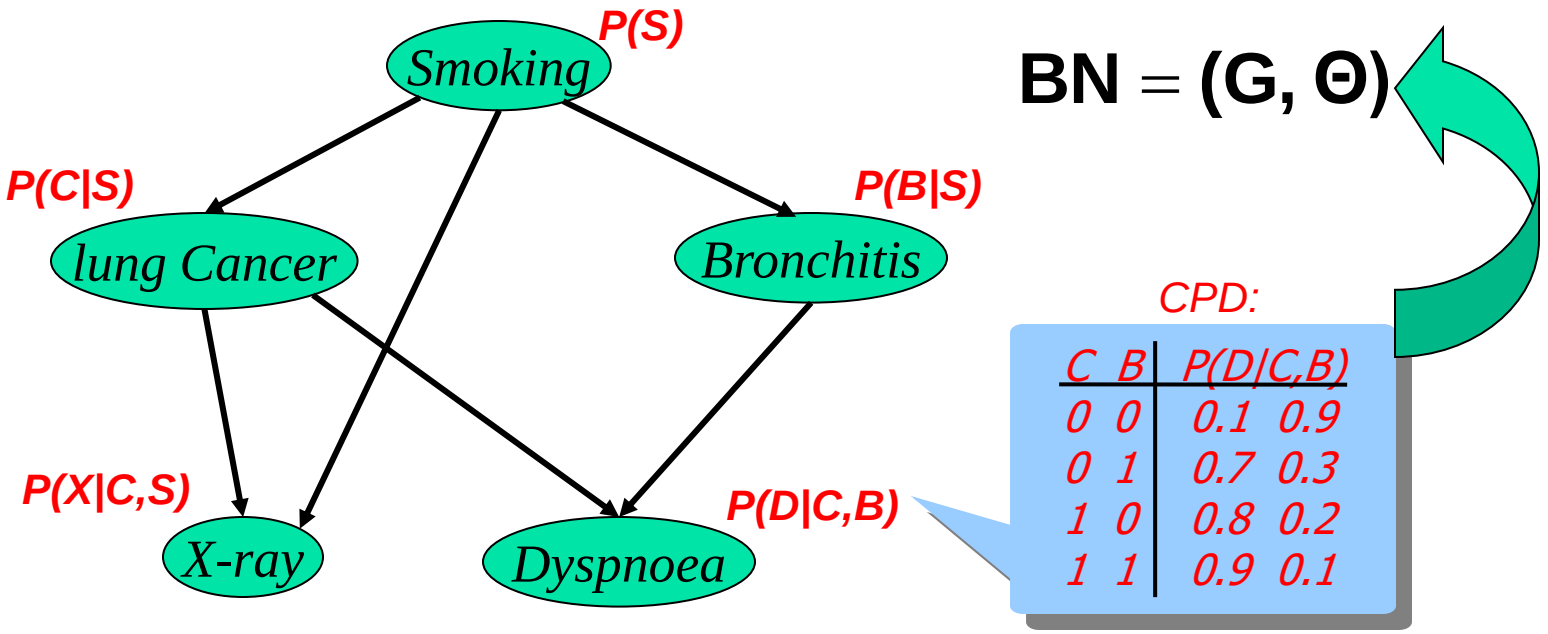
$$= P(z) P(x \mid z) P(y \mid x, z)$$

- In Markovian models, $P(v_i \mid pa_i)$ can be seen as “canonical factors”.

Inference for probabilistic networks

- Bucket elimination
  - Belief-updating, P(e), partition function
  - Marginals, probability of evidence
  - The impact of evidence
  - for MPE (→MAP)
  - for MAP (→ Marginal Map)
- Induced-Width

Bayesian Networks: Example (Pearl, 1988)

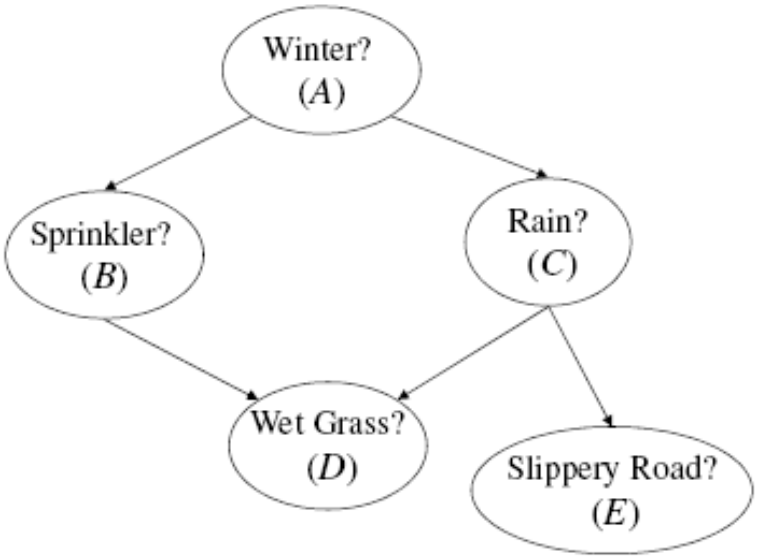


$$P(S, C, B, X, D) = P(S)\,P(C|S)\,P(B|S)\,P(X|C,S)\,P(D|C,B)$$

Belief Updating:

P (lung cancer=yes | smoking=no, dyspnoea=yes) = ?

A Bayesian Network



A	Θ_A
true	.6
false	.4

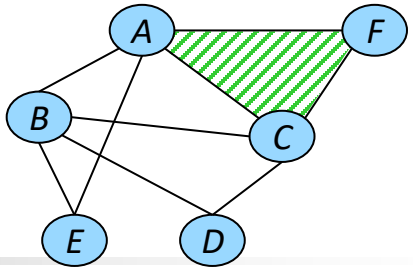
A	B	$\Theta_{B\|A}$
true	true	.2
true	false	.8
false	true	.75
false	false	.25

A	C	$\Theta_{C\|A}$
true	true	.8
true	false	.2
false	true	.1
false	false	.9

B	C	D	$\Theta_{D\|BC}$
true	true	true	.95
true	true	false	.05
true	false	true	.9
true	false	false	.1
false	true	true	.8
false	true	false	.2
false	false	true	0
false	false	false	1

C	E	$\Theta_{E\|C}$
true	true	.7
true	false	.3
false	true	0
false	false	1

Types of queries

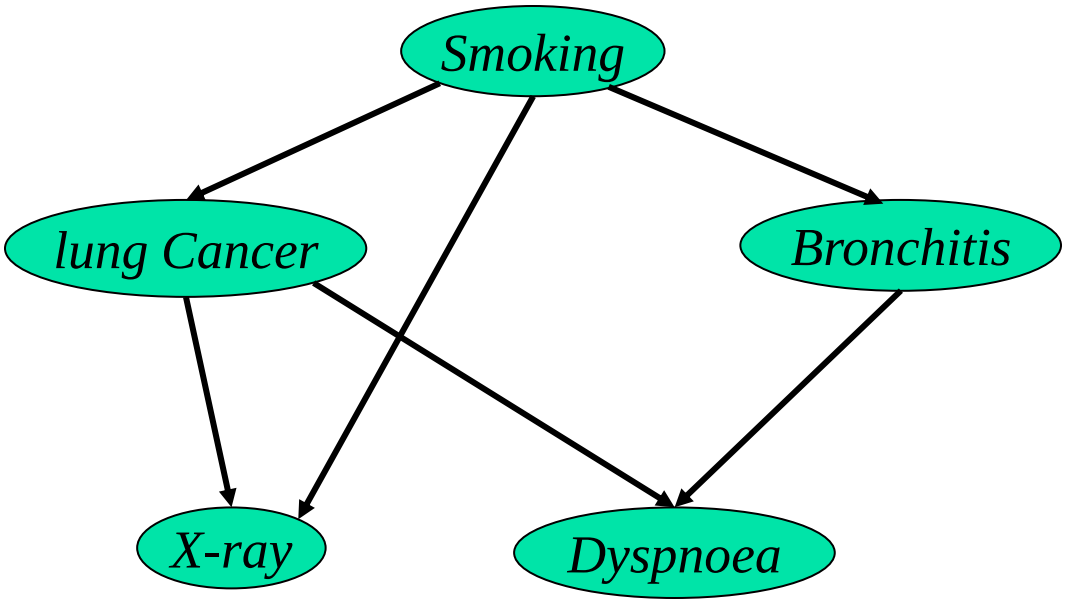


Max-Inference	$f(\mathbf{x}^*) = \max_{\mathbf{x}} \prod_{\alpha} f_\alpha(\mathbf{x}_\alpha)$
Sum-Inference	$Z = \sum_{\mathbf{x}} \prod_{\alpha} f_\alpha(\mathbf{x}_\alpha)$
Mixed-Inference	$f(\mathbf{x}_M^*) = \max_{\mathbf{x}_M} \sum_{\mathbf{x}_S} \prod_{\alpha} f_\alpha(\mathbf{x}_\alpha)$

Harder

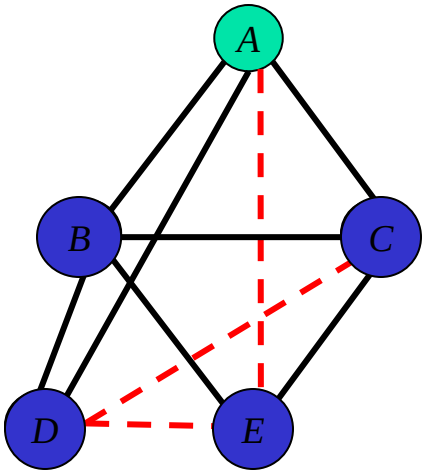
- **NP-hard**: exponentially many terms
- We will focus on exact and then on **approximation** algorithms
  - **Anytime**: very fast & very approximate ! Slower & more accurate

Belief Updating



P (lung cancer=yes | smoking=no, dyspnoea=yes) = ?

Belief updating: P(X|evidence)=?



"Moral" graph

$$P(a|e=0) \propto P(a,e=0)=$$

$$\sum_{e=0,d,c,b} P(a)P(b|a)P(c|a)P(d|b,a)P(e|b,c)=$$

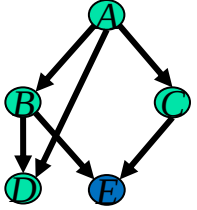
$$P(a)\sum_{e=0}\sum_{d}\sum_{c}P(c|a)\sum_{b}P(b|a)P(d|b,a)P(e|b,c)$$

Variable Elimination

$h^B(a,d,c,e)$

Bucket elimination

Algorithm *BE-bel* (Dechter 1996)



$$P(A \mid E=0) = \alpha \sum_{E=0,D,C,B} P(A)\cdot P(B \mid A)\cdot P(C \mid A)\cdot P(D \mid A,B)\cdot P(E \mid B,C)$$

$\sum\prod_{b}$ ← *Elimination operator*

bucket B: $P(b|a)$ $P(d|b,a)$ $P(e|b,c)$

bucket C: $P(c|a)$ $\lambda^{B}(a,d,c,e)$

bucket D: $\lambda^{C}(a,d,e)$

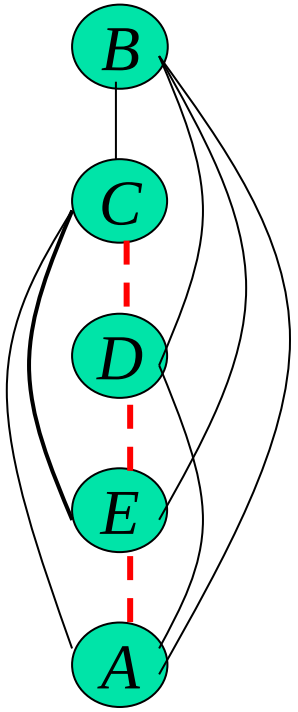
bucket E: $e=0$ $\lambda^{D}(a,e)$

bucket A: $P(a)$ $\lambda^{E}(a)$

$P(e=0)$

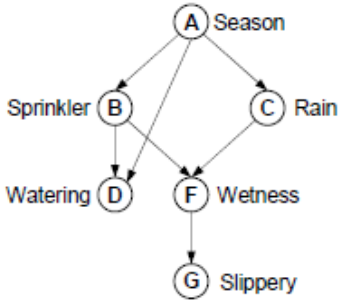
$P(a,e=0)$

W=4*
"induced width"
(max clique size)

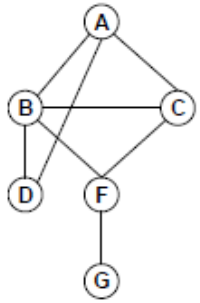


$$P(a|e=0)=\frac{P(a,e=0)}{P(e=0)}$$

A Bayesian Network
Ordering: A,C,B,E,D,G



(a) Directed acyclic graph



(b) Moral graph

$$P(a, g=1) = \sum_{c,b,e,d,g=1} P(a,b,c,d,e,g) = \sum_{c,b,f,d,g=1} P(g|f)P(f|b,c)P(d|a,b)P(c|a)P(b|a)P(a).$$

$$P(a, g=1) = P(a)\sum_{c} P(c|a)\sum_{b} P(b|a)\sum_{f} P(f|b,c)\sum_{d} P(d|b,a)\sum_{g=1} P(g|f). \quad (4.1)$$

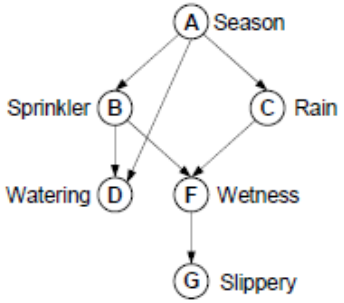
$$P(a, g=1) = P(a)\sum_{c} P(c|a)\sum_{b} P(b|a)\sum_{f} P(f|b,c)\lambda_G(f)\sum_{d} P(d|b,a). \quad (4.2)$$

$$P(a, g=1) = P(a)\sum_{c} P(c|a)\sum_{b} P(b|a)\lambda_D(a,b)\sum_{f} P(f|b,c)\lambda_G(f) \quad (4.3)$$

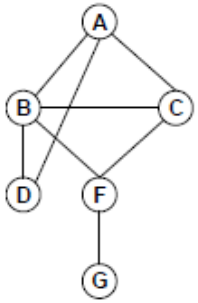
$$P(a, g=1) = P(a)\sum_{c} P(c|a)\sum_{b} P(b|a)\lambda_D(a,b)\lambda_F(b,c) \quad (4.4)$$

$$P(a, g=1) = P(a)\sum_{c} P(c|a)\lambda_B(a,c) \quad (4.5)$$

A Bayesian Network
Ordering: A,C,B,E,D,G



(a) Directed acyclic graph



(b) Moral graph

$$P(a, g=1) = \sum_{c,b,e,d,g=1} P(a,b,c,d,e,g) = \sum_{c,b,f,d,g=1} P(g|f)P(f|b,c)P(d|a,b)P(c|a)P(b|a)P(a).$$

$$P(a, g=1) = P(a)\sum_{c} P(c|a)\sum_{b} P(b|a)\sum_{f} P(f|b,c)\sum_{d} P(d|b,a)\sum_{g=1} P(g|f). \quad (4.1)$$

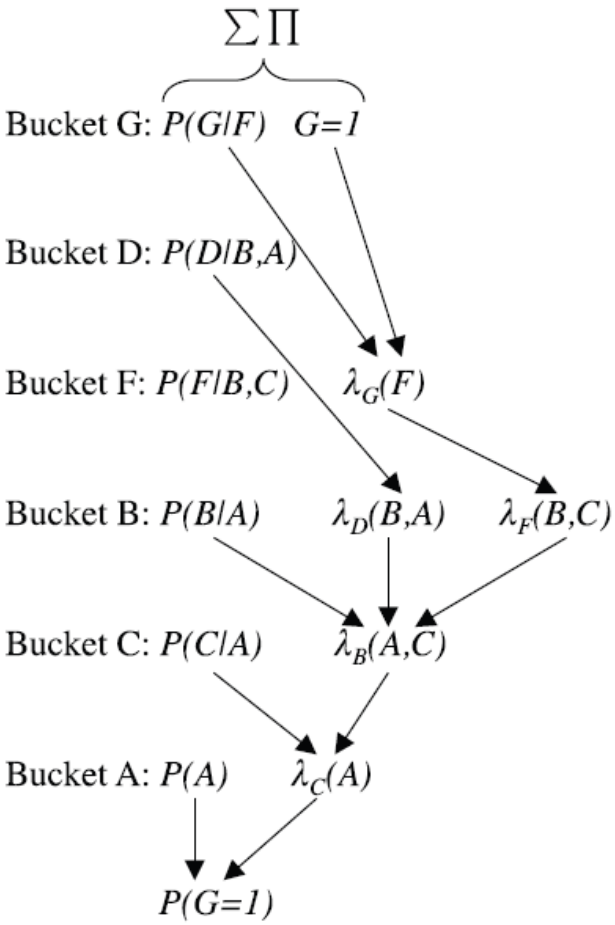
$$P(a, g=1) = P(a)\sum_{c} P(c|a)\sum_{b} P(b|a)\sum_{f} P(f|b,c)\lambda_G(f)\sum_{d} P(d|b,a). \quad (4.2)$$

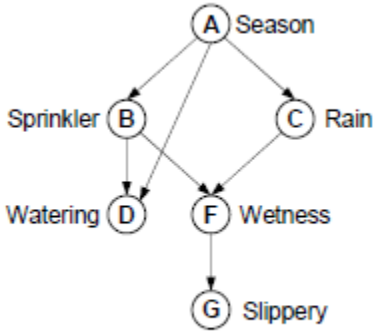
$$P(a, g=1) = P(a)\sum_{c} P(c|a)\sum_{b} P(b|a)\lambda_D(a,b)\sum_{f} P(f|b,c)\lambda_G(f) \quad (4.3)$$

$$P(a, g=1) = P(a)\sum_{c} P(c|a)\sum_{b} P(b|a)\lambda_D(a,b)\lambda_F(b,c) \quad (4.4)$$

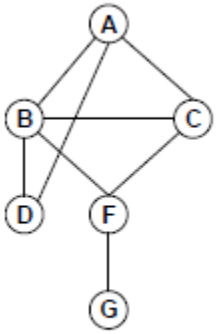
$$P(a, g=1) = P(a)\sum_{c} P(c|a)\lambda_B(a,c) \quad (4.5)$$

A Bayesian Network
Ordering: A,C,B,F,D,G



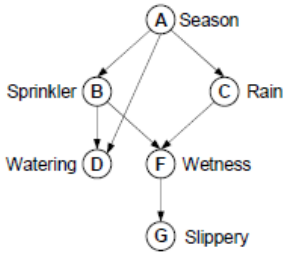


(a) Directed acyclic graph

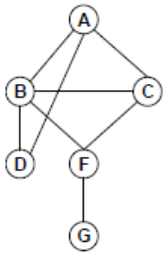


(b) Moral graph

A Different Ordering



(a) Directed acyclic graph



(b) Moral graph

Ordering: A,F,D,C,B,G

$$
\begin{aligned}
P(a, g=1) &= P(a) \sum_f \sum_d \sum_c P(c|a) \sum_b P(b|a)\ P(d|a,b) P(f|b,c) \sum_{g=1} P(g|f) \\
&= P(a) \sum_f \lambda_G(f) \sum_d \sum_c P(c|a) \sum_b P(b|a)\ P(d|a,b) P(f|b,c) \\
&= P(a) \sum_f \lambda_G(f) \sum_d \sum_c P(c|a) \lambda_B(a,d,c,f) \\
&= P(a) \sum_f \lambda_g(f) \sum_d \lambda_C(a,d,f) \\
&= P(a) \sum_f \lambda_G(f) \lambda_D(a,f) \\
&= P(a) \lambda_F(a)
\end{aligned}
$$

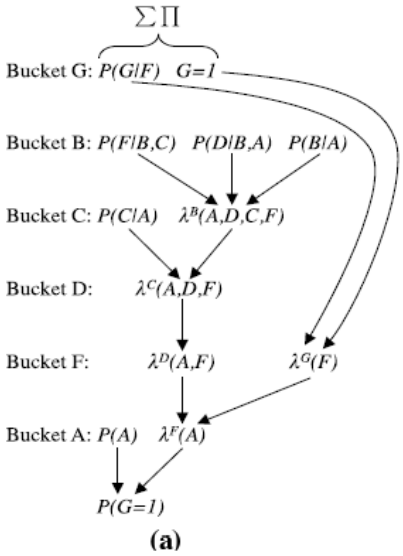
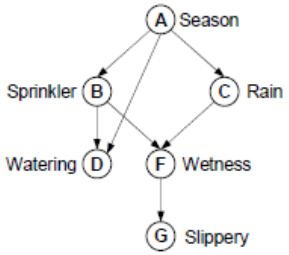
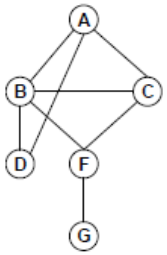


Figure 4.3: The bucket's output when processing along $d_2 = A, F, D, C, B, G$

A Different Ordering



(a) Directed acyclic graph



(b) Moral graph

Ordering: A,F,D,C,B,G

$$
\begin{aligned}
&P(a, g=1) = P(a)\sum_f \sum_d \sum_c P(c|a) \sum_b P(b|a)\ P(d|a,b) P(f|b,c) \sum_{g=1} P(g|f) \\
&= P(a)\sum_f \lambda_G(f) \sum_d \sum_c P(c|a) \sum_b P(b|a)\ P(d|a,b) P(f|b,c) \\
&= P(a)\sum_f \lambda_G(f) \sum_d \sum_c P(c|a) \lambda_B(a,d,c,f) \\
&= P(a)\sum_f \lambda_g(f) \sum_d \lambda_C(a,d,f) \\
&= P(a)\sum_f \lambda_G(f) \lambda_D(a,f) \\
&= P(a)\lambda_F(a)
\end{aligned}
$$

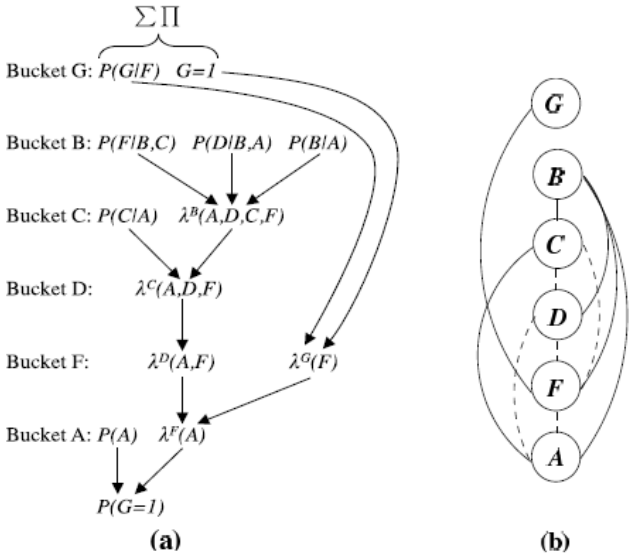
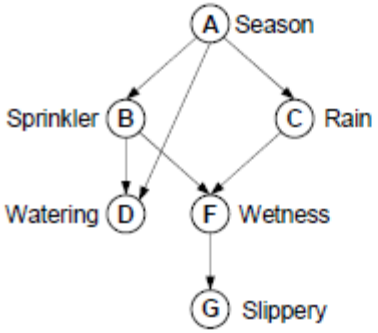
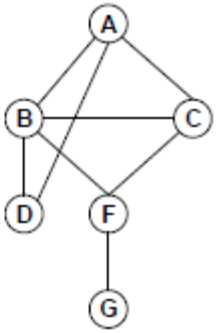


Figure 4.3: The bucket's output when processing along $d_2 = A, F, D, C, B, G$

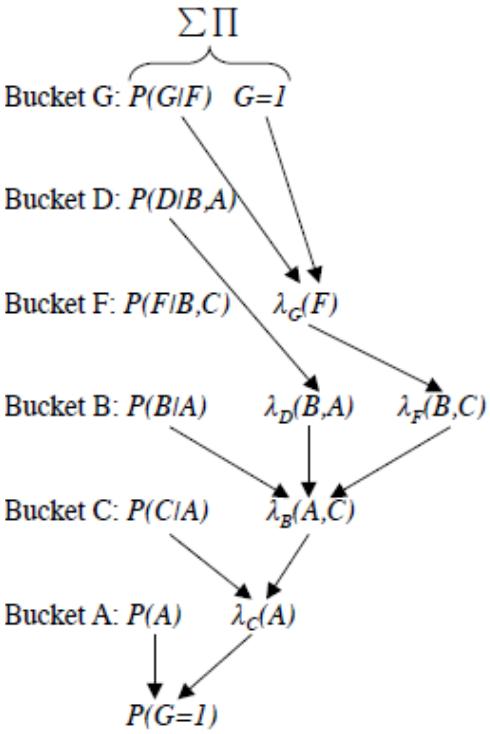
A Bayesian Network Processed Along 2 Orderings



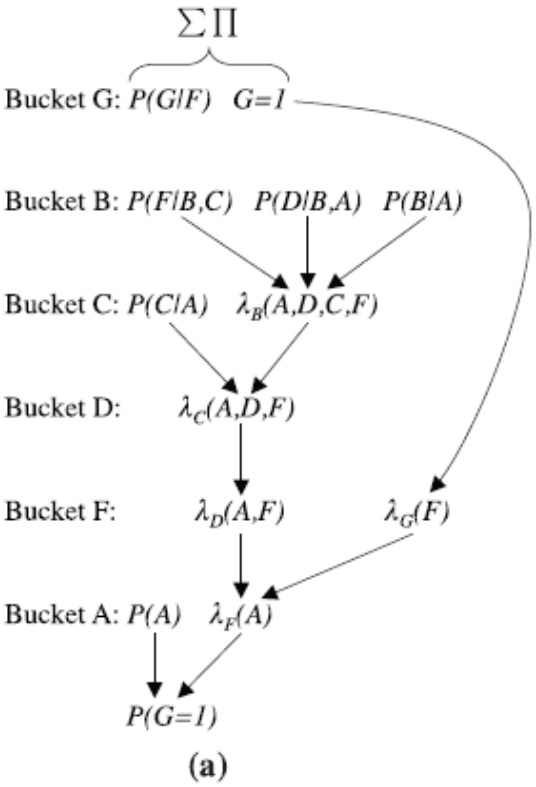
(a) Directed acyclic graph



(b) Moral graph



d1=A,C,B,F,D,G



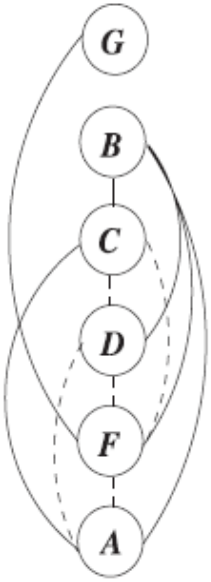
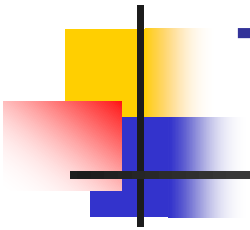


Figure 4.4: The bucket's output when processing along $d_2 = A, F, D, C, B, G$.



The Operation In a Bucket

- Multiplying functions
- Marginalizing (summing-out) functions

Combination of Cost Functions

A	B	f(A,B)
b	b	0.4
b	g	0.1
g	b	0
g	g	0.5

B	C	f(B,C)
b	b	0.2
b	g	0
g	b	0
g	g	0.8

A	B	C	f(A,B,C)
b	b	b	0.1
b	b	g	0
b	g	b	0
b	g	g	0.08
g	b	b	0
g	b	g	0
g	g	b	0
g	g	g	0.4

= 0.1 x 0.8

Factors: Sum-Out Operation

The result of summing out variable X from factor $f(\mathbf{X})$

is another factor over variables $\mathbf{Y} = \mathbf{X} \setminus \{X\}$:

$$\left(\sum_X f\right)(\mathbf{y}) \stackrel{def}{=} \sum_x f(x, \mathbf{y})$$

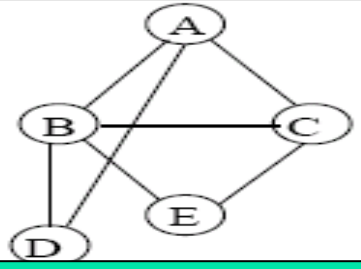
B	C	D	f_1
true	true	true	.95
true	true	false	.05
true	false	true	.9
true	false	false	.1
false	true	true	.8
false	true	false	.2
false	false	true	0
false	false	false	1

B	C	$\sum_D f_1$
true	true	1
true	false	1
false	true	1
false	false	1

	$\sum_B \sum_C \sum_D f_1$
$\top$	4

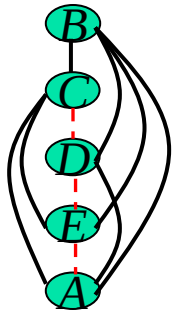
Thanks to Darwiche

Bucket Elimination and Induced Width



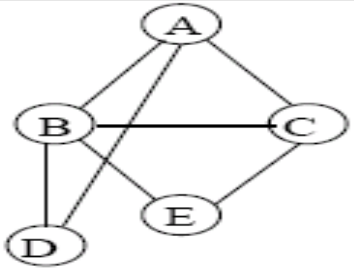
Ordering: a, e, d, c, b

$bucket(B) = \quad P(e|b,c), P(d|a,b), P(b|a)$
$bucket(C) = \quad P(c|a) \quad || \quad \lambda_B(a,c,d,e)$
$bucket(D) = \quad || \ \lambda_C(a,d,e)$
$bucket(E) = \quad e = 0 \quad || \ \lambda_D(a,c)$
$bucket(A) = \quad P(a) \quad || \quad \lambda_E(a)$



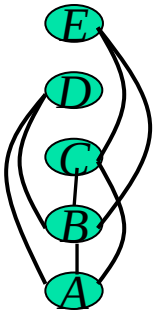
W*=4

Bucket Elimination and Induced Width



Ordering: a, b, c, d, e

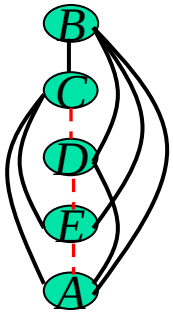
$bucket(E) = \quad P(e|b,c), \; e = 0$
$bucket(D) = \quad P(d|a,b)$
$bucket(C) = \quad P(c|a) \;\; || \;\; P(e = 0|b,c)$
$bucket(B) = \quad P(b|a) \;\; || \;\; \lambda_D(a,b), \lambda_C(b,c)$
$bucket(A) = \quad P(a) \;\; || \;\; \lambda_B(a)$



W*=2

Ordering: a, e, d, c, b

$bucket(B) = \quad P(e|b,c), P(d|a,b), P(b|a)$
$bucket(C) = \quad P(c|a) \;\; || \;\; \lambda_B(a,c,d,e)$
$bucket(D) = \quad || \; \lambda_C(a,d,e)$
$bucket(E) = \quad e = 0 \;\; || \;\; \lambda_D(a,c)$
$bucket(A) = \quad P(a) \;\; || \;\; \lambda_E(a)$



W*=4

ALGORITHM BE-BEL

Input: A belief network $\mathcal{B} = \langle \mathbf{X}, \mathbf{D}, \mathbf{P}_G, \prod \rangle$, an ordering $d = (X_1, \ldots, X_n)$; evidence e
output: The belief $P(X_1|\mathbf{e})$ and probability of evidence $P(\mathbf{e})$

1. Partition the input functions (CPTs) into $bucket_1$, ..., $bucket_n$ as follows: **for** $i \leftarrow n$ **downto** 1, put in $bucket_i$ all unplaced functions mentioning X_i. Put each observed variable in its bucket. Denote by ψ_i the product of input functions in $bucket_i$.
2. **backward: for** $p \leftarrow n$ **downto** 1 **do**
3. **for** all the functions $\psi_{S_0}, \lambda_{S_1}, \ldots, \lambda_{S_j}$ in $bucket_p$ **do**

   If (observed variable) $X_p = x_p$ appears in $bucket_p$,

   assign $X_p = x_p$ to each function in $bucket_p$ and then

   put each resulting function in the bucket of the *closest* variable in its scope.

   else,

4. $\lambda_p \leftarrow \sum_{X_p} \psi_p \cdot \prod_{i=1}^{j} \lambda_{S_i}$
5. place λ_p in bucket of the latest variable in scope(λ_p),
6. **return** (as a result of processing $bucket_1$):

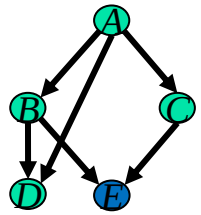
   $P(\mathbf{e}) = \alpha = \sum_{X_1} \psi_1 \cdot \prod_{\lambda \in bucket_1} \lambda$

   $P(X_1|\mathbf{e}) = \frac{1}{\alpha} \psi_1 \cdot \prod_{\lambda \in bucket_1} \lambda$

Figure 4.5: BE-bel: a sum-product bucket-elimination algorithm.

Belief Updating

Algorithm BE-bel *[Dechter 1996]*



$$p(A|E=0) = \alpha \sum_{e,d,c,b} p(A)\, p(b|A)\, p(c|A)\, p(d|A,b)\, p(e|b,c)\, \mathbb{1}[e=0]$$

$\sum_b \prod$ ← *Elimination & combination operators*

bucket B: $p(b|A)\ p(d|b,A)\ p(e|b,c)$

bucket C: $p(c|A)\quad \lambda_{B\to C}(A,d,c,e)$

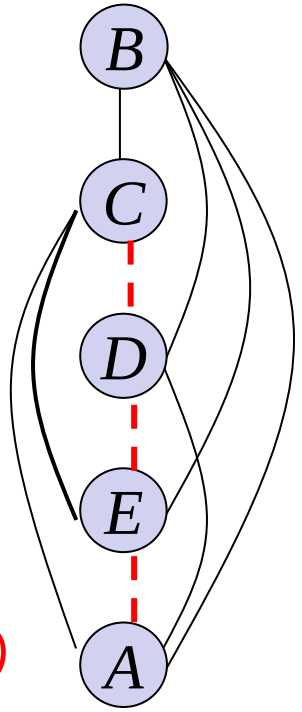
bucket D: $\lambda_{C\to D}(A,d,e)$

bucket E: $\mathbb{1}[E=0]\ \lambda_{D\to E}(A,e)$

bucket A: $p(A)\quad \lambda_{E\to A}(A)$

$p(E=0)$

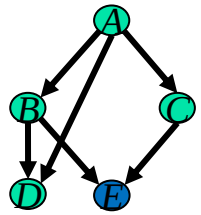
W=4*
"induced width"
(max clique size)



$p(A|E=0) = p(A,E=0)\,/\,p(E=0)$

Bucket Elimination

Algorithm BE-bel *[Dechter 1996]*



$$p(A|E=0) = \alpha \sum_{e,d,c,b} p(A)\, p(b|A)\, p(c|A)\, p(d|A,b)\, p(e|b,c)\, \mathbb{1}[e=0]$$

$\sum \prod$ ⟵ *Elimination & combination operators*

$\sum_b$

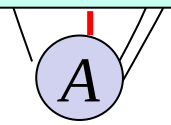
Time and space exponential in the induced-width / treewidth

bucket A: $p(A)$ $\lambda_{E \to A}(A)$

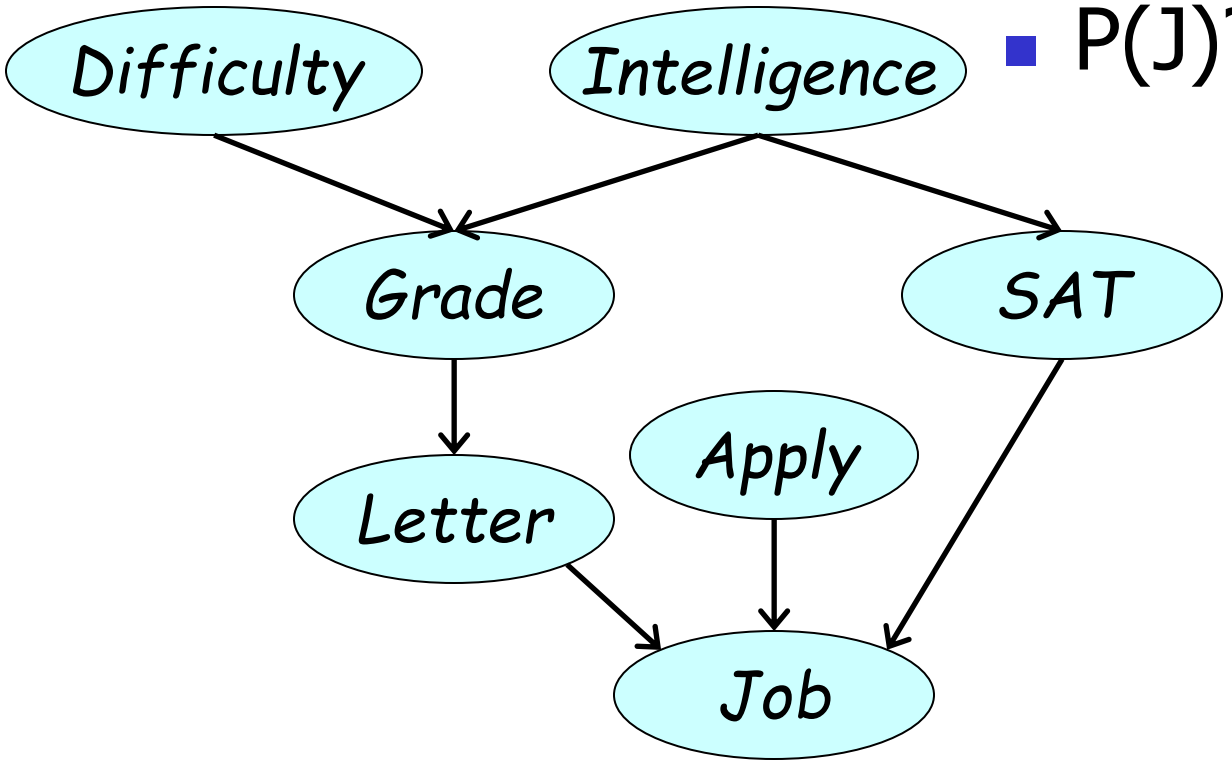
$p(E=0)$

$p(A|E=0) = p(A, E=0) \,/\, p(E=0)$

induced width (max clique size)



Student Network Example

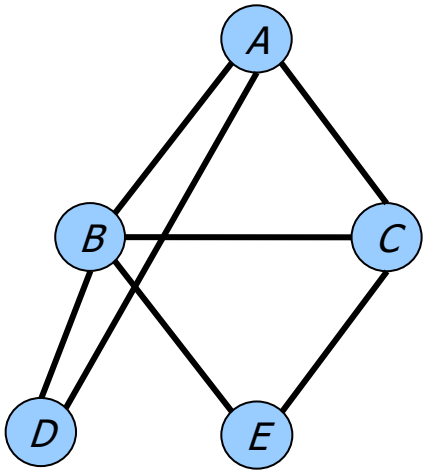


- P(J)?

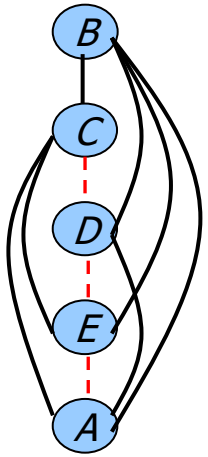
Induced Width (continued)

$w^*(d)$ – the induced width of the primal graph along ordering d

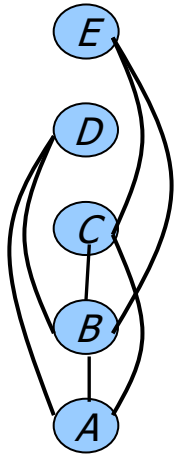
The effect of the ordering:



Primal (moraal) graph



$w^*(d_1) = 4$



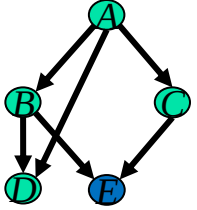
$w^*(d_2) = 2$

Inference for probabilistic networks

- Bucket elimination
  - Belief-updating, P(e), partition function
  - Marginals, probability of evidence
  - The impact of evidence
  - for MPE (→MAP)
  - for MAP (→ Marginal Map)
- Induced-Width

The impact of evidence?

Algorithm *BE-bel*



$$P(A \mid E=0) = \alpha \sum_{E=0,D,C,B} P(A)\cdot P(B \mid A)\cdot P(C \mid A)\cdot P(D \mid A,B)\cdot P(E \mid B,C)$$

$\sum\prod_{b}$ ← *Elimination operator*

bucket B: $P(b|a)$ $P(d|b,a)$ $P(e|b,c)$

bucket C: $P(c|a)$ $\lambda^{B}(a,d,c,e)$

bucket D: $\lambda^{C}(a,d,e)$

bucket E: $e=0$ $\lambda^{D}(a,e)$

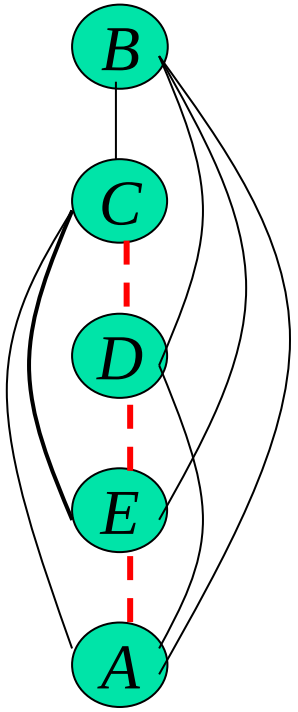
bucket A: $P(a)$ $\lambda^{E}(a)$

$P(e=0)$

$P(a|e=0)$

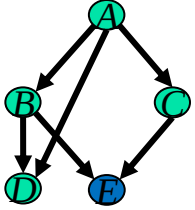
W=4*
"induced width"
(max clique size)

B=1



The impact of evidence?

Algorithm *BE-bel*



$$P(A \mid E=0) = \alpha \sum_{E=0,D,C,B} P(A)\cdot P(B \mid A)\cdot P(C \mid A)\cdot P(D \mid A,B)\cdot P(E \mid B,C)$$

P(A|E=0,B=1)?

$\sum\prod$ ← *Elimination operator*

$\sum_b$

bucket B: $P(b|a)$ $P(d|b,a)$ $P(e|b,c)$

bucket C: $P(c|a)$ *P(e/b=1,c)*

bucket D: *P(d/b=1,a)*

bucket E: $e=0$

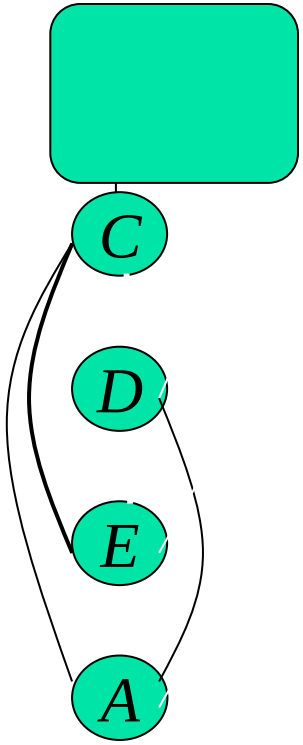
bucket A: $P(a)$ *P(b=1/a)*

P(e=0)

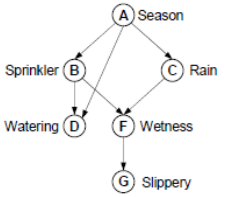
P(a|e=0)

$$P(a|e=0)=\frac{P(a,e=0)}{P(e=0)}$$

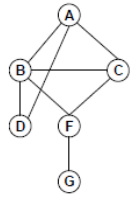
B=1



The impact of observations



(a) Directed acyclic graph



(b) Moral graph

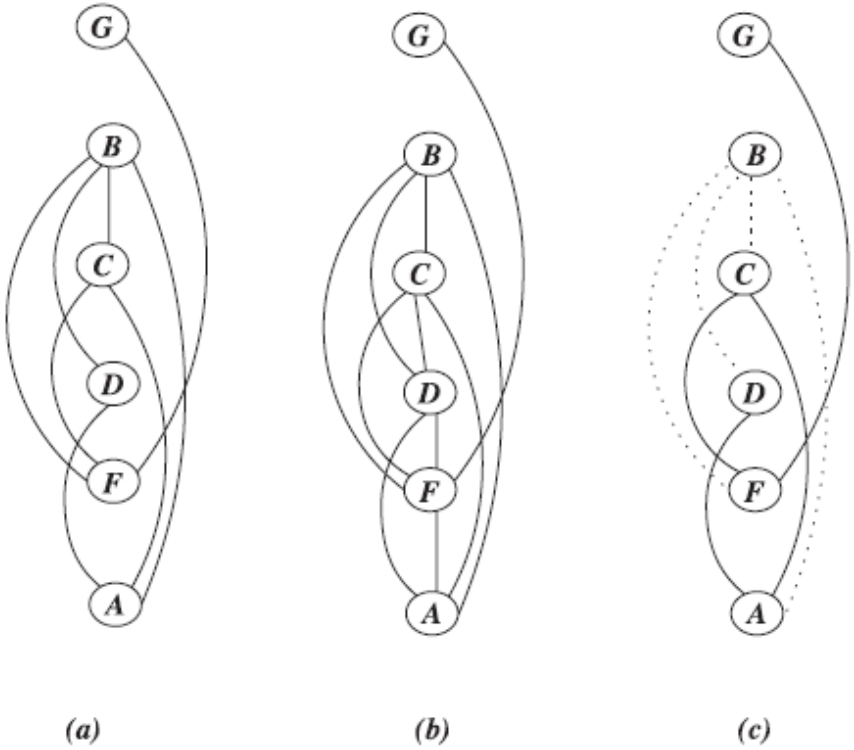
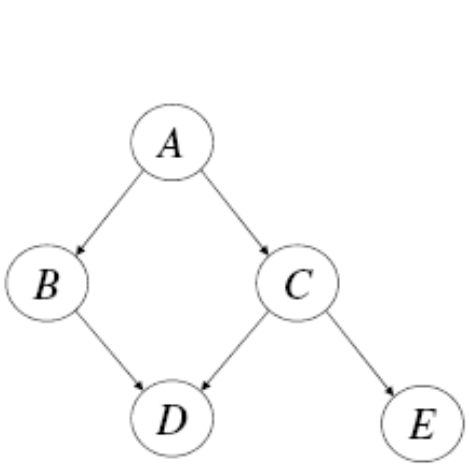


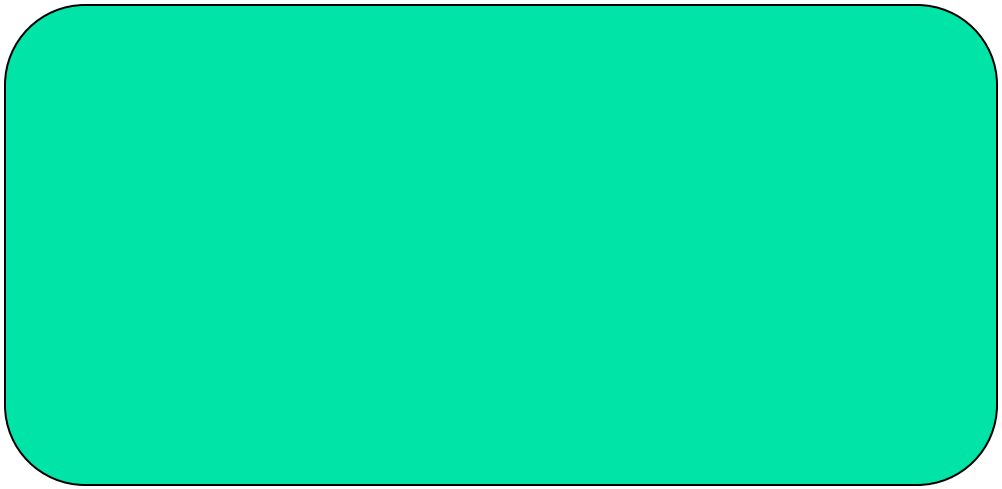
Figure 4.9: Adjusted induced graph relative to observing B.

Ordered graph *Induced graph* *Ordered conditioned graph*

Pruning Nodes: Example

Example of pruning irrelevant subnetworks

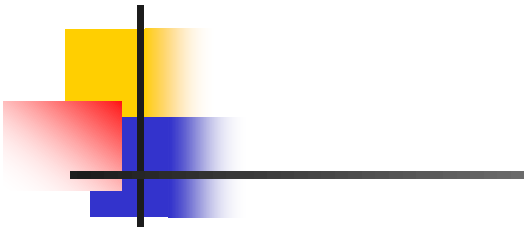




network structure joint on B, E joint on B

Inference for probabilistic networks

- Bucket elimination
  - Belief-updating, P(e), partition function
  - Marginals, probability of evidence
  - The impact of evidence
  - for MPE (→MAP)
  - for MAP (→ Marginal Map)
- Induced-Width

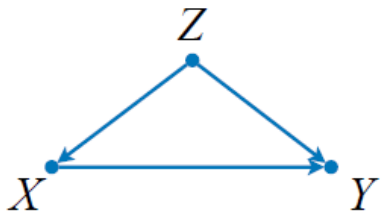


Back to SCM

Markovian Case

- The distribution $P(\mathbf{v})$ decomposes as:

$$P(\mathbf{v}) = \sum_{\mathbf{u}} P(\mathbf{u}) \prod_{V_i \in \mathbf{V}} P\left(v_i \mid v_1, \ldots, v_{i-1}, \mathbf{u}\right) = \sum_{\mathbf{u}} P(\mathbf{u}) \prod_{V_i \in \mathbf{V}} P\left(v_i \mid pa_i, u_i\right)$$



$$P(z,x,y) = \sum_{\mathbf{u}} P(\mathbf{u}) P(z \mid u_z) P(x \mid z, u_x) P(y \mid x, z, u_y)$$

$$= \left(\sum_{u_z} P(z \mid u_z) P(u_z) \right) \left(\sum_{u_x} P(x \mid z, u_x) P(u_x) \right) \left(\sum_{u_y} P(y \mid x, z, u_y) P(u_y) \right)$$

$$= P(z) P(x \mid z) P(y \mid x, z)$$

- In Markovian models, $P(v_i \mid pa_i)$ can be seen as “canonical factors”.

Markovian Case

U_z: $P(U_z)$, $P(z|U_z)$

U_x: $P(U_x)$, $P(x|U_x, z)$

U_y: $P(U_y)$, $P(y|z, x, U_y)$

X, Y, Z:

$\lambda_{U_z}(z) = \sum_{U_z} P(z|U_z) \cdot P(U_z) = ?$

$\lambda_{U_x}(x, z) = \sum_{U_x} P(x|U_x, z) \cdot P(U_x) = ?$

$\lambda_{U_y}(y, x, z) = \sum_{U_y} P(y|z, x, U_y) \cdot P(y) = ?$

$P(x, y, z) = \lambda_{U_z}(z) \cdot \lambda_{U_x}(x, z) \cdot \lambda_{U_y}(y, x, z)$

Markovian Case

U_z: $P(U_z)$, $P(z|U_z)$ →

U_x: $P(U_x)$, $P(x|U_x,z)$

U_y: $P(U_y)$, $P(y|z,x,U_y)$

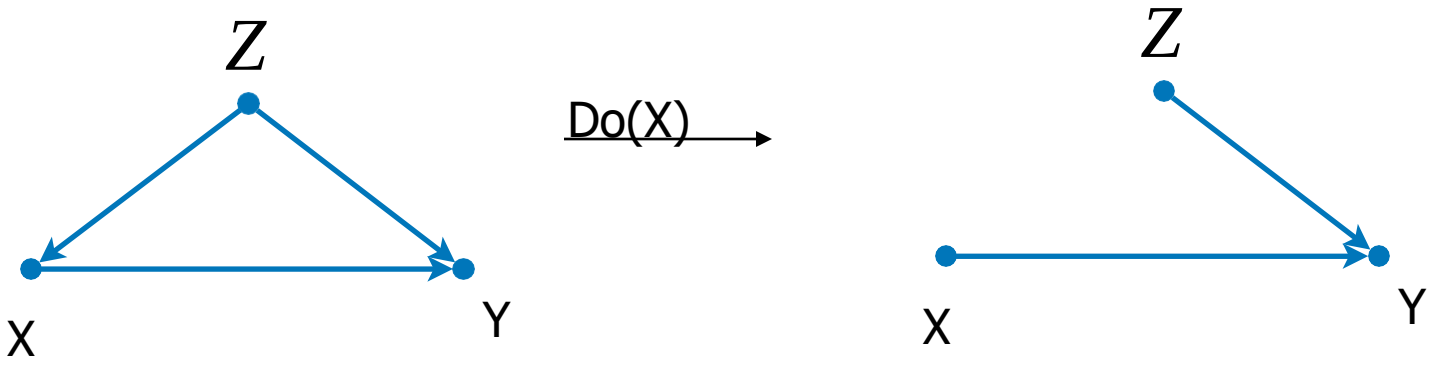
$P(z)$

x,y,z:
- $\lambda_{U_z}(z) = \sum_{U_z} P(z|U_z)\cdot P(U_z) = ?$
- $p(x|z)$
- $\lambda_{U_x}(x,z) = \sum_{U_x} P(x|U_x,z)\cdot P(U_x) = ?$
- $\lambda_{U_y}(y,x,z) = \sum_{U_y} P(y|z,x,U_y)\cdot P(y)$
- $P(y|z,x)$

$P(x,y,z) = \lambda_{U_z}(z)\cdot\lambda_{U_x}(x,z)\cdot\lambda_{U_y}(y,x,z)$

P(X,Y,Z) = P(Z)P(X|Z)P(Y|X,Y,Z)

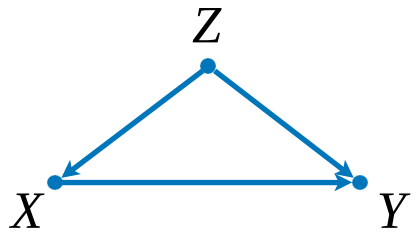
Markovian Case



Markovian Case

- *Every $P(v_i|pa_i)$ is computable from $P(\mathbf{v})$, i.e.,*

$$P(v_i/pa_i) = \frac{\sum_{v|v_i,pa_i} P(v)}{\sum_{v|pa_i} P(v)}$$



$$P(z,x,y) = P(z)P(x/z)P(y/x,z)$$

$$P(z) = \sum_{x,y} P(v)$$

$$P(y/x,z) = \frac{P(v)}{\sum_y P(v)}$$

$$P(x/z) = \frac{\sum_y P(v)}{\sum_{x,y} P(v)}$$

Complexity of Bucket-Elimination

- **Theorem:**

BE is O(n exp(w*+1)) time and O(n exp(w*)) space, when w* is the induced-width of the moral graph along d when evidence nodes are processed (edges from evidence nodes to earlier variables are removed.)

More accurately: O(r exp(w(d)) where r is the number of CPTs. For Bayesian networks r=n. For Markov networks?*