Reasoning with Bayesian Networks

Lecture 5: Complexity of Probabilistic Inference, Compiling Bayesian Networks

Jinbo Huang

NICTA and ANU

Decision Versions of Queries

- D-MAR / D-PR: $\Pr(\mathbf{q}|\mathbf{e}) > p$?
- D-MPE: In there $\mathbf{x}$ such that $\Pr(\mathbf{x}, \mathbf{e}) > p$?
- D-MAP: Given variables $\mathbf{Q} \subseteq \mathbf{X}$, in there $\mathbf{q}$ such that $\Pr(\mathbf{q}, \mathbf{e}) > p$?

Decision Versions of Queries

- D-MAR / D-PR: $\Pr(\mathbf{q}|\mathbf{e}) > p$?
- D-MPE: In there $\mathbf{x}$ such that $\Pr(\mathbf{x}, \mathbf{e}) > p$?
- D-MAP: Given variables $\mathbf{Q} \subseteq \mathbf{X}$, in there $\mathbf{q}$ such that $\Pr(\mathbf{q}, \mathbf{e}) > p$?
- D-MPE is NP-complete
- D-MAR / D-PR is PP-complete
- D-MAP is NP^{PP}-complete

NP (Nondeterministic Polynomial)

- Solvable by nondeterministic Turing machine in polynomial time
- Alternative definition: YES answers have proofs that can be verified in polynomial time
- Problem Q is NP-complete if every problem in NP can be reduced to Q in polynomial time
- Intuitively, “hardest” problems in NP; no known (deterministic) polynomial time algorithms

SAT: An NP-complete Problem

$$(X_1 \vee X_2 \vee \neg X_3) \wedge ((X_3 \wedge X_4) \vee \neg X_5)$$

- Is there assignment that *satisfies* Boolean formula?
- SAT $\in$ NP: Guess assignment and return YES if satisfying, in polynomial time
- NP-complete: Reduce computation of nondeterministic Turing machine to SAT

PP (Probabilistic Polynomial)

- Solvable in polynomial time by nondeterministic Turing machine that accepts when majority ($>$ half) of paths accept
- Alternative definition: Solvable in polynomial time with correctness probability $> 1/2$ on YES instances and $\geq 1/2$ on NO instances
- NP $\subseteq$ PP
  - SAT $\in$ PP: Guess assignment, if satisfying return YES, else return YES with probability $1/2$

MAJSAT: A PP-complete Problem

$$(X_1 \vee X_2 \vee \neg X_3) \wedge ((X_3 \wedge X_4) \vee \neg X_5)$$

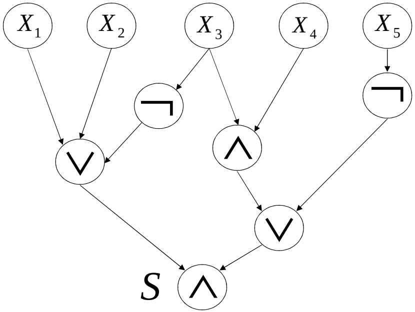
- Are majority of assignments satisfying?
- MAJSAT $\in$ PP: Why?
- PP-complete: Establish one-to-one correspondence between computation paths of nondeterministic Turing machine and satisfying assignments of Boolean formula

$\mathrm{NP}^{\mathrm{PP}}$

- Solvable in nondeterministic polynomial time given PP oracle
- E-MAJSAT ($\mathrm{NP}^{\mathrm{PP}}$-complete): Is there assignment to $X_1, \ldots, X_k$ such that majority of its completions (to full assignments) satisfy Boolean formula?

Boolean Formula to Bayesian Network

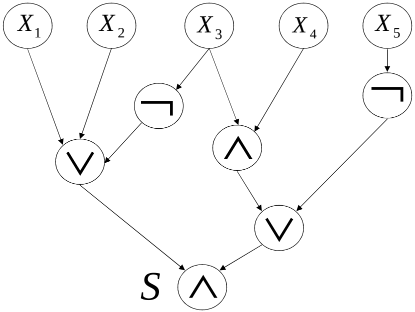
- $(X_1 \vee X_2 \vee \neg X_3) \wedge ((X_3 \wedge X_4) \vee \neg X_5)$



- Turn Boolean formula into Bayesian network with deterministic CPTs (0/1 probabilities)

Boolean Formula to Bayesian Network

- $(X_1 \vee X_2 \vee \neg X_3) \wedge ((X_3 \wedge X_4) \vee \neg X_5)$



- $\Pr(x_1, \ldots, x_n, S_\alpha = \mathit{true}) =$
  $1/2^n$ if $x_1, \ldots, x_n$ satisfying, 0 otherwise

Reducing SAT to D-MPE

$\exists$ assignment $x_1, \ldots, x_n$ satisfying formula α

iff

$\exists$ variable instantiation $\mathbf{y}$ of network $\mathcal{N}_\alpha$ such that $\Pr(\mathbf{y}, S_\alpha = \mathit{true}) > 0$

Reducing MAJSAT to D-PR/D-MAR

Majority of assignments $x_1, \ldots, x_n$ satisfy formula α

iff

$\Pr(S_\alpha = \mathit{true}) > 1/2$

Reducing E-MAJSAT to D-MAP

$\exists$ assignment $x_1, \ldots, x_k$ such that majority of assignments $x_{k+1}, \ldots, x_n$ satisfy formula α

iff

$\exists$ MAP instantiation $x_1, \ldots, x_k$ such that $\Pr(x_1, \ldots, x_k, S_\alpha = \mathit{true}) > 1/2^{k+1}$

Hardness Summary

- D-MPE is NP-hard
- D-MAR / D-PR is PP-hard
- D-MAP is NP^{PP}-hard

D-MPE $\in$ NP

In there $\mathbf{x}$ such that $\Pr(\mathbf{x}, \mathbf{e}) > p$?

- Guess network instantiation $\mathbf{x}$ (linear time)
- Compute $\Pr(\mathbf{x}, \mathbf{e})$ (linear time, by chain rule)
- Return YES iff $\Pr(\mathbf{x}, \mathbf{e}) > p$

D-MAR/D-PR $\in$ PP

$\Pr(\mathbf{q}|\mathbf{e}) > p$?

- Sample network instantiation $\mathbf{x}$ (linear time)
  - Instantiate variables $\mathbf{X}$ one at a time, parents before child
  - Sample value according to probabilities in CPT
- Return YES with probability
  - $a(p) = \min(1, 1/2p)$ if $\mathbf{x} \sim \mathbf{eq}$
  - $b(p) = \max(0, (1-2p)/(2-2p))$ if $\mathbf{x} \sim \mathbf{e}$, $\mathbf{x} \not\sim \mathbf{q}$
  - $1/2$ if $\mathbf{x} \not\sim \mathbf{e}$
- $\Pr(\text{YES}) > 1/2$ exactly when $\Pr(\mathbf{q}|\mathbf{e}) > p$

D-MAP $\in$ NP$^{\text{PP}}$

Given variables $\mathbf{Q} \subseteq \mathbf{X}$, in there $\mathbf{q}$ such that $\Pr(\mathbf{q}, \mathbf{e}) > p$?

- Guess instantiation $\mathbf{q}$ (linear time)
- Check if $\Pr(\mathbf{q}, \mathbf{e}) > p$ (a D-PR problem) using PP oracle
- Return YES iff $\Pr(\mathbf{q}, \mathbf{e}) > p$

Complexity Summary

- D-MPE is NP-complete
- D-MAR / D-PR is PP-complete
- D-MAP is NP^{PP}-complete

D-MAP on Polytrees (with bounded # of parents / node)

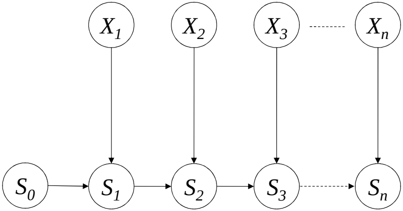
Given variables $\mathbf{Q} \subseteq \mathbf{X}$, in there $\mathbf{q}$ such that $\Pr(\mathbf{q}, \mathbf{e}) > p$?

- $\in$ NP
  - Guess instantiation $\mathbf{q}$ (linear time)
  - Check if $\Pr(\mathbf{q}, \mathbf{e}) > p$ by variable elimination (polynomial time due to bounded treewidth)
  - Return YES iff $\Pr(\mathbf{q}, \mathbf{e}) > p$
- NP-hard by reduction from MAXSAT

MAXSAT (decision version)

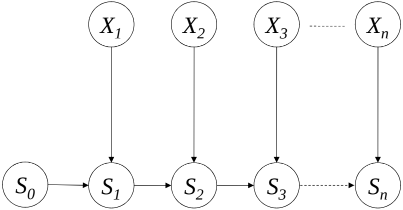
- Given clauses $\alpha_1, \ldots, \alpha_m$ over variables $X_1, \ldots, X_n$, is there assignment $x_1, \ldots, x_n$ satisfying more than k clauses?
- $\in$ NP: Guess assignment, check # of satisfied clauses, return YES if $> k$
- NP-hard: Contains SAT as special case $(k = m - 1)$

Reducing MAXSAT to D-MAP



- $S_0 : \{1, \ldots, m\}$, uniform priors on $S_0, X_1, \ldots, X_n$
- For $j > 0$, $S_j = 0$ if one of $x_1, \ldots, x_j$ satisfies α_i, where $i = S_0$; $S_j = i$ otherwise

Reducing MAXSAT to D-MAP



- $\exists$ assignment $x_1, \ldots, x_n$ satisfying more than k clauses iff $\exists$ instantiation $x_1, \ldots, x_n$ such that $\Pr(x_1, \ldots, x_n, S_n = 0) > k/(m2^n)$

Weighted Model Counting

$\sum$ of weights of models of Boolean formula

- Weight for literal: $A\,(.3), \neg A\,(.7), B\,(1), \neg B\,(1)$
- Weight of model: $w(\bar{A}B) = .7 \times 1 = .7$
- Weight of formula: $w(A \vee B) =$
  $w(AB) + w(A\bar{B}) + w(\bar{A}B) = .3 + .3 + .7 = 1.3$

Weighted Model Counting

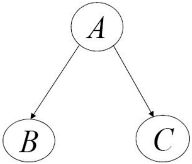
$\sum$ of weights of models of Boolean formula

- Weight for literal: $A\,(.3), \neg A\,(.7), B\,(1), \neg B\,(1)$
- Weight of model: $w(\overline{A}B) = .7 \times 1 = .7$
- Weight of formula: $w(A \vee B) =$
  $w(AB) + w(A\overline{B}) + w(\overline{A}B) = .3 + .3 + .7 = 1.3$

Use weighted model counters to compute $\Pr(\mathbf{e})$

- Encode Bayesian network into Boolean formula with weights

First Encoding



A	Θ_A
a_1	0.1
a_2	0.9

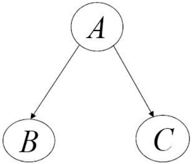
A	B	$\Theta_{B\|A}$
a_1	b_1	0.1
a_1	b_2	0.9
a_2	b_1	0.2
a_2	b_2	0.8

A	C	$\Theta_{C\|A}$
a_1	c_1	0.1
a_1	c_2	0.9
a_2	c_1	0.2
a_2	c_2	0.8

Indicator variables/clauses

- $I_{a_1} \vee I_{a_2} \quad \neg I_{a_1} \vee \neg I_{a_2}$
- $I_{b_1} \vee I_{b_2} \quad \neg I_{b_1} \vee \neg I_{b_2}$
- $I_{c_1} \vee I_{c_2} \quad \neg I_{c_1} \vee \neg I_{c_2}$

First Encoding



A	Θ_A
a_1	0.1
a_2	0.9

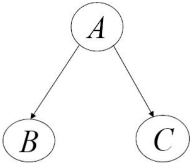
A	B	$\Theta_{B\|A}$
a_1	b_1	0.1
a_1	b_2	0.9
a_2	b_1	0.2
a_2	b_2	0.8

A	C	$\Theta_{C\|A}$
a_1	c_1	0.1
a_1	c_2	0.9
a_2	c_1	0.2
a_2	c_2	0.8

Parameter variables/clauses

- $I_{a_1} \leftrightarrow P_{a_1} \quad I_{a_2} \leftrightarrow P_{a_2}$
- $I_{a_1} \wedge I_{b_1} \leftrightarrow P_{b_1|a_1}$
  $I_{a_1} \wedge I_{b_2} \leftrightarrow P_{b_2|a_1}$
  $I_{a_2} \wedge I_{b_1} \leftrightarrow P_{b_1|a_2}$
  $I_{a_2} \wedge I_{b_2} \leftrightarrow P_{b_2|a_2}$
- $I_{a_1} \wedge I_{c_1} \leftrightarrow P_{c_1|a_1}$
  $I_{a_1} \wedge I_{c_2} \leftrightarrow P_{c_2|a_1}$
  $I_{a_2} \wedge I_{c_1} \leftrightarrow P_{c_1|a_2}$
  $I_{a_2} \wedge I_{c_2} \leftrightarrow P_{c_2|a_2}$

First Encoding



A	Θ_A
a_1	0.1
a_2	0.9

A	B	$\Theta_{B\|A}$
a_1	b_1	0.1
a_1	b_2	0.9
a_2	b_1	0.2
a_2	b_2	0.8

A	C	$\Theta_{C\|A}$
a_1	c_1	0.1
a_1	c_2	0.9
a_2	c_1	0.2
a_2	c_2	0.8

Given $\mathbf{e} = e_1, \ldots, e_k$

$\Pr(\mathbf{e}) = w(\Delta_{\mathcal{N}} \wedge I_{e_1} \wedge \ldots \wedge I_{e_k})$

where $w(P_{x|\mathbf{u}}) = \theta_{x|\mathbf{u}}$,
$w(I_x) = w(\neg I_x) =$
$w(\neg P_{x|\mathbf{u}}) = 1$

First Encoding

Network instantiation	Truth assignment sets these variables to true and all others to false	Weight of truth assignment
$a_1b_1c_1$	$\omega_0 : I_{a_1}\ I_{b_1}\ I_{c_1}\ P_{a_1}\ P_{b_1\|a_1}\ P_{c_1\|a_1}$	$0.1 \cdot 0.1 \cdot 0.1 = 0.001$
$a_1b_1c_2$	$\omega_1 : I_{a_1}\ I_{b_1}\ I_{c_2}\ P_{a_1}\ P_{b_1\|a_1}\ P_{c_2\|a_1}$	$0.1 \cdot 0.1 \cdot 0.9 = 0.009$
$a_1b_2c_1$	$\omega_2 : I_{a_1}\ I_{b_2}\ I_{c_1}\ P_{a_1}\ P_{b_2\|a_1}\ P_{c_1\|a_1}$	$0.1 \cdot 0.9 \cdot 0.1 = 0.009$
$a_1b_2c_2$	$\omega_3 : I_{a_1}\ I_{b_2}\ I_{c_2}\ P_{a_1}\ P_{b_2\|a_1}\ P_{c_2\|a_1}$	$0.1 \cdot 0.9 \cdot 0.9 = 0.081$
$a_2b_1c_1$	$\omega_4 : I_{a_2}\ I_{b_1}\ I_{c_1}\ P_{a_2}\ P_{b_1\|a_1}\ P_{c_1\|a_2}$	$0.9 \cdot 0.2 \cdot 0.2 = 0.036$
$a_2b_1c_2$	$\omega_5 : I_{a_2}\ I_{b_1}\ I_{c_2}\ P_{a_2}\ P_{b_1\|a_1}\ P_{c_2\|a_2}$	$0.9 \cdot 0.2 \cdot 0.8 = 0.144$
$a_2b_2c_1$	$\omega_6 : I_{a_2}\ I_{b_2}\ I_{c_1}\ P_{a_2}\ P_{b_2\|a_1}\ P_{c_1\|a_2}$	$0.9 \cdot 0.8 \cdot 0.2 = 0.144$
$a_2b_2c_2$	$\omega_7 : I_{a_2}\ I_{b_2}\ I_{c_2}\ P_{a_2}\ P_{b_2\|a_1}\ P_{c_2\|a_2}$	$0.9 \cdot 0.8 \cdot 0.8 = 0.576$

$$\Pr(a_1c_2) = w(\Delta_{\mathcal{N}} \wedge I_{a_1} \wedge I_{c_2}) = w(\omega_1) + w(\omega_3) = .009 + .081 = .09$$

Second Encoding

A	Θ_A
a_1	.3
a_2	.5
a_3	.2

A	B	$\Theta_{B\|A}$
a_1	b_1	.2
a_1	b_2	.8
a_2	b_1	1
a_2	b_2	0
a_3	b_1	.6
a_3	b_2	.4

Indicator variables/clauses (as 1st encoding)

- $I_{a_1} \vee I_{a_2} \vee I_{a_3} \quad \neg I_{a_1} \vee \neg I_{a_2}$
  $\neg I_{a_1} \vee \neg I_{a_3} \quad \neg I_{a_2} \vee \neg I_{a_3}$
- $I_{b_1} \vee I_{b_2} \quad \neg I_{b_1} \vee \neg I_{b_2}$

Second Encoding

A	Θ_A
a_1	.3
a_2	.5
a_3	.2

A	B	$\Theta_{B\|A}$
a_1	b_1	.2
a_1	b_2	.8
a_2	b_1	1
a_2	b_2	0
a_3	b_1	.6
a_3	b_2	.4

Parameter variables/clauses (fewer than 1st encoding)

- $Q_{a_1} \longrightarrow I_{a_1} \quad \neg Q_{a_1} \wedge Q_{a_2} \longrightarrow I_{a_2}$
  $\neg Q_{a_1} \wedge \neg Q_{a_2} \longrightarrow I_{a_3}$
- $I_{a_1} \wedge Q_{b_1|a_1} \longrightarrow I_{b_1}$
  $I_{a_1} \wedge \neg Q_{b_1|a_1} \longrightarrow I_{b_2}$
  $I_{a_2} \wedge Q_{b_1|a_2} \longrightarrow I_{b_1}$
  $I_{a_2} \wedge \neg Q_{b_1|a_2} \longrightarrow I_{b_2}$
  $I_{a_3} \wedge Q_{b_1|a_3} \longrightarrow I_{b_1}$
  $I_{a_3} \wedge \neg Q_{b_1|a_3} \longrightarrow I_{b_2}$

Second Encoding

A	Θ_A
a_1	.3
a_2	.5
a_3	.2

A	B	$\Theta_{B\|A}$
a_1	b_1	.2
a_1	b_2	.8
a_2	b_1	1
a_2	b_2	0
a_3	b_1	.6
a_3	b_2	.4

Multiple models for same instantiation of $I_{a_1}, I_{a_2}, I_{a_3}, I_{b_1}, I_{b_2}$

Eight Models for $\overline{I_{a_1}}\ I_{a_2}\ \overline{I_{a_3}}\ \overline{I_{b_1}}\ I_{b_2}$

- $\overline{Q_{a_1}}\ Q_{a_2}\ \overline{Q_{b_1|a_2}}$
- $Q_{a_3},\ Q_{b_1|a_1},\ Q_{b_1|a_3}$ unconstrained

Given appropriate weights,
$\Pr(\mathbf{x}) = \sum_{model \sim \mathbf{x}} w(model)$

Second Encoding

A	Θ_A
a_1	.3
a_2	.5
a_3	.2

A	B	$\Theta_{B\|A}$
a_1	b_1	.2
a_1	b_2	.8
a_2	b_1	1
a_2	b_2	0
a_3	b_1	.6
a_3	b_2	.4

Given $\mathbf{e} = e_1, \ldots, e_k$

$$\Pr(\mathbf{e}) = w(\Delta_{\mathcal{N}} \wedge I_{e_1} \wedge \ldots \wedge I_{e_k})$$

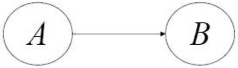
where $w(Q_{x|\mathbf{u}}) = \frac{\theta_{x|\mathbf{u}}}{1-\sum_{x'<x} \theta_{x'|\mathbf{u}}}$,
$w(\neg Q_{x|\mathbf{u}}) = 1 - w(Q_{x|\mathbf{u}})$,
$w(I_x) = w(\neg I_x) = 1$

MPE by Weighted MAXSAT

$(X \vee \neg Y \vee \neg Z)^3$, $(\neg X)^{10.1}$, $(Y)^{.5}$, $(Z)^{2.5}$

- Clauses have weights
- W-MAXSAT: Find assignment maximizing sum of weights of satisfied clauses
- Alternatively: Minimize *penalty*—sum of weights of violated clauses

MPE by Weighted MAXSAT



A	Θ_A
a_1	.3
a_2	.5
a_3	.2

A	B	$\Theta_{B\|A}$
a_1	b_1	.2
a_1	b_2	.8
a_2	b_1	1
a_2	b_2	0
a_3	b_1	.6
a_3	b_2	.4

Indicator clauses, all having weight W (some big number)

- $I_{a_1} \vee I_{a_2} \vee I_{a_3}$ $\quad \neg I_{a_1} \vee \neg I_{a_2}$ $\quad \neg I_{a_1} \vee \neg I_{a_3}$ $\quad \neg I_{a_2} \vee \neg I_{a_3}$
- $I_{b_1} \vee I_{b_2}$ $\quad \neg I_{b_1} \vee \neg I_{b_2}$

These *hard* clauses ensure assignment corresponds to network instantiation

MPE by Weighted MAXSAT

A	Θ_A
a_1	.3
a_2	.5
a_3	.2

A	B	$\Theta_{B\|A}$
a_1	b_1	.2
a_1	b_2	.8
a_2	b_1	1
a_2	b_2	0
a_3	b_1	.6
a_3	b_2	.4

Parameter clauses

- $(\neg I_{a_1})^{-\log .3}$
  $(\neg I_{a_2})^{-\log .5}$
  $(\neg I_{a_3})^{-\log .2}$
- $(\neg I_{a_1} \vee \neg I_{b_1})^{-\log .2}$
  $(\neg I_{a_1} \vee \neg I_{b_2})^{-\log .8}$
  $(\neg I_{a_2} \vee \neg I_{b_1})^{-\log 1}$
  $(\neg I_{a_2} \vee \neg I_{b_2})^{W}$
  $(\neg I_{a_3} \vee \neg I_{b_1})^{-\log .6}$
  $(\neg I_{a_3} \vee \neg I_{b_2})^{-\log .4}$

MPE by Weighted MAXSAT

A	Θ_A
a_1	.3
a_2	.5
a_3	.2

A	B	$\Theta_{B\|A}$
a_1	b_1	.2
a_1	b_2	.8
a_2	b_1	1
a_2	b_2	0
a_3	b_1	.6
a_3	b_2	.4

$I_{a_1}\ \overline{I_{a_2}}\ \overline{I_{a_3}}\ \overline{I_{b_1}}\ I_{b_2}$ has penalty

$-\log .3 - \log .8$

$= -\log(.3 \times .8)$

$= -\Pr(a_1, b_2)$

Parameter clauses

- $(\neg I_{a_1})^{-\log .3}$
  $(\neg I_{a_2})^{-\log .5}$
  $(\neg I_{a_3})^{-\log .2}$
- $(\neg I_{a_1} \vee \neg I_{b_1})^{-\log .2}$
  $(\neg I_{a_1} \vee \neg I_{b_2})^{-\log .8}$
  $(\neg I_{a_2} \vee \neg I_{b_1})^{-\log 1}$
  $(\neg I_{a_2} \vee \neg I_{b_2})^{W}$
  $(\neg I_{a_3} \vee \neg I_{b_1})^{-\log .6}$
  $(\neg I_{a_3} \vee \neg I_{b_2})^{-\log .4}$

MPE by Weighted MAXSAT

A	Θ_A
a_1	.3
a_2	.5
a_3	.2

A	B	$\Theta_{B\|A}$
a_1	b_1	.2
a_1	b_2	.8
a_2	b_1	1
a_2	b_2	0
a_3	b_1	.6
a_3	b_2	.4

$I_{a_1}\,\overline{I_{a_2}}\,\overline{I_{a_3}}\,\overline{I_{b_1}}\,I_{b_2}$ has penalty

$- \log .3 - \log .8$

$= -\log(.3 \times .8)$

$= -\Pr(a_1, b_2)$

Any assignment Γ (satisfying hard clauses) violates exactly one clause from each CPT

$$Pn(\Gamma) = \sum_{\theta_{x|\mathbf{u}} \sim \mathbf{x}} -\log \theta_{x|\mathbf{u}}$$

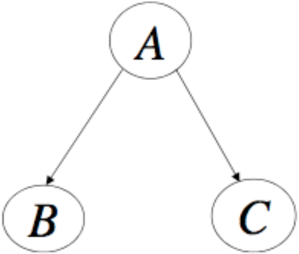
$$= -\log \prod_{\theta_{x|\mathbf{u}} \sim \mathbf{x}} \theta_{x|\mathbf{u}} = -\log \Pr(\mathbf{x})$$

Min penalty $\Leftrightarrow$ max $\Pr(\mathbf{x})$

Complexity of Probabilistic Inference: Summary

- D-MPE NP-complete, D-MAR / D-PR PP-complete, D-MAP NP^{PP}-complete
- D-MAP on polytrees NP-complete
- Two encodings for reducing $\Pr(\mathbf{e})$ to weighted model counting
- Reducing MPE to weighted MAXSAT

Arithmetic Circuits

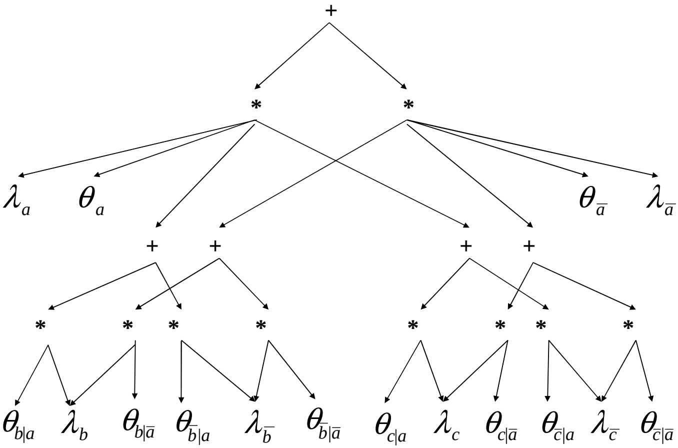


A	Θ_A
true	.5
false	.5

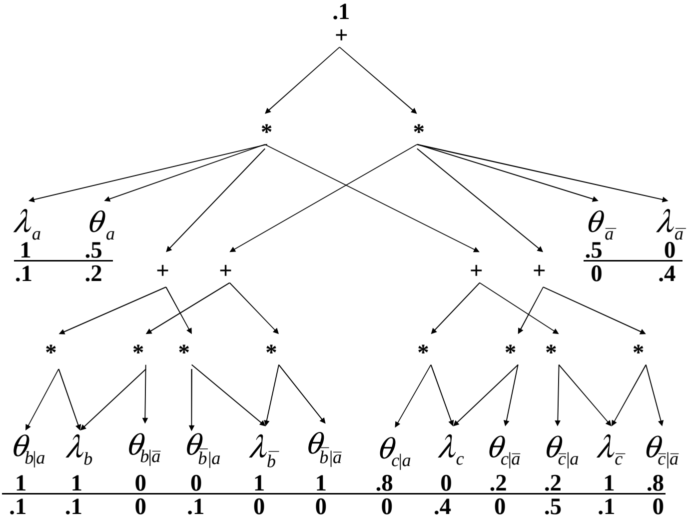
A	B	$\Theta_{B\|A}$
true	true	1
true	false	0
false	true	0
false	false	1

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

Arithmetic Circuits



Arithmetic Circuits



Network Polynomial

A	Θ_A
true	$\theta_a = .3$
false	$\theta_{\bar{a}} = .7$

A	B	$\Theta_{B\|A}$
true	true	$\theta_{b\|a} = .1$
true	false	$\theta_{\bar{b}\|a} = .9$
false	true	$\theta_{b\|\bar{a}} = .8$
false	false	$\theta_{\bar{b}\|\bar{a}} = .2$

A	B	$\Pr(A, B)$
a	b	$\theta_a \theta_{b\|a}$
a	$\bar{b}$	$\theta_a \theta_{\bar{b}\|a}$
$\bar{a}$	b	$\theta_{\bar{a}} \theta_{b\|\bar{a}}$
$\bar{a}$	$\bar{b}$	$\theta_{\bar{a}} \theta_{\bar{b}\|\bar{a}}$

Network Polynomial

A	Θ_A
true	$\theta_a = .3$
false	$\theta_{\bar{a}} = .7$

A	B	$\Theta_{B\|A}$
true	true	$\theta_{b\|a} = .1$
true	false	$\theta_{\bar{b}\|a} = .9$
false	true	$\theta_{b\|\bar{a}} = .8$
false	false	$\theta_{\bar{b}\|\bar{a}} = .2$

A	B	$\Pr(A, B)$
a	b	$\lambda_a \lambda_b \theta_a \theta_{b\|a}$
a	$\bar{b}$	$\lambda_a \lambda_{\bar{b}} \theta_a \theta_{\bar{b}\|a}$
$\bar{a}$	b	$\lambda_{\bar{a}} \lambda_b \theta_{\bar{a}} \theta_{b\|\bar{a}}$
$\bar{a}$	$\bar{b}$	$\lambda_{\bar{a}} \lambda_{\bar{b}} \theta_{\bar{a}} \theta_{\bar{b}\|\bar{a}}$

Network Polynomial

A	Θ_A
true	$\theta_a = .3$
false	$\theta_{\bar{a}} = .7$

A	B	$\Theta_{B\|A}$
true	true	$\theta_{b\|a} = .1$
true	false	$\theta_{\bar{b}\|a} = .9$
false	true	$\theta_{b\|\bar{a}} = .8$
false	false	$\theta_{\bar{b}\|\bar{a}} = .2$

A	B	$\Pr(A, B)$
a	b	$\lambda_a \lambda_b \theta_a \theta_{b\|a}$
a	$\bar{b}$	$\lambda_a \lambda_{\bar{b}} \theta_a \theta_{\bar{b}\|a}$
$\bar{a}$	b	$\lambda_{\bar{a}} \lambda_b \theta_{\bar{a}} \theta_{b\|\bar{a}}$
$\bar{a}$	$\bar{b}$	$\lambda_{\bar{a}} \lambda_{\bar{b}} \theta_{\bar{a}} \theta_{\bar{b}\|\bar{a}}$

$$f = \lambda_a \lambda_b \theta_a \theta_{b|a} + \lambda_a \lambda_{\bar{b}} \theta_a \theta_{\bar{b}|a} + \lambda_{\bar{a}} \lambda_b \theta_{\bar{a}} \theta_{b|\bar{a}} + \lambda_{\bar{a}} \lambda_{\bar{b}} \theta_{\bar{a}} \theta_{\bar{b}|\bar{a}}$$

Network Polynomial

A	Θ_A
true	$\theta_a = .3$
false	$\theta_{\bar{a}} = .7$

A	B	$\Theta_{B\mid A}$
true	true	$\theta_{b\mid a} = .1$
true	false	$\theta_{\bar{b}\mid a} = .9$
false	true	$\theta_{b\mid \bar{a}} = .8$
false	false	$\theta_{\bar{b}\mid \bar{a}} = .2$

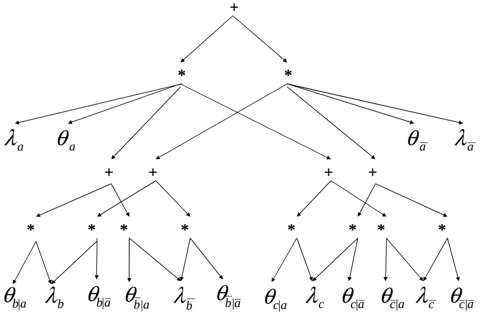
A	B	$\Pr(A,B)$
a	b	$\lambda_a\lambda_b\theta_a\theta_{b\mid a}$
a	$\bar{b}$	$\lambda_a\lambda_{\bar{b}}\theta_a\theta_{\bar{b}\mid a}$
$\bar{a}$	b	$\lambda_{\bar{a}}\lambda_b\theta_{\bar{a}}\theta_{b\mid \bar{a}}$
$\bar{a}$	$\bar{b}$	$\lambda_{\bar{a}}\lambda_{\bar{b}}\theta_{\bar{a}}\theta_{\bar{b}\mid \bar{a}}$

$$f = \lambda_a\lambda_b\theta_a\theta_{b\mid a} + \lambda_a\lambda_{\bar{b}}\theta_a\theta_{\bar{b}\mid a} + \lambda_{\bar{a}}\lambda_b\theta_{\bar{a}}\theta_{b\mid \bar{a}} + \lambda_{\bar{a}}\lambda_{\bar{b}}\theta_{\bar{a}}\theta_{\bar{b}\mid \bar{a}}$$

$$\begin{aligned} f(\mathbf{e} = \bar{a}) &= (0)(1)\theta_a\theta_{b\mid a}(0)(1)\theta_a\theta_{\bar{b}\mid a} + (1)(1)\theta_{\bar{a}}\theta_{b\mid \bar{a}} + (1)(1)\theta_{\bar{a}}\theta_{\bar{b}\mid \bar{a}} \\ &= \theta_{\bar{a}}\theta_{b\mid \bar{a}} + \theta_{\bar{a}}\theta_{\bar{b}\mid \bar{a}} = \Pr(\mathbf{e}) \end{aligned}$$

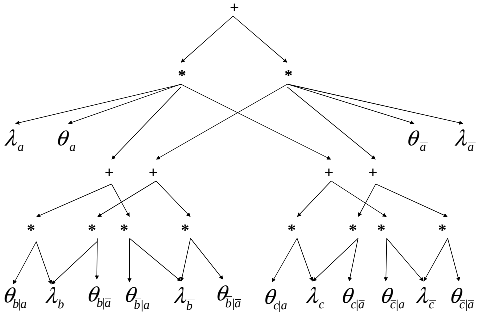
Network Polynomial as Arithmetic Circuit

$$f = \lambda_a\lambda_b\lambda_c\theta_a\theta_{b|a}\theta_{c|a} + \lambda_a\lambda_b\lambda_{\bar{c}}\theta_a\theta_{b|a}\theta_{\bar{c}|a} + \ldots + \lambda_{\bar{a}}\lambda_{\bar{b}}\lambda_{\bar{c}}\theta_{\bar{a}}\theta_{\bar{b}|\bar{a}}\theta_{\bar{c}|\bar{a}}$$



Network Polynomial as Arithmetic Circuit

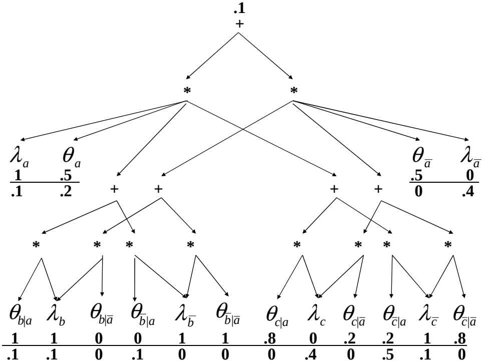
$$f = \lambda_a\lambda_b\lambda_c\theta_a\theta_{b|a}\theta_{c|a} + \lambda_a\lambda_b\lambda_{\bar{c}}\theta_a\theta_{b|a}\theta_{\bar{c}|a} + \ldots + \lambda_{\bar{a}}\lambda_{\bar{b}}\lambda_{\bar{c}}\theta_{\bar{a}}\theta_{\bar{b}|\bar{a}}\theta_{\bar{c}|\bar{a}}$$



f has exponential size, circuit may not

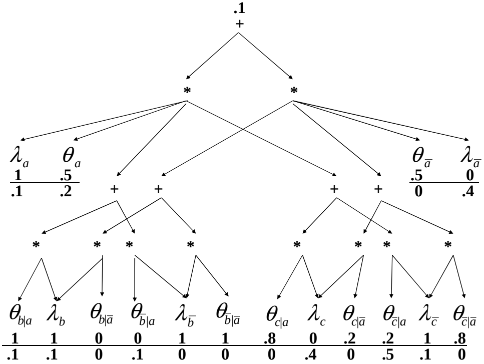
Partial Derivatives

$$\Pr(\mathbf{e} = a\bar{c}) = .1$$



Partial Derivatives

$\Pr(\mathbf{e} = a\bar{c}) = .1$



f is linear in every variable

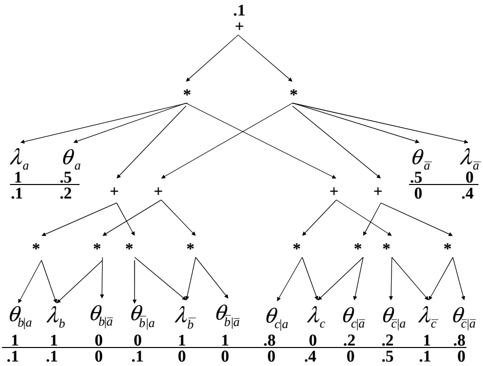
$$\frac{\partial f}{\partial \lambda_{\bar{a}}}(\mathbf{e}) = .4$$

f will increase by .4 if $\lambda_{\bar{a}}$ changes from 0 to 1 ($\mathbf{e}$ changes from $a\bar{c}$ to $\bar{c}$)

$\Pr(\bar{c}) = .1 + .4 = .5$

Partial Derivatives

$\Pr(\mathbf{e} = a\bar{c}) = .1$



$$\frac{\partial f}{\partial \lambda_x}(\mathbf{e}) = \Pr(x, \mathbf{e} - X)$$

Flipping variable: $\bar{x} \in \mathbf{e}$

Adding literal: $x, \bar{x} \notin \mathbf{e}$

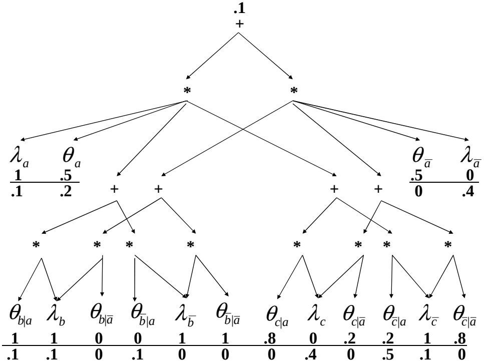
$\Pr(\mathbf{e})$: $x \in \mathbf{e}$

$$\Pr(\bar{a}\,\bar{c}) = \frac{\partial f}{\partial \lambda_{\bar{a}}} = .4$$

$$\Pr(ab\bar{c}) = \frac{\partial f}{\partial \lambda_b} = .1$$

Partial Derivatives

$\Pr(\mathbf{e} = a\bar{c}) = .1$

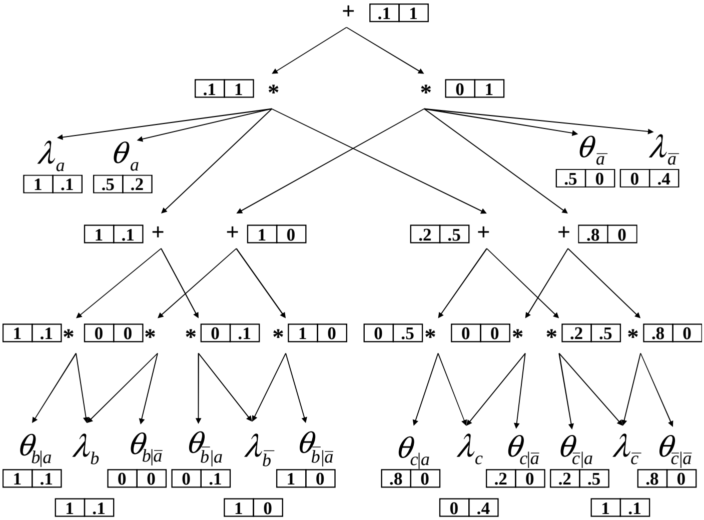


$$\theta_{x|\mathbf{u}} \frac{\partial f}{\partial \theta_{x|\mathbf{u}}}(\mathbf{e}) = \Pr(x, \mathbf{u}, \mathbf{e})$$

Gives family marginals $\Pr(x, \mathbf{u}, \mathbf{e})\ \forall x\mathbf{u}$

$\frac{\partial f}{\partial \theta_{x|\mathbf{u}}}$ also useful in sensitivity analysis and parameter learning

Evaluation and Differentiation



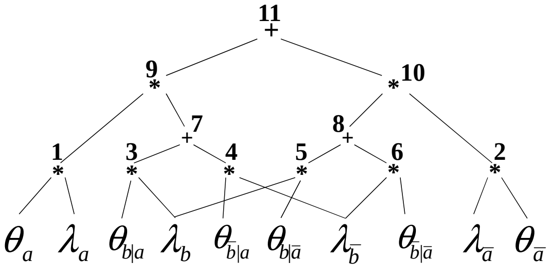
Evaluation and Differentiation

- Bottom-up pass evaluates circuit, computes $\Pr(\mathbf{e})$
- Top-down pass computes all partial derivatives
- Linear in circuit size
  - Use division instead of multiplication
  - Handle 0s with extra bit per node, set to 1 when exactly one child has 0 value

Maximizer Circuits

- Turn addition into maximization in both passes
- Bottom-up pass computes $f^m = \text{MPE}_P(\mathbf{e})$
  - Can recover MPE instantiation
- Top-down pass computes all partial derivatives
  - $\frac{\partial f_x^m}{\partial \lambda_x}(\mathbf{e}) = \text{MPE}_P(x, \mathbf{e} - X)$
  - $\theta_{x|\mathbf{u}} \frac{\partial f_{x,\mathbf{u}}^m}{\partial \theta_{x|\mathbf{u}}}(\mathbf{e}) = \text{MPE}_P(\mathbf{e}, x, \mathbf{u})$
- f_x^m $(f_{x,\mathbf{u}}^m)$ is max over terms $\sim x$ $(x\mathbf{u})$

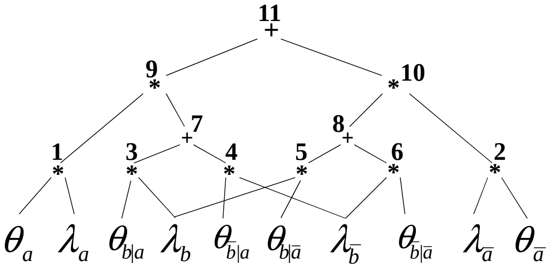
Circuits from Variable Elimination



A	B	$\Theta_{B\|A}$
true	true	$n_3 = \star(\lambda_b, \theta_{b\|a})$
true	false	$n_4 = \star(\lambda_{\bar{b}}, \theta_{\bar{b}\|a})$
false	true	$n_5 = \star(\lambda_b, \theta_{b\|\bar{a}})$
false	false	$n_6 = \star(\lambda_{\bar{b}}, \theta_{\bar{b}\|\bar{a}})$

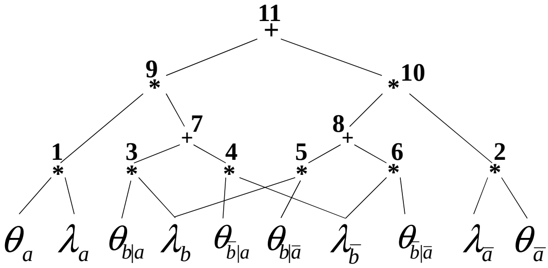
A	Θ_A
true	$n_1 = \star(\lambda_a, \theta_a)$
false	$n_2 = \star(\lambda_{\bar{a}}, \theta_{\bar{a}})$

Circuits from Variable Elimination



A	B	$\Theta_{B\|A}$
true	true	$n_3 = \star(\lambda_b, \theta_{b\|a})$
true	false	$n_4 = \star(\lambda_{\bar{b}}, \theta_{\bar{b}\|a})$
false	true	$n_5 = \star(\lambda_b, \theta_{b\|\bar{a}})$
false	false	$n_6 = \star(\lambda_{\bar{b}}, \theta_{\bar{b}\|\bar{a}})$

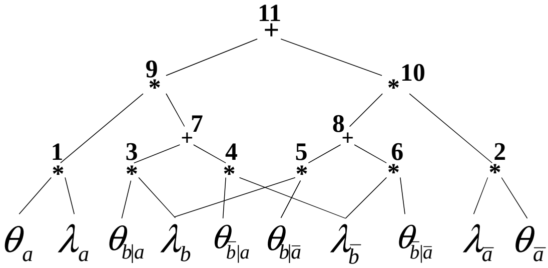
Circuits from Variable Elimination



A	B	$\Theta_{B\|A}$
true	true	$n_3 = \star(\lambda_b, \theta_{b\|a})$
true	false	$n_4 = \star(\lambda_{\bar{b}}, \theta_{\bar{b}\|a})$
false	true	$n_5 = \star(\lambda_b, \theta_{b\|\bar{a}})$
false	false	$n_6 = \star(\lambda_{\bar{b}}, \theta_{\bar{b}\|\bar{a}})$

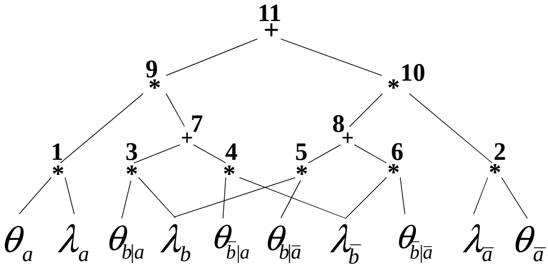
A	$\sum_B \Theta_{B\|A}$
true	$n_7 = +(n_3, n_4)$
false	$n_8 = +(n_5, n_6)$

Circuits from Variable Elimination



A	$\Theta_A \sum_B \Theta_{B\|A}$
true	$n_9 = \star(n_1, n_7)$
false	$n_{10} = \star(n_2, n_8)$

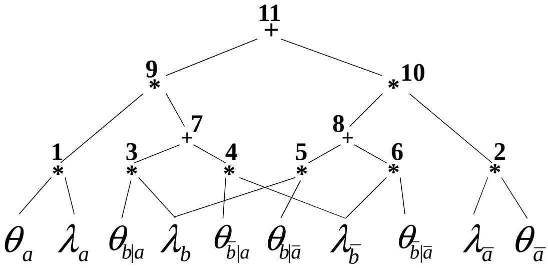
Circuits from Variable Elimination



A	$\Theta_A \sum_B \Theta_{B\|A}$
true	$n_9 = \star(n_1, n_7)$
false	$n_{10} = \star(n_2, n_8)$

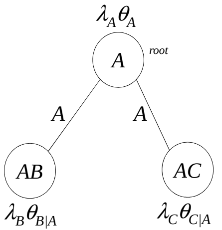
	$\sum_A \Theta_A \sum_B \Theta_{B\|A}$
	$n_{11} = +(n_9, n_{10})$

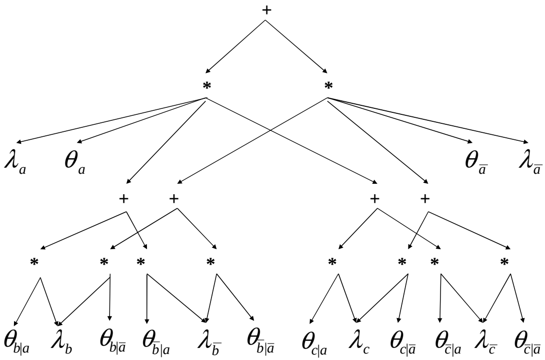
Circuits from Variable Elimination



Circuit size $O(n\, exp(w))$ as complexity of variable elimination

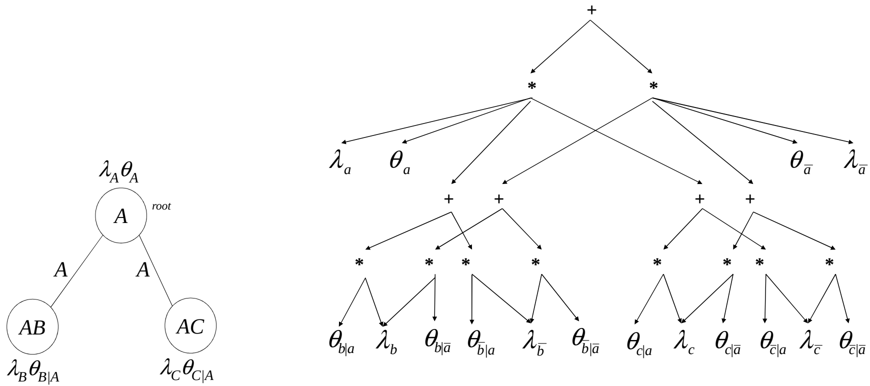
Circuits Embedded in Jointree





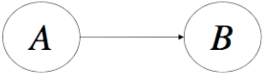
Cluster instantiation to $*$ node, separator instantiation to $+$ node

Circuits Embedded in Jointree



Circuit size $O(n\,exp(w))$ as complexity of jointree propagation

Circuits from Boolean Encoding



A	Θ_A
true	$\theta_a = .3$
false	$\theta_{\bar{a}} = .7$

A	B	$\Theta_{B\|A}$
true	true	$\theta_{b\|a} = .1$
true	false	$\theta_{\bar{b}\|a} = .9$
false	true	$\theta_{b\|\bar{a}} = .8$
false	false	$\theta_{\bar{b}\|\bar{a}} = .2$

$$I_a \vee I_{\bar{a}} \quad \neg I_a \vee \neg I_{\bar{a}}$$
$$I_b \vee I_{\bar{b}} \quad \neg I_b \vee \neg I_{\bar{b}}$$

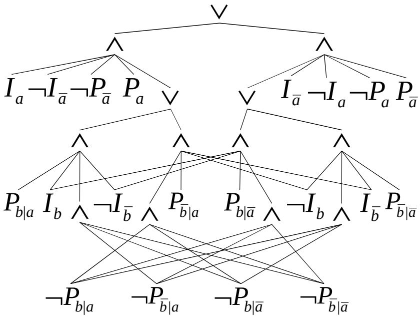
$$I_a \leftrightarrow P_a$$
$$I_{\bar{a}} \leftrightarrow P_{\bar{a}}$$

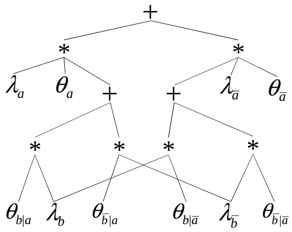
$$I_a \wedge I_b \leftrightarrow P_{b|a}$$
$$I_a \wedge I_{\bar{b}} \leftrightarrow P_{\bar{b}|a}$$
$$I_{\bar{a}} \wedge I_b \leftrightarrow P_{b|\bar{a}}$$
$$I_{\bar{a}} \wedge I_{\bar{b}} \leftrightarrow P_{\bar{b}|\bar{a}}$$

Circuits from Boolean Encoding

- Boolean encoding (1st version) of Bayesian network encodes network polynomial
  - Each model corresponds to one term
- Convert Boolean formula to Boolean circuit satisfying four properties
  - NNF: literals as leaves, $\vee, \wedge$ as internal nodes
  - Decomposable: children of $\wedge$ node do not share variables
  - Determinism: children of $\vee$ node mutually inconsistent
  - Smooth: children of $\vee$ node have same variables
  - Conversion algorithms studied in *Knowledge Compilation*
- Convert Boolean circuit to arithmetic circuit

Boolean Circuit to Arithmetic Circuit





- $\vee$ to $+$, $\wedge$ to $*$, negative literals to 1
- I_x to λ_x, $P_{x|\mathbf{u}}$ to $\theta_{x|\mathbf{u}}$

Compiling Bayesian Networks: Summary

- Network polynomials and arithmetic circuits
- Differential semantics of arithmetic circuits
- Circuit evaluation and differentiation, maximizer circuits
- Circuits compiled from variable elimination, jointree, Boolean encoding