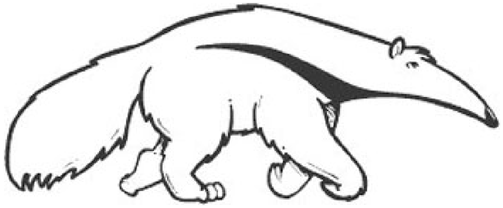
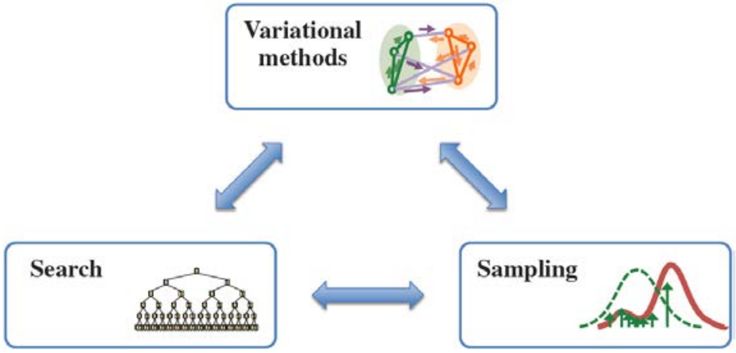
Anytime Probabilistic Reasoning (Solving the Marginal MAP)

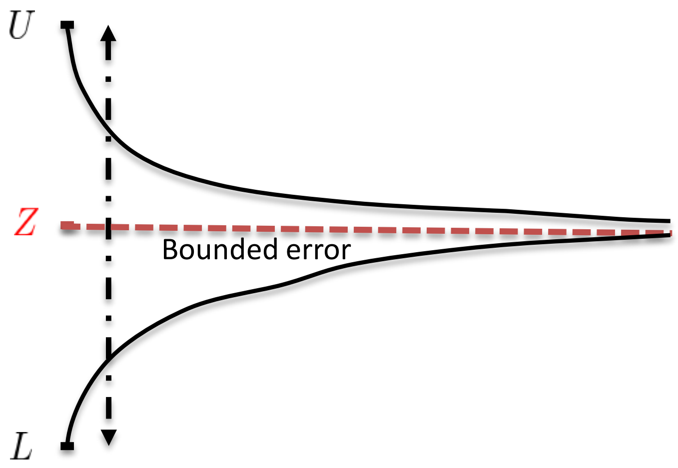
Rina Dechter
Information and Computer Sciences
UC-Irvine



Outline

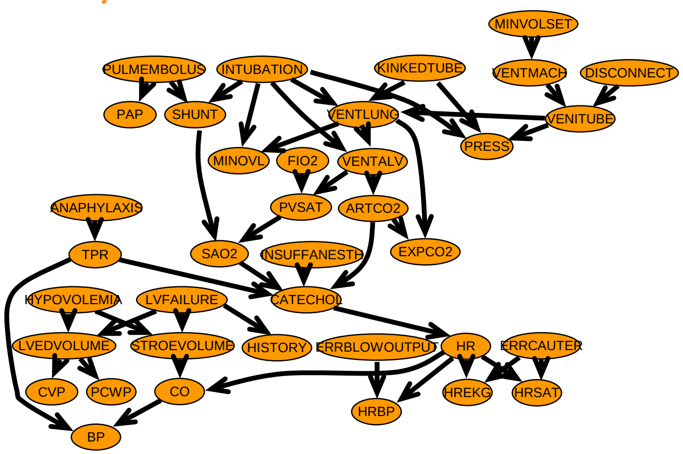
- Graphical models, the Marginal Map task, anytime reasoning
- Inference and variational bounds
- AND/OR search spaces
- Combining methods: Heuristic Search
- Combining methods: Sampling
- Conclusion



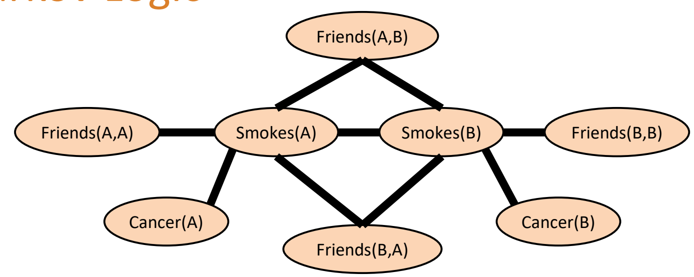


Overview: Graphical Models

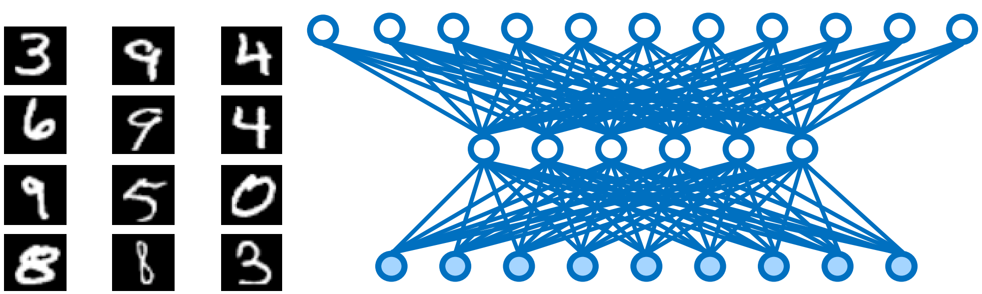
Bayesian Networks



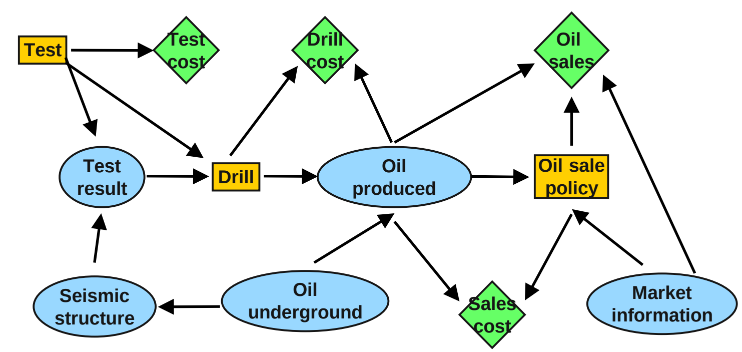
Markov Logic



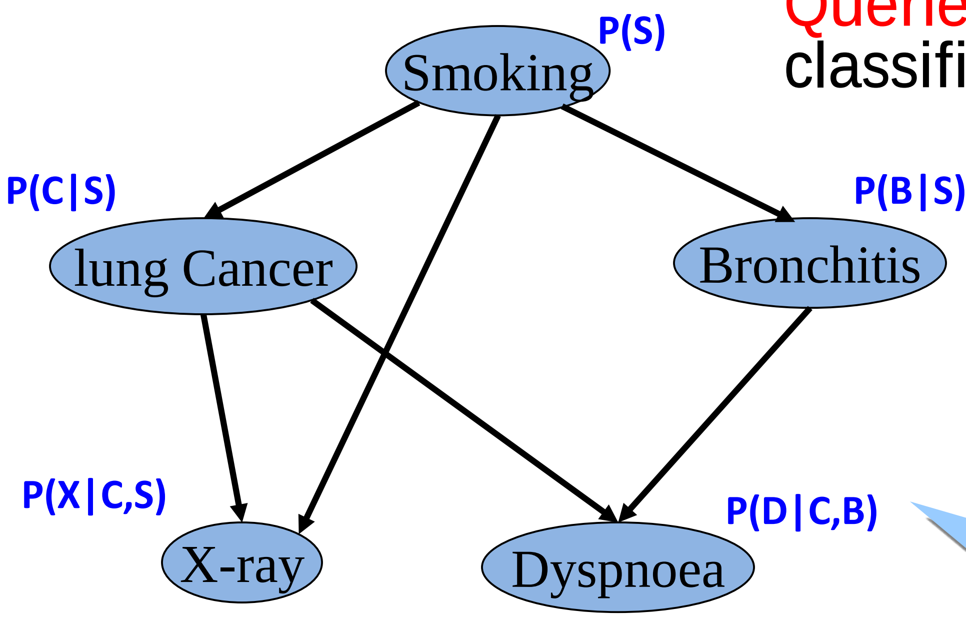
Deep Boltzmann Machines



Influence Diagrams



Bayesian Networks (Pearl 1988)



Queries: prediction, diagnosis, classification, decision making

CPD:

C	B	P(D|C,B)	
0	0	0.1	0.9
0	1	0.7	0.3
1	0	0.8	0.2
1	1	0.9	0.1

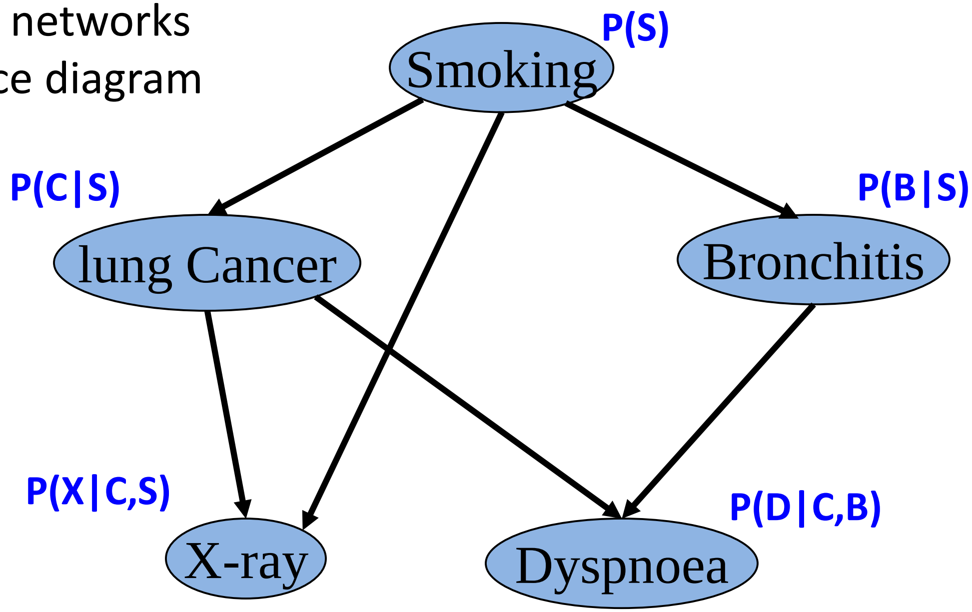
P(S, C, B, X, D)* = *P(S) P(C|S) P(B|S) P(X|C,S) P(D|C,B)

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

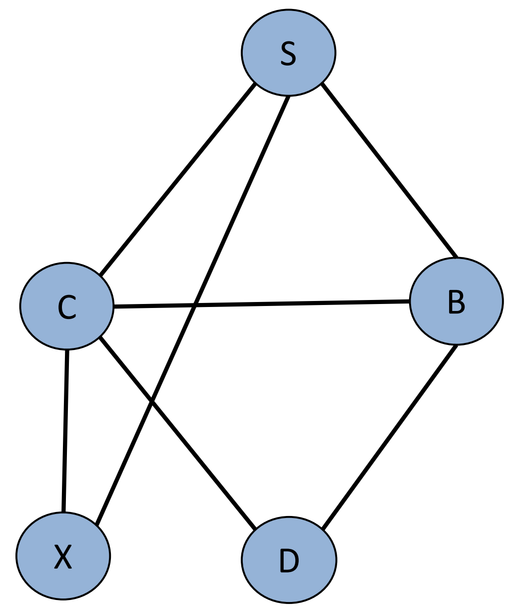
- P(D= 0) = $\sum_{S,L,B,X}$ P(S)· P(C|S)· P(B|S)· P(X|C,S)· P(D|C,B)
- MAP/MPE = $max_{S,L,B,X}$ P(S)· P(C|S)· P(B|S)· P(X|C,S)· P(D|C,B)

Graphical Models

Bayesian networks
Markov networks
Influence diagram



Primal Graph



A *graphical model* consists of:

$X = \{X_1, \ldots, X_n\}$ -- variables

$D = \{D_1, \ldots, D_n\}$ -- domains

$F = \{f_{\alpha_1}, \ldots, f_{\alpha_m}\}$ -- (non-negative) functions or “factors”

CPD:

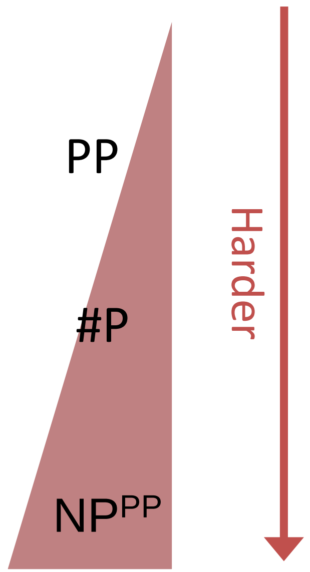
C	B	f(D,B,C)	
0	0	0.1	0.9
0	1	0.7	0.3
1	0	0.8	0.2
1	1	0.9	0.1

P(S, C, B, X, D) = ***P(S) P(C|S) P(B|S) P(X|C,S) P(D|C,B)***

Types of Queries

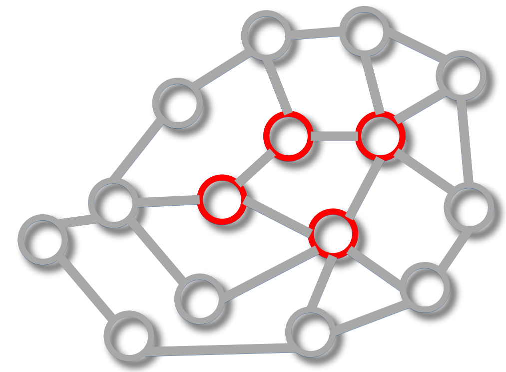
- Tasks:

Max-Inference (most likely config.)	$f(\mathbf{x}^*) = \max_{\mathbf{x}} \prod_{\alpha} f_\alpha(\mathbf{x}_\alpha)$
Sum-Inference (data likelihood)	$Z = \sum_{\mathbf{x}} \prod_{\alpha} f_\alpha(\mathbf{x}_\alpha)$
Mixed-Inference (optimal prediction)	$f(\mathbf{x}_M^*) = \max_{\mathbf{x}_M} \sum_{\mathbf{x}_S} \prod_{\alpha} f_\alpha(\mathbf{x}_\alpha)$



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

Why Marginal MAP?

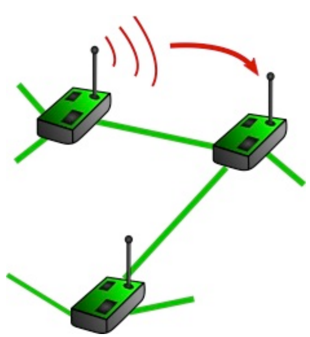


- Often, Marginal MAP is the “right” task:
  - We have a model describing a large system
  - We care about predicting the state of some part

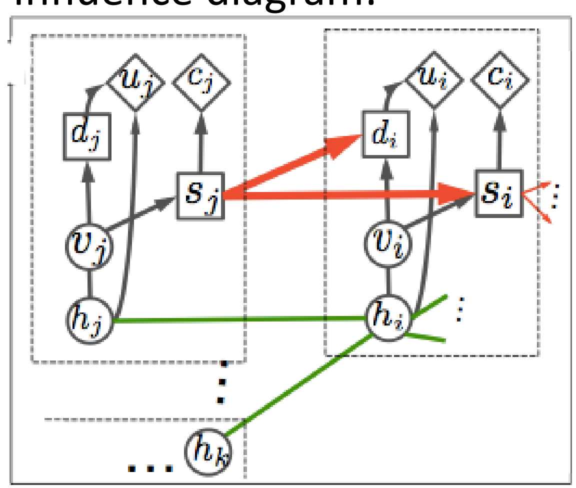
- Complexity: NP^{PP} complete
- Not necessarily easy on trees

- Example: decision making
  - Sum over random variables (random effects, etc.)
  - Max over decision variables (specify action policies)

Sensor network



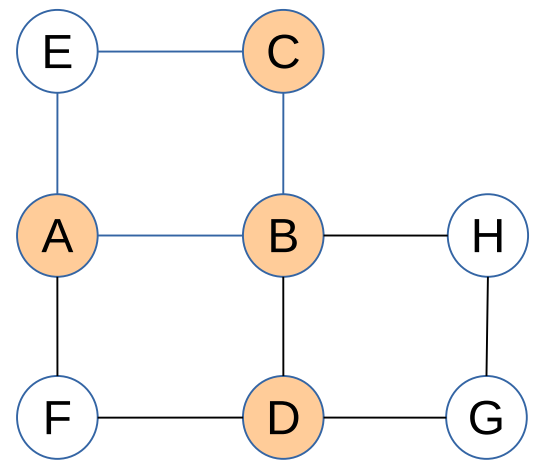
Influence diagram:



An Influence diagram

Marginal Map

primal graph



- Graphical Model: $\mathcal{M} = \langle \mathbf{X}, \mathbf{D}, \mathbf{F} \rangle$
  - –variables $\mathbf{X} = \{X_1, \ldots, X_n\}$
  - –domains $\mathbf{D} = \{D_1, \ldots, D_n\}$
  - –functions $\mathbf{F} = \{f_1, \ldots, f_r\}$

$$P(\mathbf{X}) = \frac{1}{Z} \prod_j f_j$$

- Marginal MAP task:

$$\mathbf{X} = \mathbf{X}_M \cup \mathbf{X}_S$$

$$\mathbf{X}_M = \{A, B, C, D\}$$

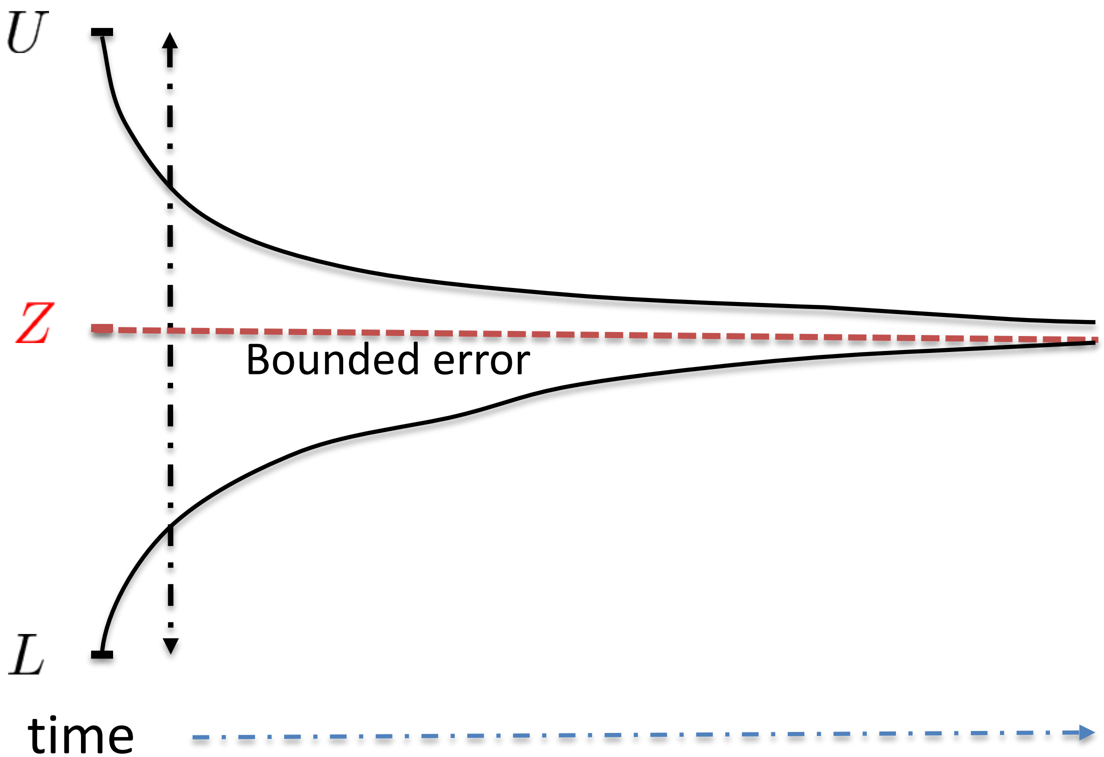
$$\mathbf{X}_S = \{E, F, G, H\}$$

$$x_M^* = argmax_{X_M} \sum_{X_S} \prod_j f_j$$

Why is it harder? intuitively

Goal: Anytime Bounds

- Desiderata
  - Meaningful confidence interval
  - Responsive
  - Complete

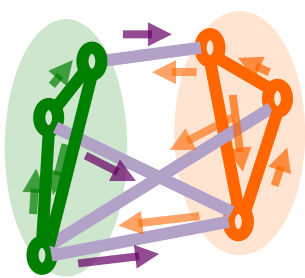


Building Blocks of Approximate Inference

- Three building blocks paradigms
  - Effective at different types of problems

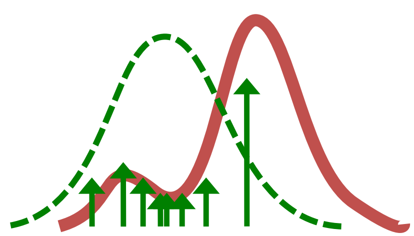
Variational methods

Reason over small subsets of variables at a time



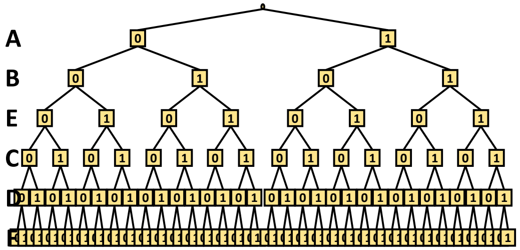
(Monte Carlo) Sampling

Use randomization to estimate averages over the state space



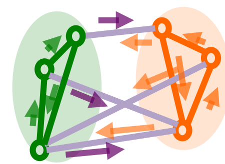
(Heuristic) Search

Structured enumeration over all possible states



Combining Approaches

Inference, Variational methods



Bucket-elimination
weighted mini-bucket (WMB)

[Dechter 1999, Dechter and Rish, 2003
Liu and Ihler, ICML 2011]

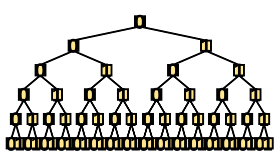
provide heuristic

provide proposal

WMB-IS

[Liu et al., NIPS 2015]

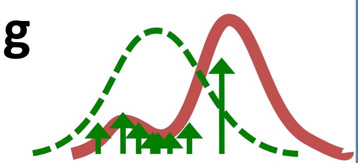
Search



refine proposal

help search

Sampling



AND/OR search

[Marinescu et al 2009, Lou et al., AAAI 2017]

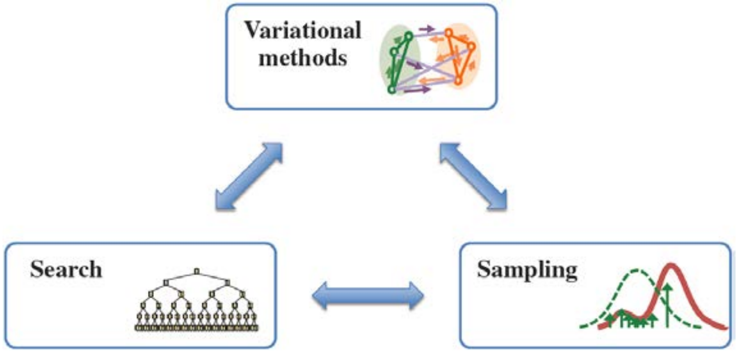
Marinescu et al., IJCAI 2018]

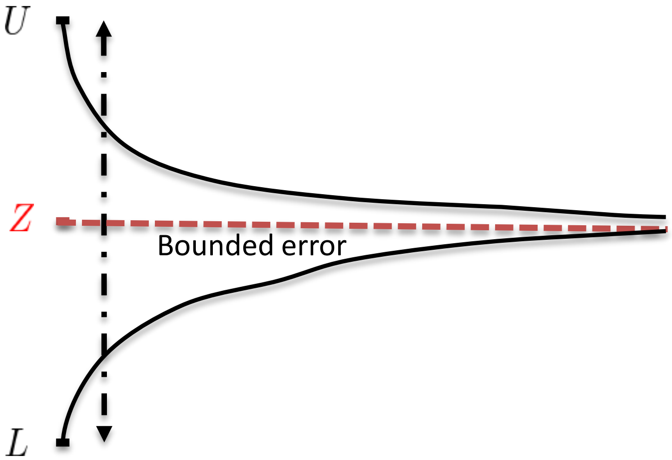
dynamic importance sampling (DIS)

[Lou et al., NIPS 2017]

Outline

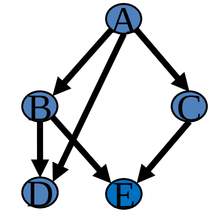
- Graphical models, The Marginal Map task
- Inference and variational bounds.
- AND/OR search spaces
- Combining methods: Heuristic Search for Marginal Map
- Combining methods: Sampling
- Conclusion





Bucket Elimination

Algorithm *BE-bel* [Dechter 1996]



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

$\sum_b \prod$ ← 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: $\lambda_{C\to D}(A,d,e)$

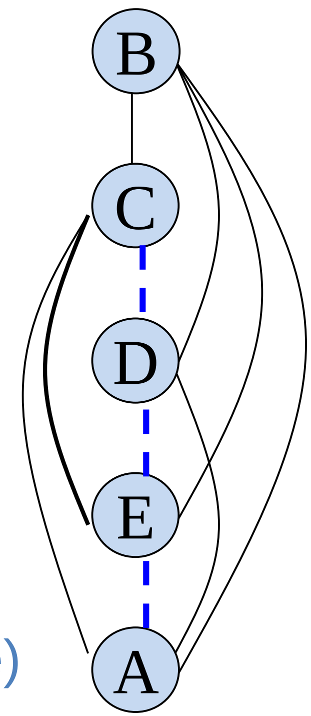
bucket E: $\mathbb{1}[E=0]\ \lambda_{D\to E}(A,e)$

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

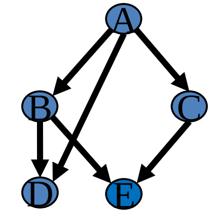
$p(E=0)$

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

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



Bucket Elimination



Algorithm *BE-bel* [Dechter 1996]

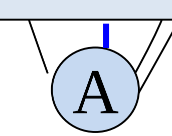
$$p(A|E=0) = \alpha \sum_{e,d,c,b} p(A)\, p(b|A)\, p(c|A)\, p(d|A,b)\, p(e|b,c)\, \mathbb{1}[e=0]$$

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

Time and space exponential in the induced-width / treewidth

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

(max clique size)

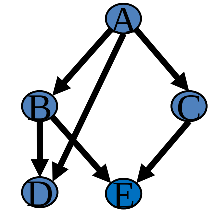


$$p(E=0)$$

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

Bucket Elimination

Optimization



Algorithm *BE-map* [Dechter 1996]

$$MPE = \max_{a,e,d,c,b} P(a)P(c \mid a)P(b \mid a)P(d \mid a,b)P(e \mid b,c)$$

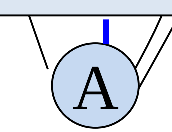
$$\max_X \prod$$

Elimination & combination operators

Time and space exponential in the induced-width / treewidth

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

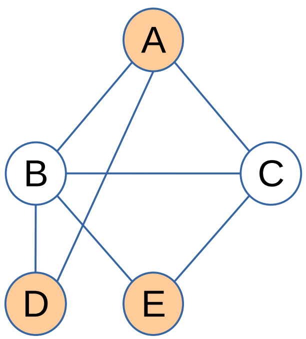
(max clique size)



OPT

Why is MMAP Harder for Inference (BE)?

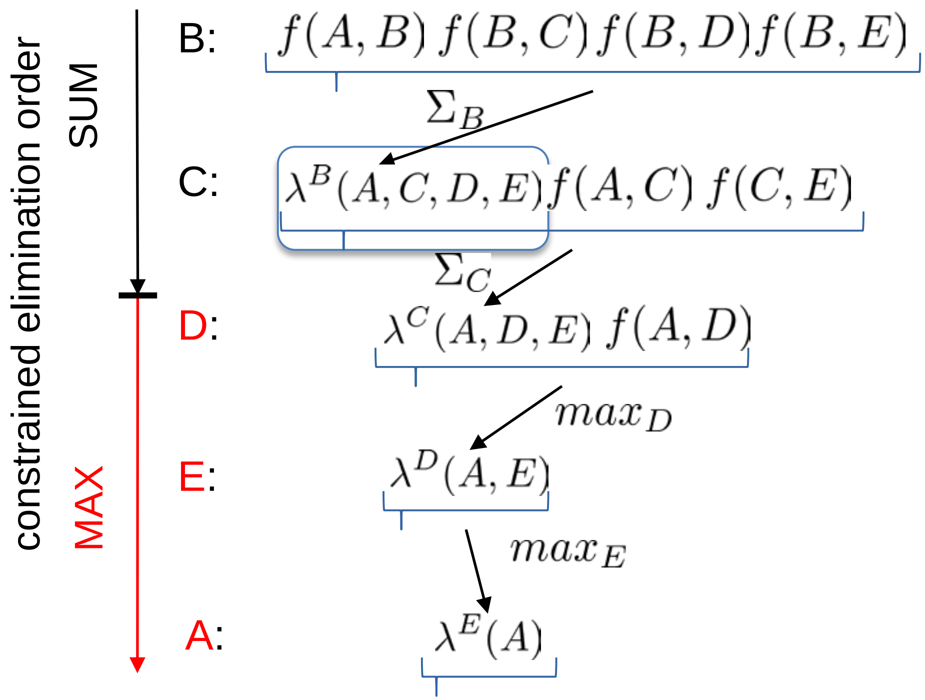
Let's apply Bucket-elimination: Complexity is exponential in the *constrained* induced-width



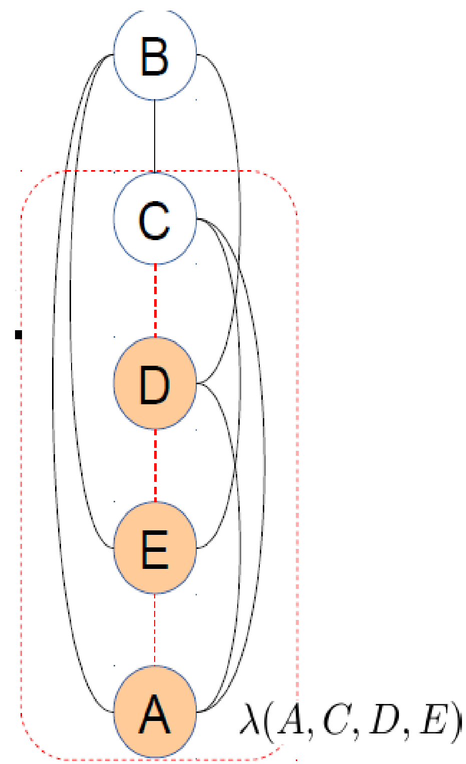
$$\mathbf{X}_M = \{A, D, E\}$$
$$\mathbf{X}_S = \{B, C\}$$

$$\max_{\mathbf{X}_M} \sum_{\mathbf{X}_S} P(\mathbf{X})$$

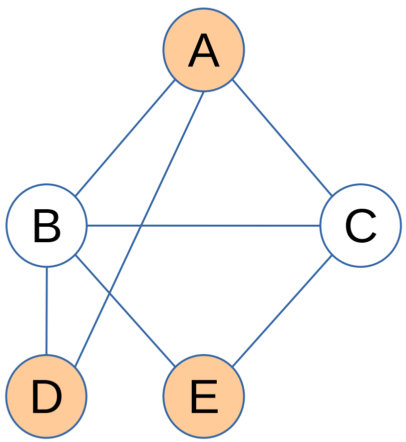
$$P(X) = \prod_j f_j$$



MAP* is the marginal MAP value

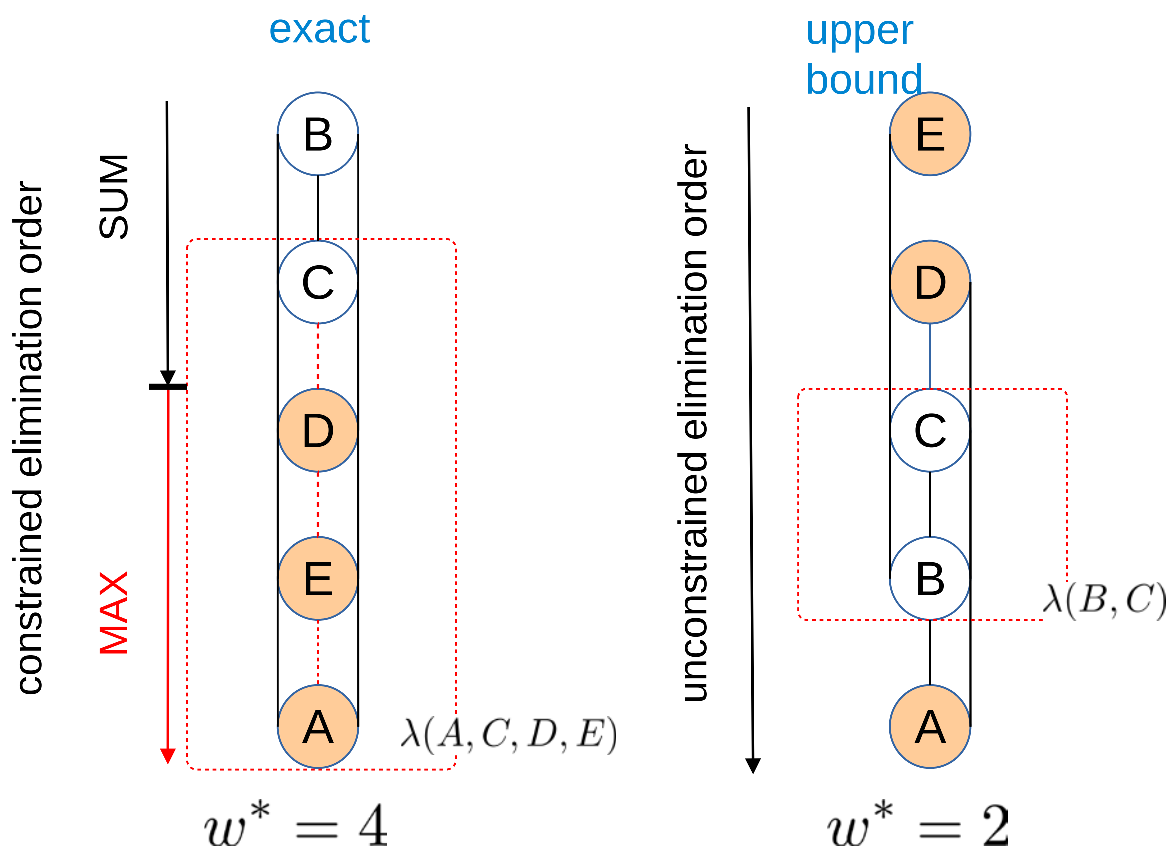


Why is MMAP Harder for Inference (BE)?



$\mathbf{X}_M = \{A, D, E\}$

$\mathbf{X}_S = \{B, C\}$



In practice, constrained induced is much larger!

$$\max_X \sum_Y \phi \leq \sum_Y \max_X \phi$$

(Park & Darwiche, 2003)
(Yuan & Hansen, 2009)

Decomposition-bounds:

Mini-bucket and weighted Mini-bucket

Tightening by Cost-shifting

Mini-Bucket Approximation

For optimization

Split a bucket into mini-buckets —> bound complexity

bucket (X) =

$$\{ f_1,\ f_2,\ \dots\ f_r,\ f_{r+1},\ \dots\ f_n \}$$

$$\lambda_X(\cdot) = \max_x \prod_{i=1}^{n} f_i(x, \dots)$$

$$\{ f_1,\ \dots\ f_r \} \qquad \{ f_{r+1},\ \dots\ f_n \}$$

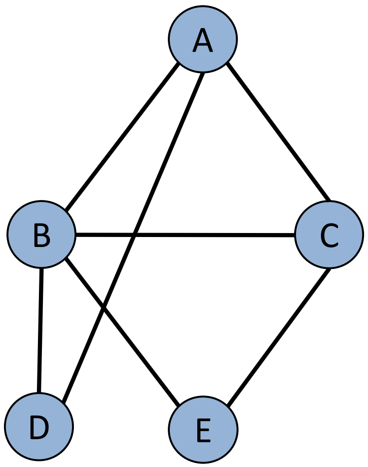
$$\lambda_{X,1}(\cdot) = \max_x \prod_{i=1}^{r} f_i(x, \dots) \qquad \lambda_{X,2}(\cdot) = \max_x \prod_{i=r+1}^{n} f_i(x, \dots)$$

$$\lambda_X(\cdot) \ \leq \ \lambda_{X,1}(\cdot)\ \lambda_{X,2}(\cdot)$$

Exponential complexity decrease: $O(e^n) \longrightarrow O(e^r) + O(e^{n-r})$

Mini-Bucket Elimination [Dechter & Rish 2003]

For optimization



mini-buckets

bucket B: $\underbrace{f(a,b)\ f(b,c)}$ $\underbrace{f(b,d)\ f(b,e)}$

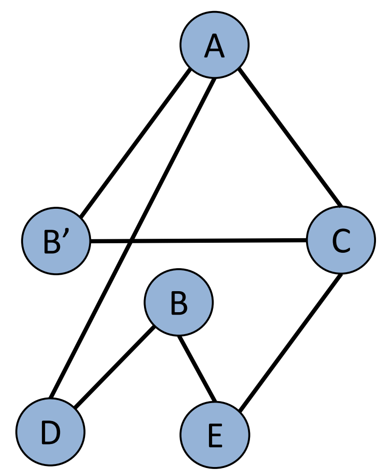
bucket C: $\underbrace{f(c,a)\ f(c,e)\ \lambda_{B\to C}(a,c)}$

bucket D: $\underbrace{f(a,d)\ \ \lambda_{B\to D}(d,e)}$

bucket E: $\underbrace{\lambda_{C\to E}(a,e)\ \lambda_{E\to E}(a,e)}$

bucket A: $\underbrace{f(a)\ \lambda_{E\to A}(a)}$

U = upper bound



$$\lambda_{B\to C}(a,c) = \max_b f(a,b)\ f(b,c)$$

$$\lambda_{B\to D}(d,e) = \max_b f(b,d)\ f(b,e)$$

$$\lambda_{C\to E}(a,e) = \max_c \ldots$$

Mini-Bucket Decoding

For optimization

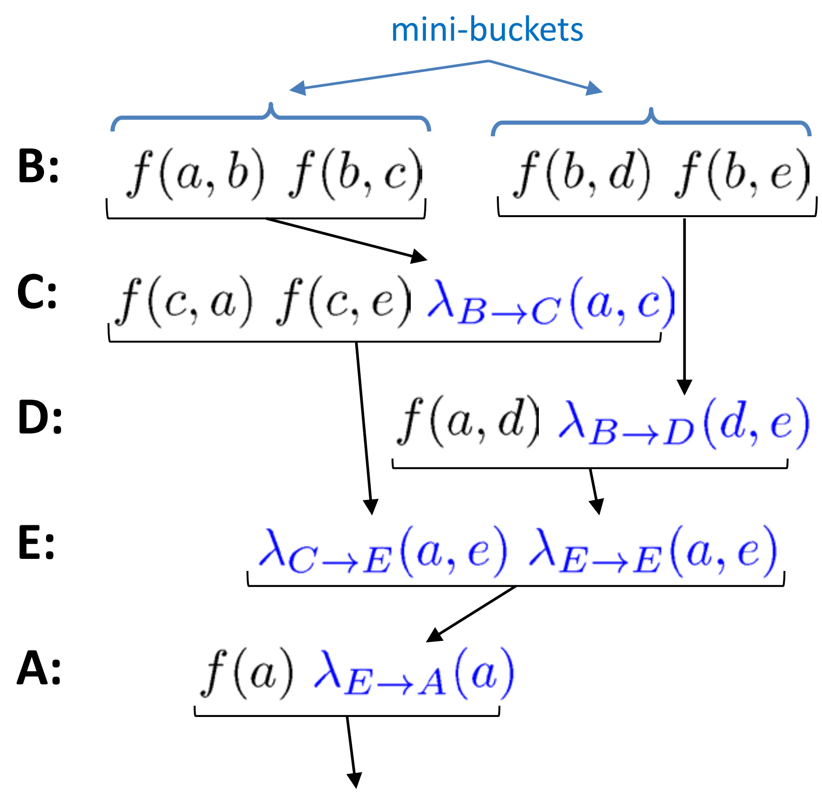
$$\mathbf{b}^* = \arg\max_b f(a^*, b) \cdot f(b, c^*) \cdot f(b, d^*) \cdot f(b, e^*)$$

$$\mathbf{c}^* = \arg\max_c f(c, a^*) \cdot f(c, e^*) \cdot \lambda_{B \to C}(a^*, c)$$

$$\mathbf{d}^* = \arg\max_d f(a^*, d) \cdot \lambda_{B \to D}(d, e^*)$$

$$\mathbf{e}^* = \arg\max_e \lambda_{C \to E}(a^*, e) \cdot \lambda_{D \to E}(a^*, e)$$

$$\mathbf{a}^* = \arg\max_a f(a) \cdot \lambda_{E \to A}(a)$$



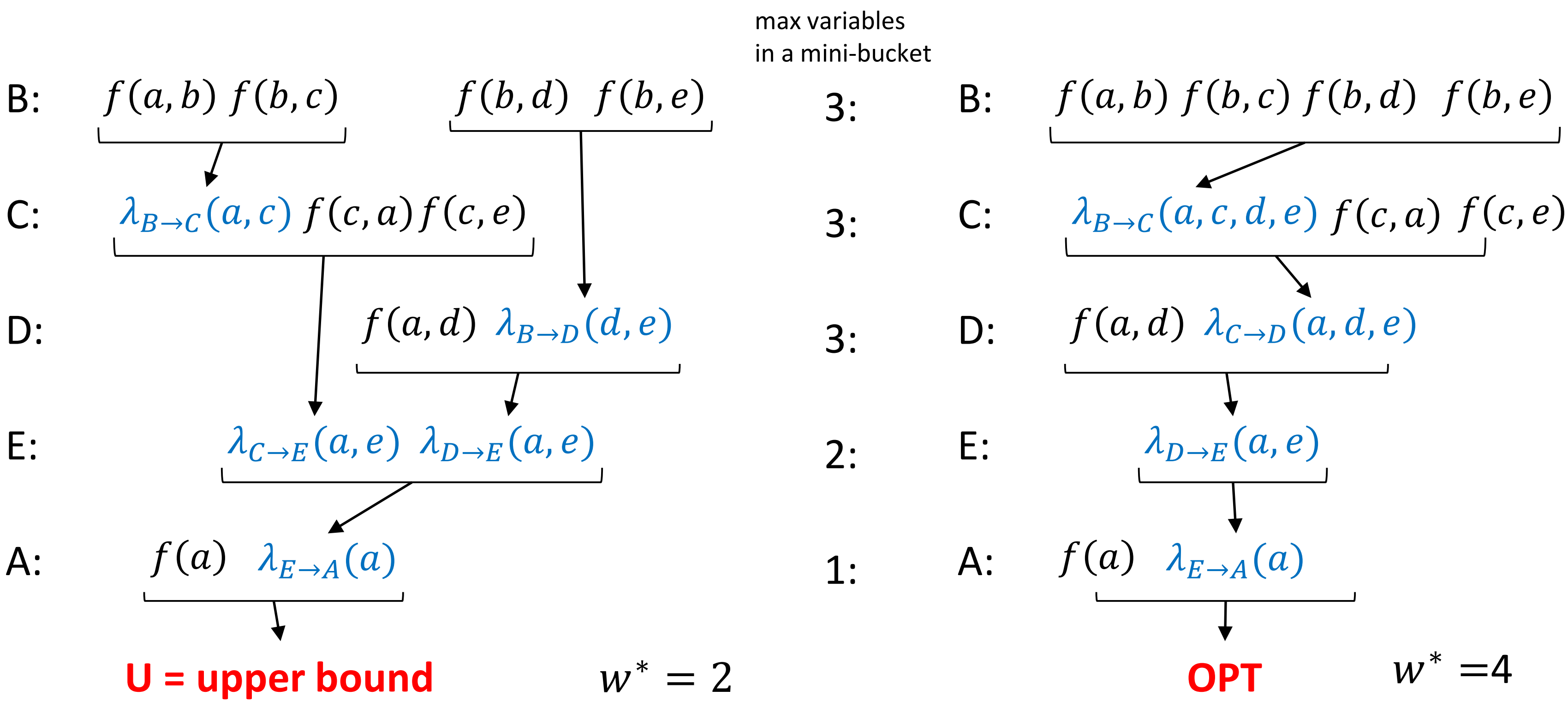
U = upper bound

Greedy configuration = lower bound

Bucket and Mini-Bucket Elimination

- **Input**: I – max number of variables allowed in a mini-bucket
- **Output**: [Lower bound (P of suboptimal solution), upper bound]

Example: MBE-mpe(3) versus **BE-mpe**

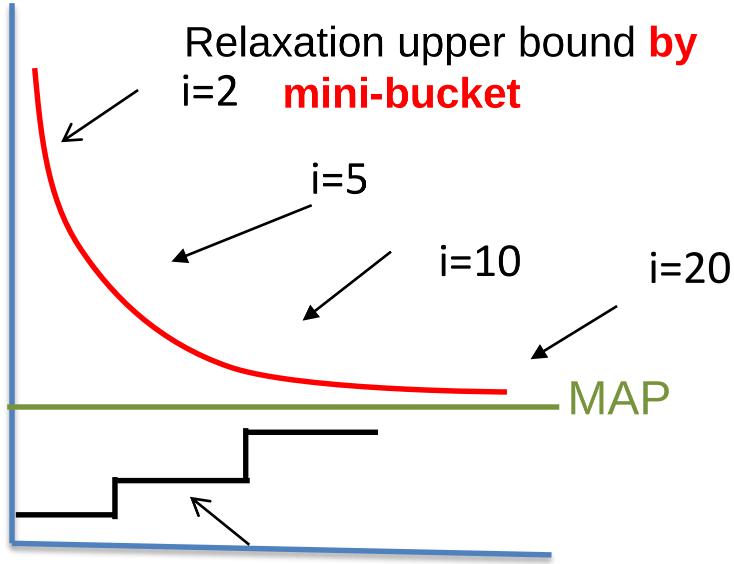


[Dechter and Rish, 1997]

Properties of Mini-Bucket Eliminaton

(For optimization)

- Bounding from above and below

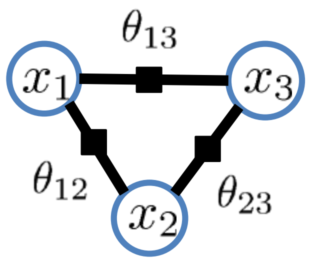


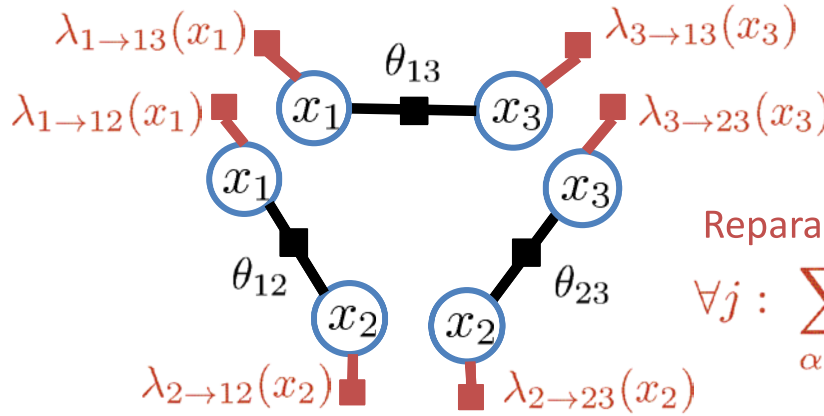
Consistent solutions (greedy search)

- Complexity: O(r exp(i)) time and O(exp(i)) space.
- Accuracy: determined by Upper/Lower bound.
- As i increases, both accuracy and complexity increase.
- Possible use of mini-bucket approximations:
  - As anytime algorithms
  - As heuristics in search
-

Tightening the Bound

Add factors that "adjust" each local term, but cancel out in total





Reparameterization:

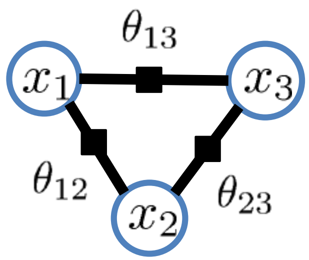
$$\forall j : \sum_{\alpha \ni j} \lambda_{j\to\alpha}(x_j) = 0$$

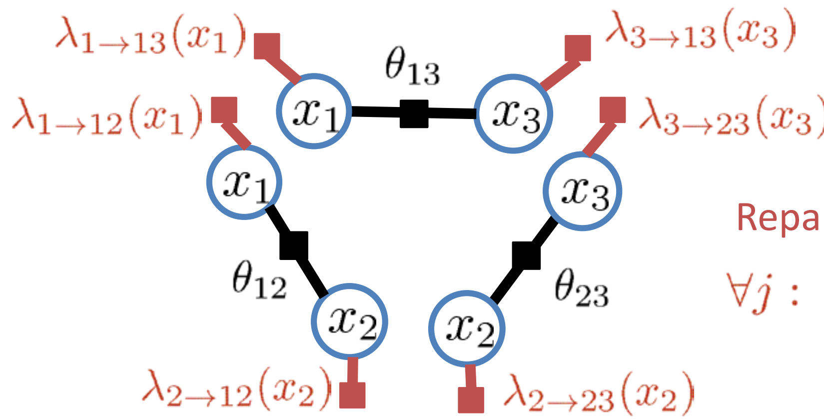
$$\log f(\mathbf{x}^*) = \max_{\mathbf{x}} \sum_{\alpha} \theta_\alpha(\mathbf{x}_\alpha) \quad \leq \min_{\{\lambda_{i\to\alpha}\}} \sum_{\alpha} \max_{\mathbf{x}_\alpha} \left[\theta_\alpha(\mathbf{x}_\alpha) + \sum_{i\in\alpha} \lambda_{i\to\alpha}(x_i) \right]$$

- Bound solution using decomposed optimization
  - Solve independently: optimistic bound
- Tighten the bound by re-parameterization
  - Enforces lost equality constraints using Lagrange multipliers

Tightening the Bound

Add factors that "adjust" each local term, but cancel out in total





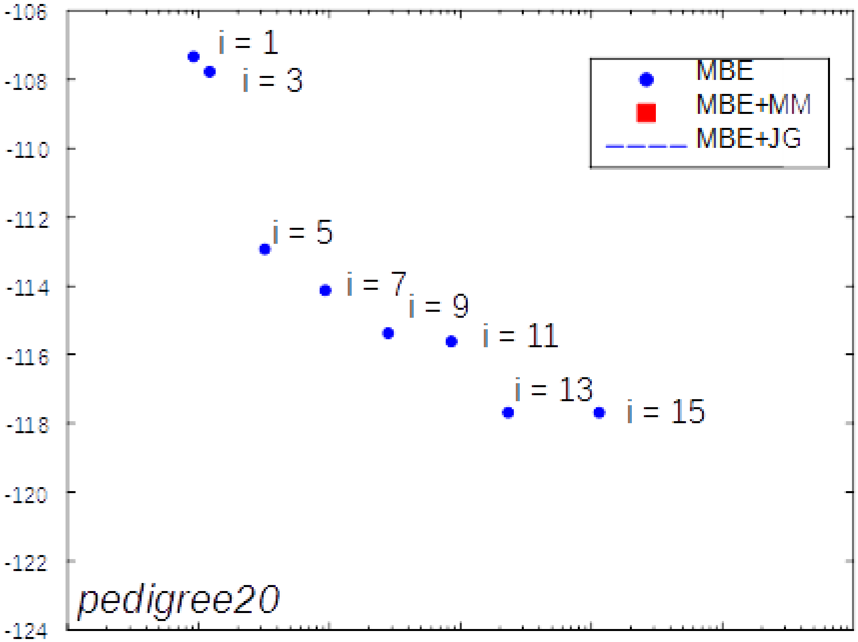
Reparameterization:

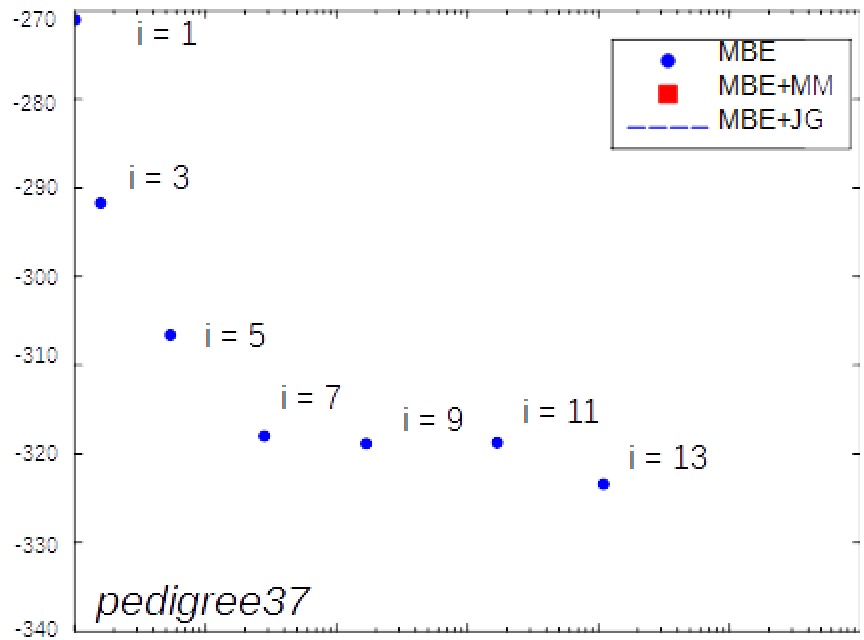
$$\forall j : \sum_{\alpha \ni j} \lambda_{j\to\alpha}(x_j) = 0$$

$$\log f(\mathbf{x}^*) = \max_{\mathbf{x}} \sum_\alpha \theta_\alpha(\mathbf{x}_\alpha) \quad \le \min_{\{\lambda_{i\to\alpha}\}} \sum_\alpha \max_{\mathbf{x}_\alpha} \left[\theta_\alpha(\mathbf{x}_\alpha) + \sum_{i\in\alpha} \lambda_{i\to\alpha}(x_i) \right]$$

- Many names for the same class of bounds
  - Dual decomposition [Komodakis et al. 2007]
  - TRW, MPLP [Wainwright et al. 2005; Globerson & Jaakkola 2007]
  - Soft arc consistency [Cooper & Schiex 2004]
  - Max-sum diffusion [Warner 2007]

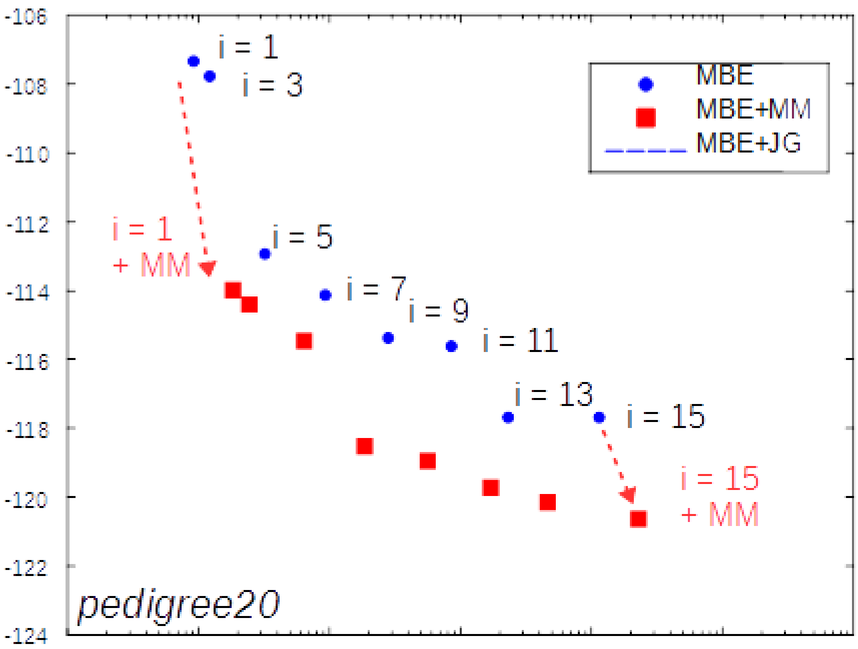
Anytime Approximation

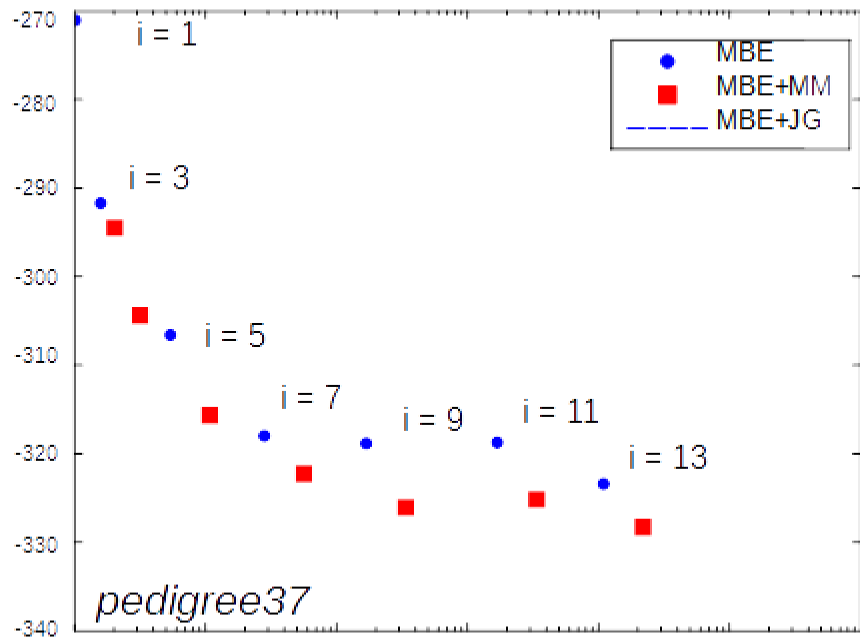




- Can tighten the bound in various ways
  - Cost-shifting (improve consistency between cliques)
  - Increase i-bound (higher order consistency)
- Simple moment-matching step improves bound significantly

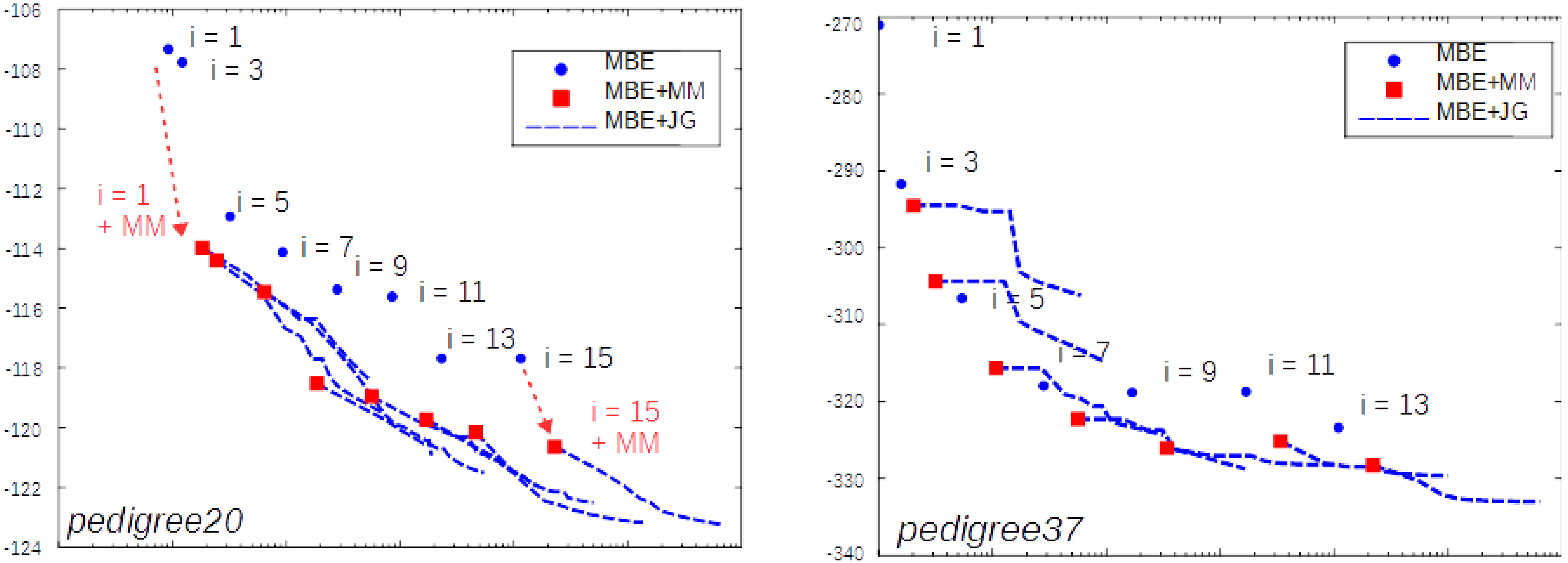
Anytime Approximation





- Can tighten the bound in various ways
  - Cost-shifting (improve consistency between cliques)
  - Increase i-bound (higher order consistency)
- Simple moment-matching step improves bound significantly

Anytime Approximation



- Can tighten the bound in various ways
  - Cost-shifting (improve consistency between cliques)
  - Increase i-bound (higher order consistency)
- Simple moment-matching step improves bound significantly

WMB for Marginal MAP

$$\sum_x f_1(x) \cdot f_2(x) \quad \leq \quad \left[\sum_x f_1(x)^{\frac{1}{w_1}}\right]^{w_1} \cdot \left[\sum_x f_2(x)^{\frac{1}{w_2}}\right]^{w_2}$$

Holder Inequality

$$(w_1 + w_2 = 1)$$

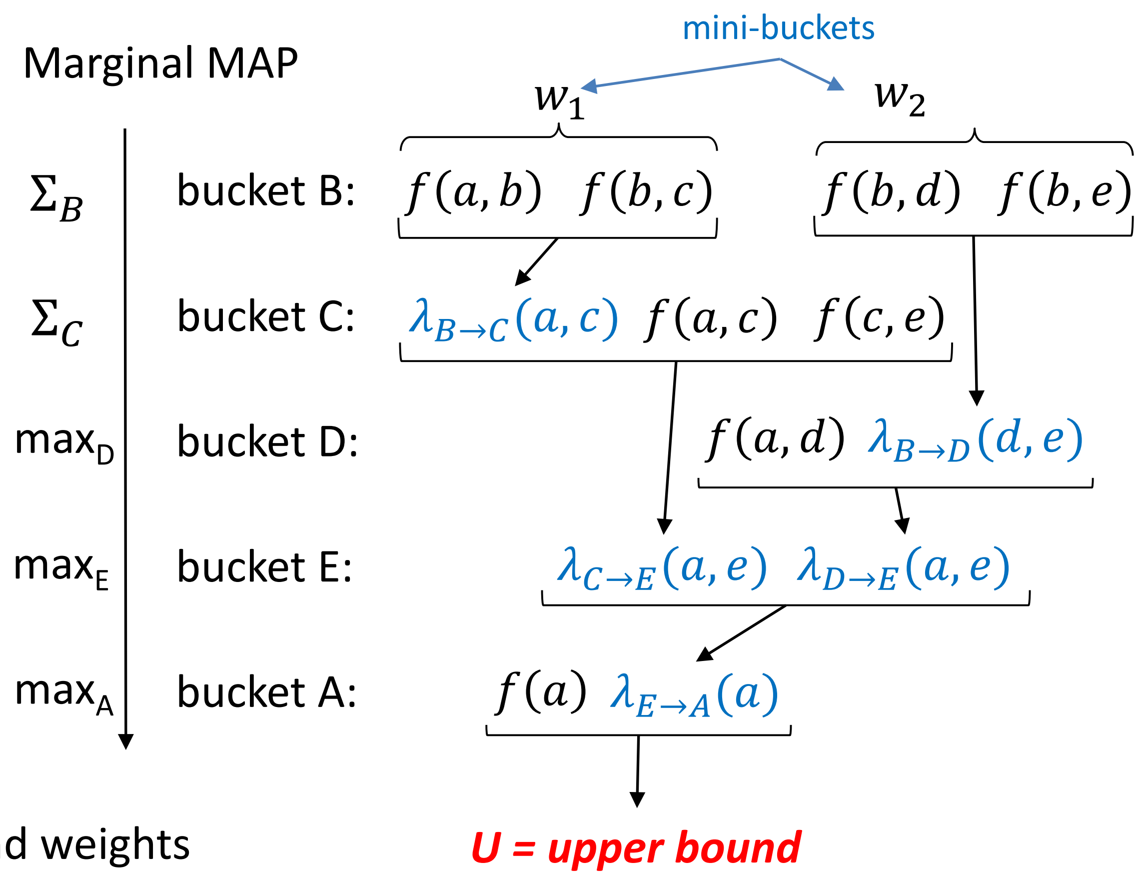
$$\lambda_{B\to C}(a,c) = \sum_b^{w_1} f(a,b) f(b,c)$$

$$\lambda_{B\to D}(d,e) = \sum_b^{w_2} f(b,d) f(b,e)$$

⋮

$$\lambda_{E\to A}(a) = \max_e \lambda_{C\to E}(a,e) \lambda_{D\to E}(a,e)$$

$$U = \max_a f(a) \lambda_{E\to A}(a)$$



Can optimize over cost-shifting and weights
(single pass "MM" or iterative message passing)

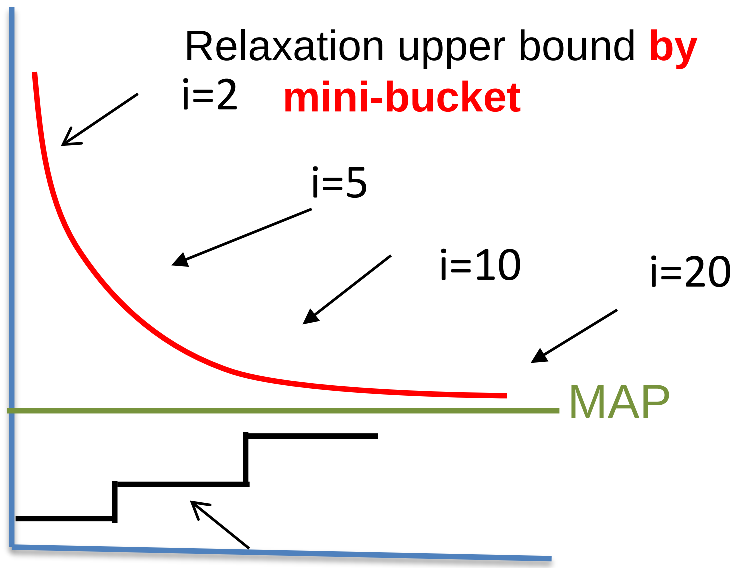
[Liu and Ihler, 2011; 2013]
[Dechter and Rish, 2003]

Bucket Elimination (BE) and WMB

[Dechter 1999, Ihler et. Al. 2013]

Holder inequality facilitated weighted MB for summation

- Bounding from above and below



Consistent solutions (greedy search)

Pros:

- Computationally bounded
- Gives upper or lower bound
- Cost-shifting Message passing
- improves bound

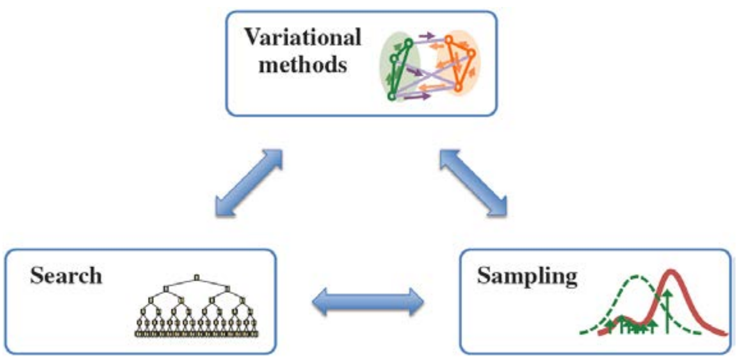
Cons:

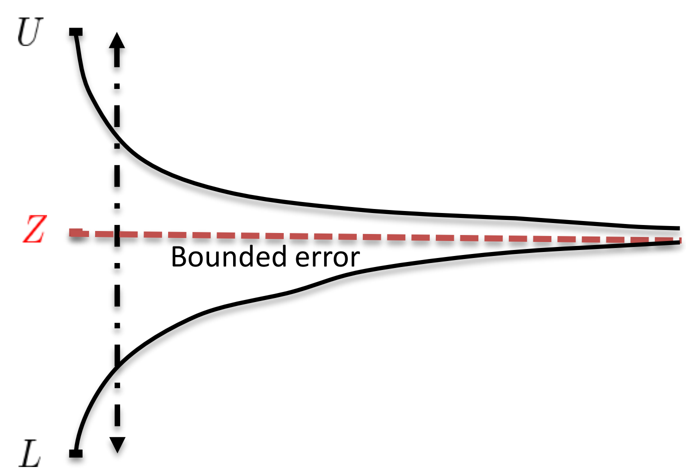
- Not anytime!

not asymp. tight w/o more memory

Outline

- Graphical models: definition, examples, methodology
- Inference and variational bounds
- AND/OR search spaces
- Variational bounds as search heuristics
- Combining methods: Heuristic Search for Marginal Map
- Conclusion





Potential Search Spaces

A	B	f_1
0	0	2
0	1	0
1	0	1
1	1	4

A	C	f_2
0	0	3
0	1	0
1	0	0
1	1	1

A	E	f_3
0	0	0
0	1	3
1	0	2
1	1	0

A	F	f_4
0	0	2
0	1	0
1	0	0
1	1	2

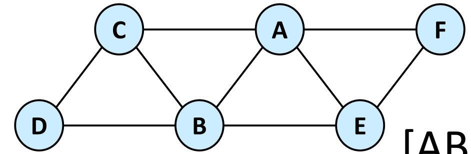
B	C	f_5
0	0	0
0	1	1
1	0	2
1	1	4

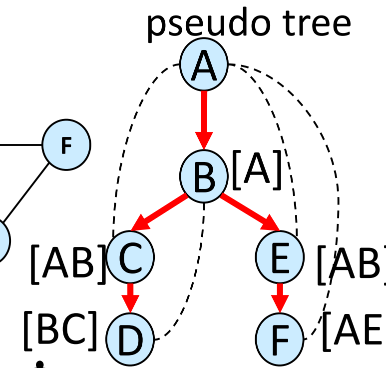
B	D	f_6
0	0	4
0	1	2
1	0	1
1	1	0

B	E	f_7
0	0	3
0	1	2
1	0	1
1	1	0

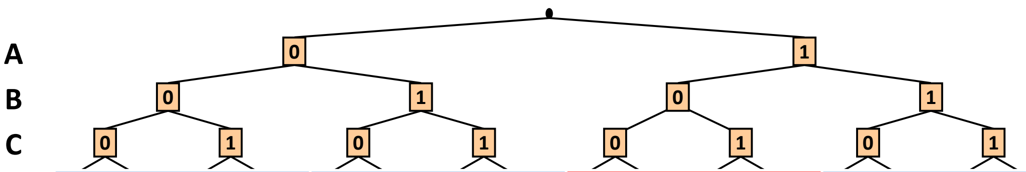
C	D	f_8
0	0	1
0	1	4
1	0	0
1	1	0

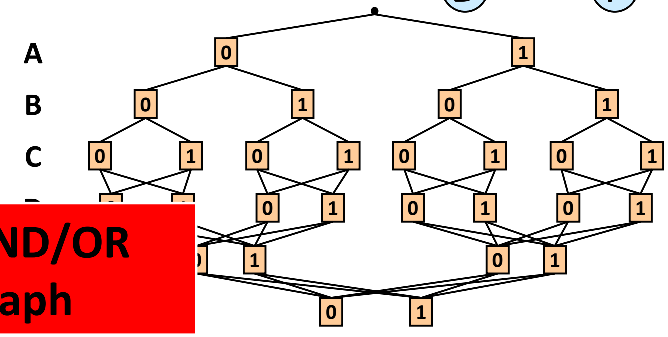
E	F	f_9
0	0	1
0	1	0
1	0	0
1	1	2





[Dechter & Mateescu, 2007]

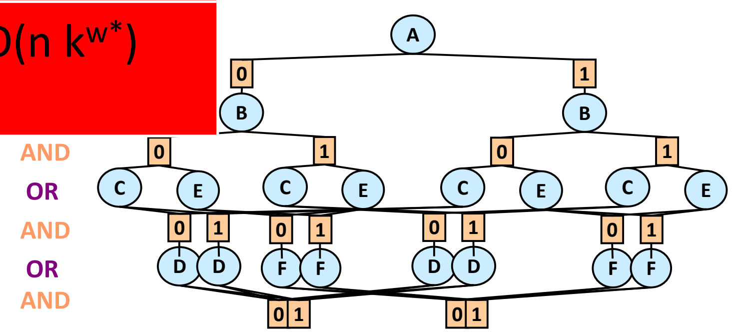




	OR tree	AND/OR tree	OR graph	AND/OR graph
time	$O(k^n)$	$O(nk^h)$	$O(n\ k^{pw^*})$	$O(n\ k^{w^*})$
memory	O(n)	O(n)	$O(n\ k^{pw^*})$	$O(n\ k^{w^*})$

Context minimal OR search graph

28 nodes



Context minimal AND/OR search graph

18 AND nodes

Computes any query:

- Constraint satisfaction
- Optimization (MAP)
- Marginal (P(e))
- Marginal map

Any query is best computed
Over the c-minimal AO search space

AND/OR Tree: An Alternative Model

$P(E \mid A, B)$

A	B	E=0	E=1
0	0	.4	.6
0	1	.5	.5
1	0	.7	.3
1	1	.2	.8

Evidence: E=0

$P(B \mid A)$

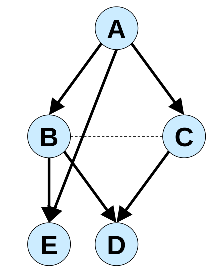
A	B=0	B=1
0	.4	.6
1	.1	.9

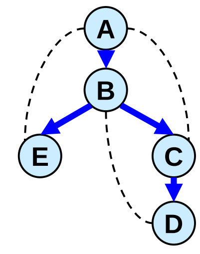
$P(C \mid A)$

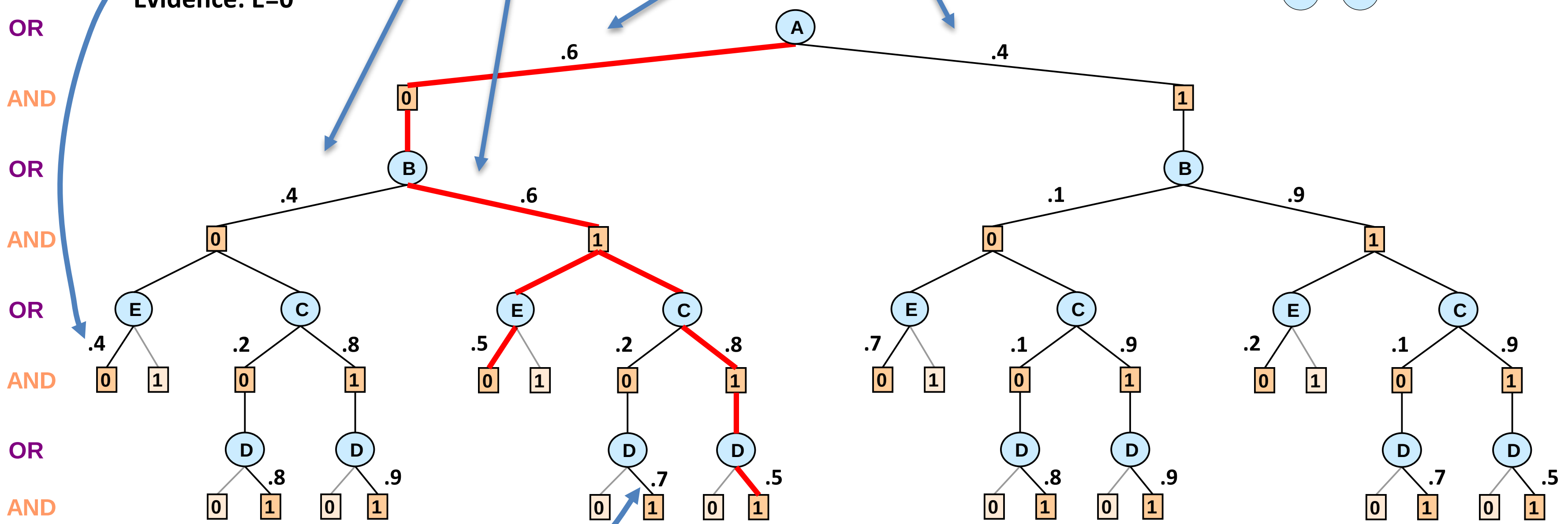
A	C=0	C=1
0	.2	.8
1	.7	.3

$P(A)$

A	P(A)
0	.6
1	.4







$P(D \mid B, C)$

B	C	D=0	D=1
0	0	.2	.8
0	1	.1	.9
1	0	.3	.7
1	1	.5	.5

Evidence: D=1

Cost of the solution tree: the product of weights on its arcs

Cost of (A=0,B=1,C=1,D=1,E=0) = 0.6 · 0.6 · 0.5 · 0.8 · 0.5 = 0.0720

Value of a Node (e.g., Probability of Evidence)

$P(E \mid A,B)$

A	B	E=0	E=1
0	0	.4	.6
0	1	.5	.5
1	0	.7	.3
1	1	.2	.8

Evidence: E=0

$P(B \mid A)$

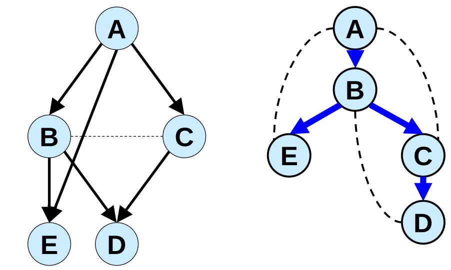
A	B=0	B=1
0	.4	.6
1	.1	.9

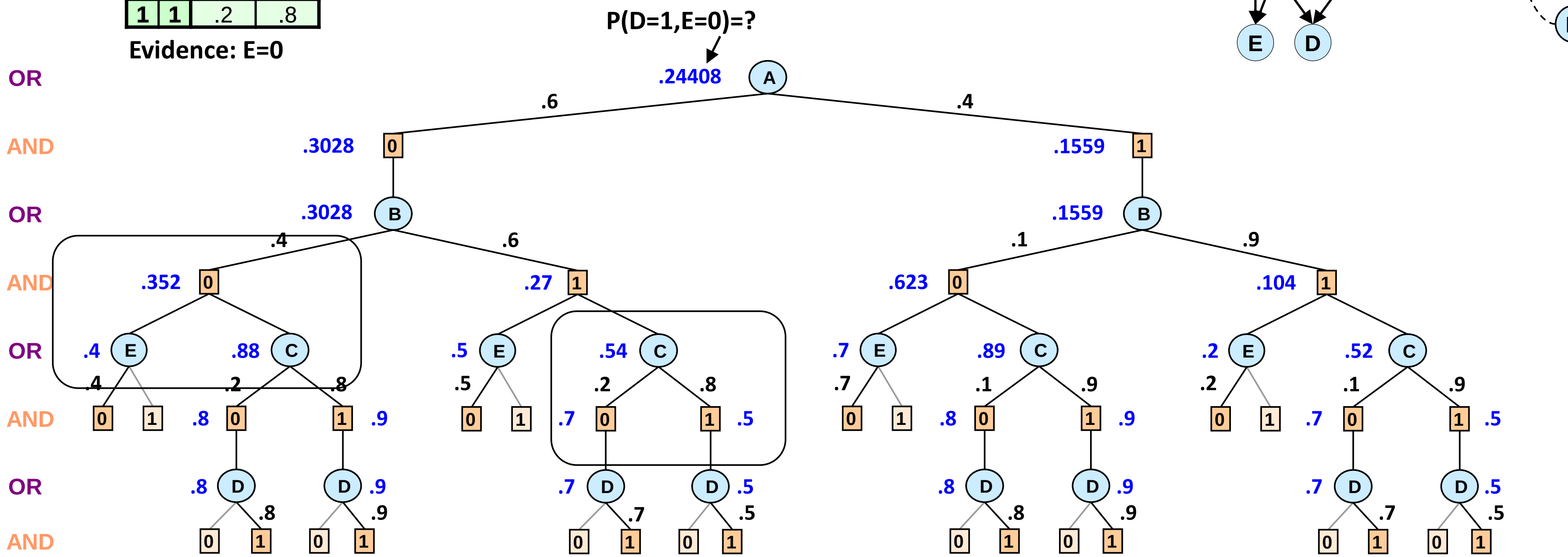
$P(C \mid A)$

A	C=0	C=1
0	.2	.8
1	.7	.3

$P(A)$

A	P(A)
0	.6
1	.4





$P(D \mid B,C)$

B	C	D=0	D=1
0	0	.2	.8
0	1	.1	.9
1	0	.3	.7
1	1	.5	.5

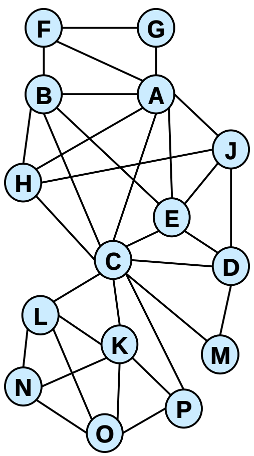
Evidence: D=1

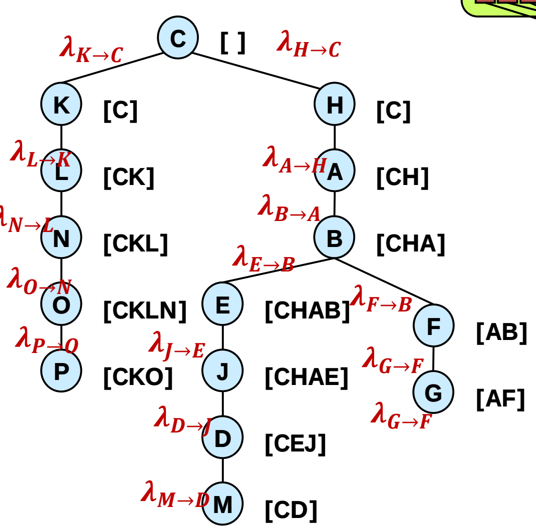
Value of node = updated belief for sub-problem below

AND node: product $\prod_{n' \in children(n)} v(n')$

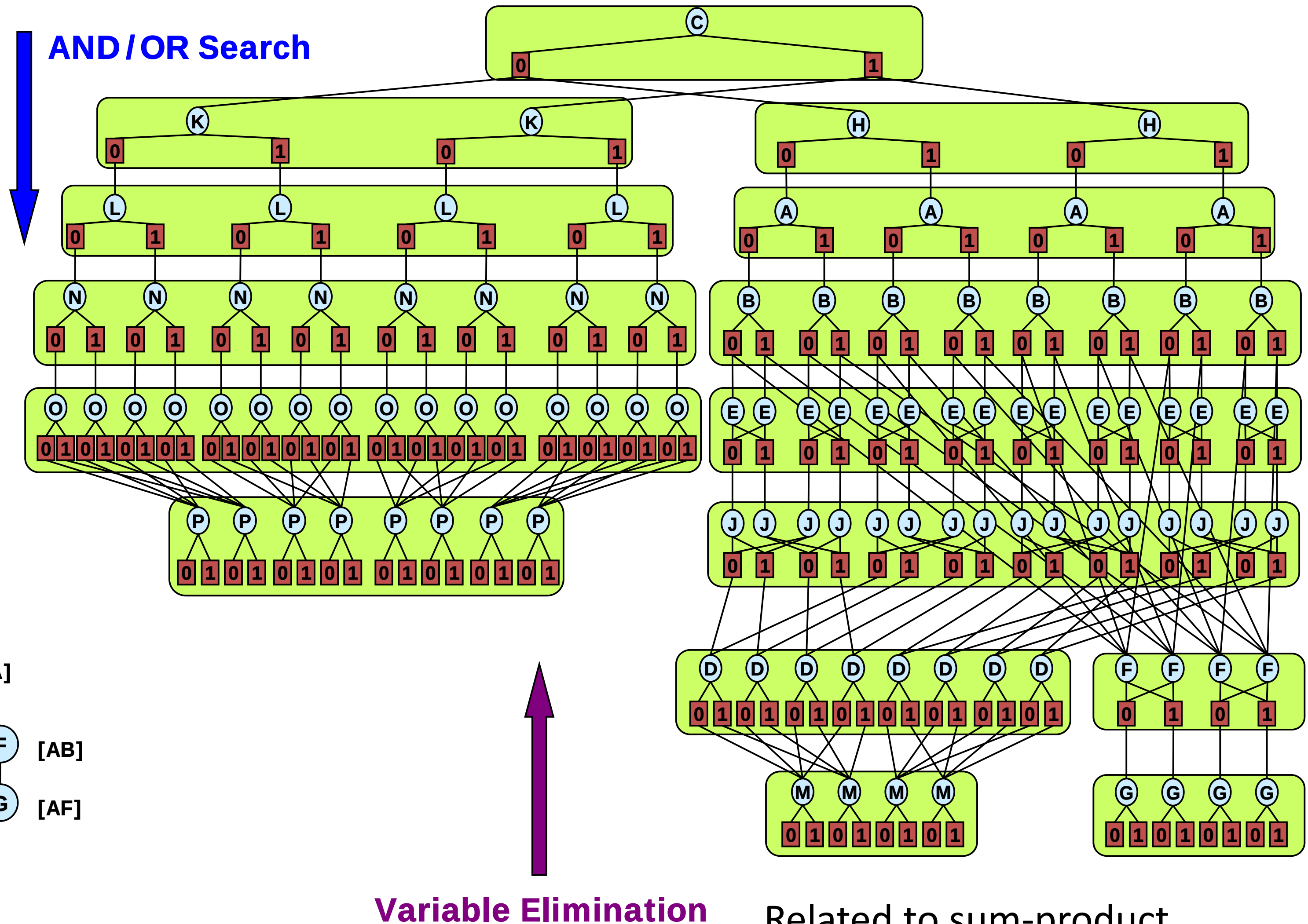
OR node: Marginalization by summation $\sum_{n' \in children(n)} w(n,n')\, v(n')$

AND/OR Search and Variable Elimination



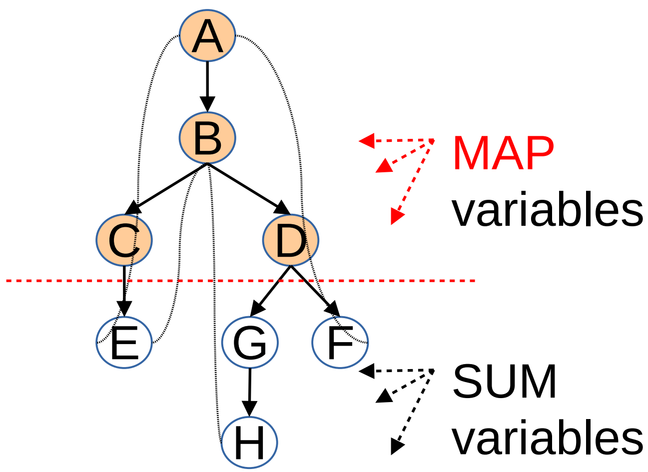


(C K H A B E J L N O D P M F G)

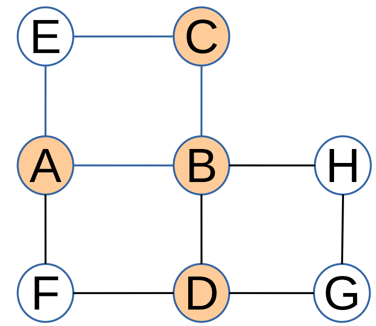


Related to sum-product Networks or Arithmetic circuit

AND/OR Search for Marginal MAP



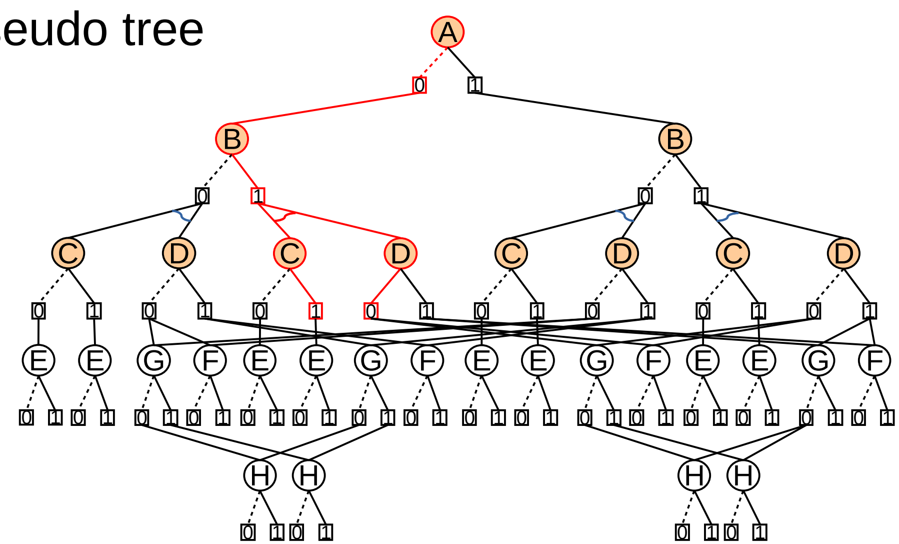
constrained
pseudo tree



primal graph

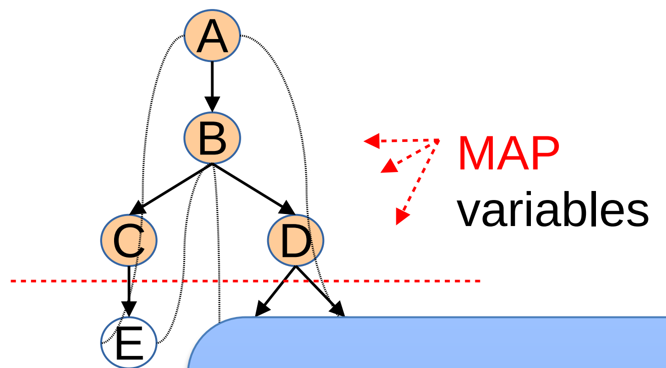
$$X_M = \{A, B, C, D\}$$

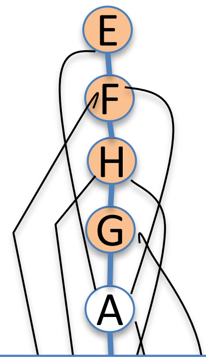
$$X_S = \{E, F, G, H\}$$

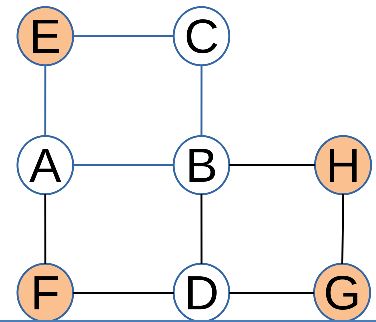


- Node types
  - OR (MAP): max
  - OR (SUM): sum
  - AND: multiplication

AND/OR Search for Marginal MAP







C-induced-width = 6

Induced-width = 2

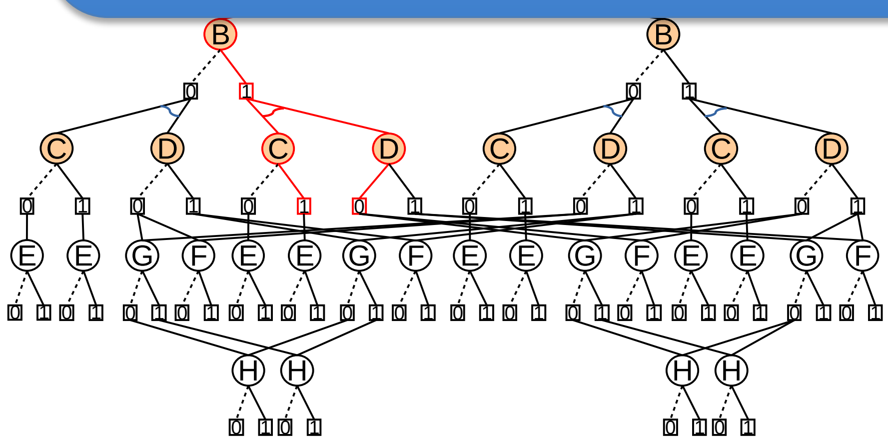
const

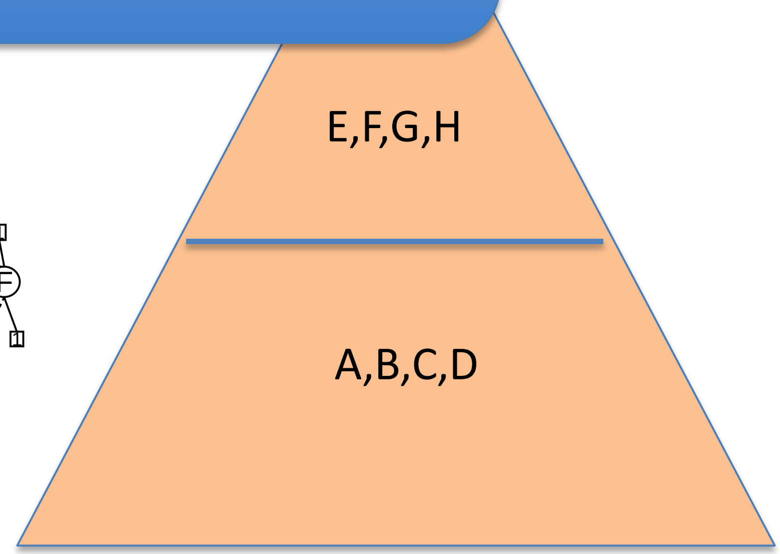
pseu

For MMAP search space is:

k^{h_c} on a AND/OR tree

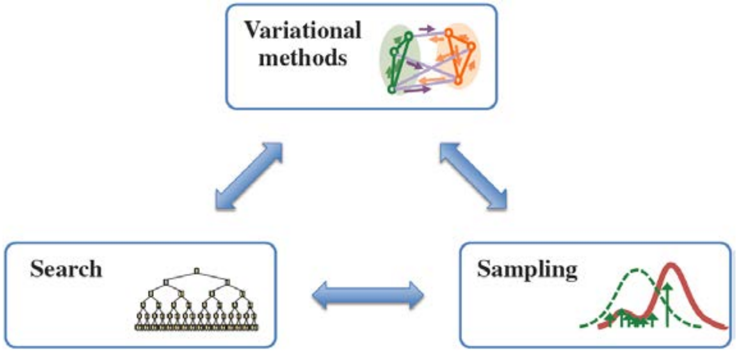
k^{w_c} on AND/OR graph

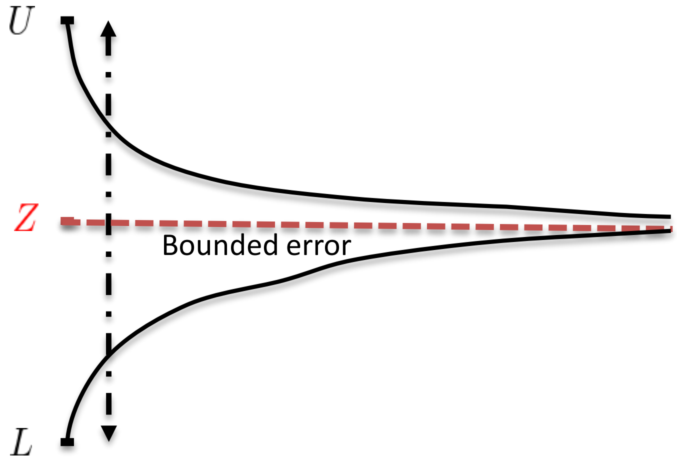




Outline

- Graphical models, The Marginal Map task
- Inference and variational bounds
- AND/OR search spaces
- Combining methods: Heuristic Search for Marginal Map
- Combining methods: Sampling
- Conclusion

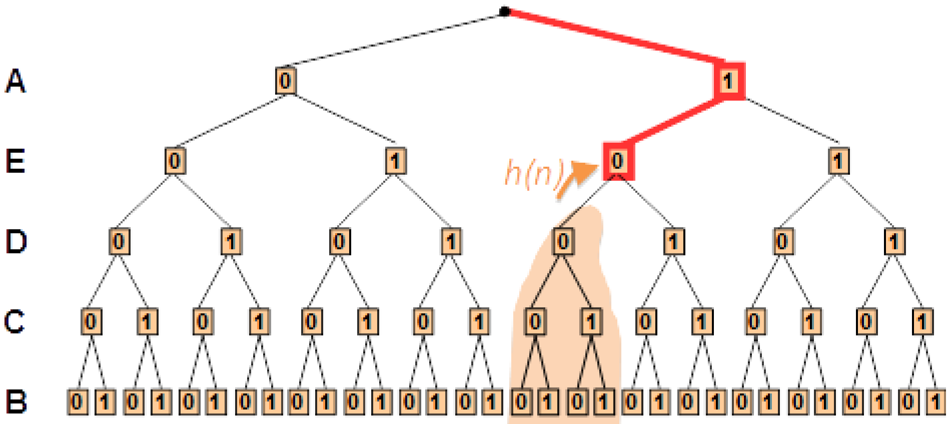


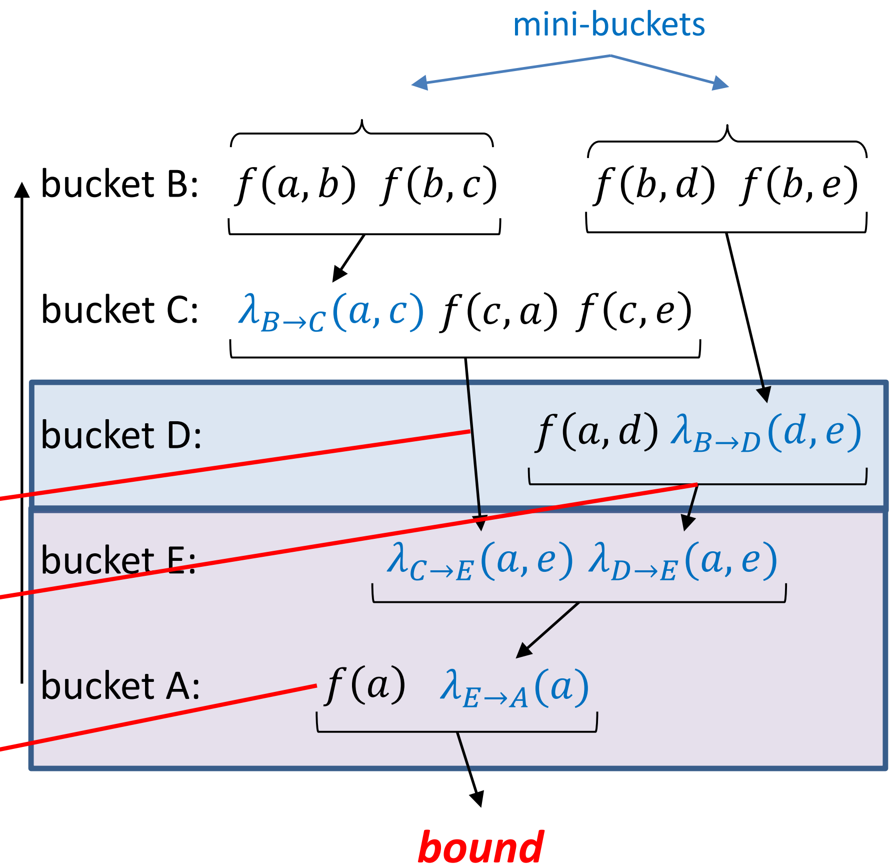


Search Aided by Variational Heuristics

[Kask, Dechter, AIJ 2001]

Given a partial assignment, $[\hat{a} = 1, \hat{e} = 0]$
(weighted) mini-bucket gives an admissible heuristic:





"cost to go":

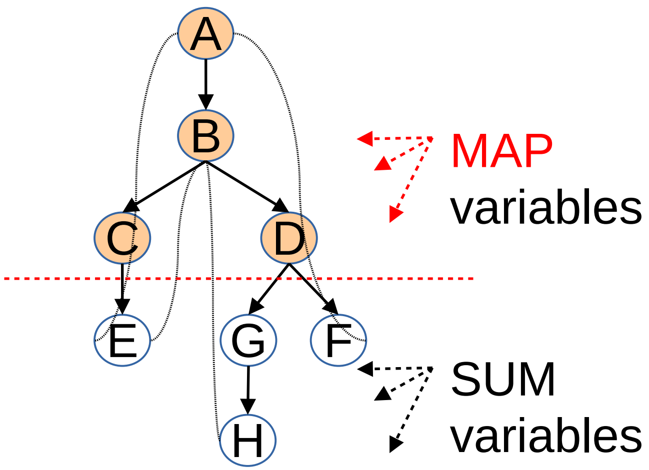
$$\tilde{h}(\hat{a},\hat{e},D) = \lambda_{C\to E}(\hat{a},\hat{e}) + f(\hat{a},D) + \lambda_{B\to D}(D,\hat{e})$$

"cost so far":

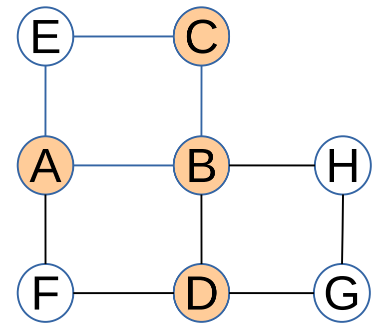
$$g(\hat{a},\hat{e},D) = f(A=\hat{a})$$

For MAP, marginal map and partition function

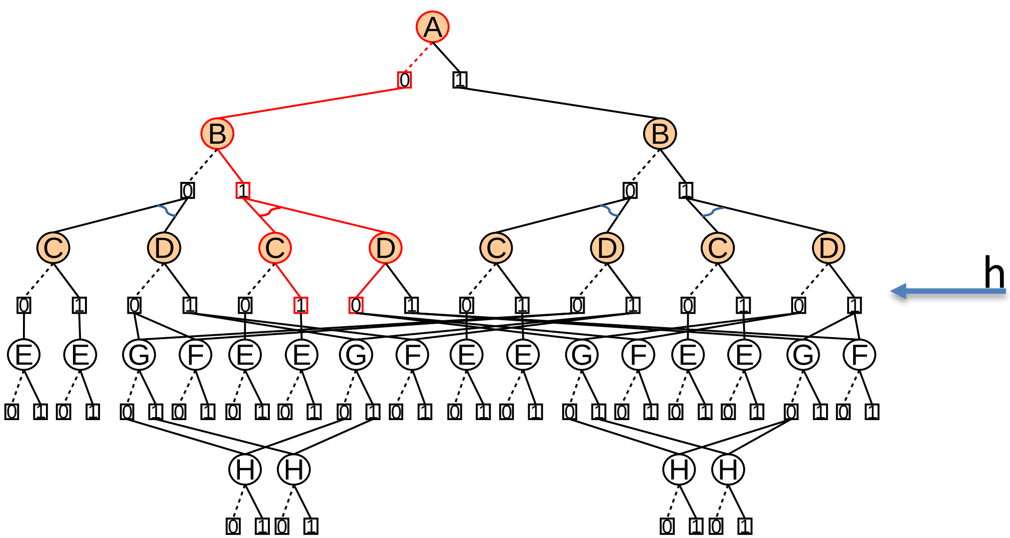
AND/OR Search for Marginal MAP

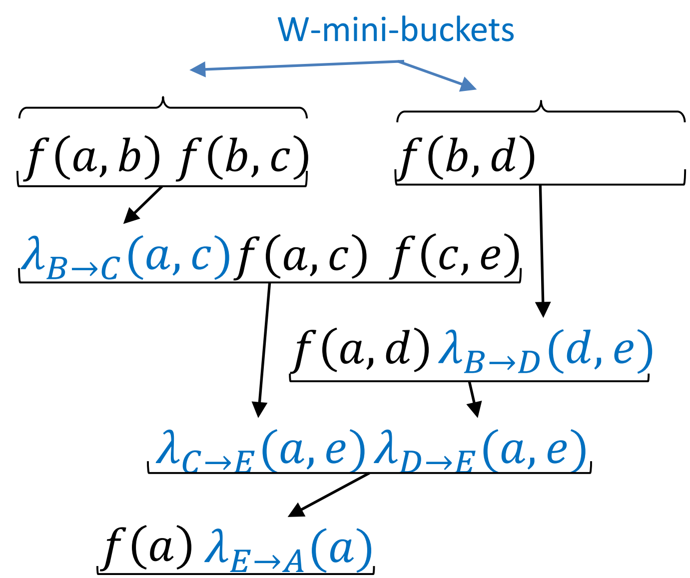


constrained
pseudo tree



primal graph

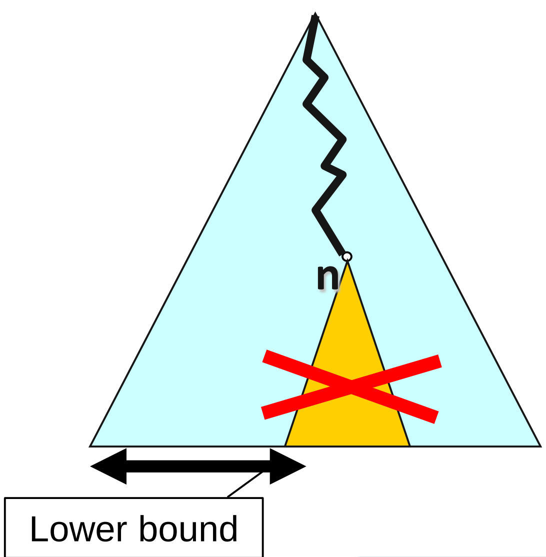




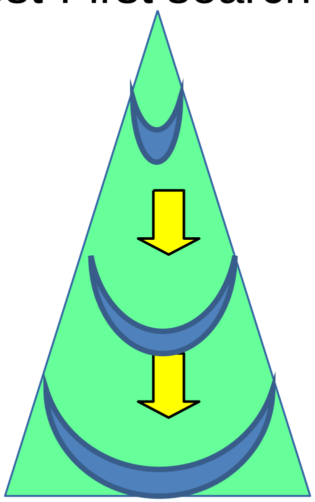
Exact MMAP Solvers: Best or Depth-First Search?

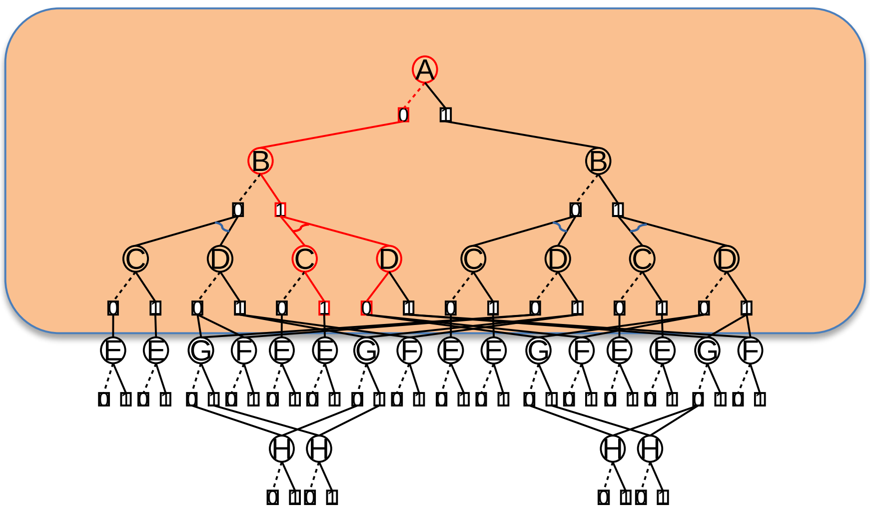
[Marinescu, Dechter, Ihler, AAAI 2014]

Depth-First search



Best-First search





The MAP search space

Best-first search is superior
Expanding fewer full MAP
Solutions, thus less
conditional sums

Anytime Solvers for Marginal MAP

[Marinsecu, Lee, Dechter, Ihler, AAAI-2017, JAIR 2019]

- **Weighted Best-First search:**
  - Weighted Restarting AOBF (WAOBF)
  - Weighted Restarting RBFAOO (WRBFAOO)
  - Weighted Repairing AOBF (WRAOBF)

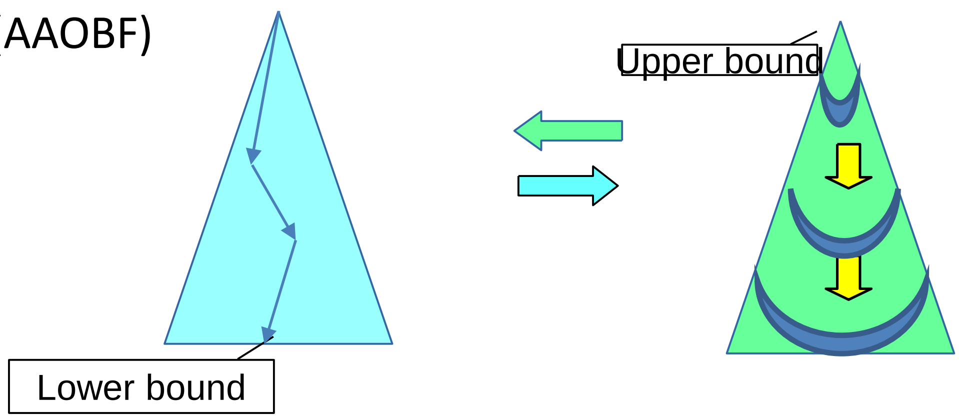
Weighted A* search [Pohl 1970]

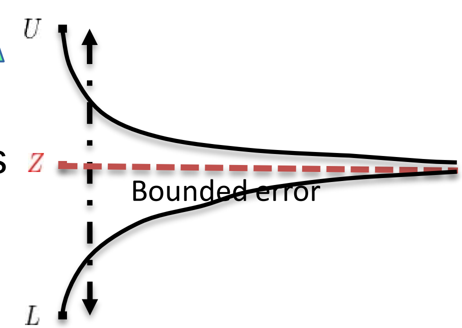
- non-admissible heuristic
- Evaluation function:

  f(n)=g(n)+w·h(n)

- **Guaranteed w-optimal solution, cost $C \leq w{\cdot}C^*$**

- **Interleaving Best-first and depth-first search**:
  - Look-ahead (LAOBF),
  - alternating (AAOBF)

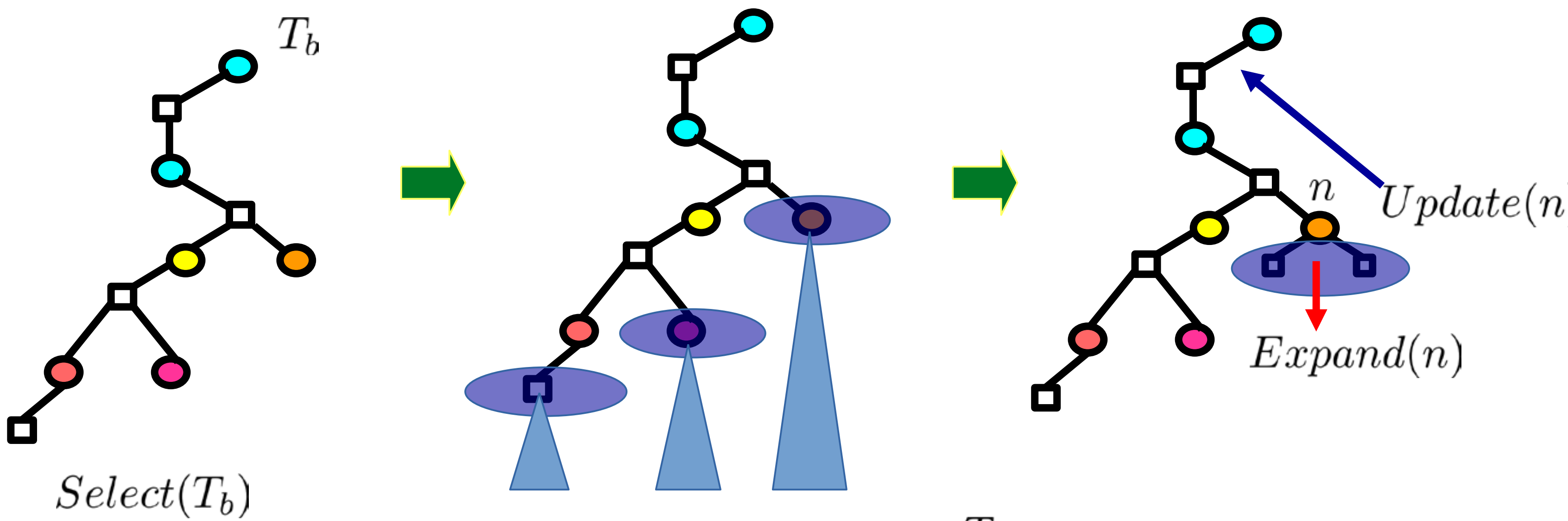




- Better guidance for depth-first dives using improved heuristics
- Memory robust best-first search
- using improved lower bounds

LAOBF (Best-first AND/OR Search with Depth-First lookaheads)

Best-first selection | Depth-first lookahead | Best-first expansion & update



- depth-first dive at the tip of T_b
- compute global lower bound
- cache summation subproblems

- Select a tip node n
- Expand and Update n

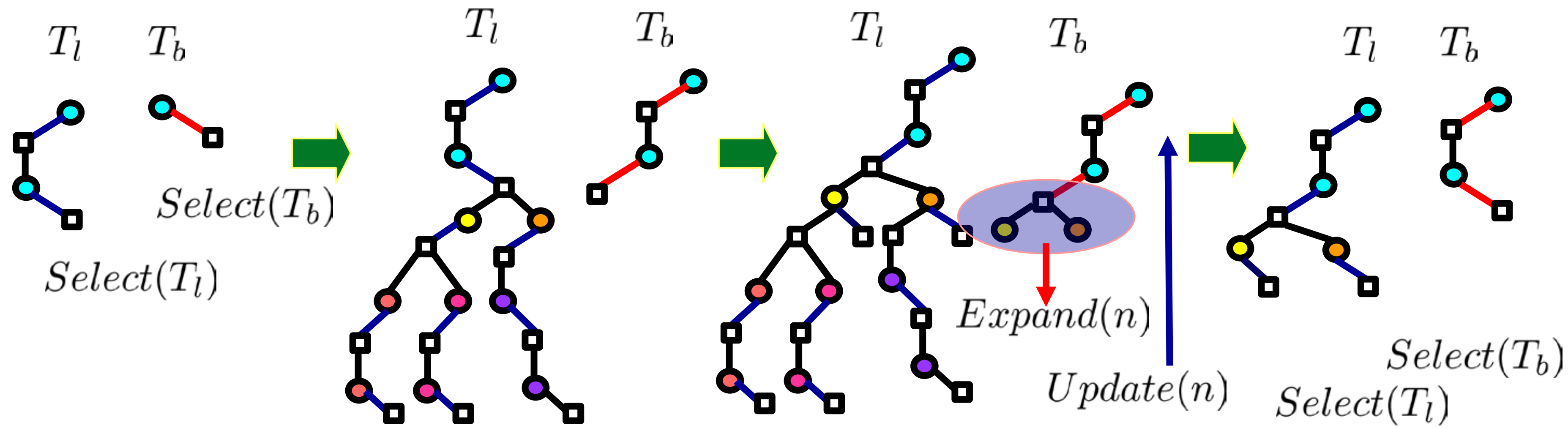
Cutoff parameter: perform depth-first dive at every θ number of node expansions.
best partial solution tree: T_b

AAOBF (Alternating Best-First and Depth-First)

Depth-first selection
Best-first selection

Depth-first greedy expansion
Best-first re-direct T_b

Depth-first re-direct T_l
Best-first expansion & update



Benchmarks and Evaluation Methods

Benchmark		#. inst	n	k	w_c	h_c	w_u	h_u
grid	easy	15	144 – 1156	2 – 2	16 – 52	50 – 164	15 – 49	48 –198
	hard	60	144 – 1156	2 – 2	25 – 375	42 – 421	–	–
pedigree	easy	10	334 – 1289	4 – 7	35 – 237	51 – 134	15 – 29	60 –160
	hard	40	334 – 1289	4 – 7	35 – 237	63 – 259	–	–
promedas	easy	10	453 – 1849	2 – 2	10 – 122	42 – 174	10 – 106	43 –157
	hard	40	453 – 1849	2 – 2	11 – 490	36 – 507	–	–

Table 1: Benchmark instances. #. inst is the number of instances in each domain. We also distinguish easy and hard instances. The minimum and the maximum values from the set of problems are shown in the following parameters: n is the number of variables, k is the maximum domain size, w_c is the constrained induced width, h_c is the height of the pseudo tree corresponding to the constrained elimination ordering. The unconstrained induced width, w_u and pseudo tree height, h_u are also shown to highlight the difficulty of hard Marginal MAP problem instances.

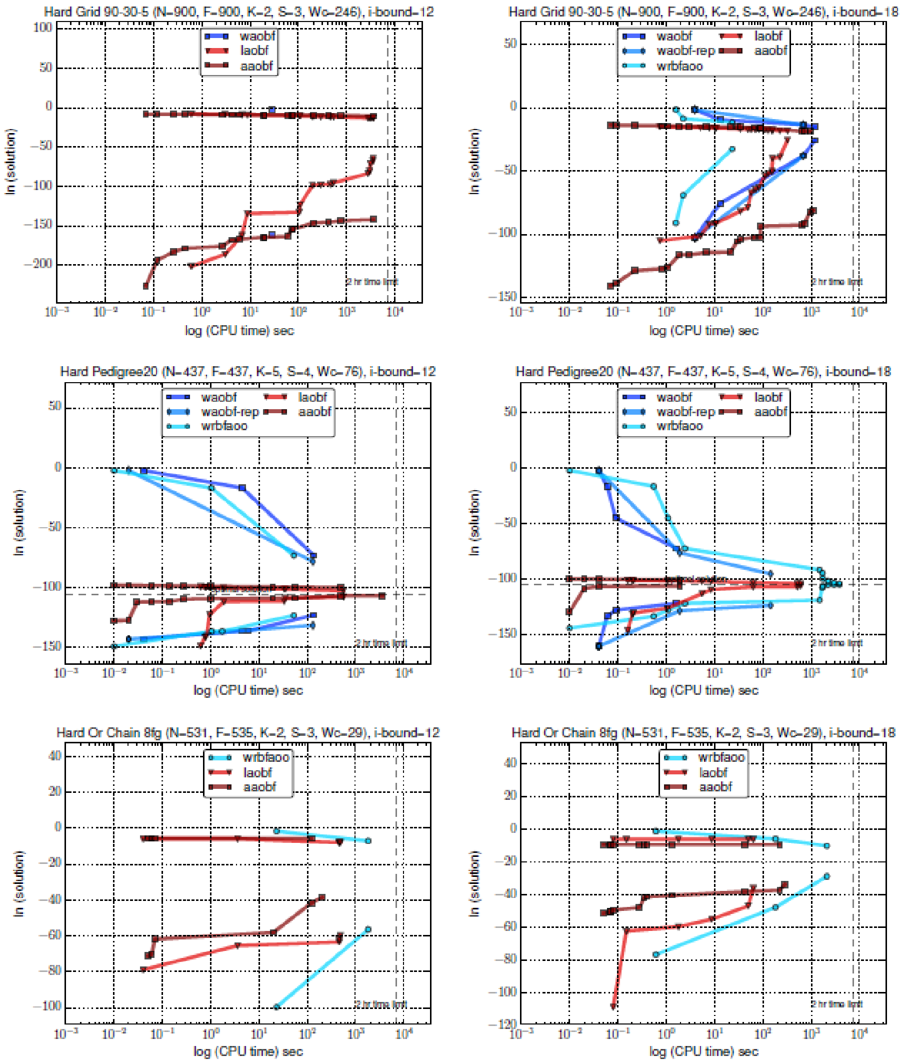
Anytime Bounds of Marginal MAP

- Search: LAOBF, AAOBF, BRAOBB, WAOBF,WAOBF-rep
- heuristic: WMB-MM (20)
- memory: 24 GB

- Anytime lower and upper bounds from hard problem instances with **i-bound 12 (left) and 18 (right)**.

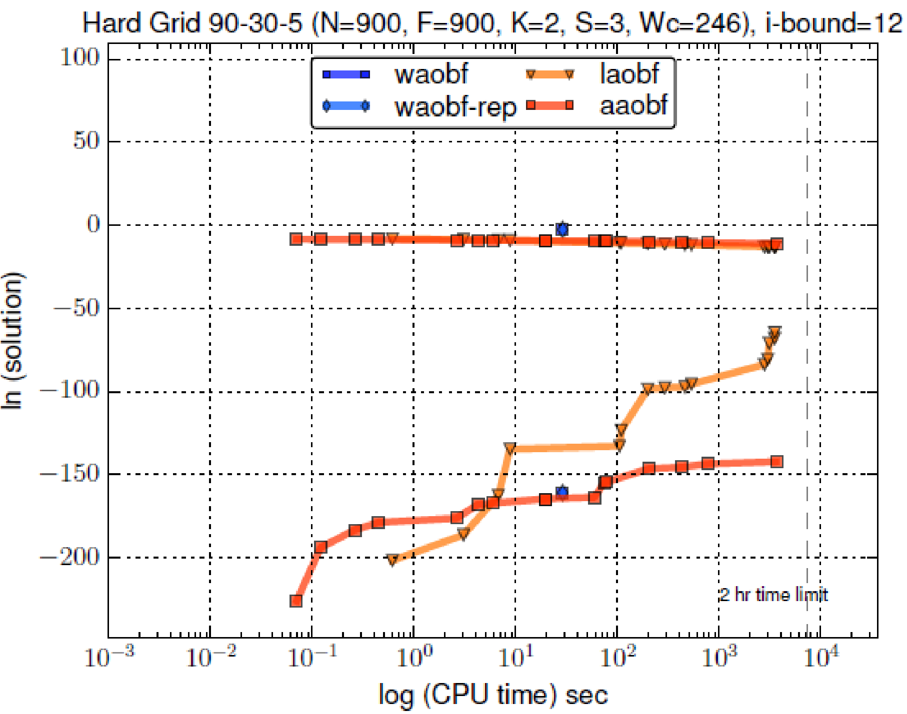
- The horizontal axis is the CPU time in log scale and the vertical axis is the value of marginal MAP in log scale.

- Benchmarks: Pedigrees, promedas, grids, planning. A fraction of variables selected as MAP (10% hard instances).

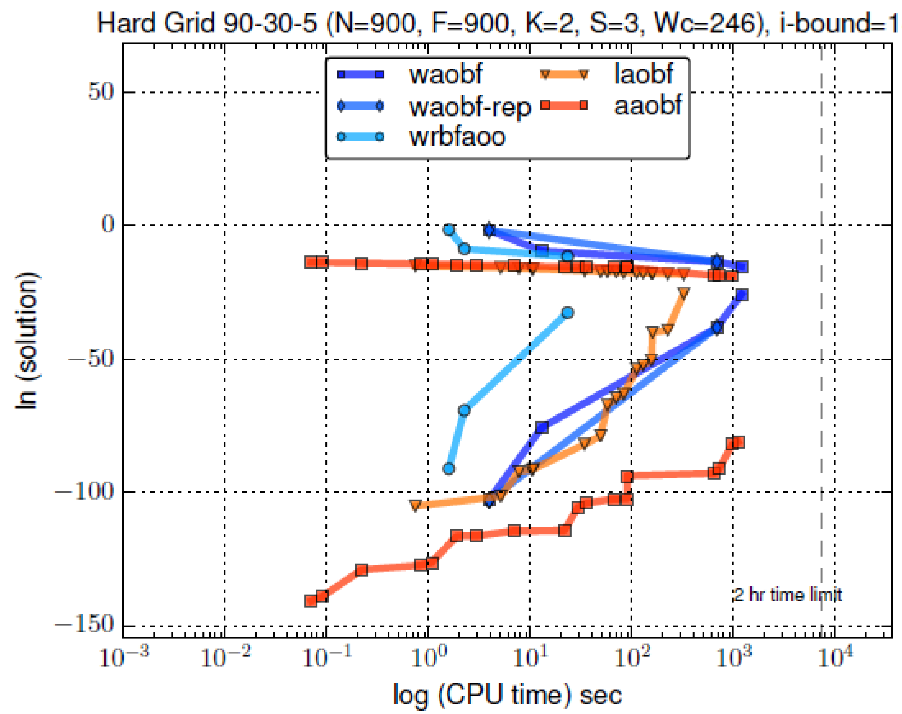


Anytime Bounds of Marginal MAP

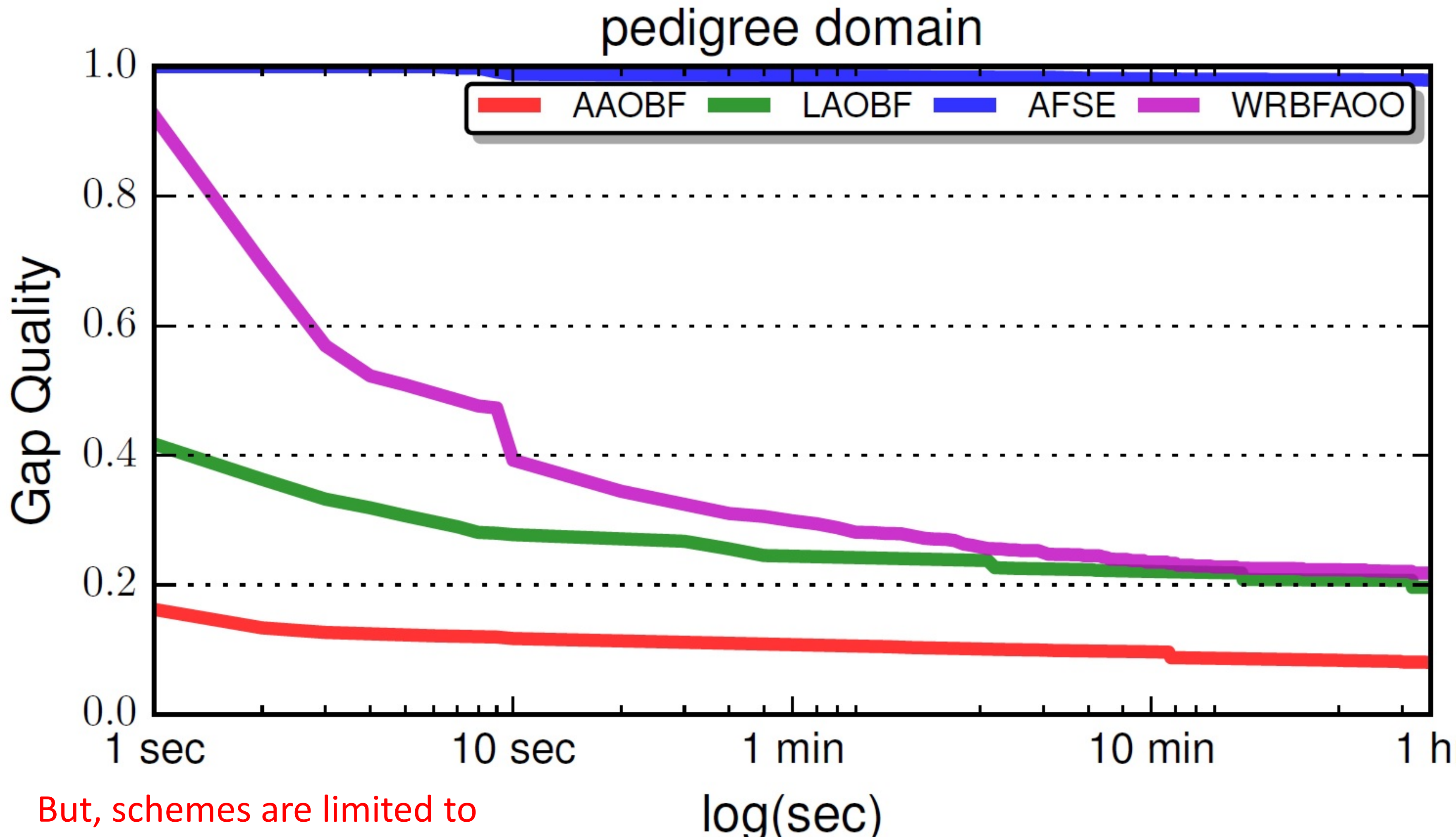
i-bound = 12



i-bound = 18



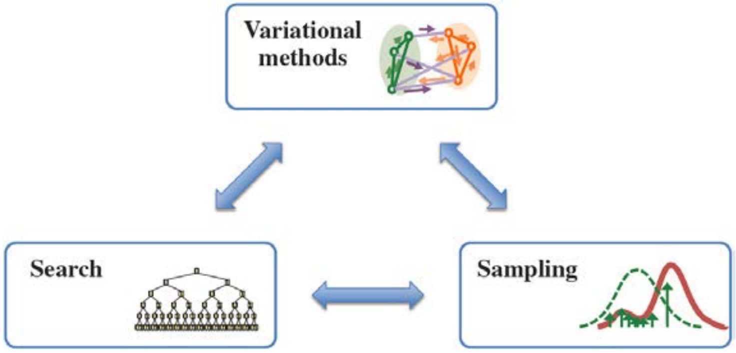
Average Gap Quality

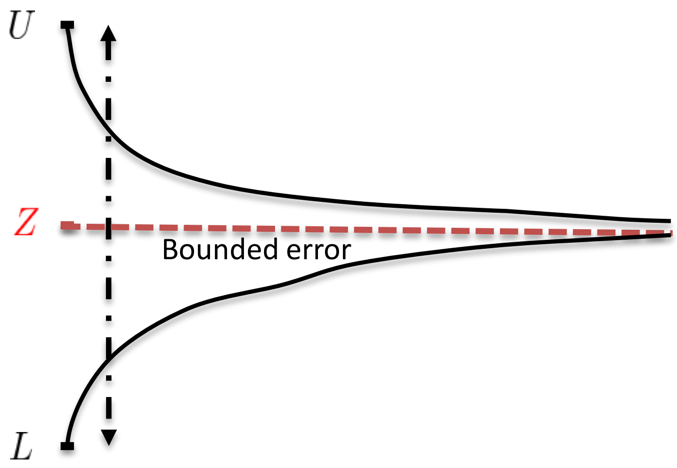


But, schemes are limited to tractable conditioned-summation

Outline

- Graphical models, The Marginal Map task
- Inference and variational bounds
- AND/OR search spaces
- Combining methods: Heuristic Search for Marginal Map
- Combining methods: Sampling
- Conclusion





Importance Sampling

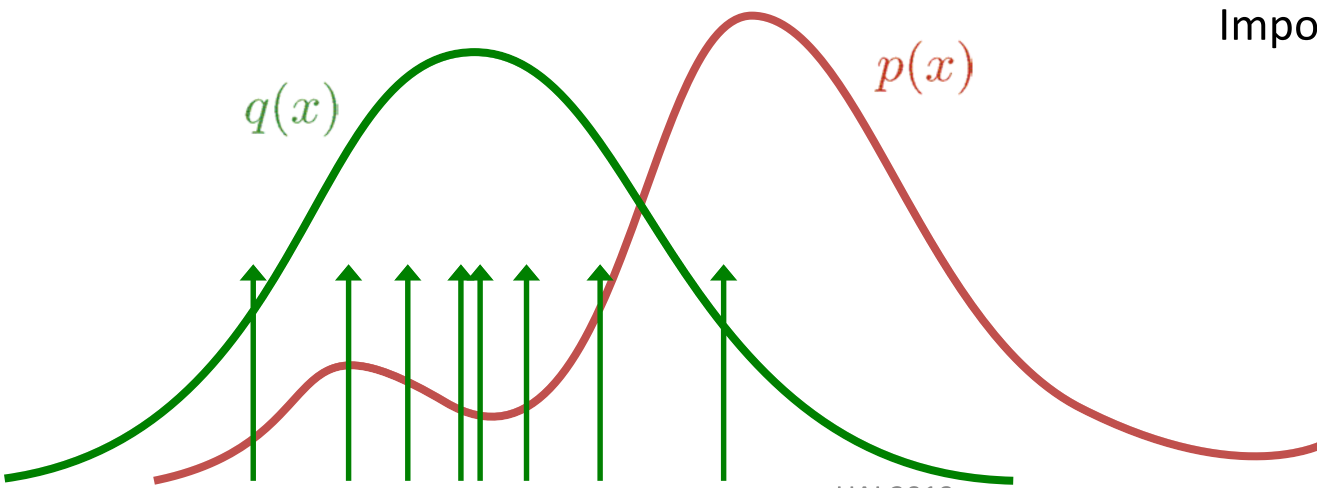
- Basic empirical estimate of probability:

$$\mathbb{E}[u(x)] = \int p(x)u(x) \;\approx\; \hat{u} = \frac{1}{m}\sum_i u(\tilde{x}^{(i)}) \qquad \tilde{x}^{(i)} \sim p(x)$$

- Importance sampling:

$$\int p(x)u(x) = \int q(x)\frac{p(x)}{q(x)}u(x) \approx \frac{1}{m}\sum_i \frac{p(\tilde{x}^{(i)})}{q(\tilde{x}^{(i)})}u(\tilde{x}^{(i)}) \qquad \tilde{x}^{(i)} \sim q(x)$$

Importance weights



Choosing a Proposal- WMB-IS

[Liu, Fisher, Ihler 2015]

$$\Pr\left[|\hat{Z} - Z| > \epsilon\right] \le 1 - \delta \qquad \epsilon = \sqrt{\frac{2\hat{V}\log(4/\delta)}{m}} + \frac{7\,U\,\log(4/\delta)}{3(m-1)}$$

- Can use WMB upper bound to define a proposal $q_{\text{wmb}}(x)$

$$\tilde{\mathbf{b}} \sim w_1\, q_1(b|\tilde{a}, \tilde{c}) \;+\; w_2\, q_2(b|\tilde{d}, \tilde{e})$$

Weighted mixture:

use minibucket 1 with probability w_1

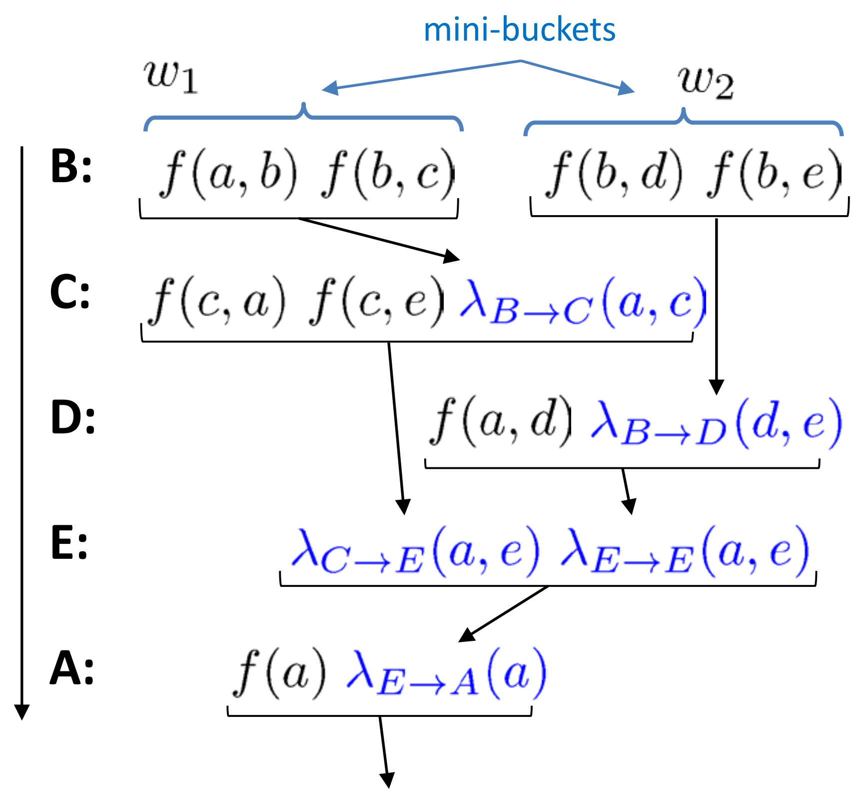
or, minibucket 2 with probability $w_2 = 1 - w_1$

where

$$q_1(b|a,c) = \left[\frac{f(a,b)\cdot f(b,c)}{\lambda_{B\to C}(a,c)}\right]^{\frac{1}{w_1}}$$

⋮

$$\tilde{\mathbf{a}} \sim q(A) = f(a)\cdot \lambda_{E\to A}(a)/U$$



Key insight: provides bounded importance weights!

U = upper bound

$$0 \le f(x)/q_{\text{wmb}}(x) \le U \quad \forall x$$

Probabilistic Lower Bounds for MMAP

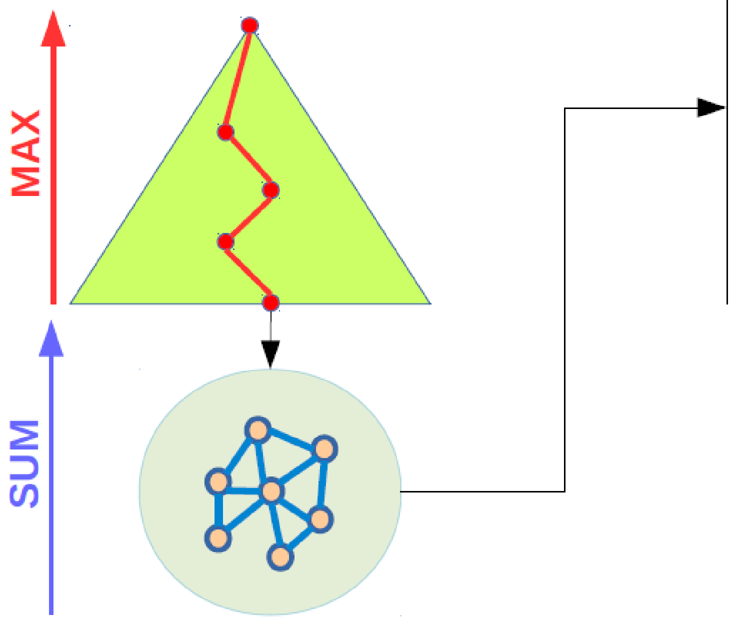
ANYSBFS: Anytime Stochastic Best-First Search

ANYLDFS: Anytime Learning Depth-First Search

[Marinsecu, Dechter, Ihler Ijcai 2018]

WMB-IS [Liu et. Al. 2015]

$$\max_{\mathbf{X}_M} \sum_{\mathbf{X}_S} P(\mathbf{X})$$



$$Z = \sum_{\mathbf{X}_S} P(X)|_{\bar{x}_M}$$

Solving the conditioned SUM subproblem is hard!

$$\#P - complete$$

Compute a (probabilistic) lower bound on the conditioned sum subproblem

$$Pr(\hat{Z} - \Delta(n, \delta) \leq Z) \geq (1 - \delta)$$

WMB based importance sampling scheme:

- n - number of samples
- δ - confidence value
- Z_{wmb} - result of WMB
- $\hat{Z}$ - Importance Sampling estimate

$$\Delta(n, \delta) = \sqrt{\frac{2\hat{var}(w(x)) \log(2/\delta)}{n}} + \frac{7Z_{wmb} \log(2/\delta)}{3(n-1)}$$

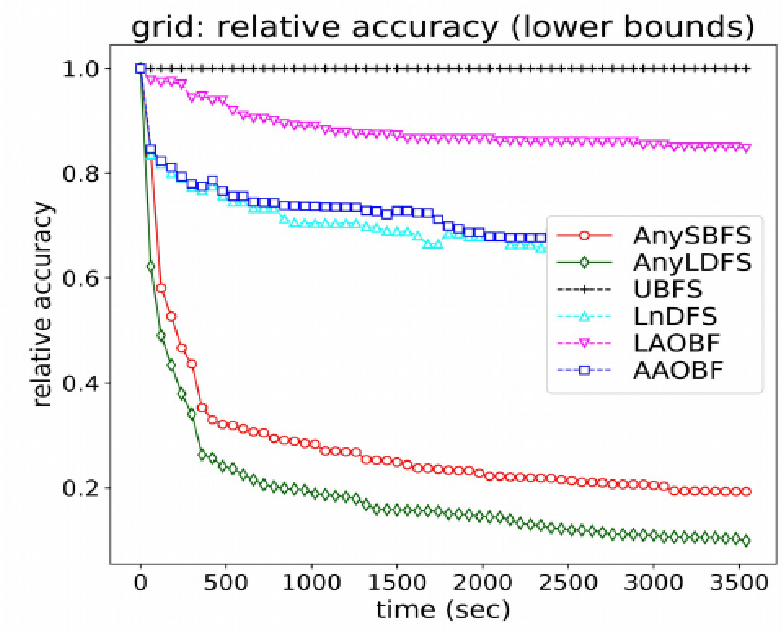
Stochastic Anytime Search for MMAP (Grids)

[Marinescu, Ihler, Dechter IJCAI-2018, Lou, Dechter, Ihler AAAI-2018]

ANYSBFS: Anytime Stochastic Best-First Search

ANYLDFS: Anytime Learning Depth-First Search

$$acc_{lb} = \frac{|l_t - l^*|}{l^*}$$

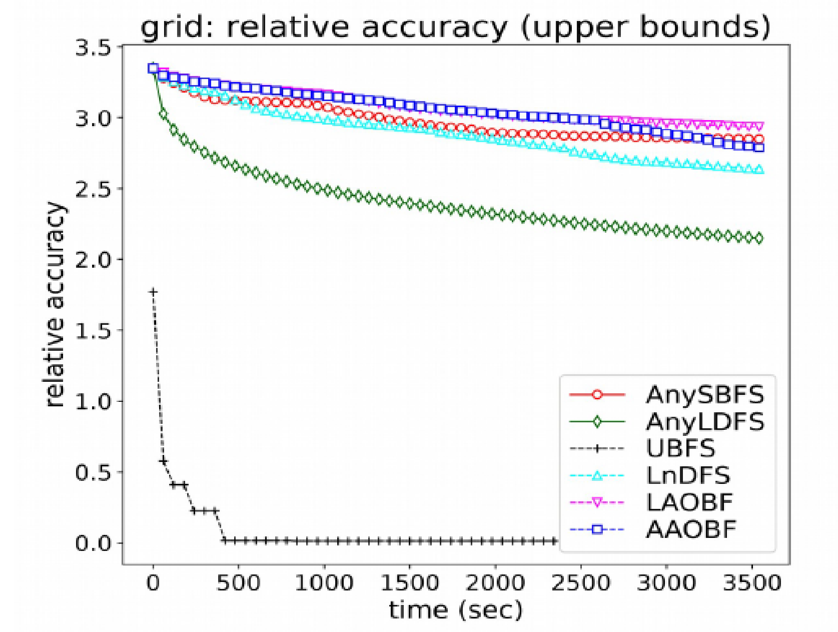


l_t – lower bound at time t

l^* – tightest lower bound found

Average over 150 instances

$$acc_{ub} = \frac{|u_t - u^*|}{u^*}$$



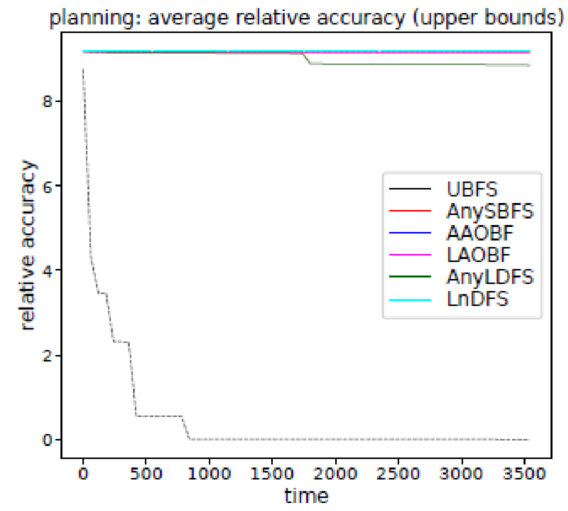
u_t – upper bound at time t

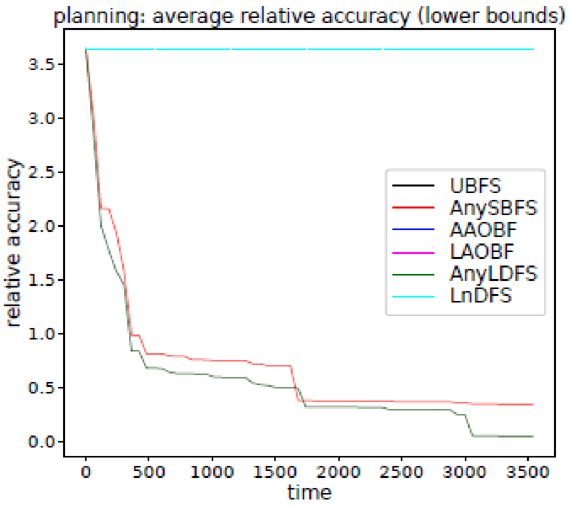
u^* – tightest upper bound found

Average over 150 instances

(Lower plots are better)

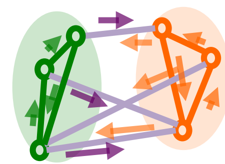
Stochastic Anytime Search for MMAP (Planning)





Summary

Inference, Variational methods



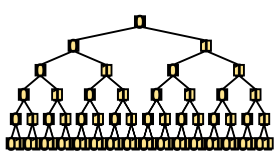
Bucket-elimination
weighted mini-bucket (WMB)
[Dechter 1999, Dechter and Rish, 2003
Liu and Ihler, ICML 2011]

provide heuristic

provide proposal

WMB-IS
[Liu et al., NIPS 2015]

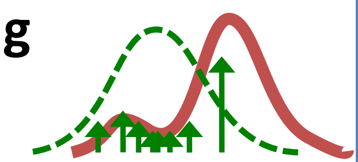
Search



refine proposal

help search

Sampling



AND/OR search (AODFS)
[Marinescu et al 2009, Lou et al., AAAI 2017]
Marinescu et al., IJCAI 2018]

dynamic importance sampling (DIS)
[Lou et al., NIPS 2017]

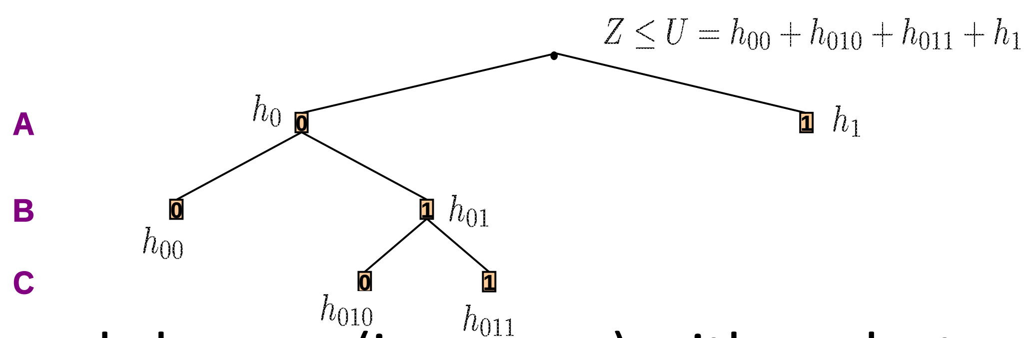
Dynamic Importance Sampling

(For partition function)

[Lou, Dechter, Ihler, NIPS 2017, AAAI 2019]

- Interleave
  - Building search tree (expand Nd nodes)
  - Draw samples given search bound (Nl samples)

(For partition function)



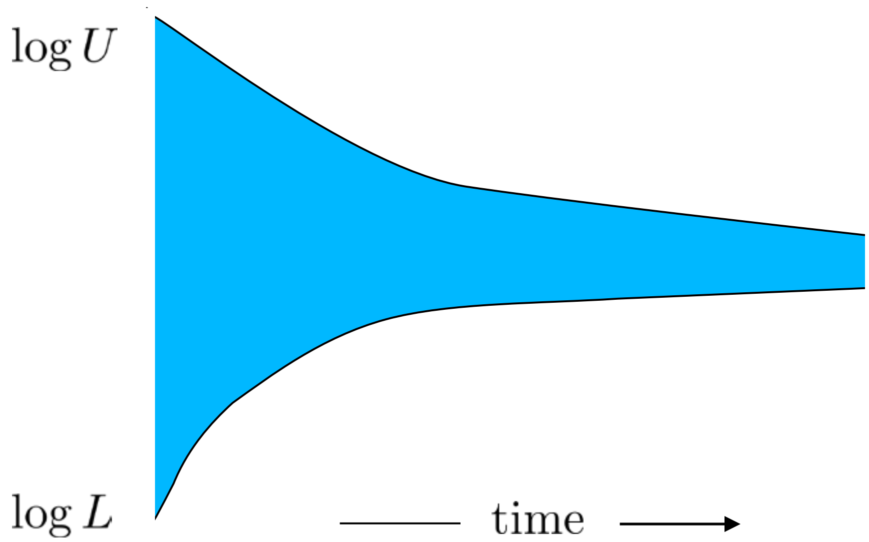
- Key insight: proposal changes (improves) with each step
  - Use weighted average: better samples get more weight

$$\widehat{Z} = \frac{\mathrm{HM}(\boldsymbol{U})}{N} \sum_{i=1}^{N} \frac{\widehat{Z}_i}{U_i}, \quad \mathrm{HM}(\boldsymbol{U}) = \left[\frac{1}{N} \sum_{i=1}^{N} \frac{1}{U_i}\right]^{-1}$$

  - Derive corresponding concentration bound on Z

Aggregated Results

[Lou, Dechter, Ihler, NIPS 2017, AAAI 2019]



area between upper and lower bounds of log$Z$

Table 1: Mean area between upper and lower bounds of $\log Z$, normalized by WMB-IS, for each benchmark. Smaller numbers indicate better anytime bounds. The best for each benchmark is bolded.

	AOBFS	WMB-IS	DIS (N_l=1, N_d=1)	DIS (N_l=1, N_d=10)	two-stage
pedigree	16.638	1	0.711	**0.585**	1.321
protein	1.576	1	0.110	**0.095**	2.511
BN	0.233	1	0.340	**0.162**	0.865

Anytime Bounds in UAI Evaluations

- **2006**  (aolib)
- **2008**  (aolib)
- **2011**  (daoopt)
- **2014**  (daoopt) |  (daoopt)  (merlin)

MPE/MAP | MMAP

Winning first or second place UAI competitions, 2006, 2008, 2011, 2014, 2016

Software

- **daoopt**
  - https://github.com/lotten/daoopt

  (distributed and standalone AOBB solver)
- **merlin**
  - https://developer.ibm.com/open/merlin

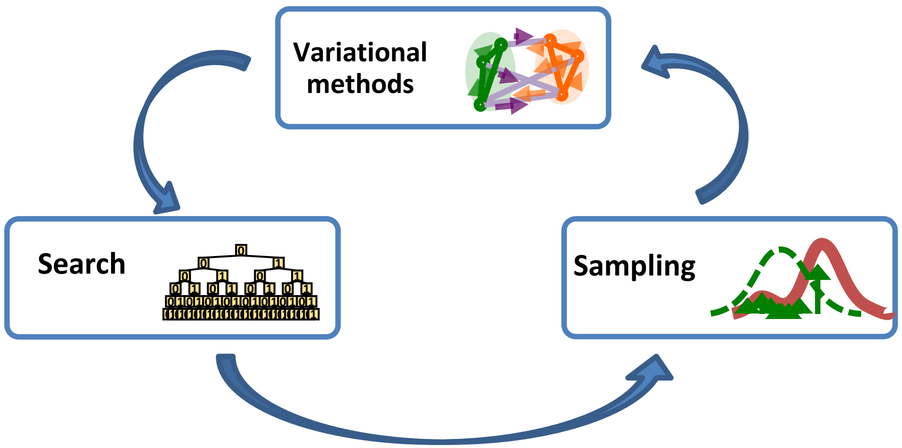
  (standalone WMB, AOBB, AOBF, RBFAOO solvers)

  open source, BSD license

pyGM : Python Toolbox for Graphical Models by Alexander Iher.

Continuing Work

- Combining approaches:
  - Tune the hyper-parameters automatically
  - Extend to decision networks



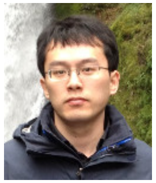
- Languages and Tools:
  - Relational languages
  - Handle constraints specification and continuous functions
  - Temporal domains; Planning, e.g., Influence diagrams, MDPs, POMDPs
  - Cross interaction of deep learning and graphical models → Reinforcement

See our poster on Thursday
(Lee, Marinescu, Ihler, Dechter)

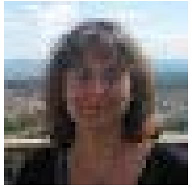
Thank You !

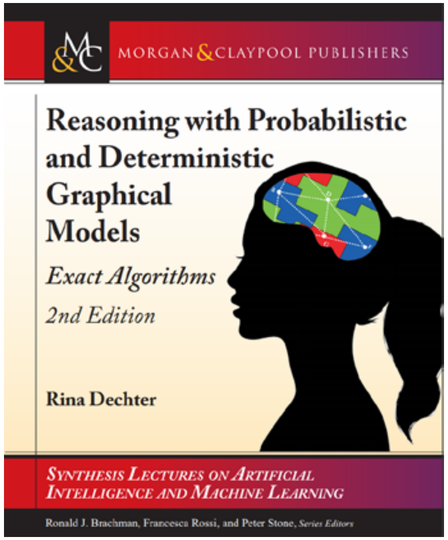
See our poster on Thursday





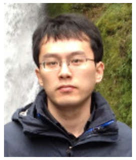






Alex Ihler
Kalev Kask
Irina Rish
Bozhena Bidyuk
Robert Mateescu
Radu Marinescu
Vibhav Gogate
Emma Rollon
Lars Otten
Natalia Flerova
Andrew Gelfand
William Lam
Junkyu Lee
Qi Lou





UNIVERSITY of CALIFORNIA IRVINE