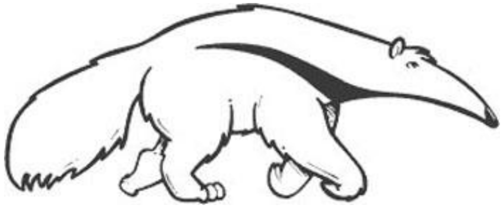
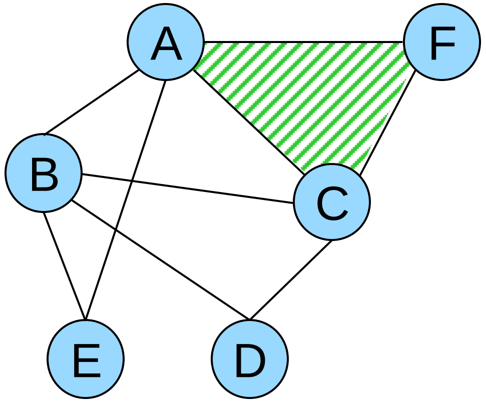
Probabilistic Reasoning Meets Heuristic Search

"We have to equip machines with a model of the environment. If a machine does not have a model of reality, you cannot expect the machine to behave intelligently in that reality". (Pearl 2018, interview for the "book of why")

Rina Dechter

Collaborators:
Radu Marinescu
Alex Ihler
Junkyu Lee

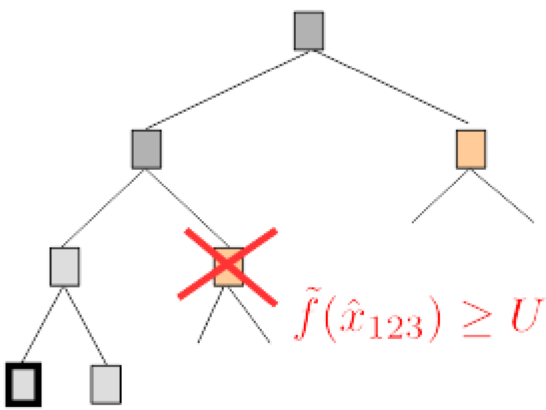




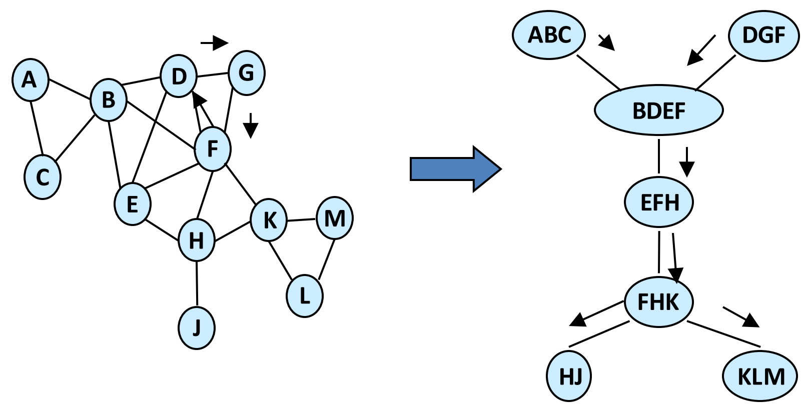
Search Collaborate Inference

- Heuristic Search
- Probabilistic reasoning, graphical models

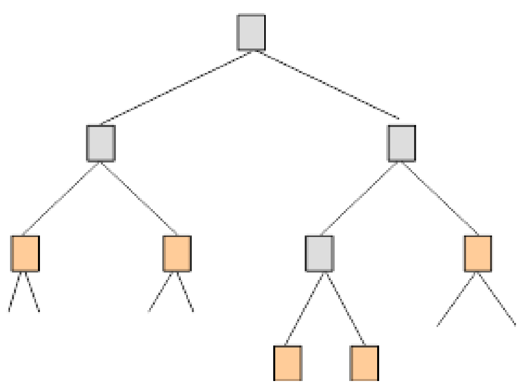
Branch-and-Bound



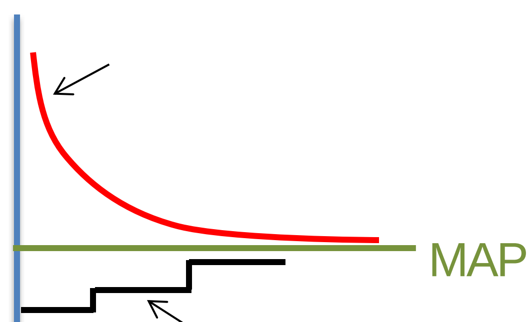
$f(\hat{x}) = U$



A* search



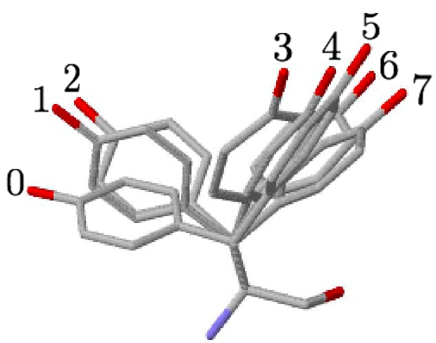
Anytime algorithms.



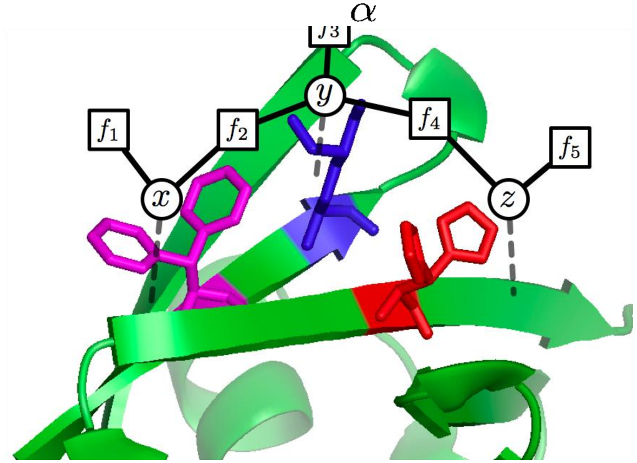
Probabilistic Graphical Models

- Describe structure in large problems
  - Large complex system $F(X)$
  - Made of “smaller”, “local” interactions $f_\alpha(x_\alpha)$
  - Complexity emerges through interdependence
- Examples & Tasks
  - Maximization (MAP): find the most probable configuration

$$\mathbf{x}^* = \arg\max_{\mathbf{x}} \prod_\alpha f_\alpha(\mathbf{x}_\alpha) \qquad f(\mathbf{x}^*) = \max_{\mathbf{x}} \prod_\alpha f_\alpha(\mathbf{x}_\alpha)$$



Phenylalanine



[Yanover & Weiss 2002]

Probabilistic Graphical Models

- Describe structure in large problems
  - Large complex system $F(X)$
  - Made of "smaller", "local" interactions $f_\alpha(x_\alpha)$
  - Complexity emerges through interdependence
- Examples & Tasks
  - Summation & marginalization

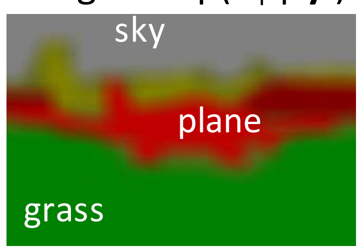
"partition function"

$$p(x_i) = \frac{1}{Z} \sum_{\mathbf{x} \setminus x_i} \prod_\alpha f_\alpha(\mathbf{x}_\alpha) \quad \text{and} \quad Z = \sum_{\mathbf{x}} \prod_\alpha f_\alpha(\mathbf{x}_\alpha)$$

Observation **y**

$\Rightarrow$

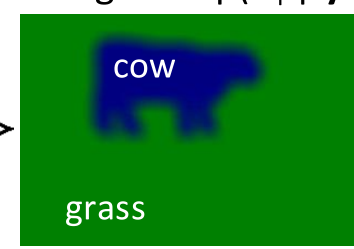
Marginals p(x_i | **y**)



Observation **y**

$\Rightarrow$

Marginals p(x_i | **y**)



e.g., [Plath et al. 2009]

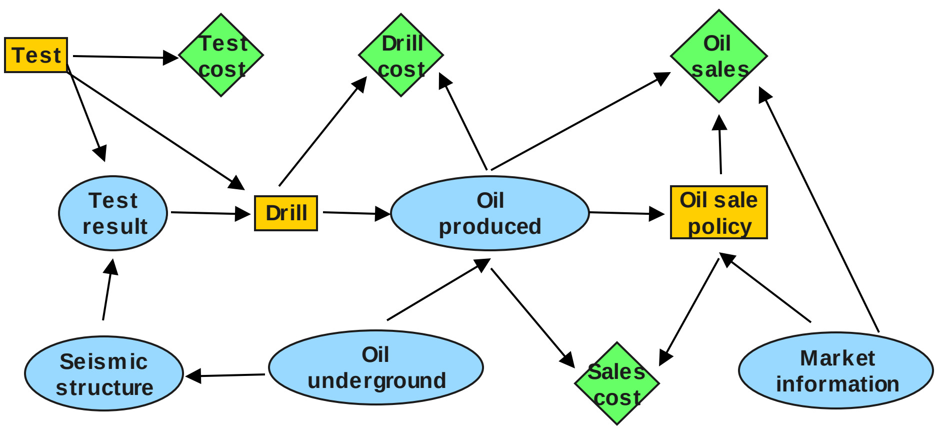
Graphical Models

- Describe structure in large problems
  - Large complex system $F(X)$
  - Made of "smaller", "local" interactions $f_\alpha(x_\alpha)$
  - Complexity emerges through interdependence
- Examples & Tasks
  - Mixed inference (marginal MAP, MEU, …)

$$f(\mathbf{x}_M^*) = \max_{\mathbf{x}_M} \sum_{\mathbf{x}_S} \prod_{\alpha} f_\alpha(\mathbf{x}_\alpha)$$

Influence diagrams & optimal decision-making

(the "oil wildcatter" problem)



e.g., [Raiffa 1968; Shachter 1986]

Sample Applications for Graphical Models

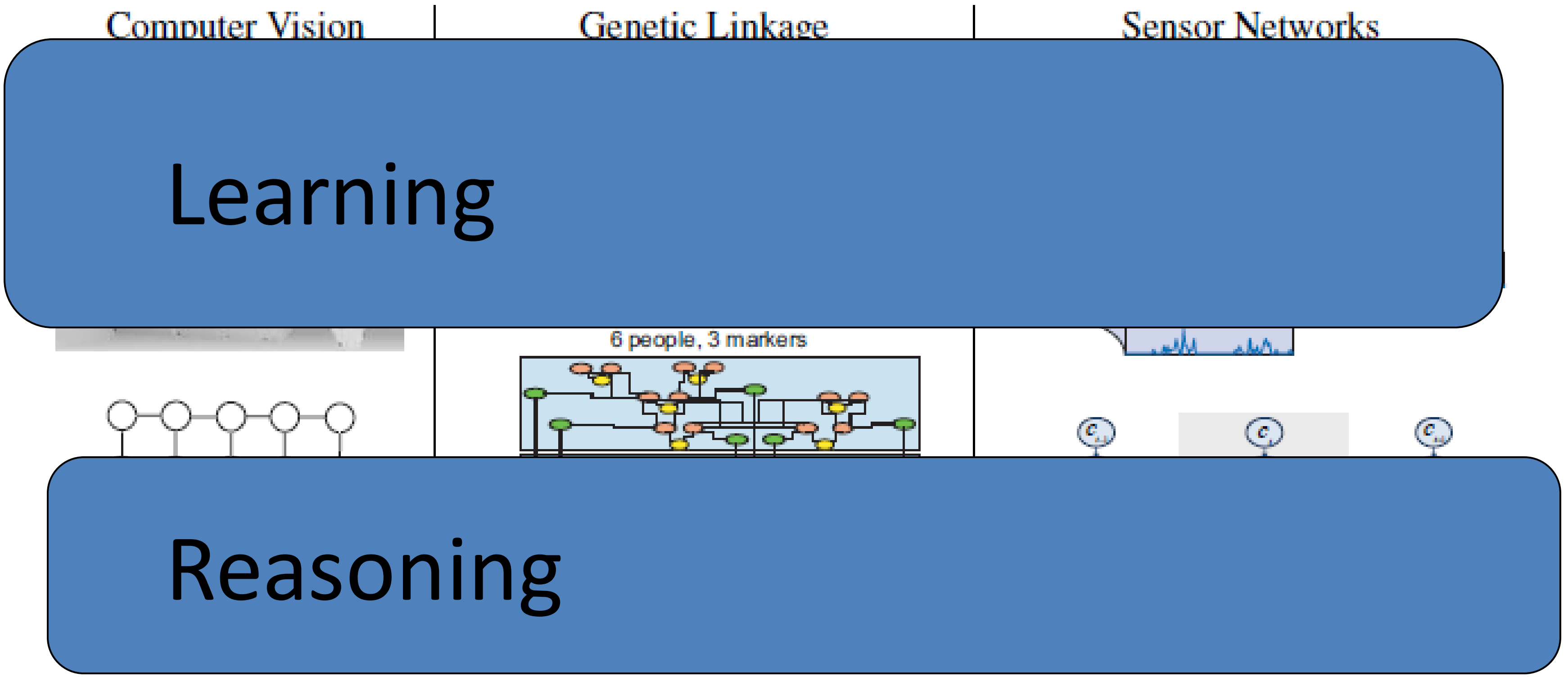
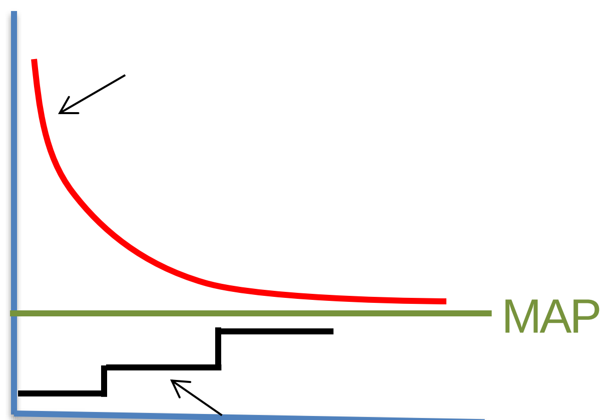


Figure 1: Application areas and graphical models used to represent their respective systems: (a) Finding correspondences between images, including depth estimation from stereo; (b) Genetic linkage analysis and pedigree data; (c) Understanding patterns of behavior in sensor measurements using spatio-temporal models.

Outline

- Graphical models, Queries, Inference vs search
- AND/OR search spaces
- Bounded Inference: a) mini-bucket, b) cost-shifting
- Generating heuristics using mini-bucket elimination
- AND/OR Heuristic Search for Map and Marginal Map
- Conclusion

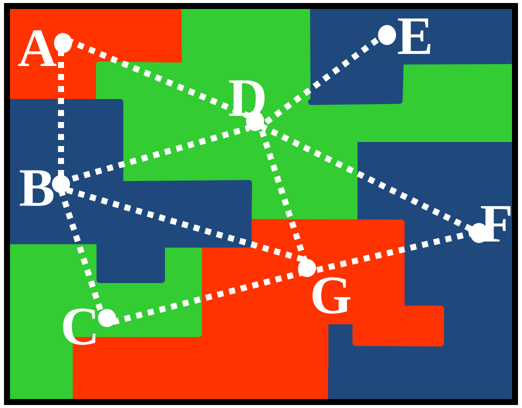


Constraint Satisfaction/Satisfiability

Constraint Networks

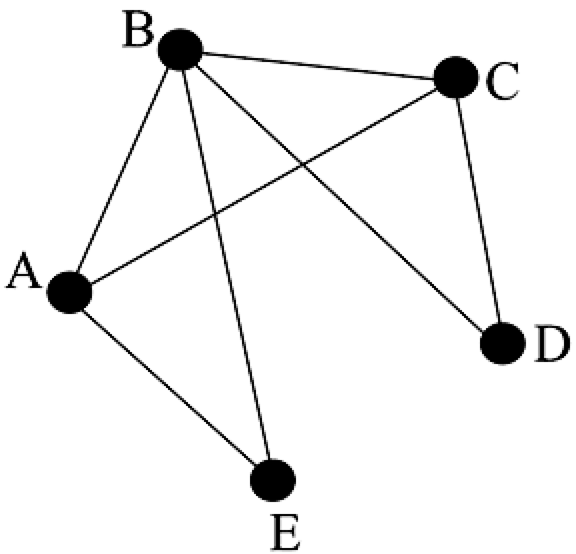
- Variables - countries (A,B,C,etc.)
- Values - colors (red, green, blue)
- Constraints:

$\mathbf{A} \neq \mathbf{B}$, $\mathbf{A} \neq \mathbf{D}$, $\mathbf{D} \neq \mathbf{E}$, *etc.*



Propositional Satisfiability

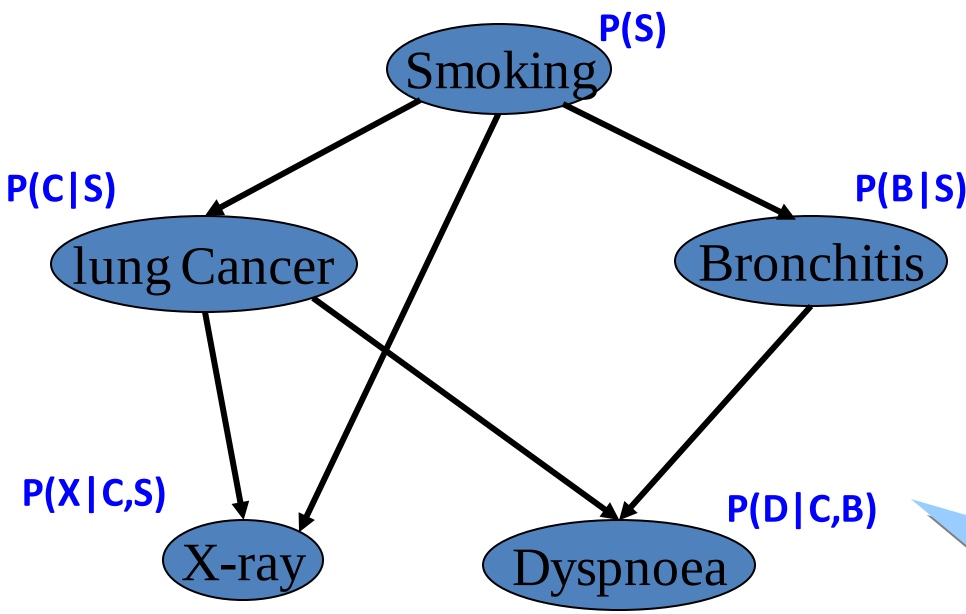
- $\varphi = \{(\neg C), (A \vee B \vee C), (\neg A \vee B \vee E), (\neg B \vee C \vee D)\}$.



Semantics: set of all solutions

Primary task: find a solution

Bayesian Networks (Pearl 1988)



$$BN = (G, \Theta)$$

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

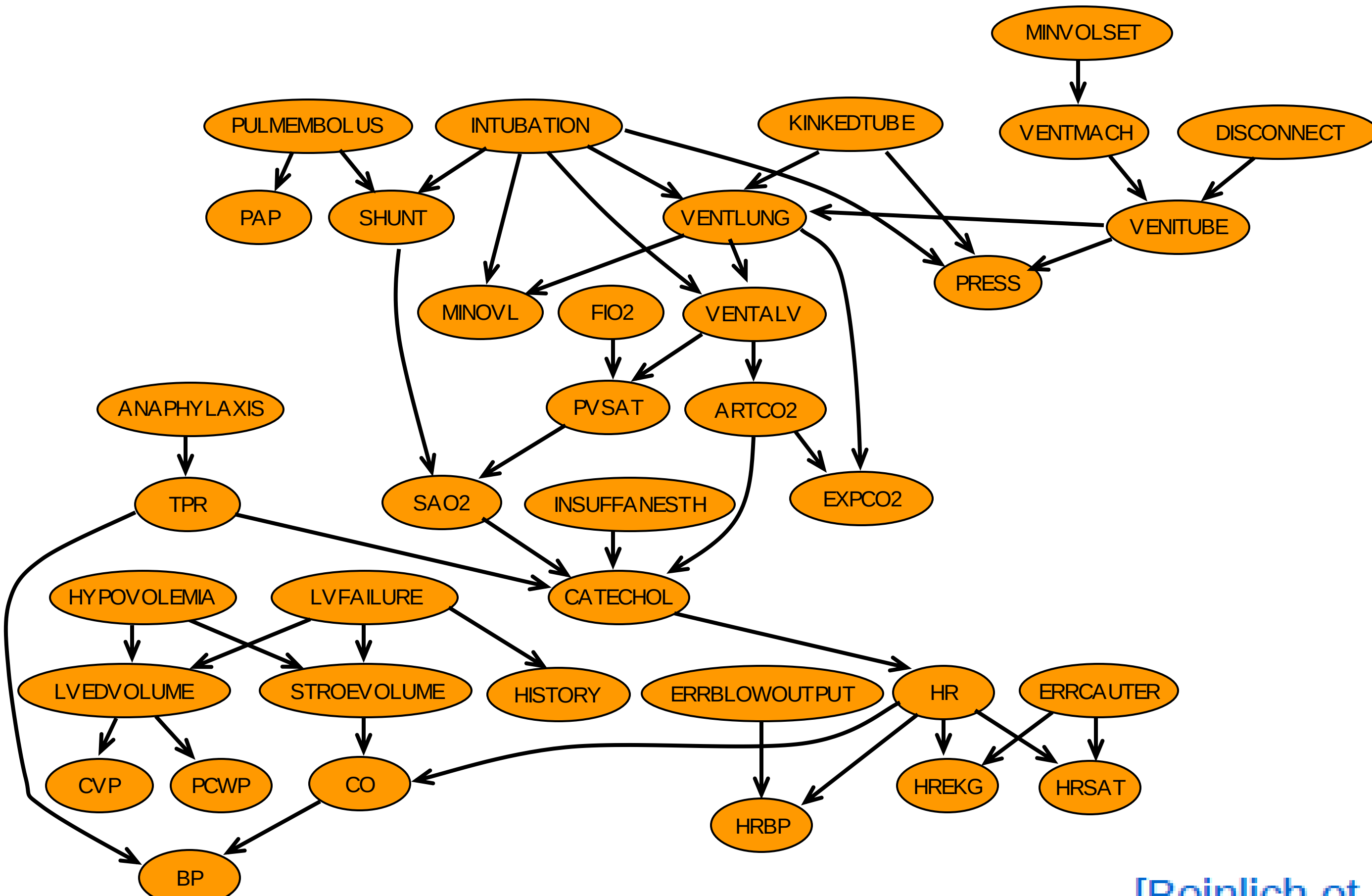
Combination: Product
Marginalization: sum/max

- Posterior marginals, probability of evidence, MPE
- 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(P)= $max_{S,L,B,X}$ P(S)· P(C|S)· P(B|S)· P(X|C,S)· P(D|C,B)

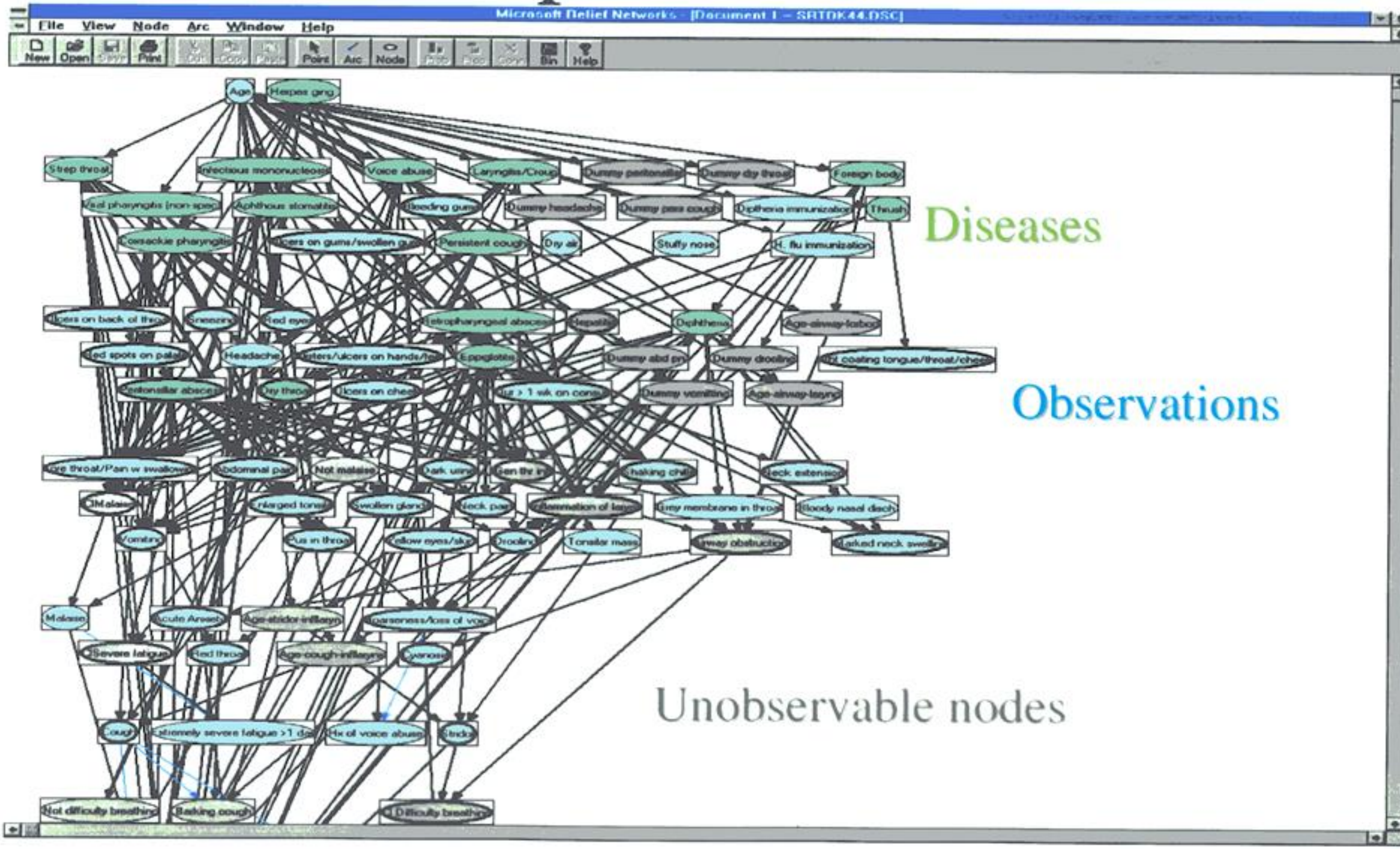
Monitoring Intensive-Care Patients

The "alarm" network - 37 variables, 509 parameters (instead of 2^{37})



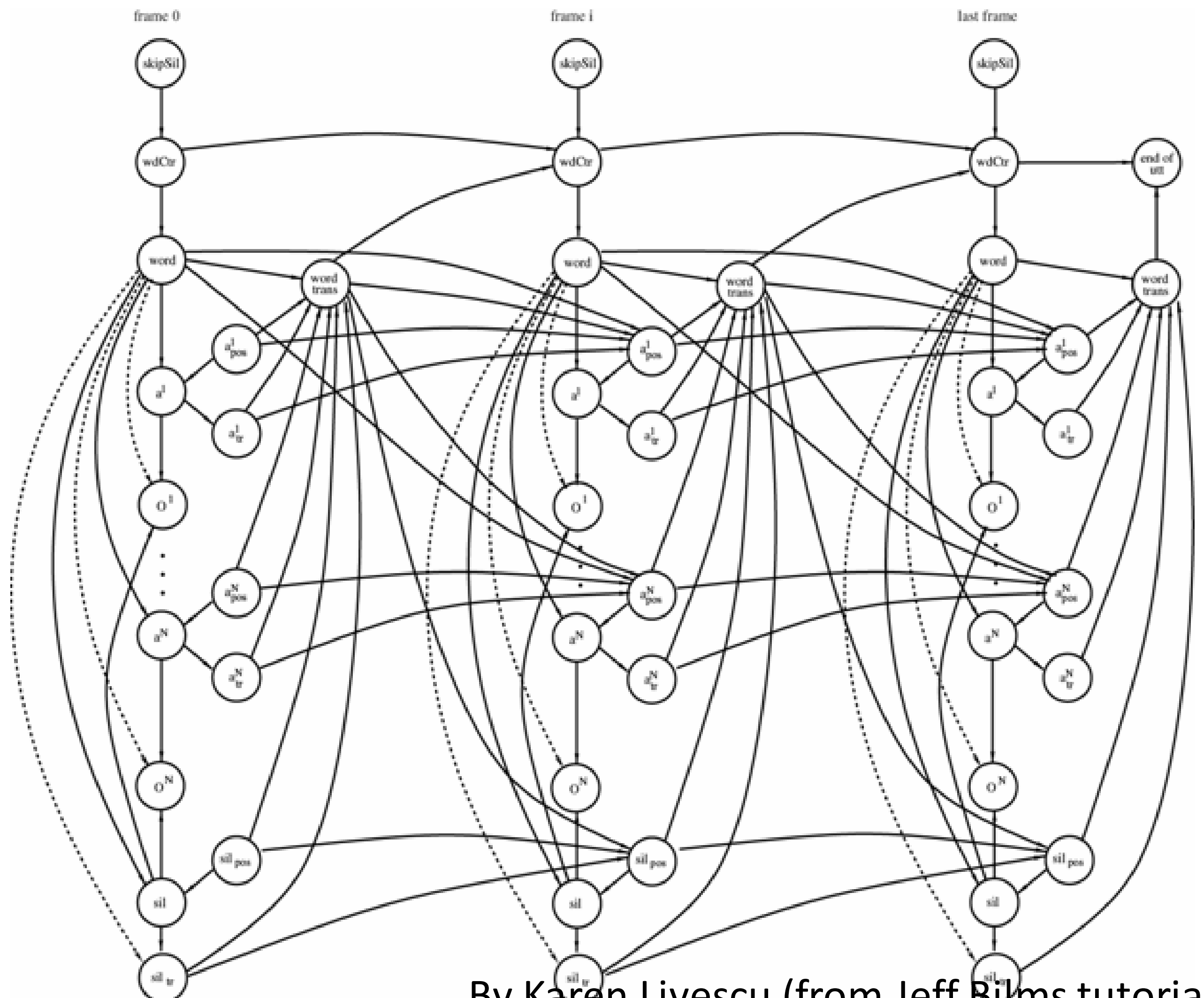
[Beinlich et al., 1989]

Chief Complaint: Sore Throat



KI

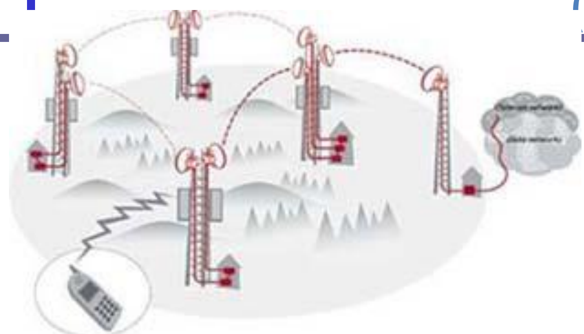
Phone-free Articulatory Graph



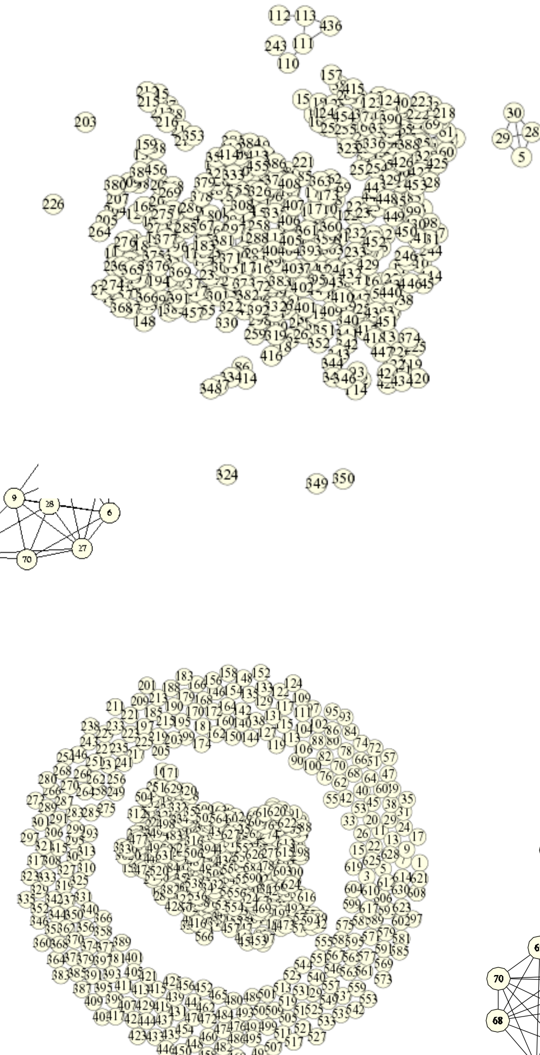
By Karen Livescu (from Jeff Bilms tutorial

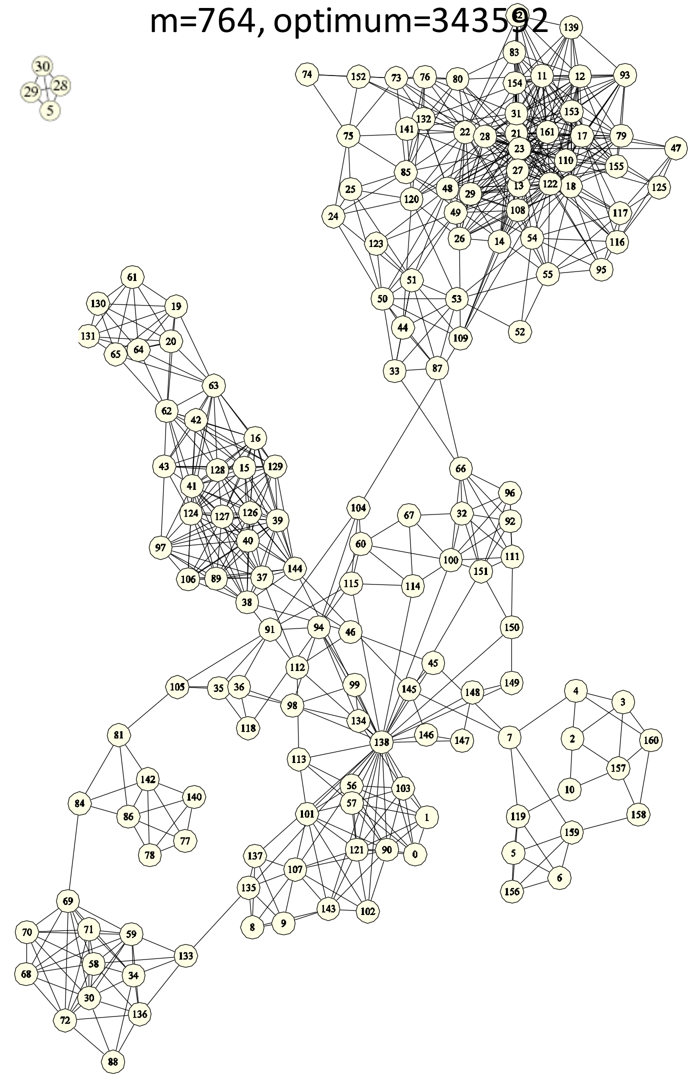
Radio Link Frequency Assignment Problem

(Cabon et al., *Constraints* 1999) (Koster *et al.*, *4OR* 2003)

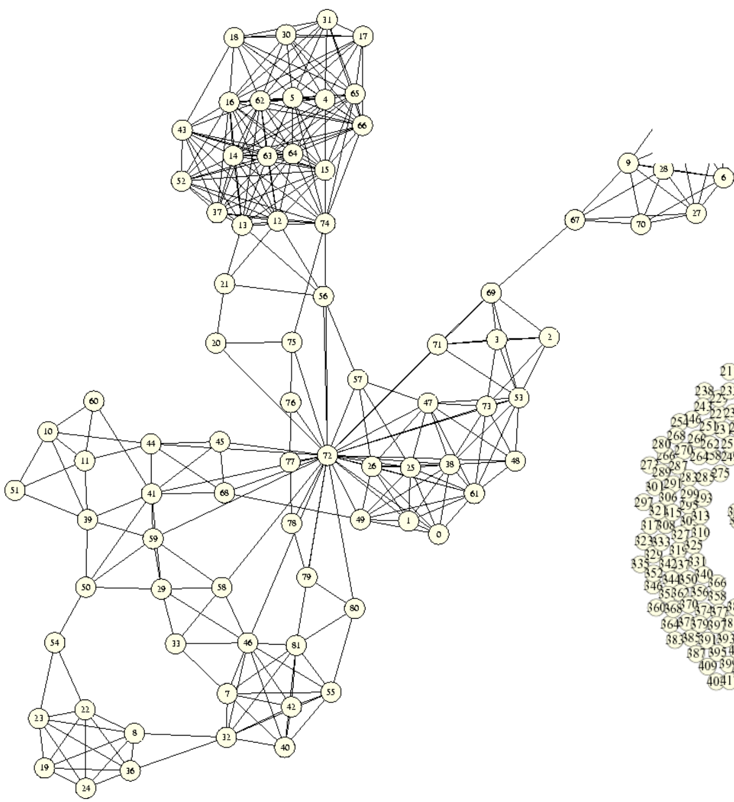


- CELAR SCEN-07r
  n=162, d=44,
  m=764, optimum=343592





CELAR SCEN-06
n=100, d=44,
m=350, optimum=3389



Graphical models

A ***graphical model*** consists of:

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

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

$F = \{f_{\alpha_1}, \ldots, f_{\alpha_m}\}$ -- functions

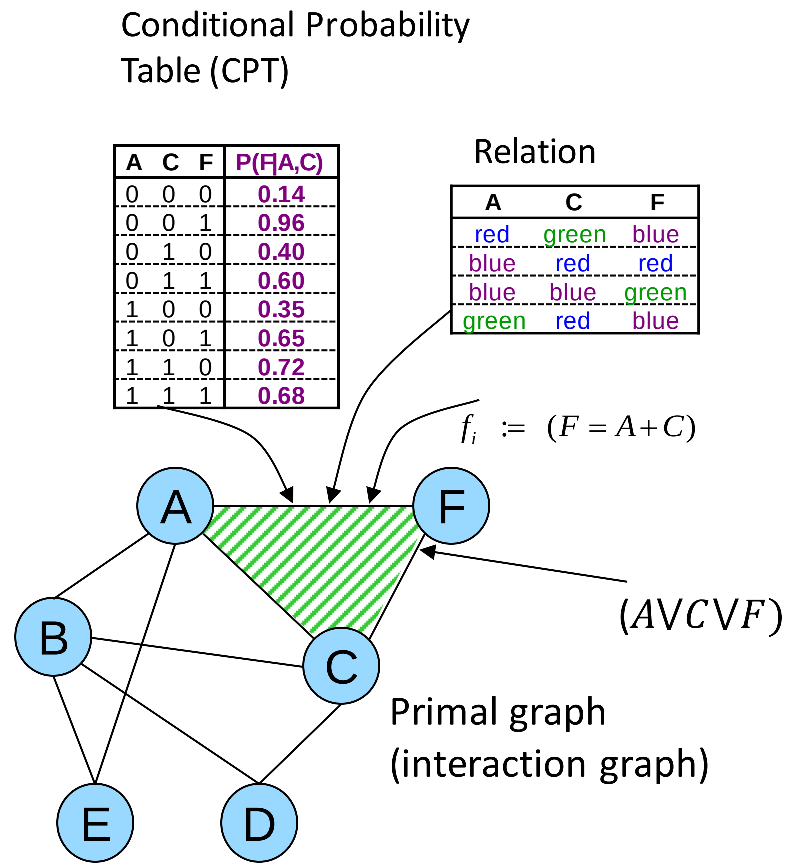
Operators:

combination operator

(sum, product, join, ...)

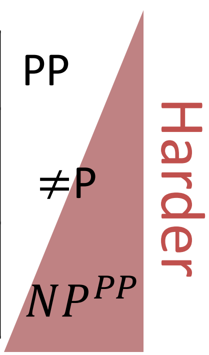
elimination operator

(projection, sum, max, min, ...)



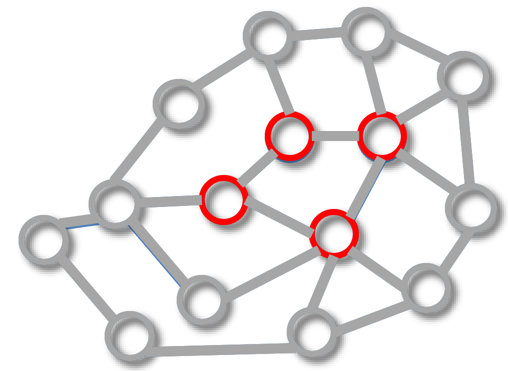
Types of queries:

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



- All these tasks are NP-hard
  - exploit problem structure
  - identify special cases
  - approximate

Why Marginal MAP?

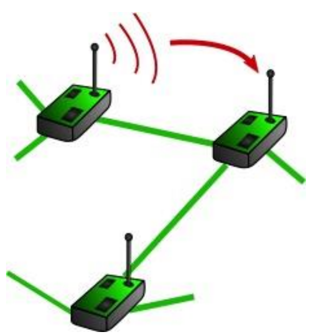


- Often, Marginal MAP is the "right" task:
  - We have a model describing a large system
  - We care about predicting the state of some part

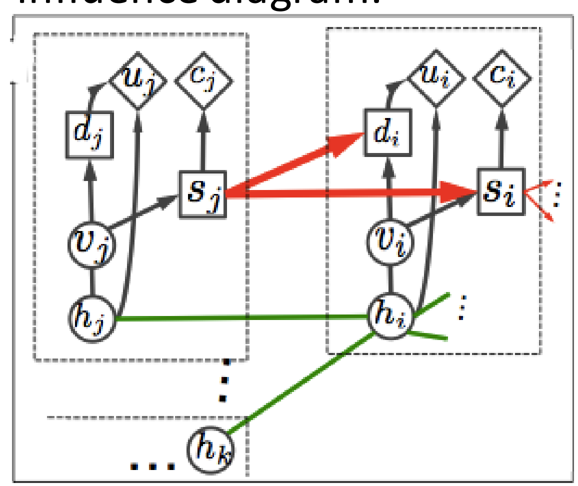
- Example: decision making
  - Sum over random variables (random effects, etc.)
  - Max over decision variables (specify action policies)

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

Sensor network

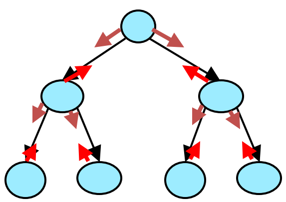


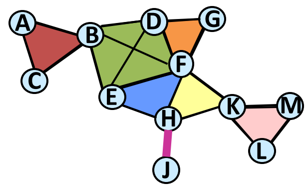
Influence diagram:

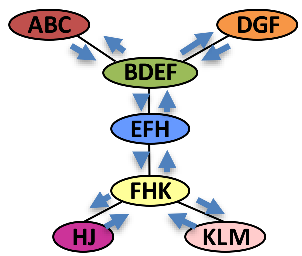


Search Collaborates with Inference

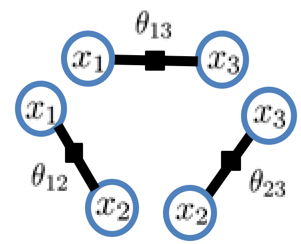
- Inference: message-passing on cluster-tree





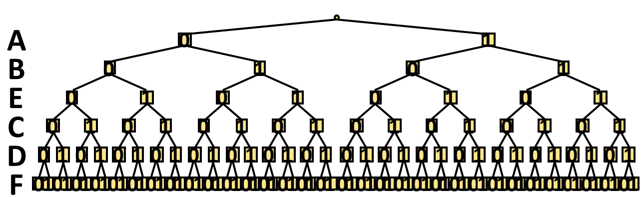


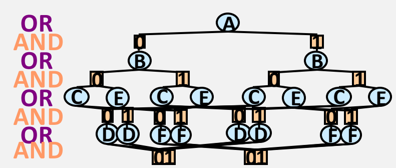
decomposition bounds



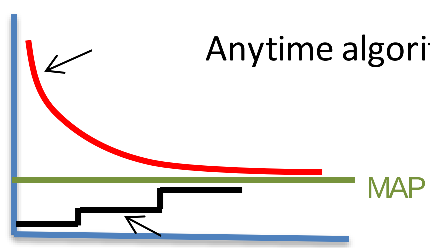
Heuristics
To guide AND/OR
search

- Search → exploiting structure



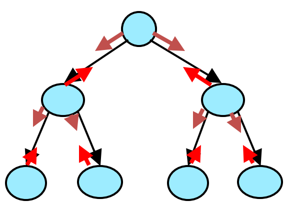


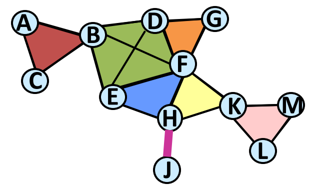
Context minimal AND/OR search graph

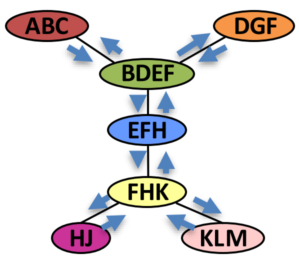


Anytime algorithms.

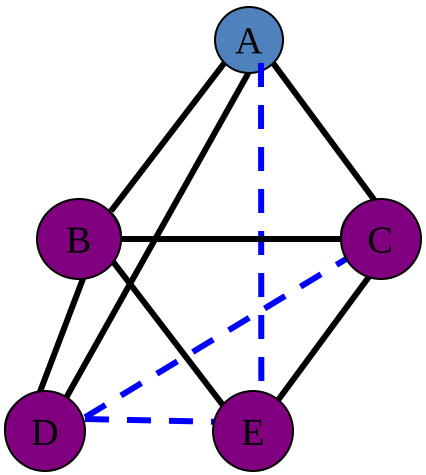
More on Inference: Bucket Elimination







Query 1: Belief updating: P(X|evidence)=?



"primal" graph

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

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

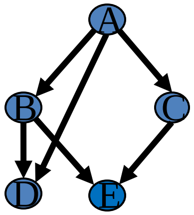
$$P(a)\sum_{e=0}\sum_{d}\sum_{c} P(c|a \sum_{b} P(b|a)P(d|b,a)P(e|b,c)$$

Variable Elimination

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

Query 1: Marginals by Bucket elimination

Algorithm *BE-bel* (Dechter 1996)



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

$\sum_b \prod$ ← Elimination operator

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

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

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

bucket E: $e=0$ $\lambda_{D\to A}(a,e)$

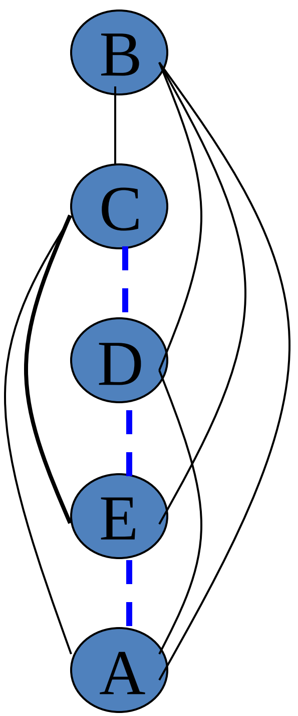
bucket A: $P(a)$ $\lambda_{E\to A}(a)$

$P(e=0)$

$P(a|e=0)$

W*=4

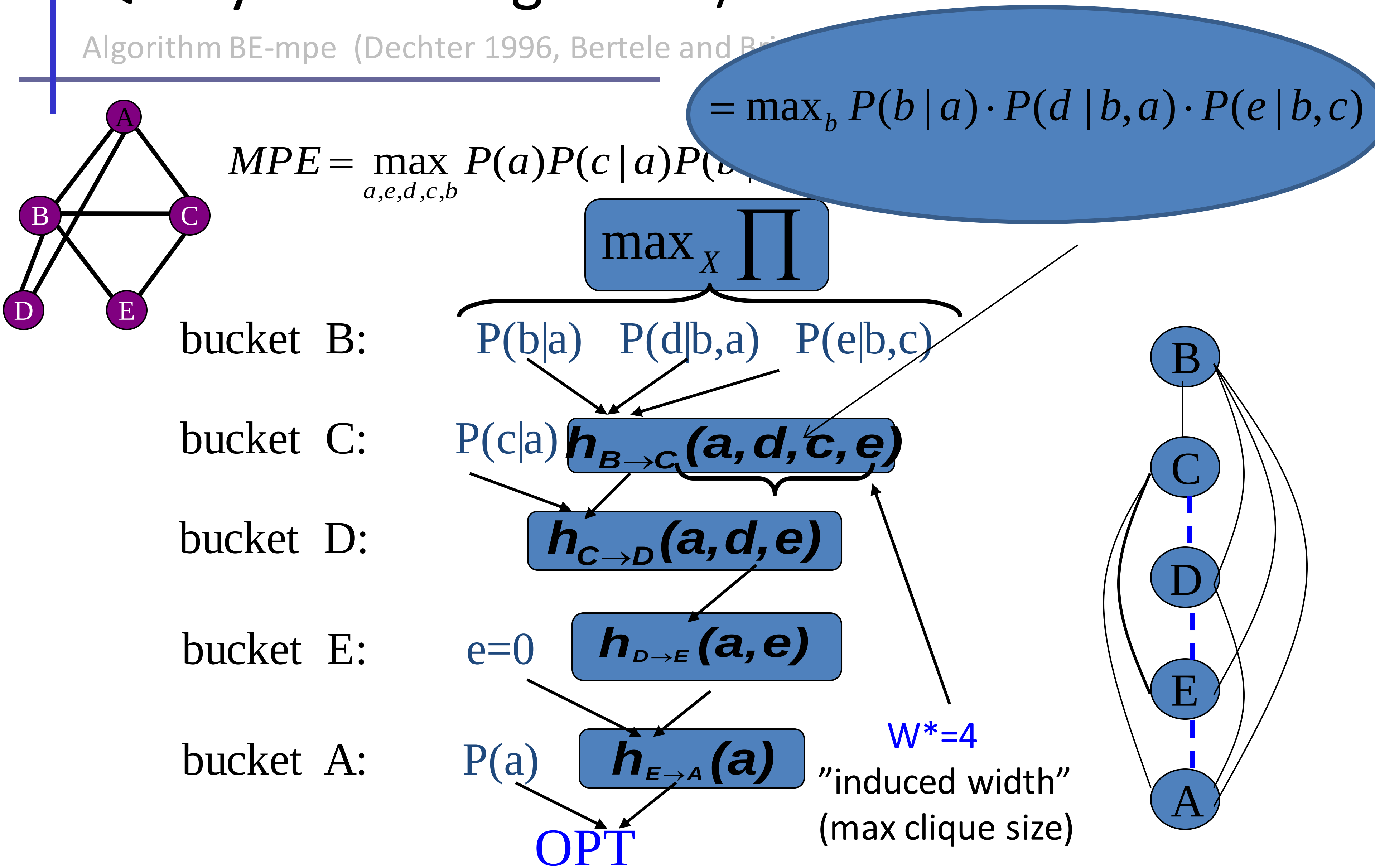
"induced width" (max clique size)



Complexity time and space O(nk^{w*+1})

Query 2: Finding MAP by Bucket Elimination

Algorithm BE-mpe (Dechter 1996, Bertele and Bri



Query 2: Decoding the MAP-Tuple

5. $b' = \arg\max_b P(b \mid a') \times P(d' \mid b, a') \times P(e' \mid b, c')$

4. $c' = \arg\max_c P(c \mid a') \times h^B(a', d', c, e')$

3. $d' = \arg\max_d h^C(a', d, e')$

2. $e' = 0$

1. $a' = \arg\max_a P(a) \cdot h^E(a)$

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

C: P(c|a) $h^B(a,d,c,e)$

D: $h^C(a,d,e)$

E: e=0 $h^D(a,e)$

A: P(a) $h^E(a)$

Return (a', b', c', d', e')

Complexity of Bucket Elimination;

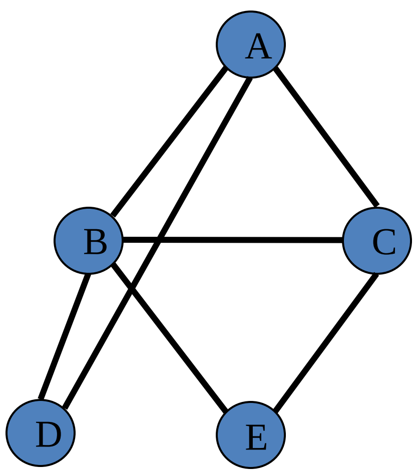
Bucket Elimination is **time** and **space**

$$O\left(r \exp\left(w^*(d) \right)\right)$$

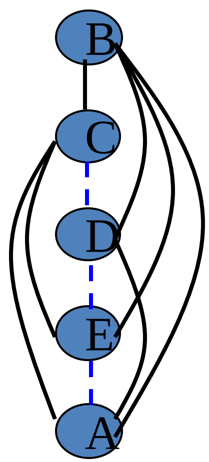
$w^*(d)$ – the induced width of graph along ordering d

r = number of functions

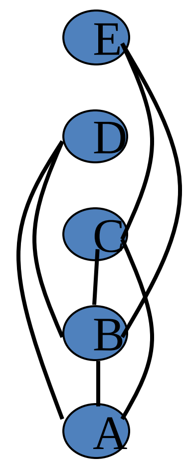
The effect of the ordering:



"Moral" graph



$w^*(d_1) = 4$

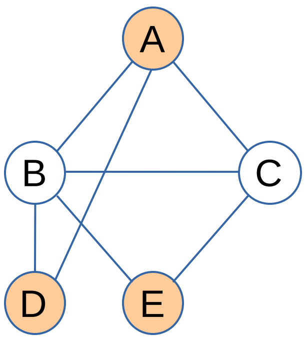


$w^*(d_2) = 2$

Finding the smallest induced width is hard!

Query 3: Finding the Marginal Map (MMAP)

Bucket Elimination



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

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

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

[Dechter, 1999]

constrained elimination order

SUM

B: $f(A,B)\, f(B,C)\, f(B,D) f(B,E)$

Σ_B

C: $\lambda^B(A,C,D,E) f(A,C)\, f(C,E)$

Σ_C

MAX

D: $\lambda^C(A,D,E)\, f(A,D)$

max_D

E: $\lambda^D(A,E)$

max_E

A: $\lambda^E(A)$

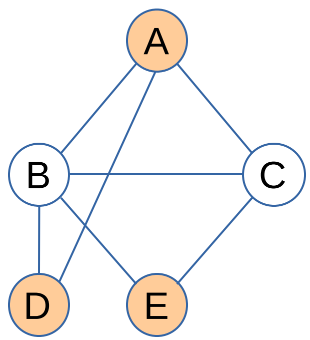
MAP* is the marginal MAP value

Bucket Elimination for MMAP

Bucket Elimination

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

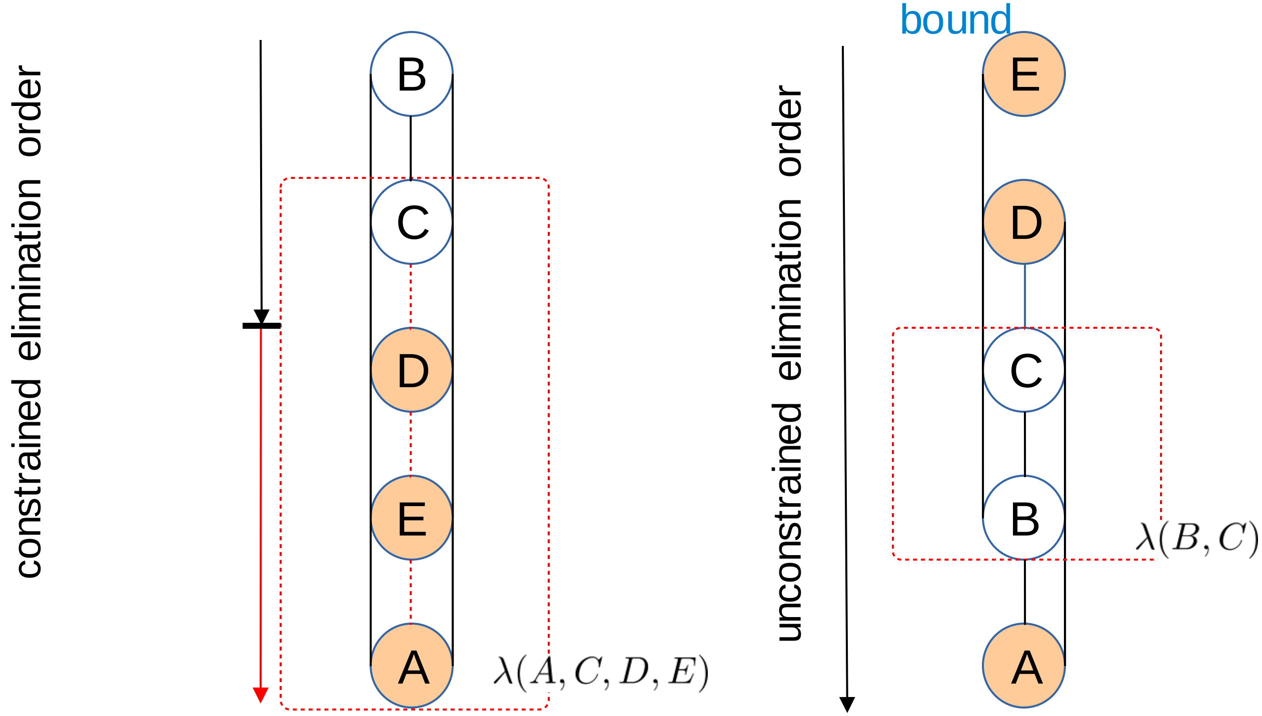
exact   upper bound



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

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

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

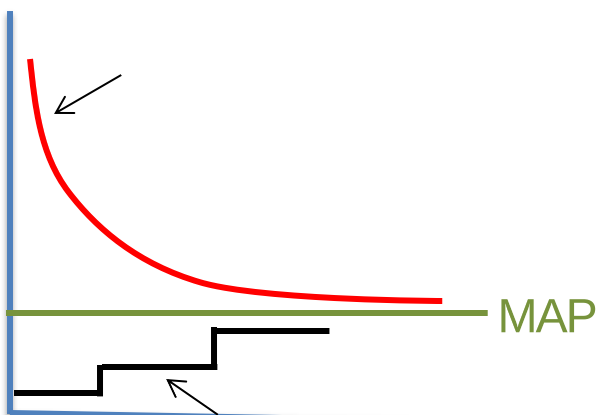


$w^* = 4$ $w^* = 2$

Complexity exponential in the constrained induced-width

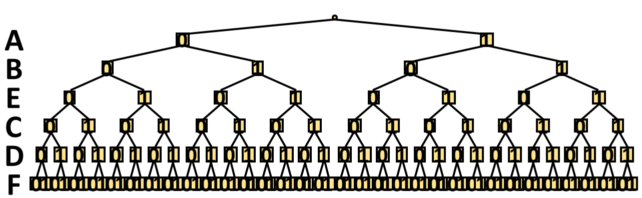
Outline

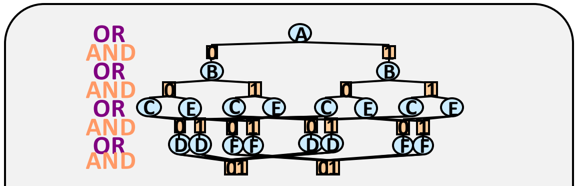
- Graphical models, Queries, Inference vs search
- AND/OR search spaces
- Bounded Inference: a) mini-bucket, b) cost-shifting
- Generating heuristics using mini-bucket elimination
- AND/OR Heuristic Search for Map and Marginal Map
- Conclusion



Dechter, Flairs-2018

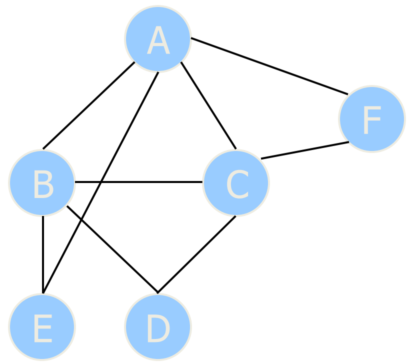
Search Collaborates with Inference



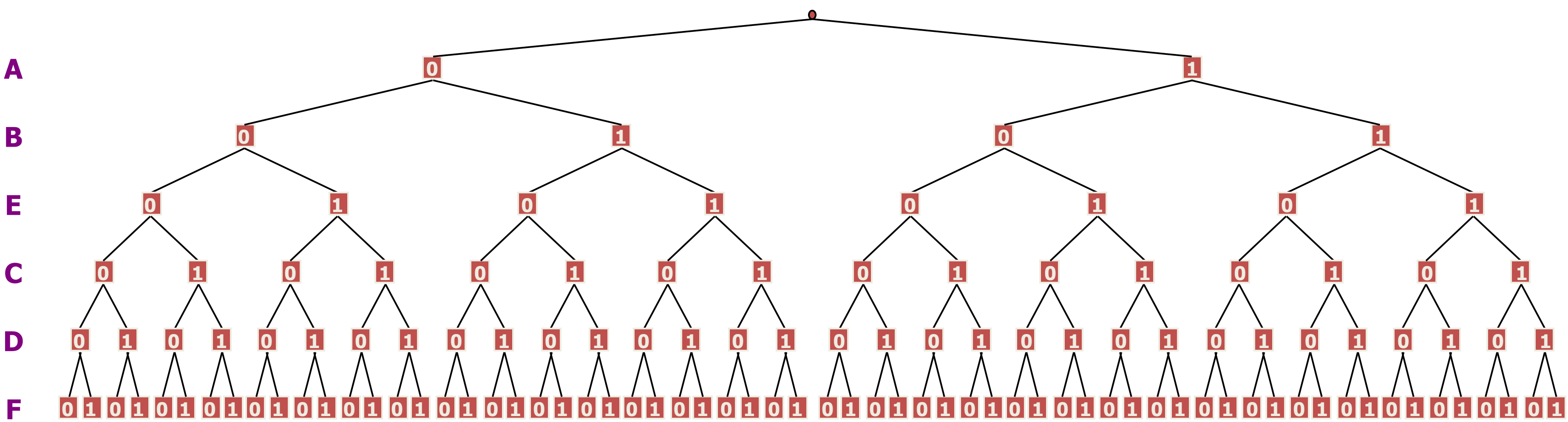


Context minimal AND/OR search graph

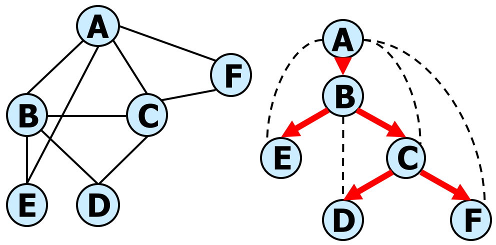
Classic OR Search Space

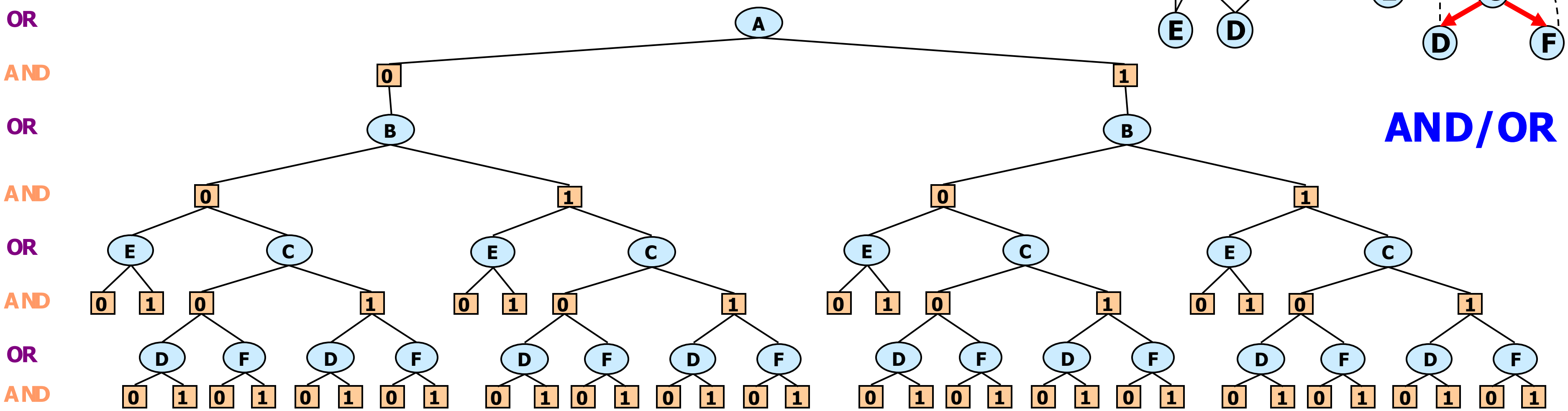


Ordering: A B E C D F

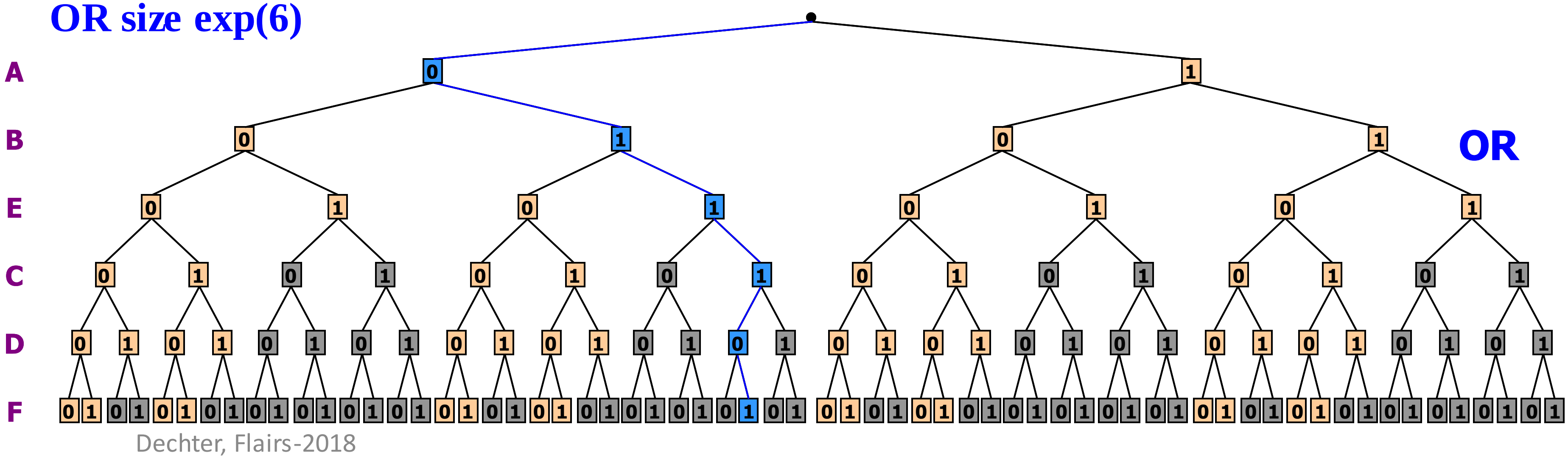


AND/OR vs. OR

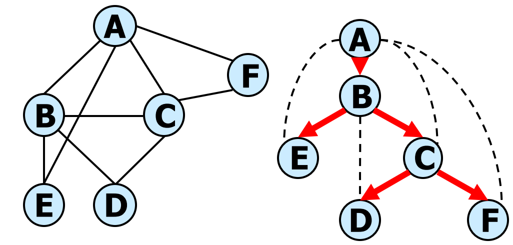


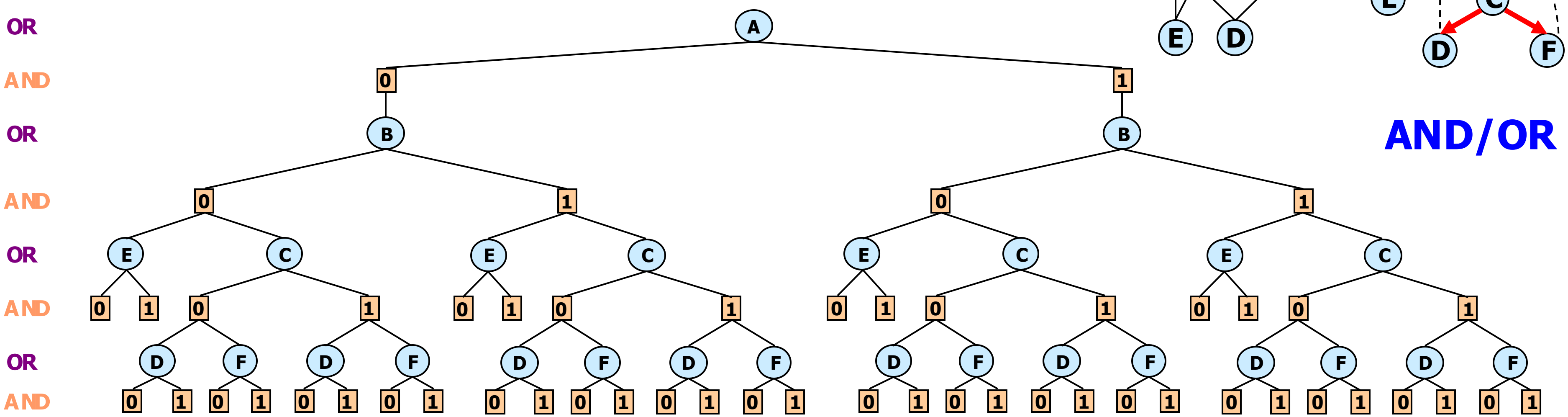


AND/OR size: exp(4),
OR size exp(6)

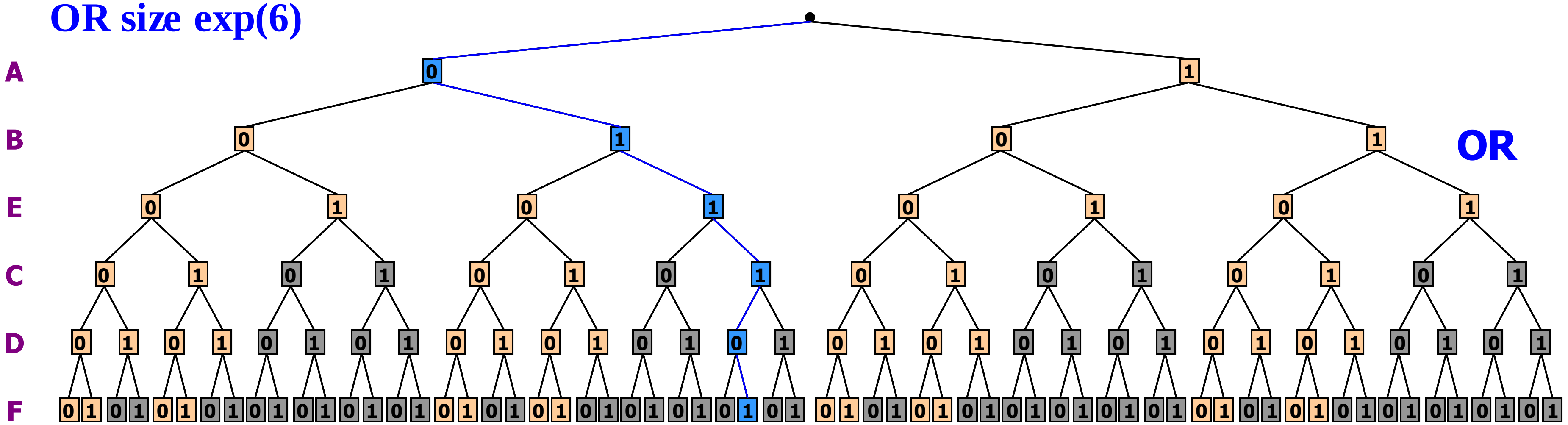


AND/OR vs. OR

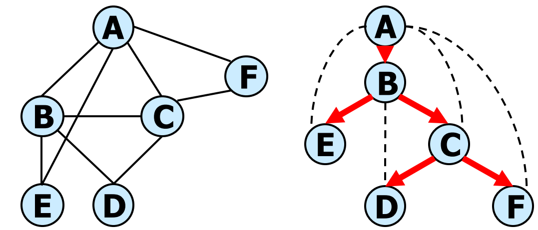


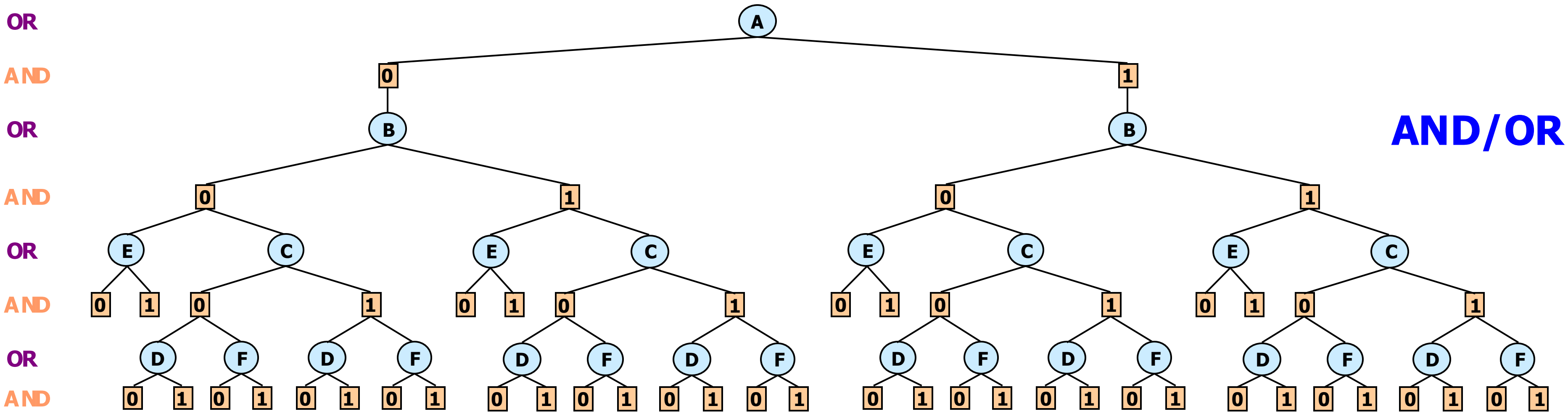


AND/OR size: exp(4),
OR size exp(6)



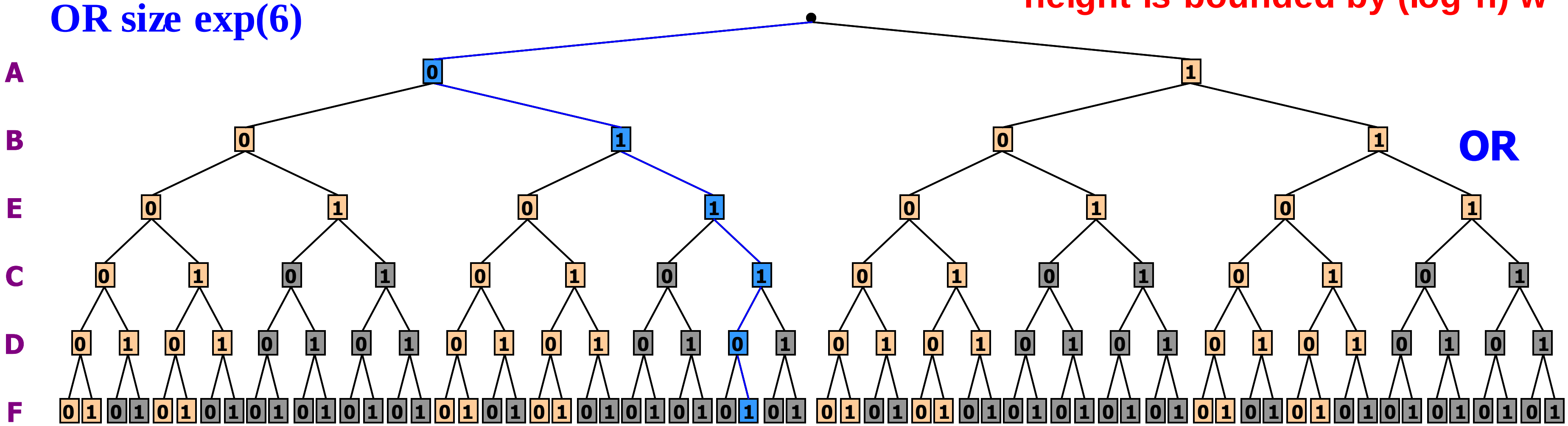
AND/OR vs. OR





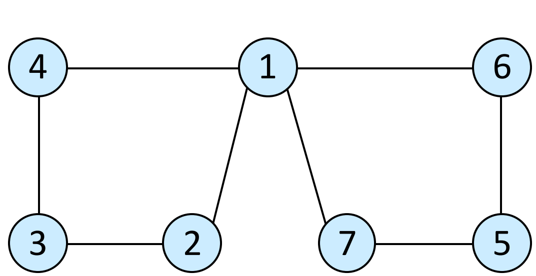
AND/OR size: exp(4),
OR size exp(6)

Time $O(nk^h)$, **Space O(n)**
height is bounded by (log n) w*



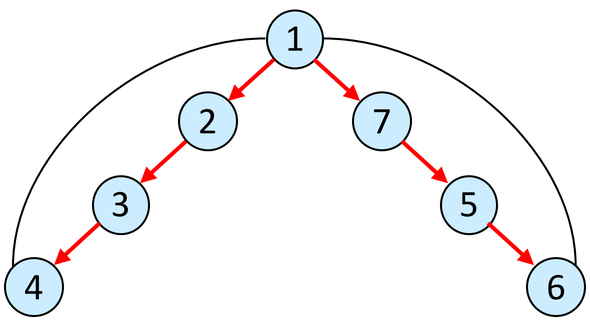
Pseudo-Trees

(Freuder 85, Bayardo 95, Bodlaender and Gilbert, 91)

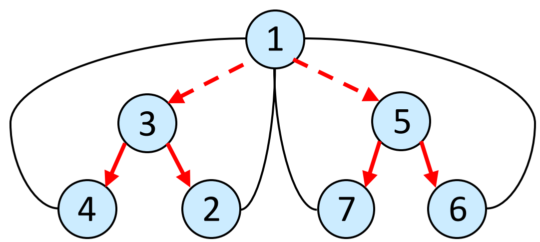


(a) Graph

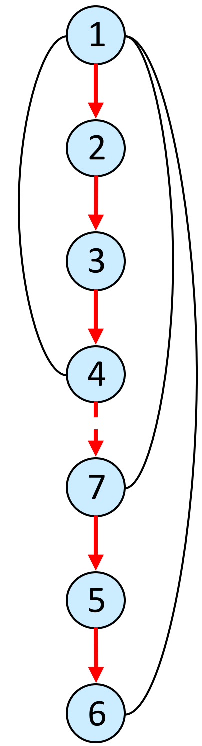
h <= w* log n



(b) DFS tree
height=3



(c) pseudo- tree
height=2



(d) Chain
height=6

Cost of a Solution Tree

$P(E \mid A, B)$

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

$P(B \mid A)$

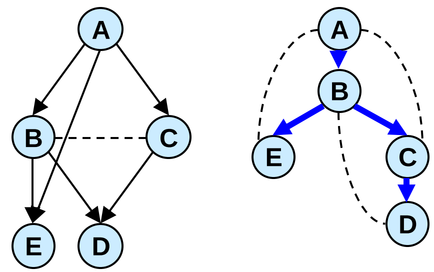
A	B=0	B=1
0	.4	.6
1	.1	.9

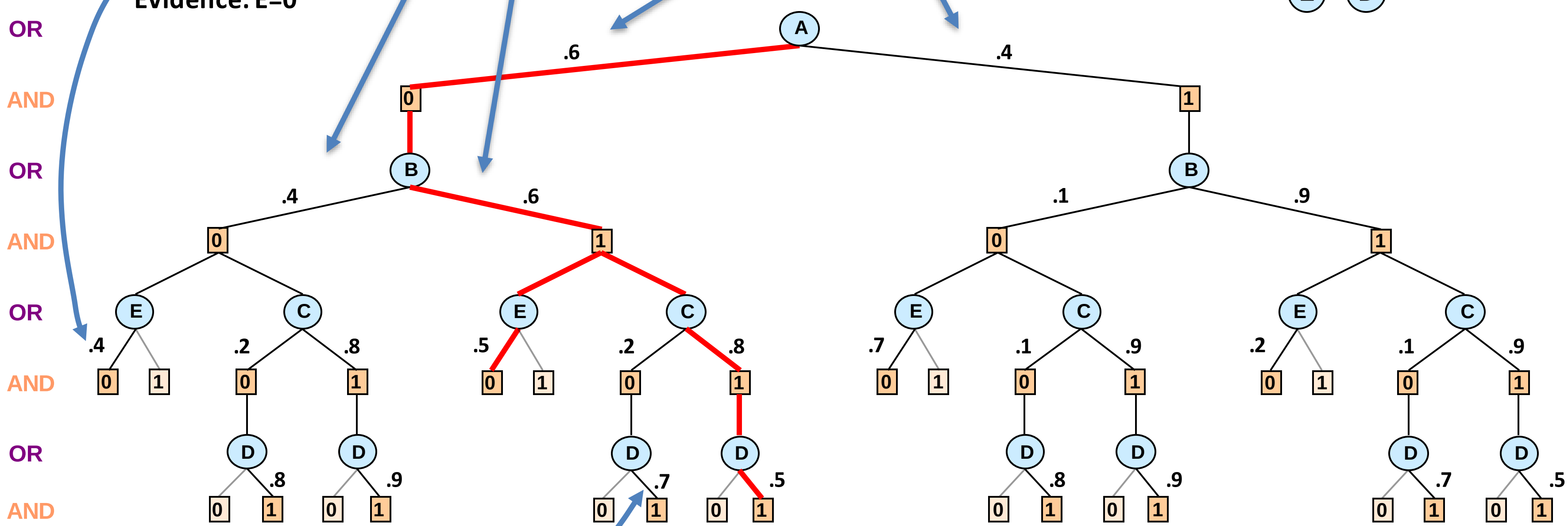
$P(C \mid A)$

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) $= 0.6 \cdot 0.6 \cdot 0.5 \cdot 0.8 \cdot 0.5 = 0.0720$

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

$P(B \mid A)$

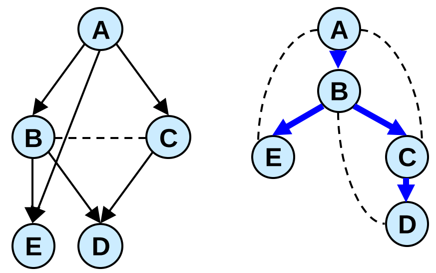
A	B=0	B=1
0	.4	.6
1	.1	.9

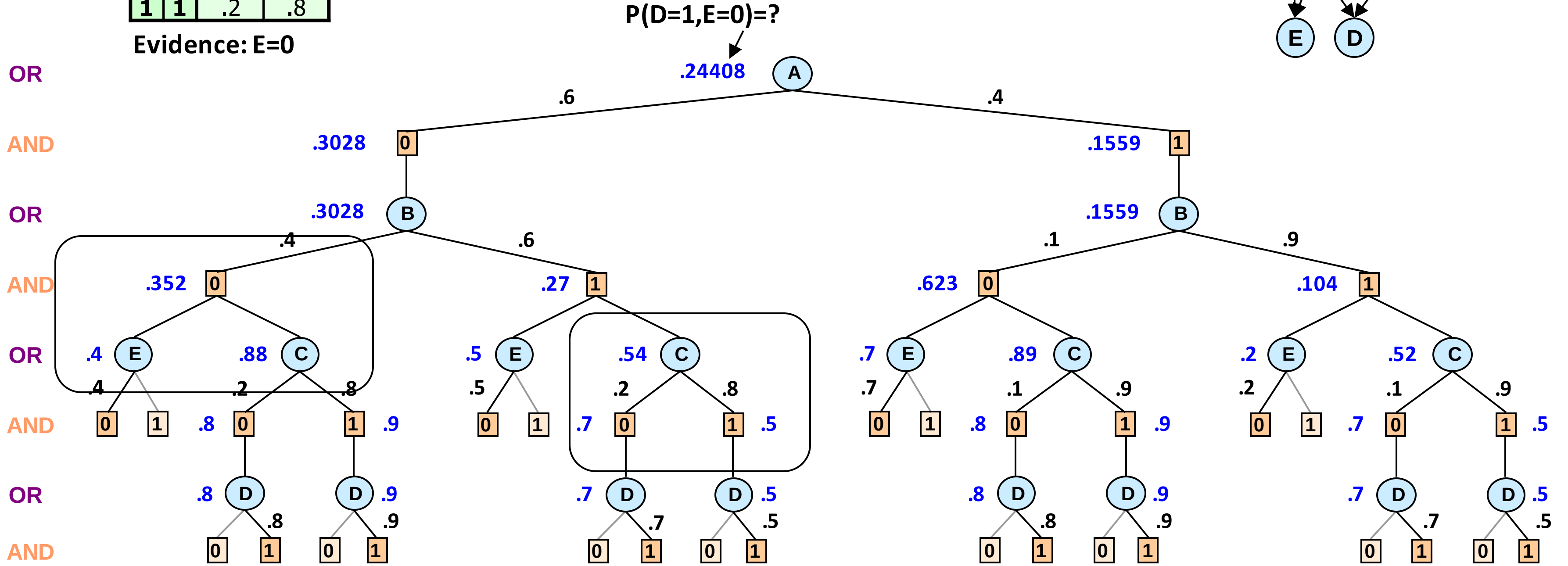
$P(C \mid A)$

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

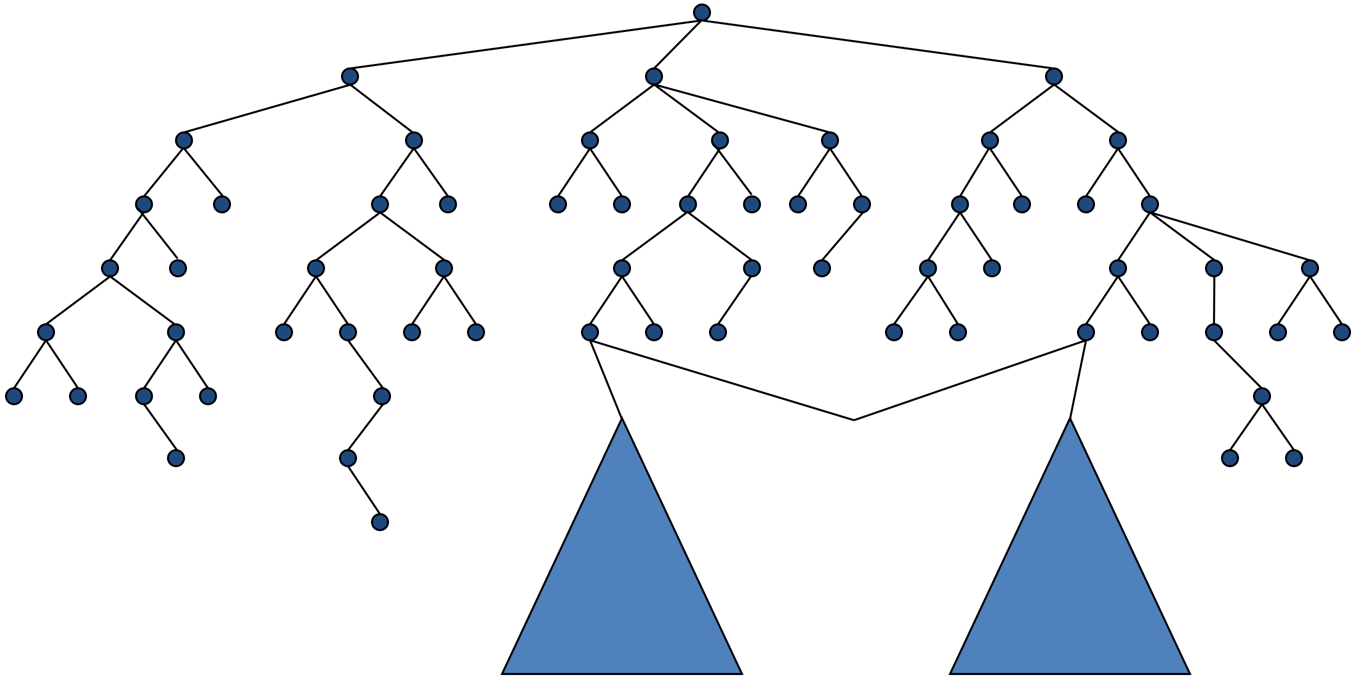
Value of node = updated belief for sub-problem below

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

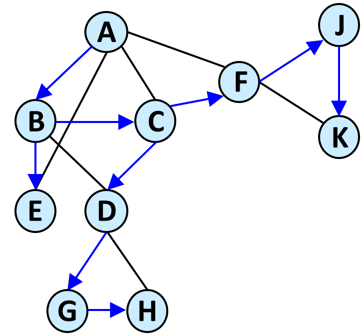
OR node: Marginalization by summation $\sum_{n' \in children(n)} w(n,n')\, v(n')$

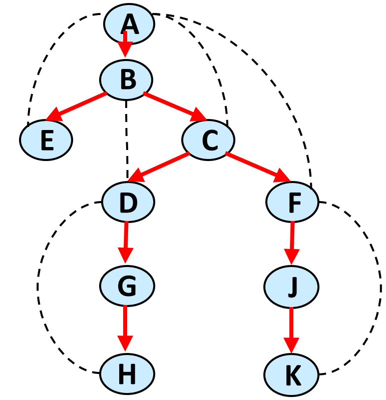
From Search Trees to Search Graphs

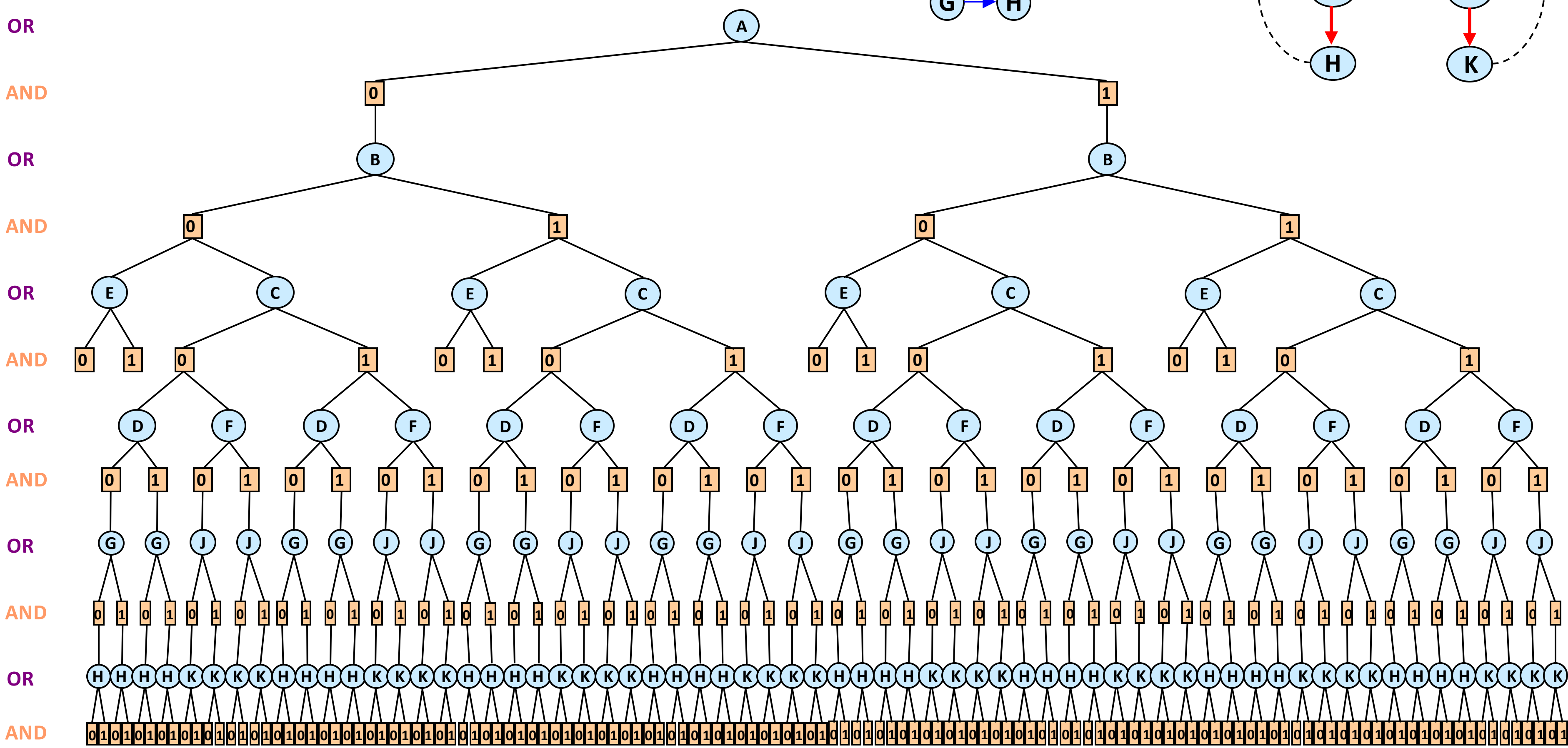
- Any two nodes that root identical sub-trees or sub-graphs can be merged



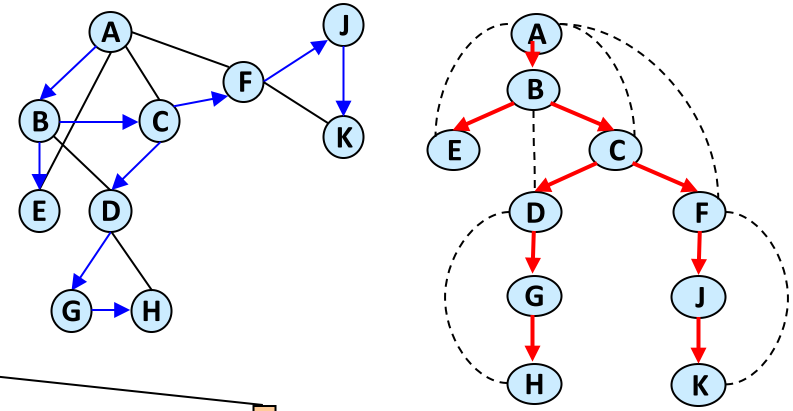
From AND/OR Tree

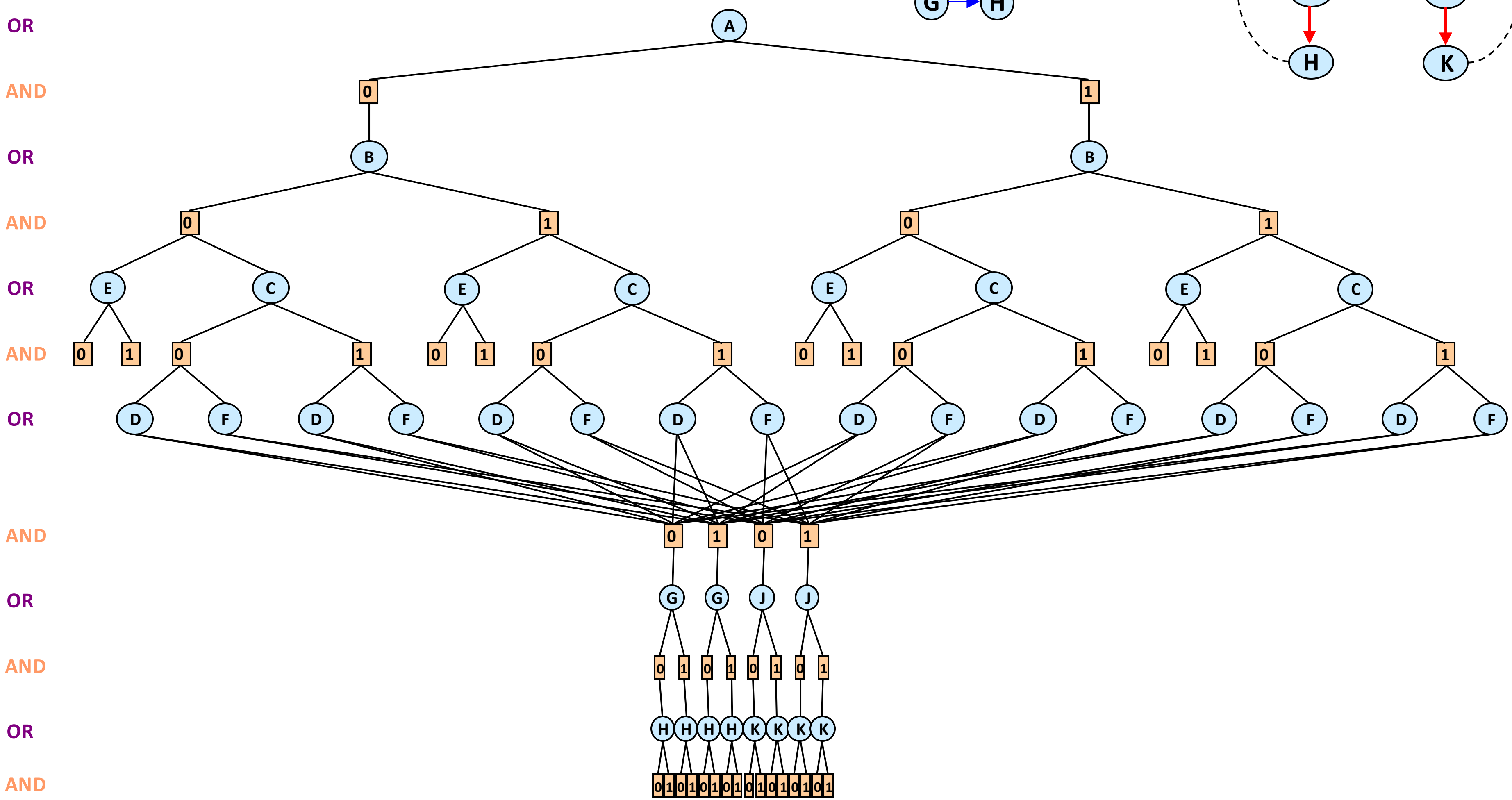






To an AND/OR Graph

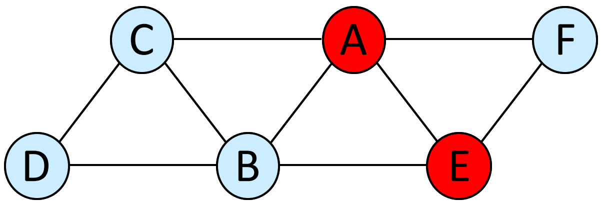


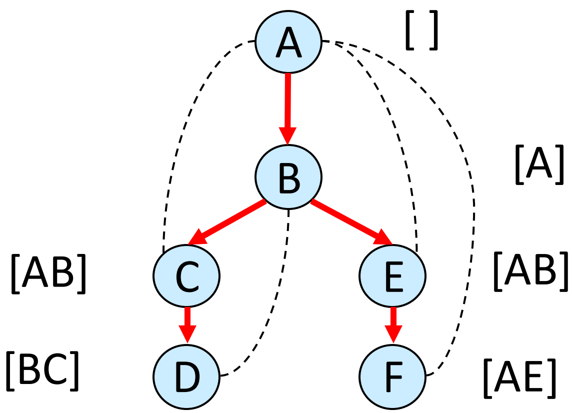


Merging Based on Context

- One way of recognizing nodes that can be merged (based on graph structure)

  context(X) = ancestors of X in the pseudo tree that are connected to X, or to descendants of X





pseudo tree

Answering Queries: Sum-Product (Belief Updating)

$P(E \mid A, B)$

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

$P(B \mid A)$

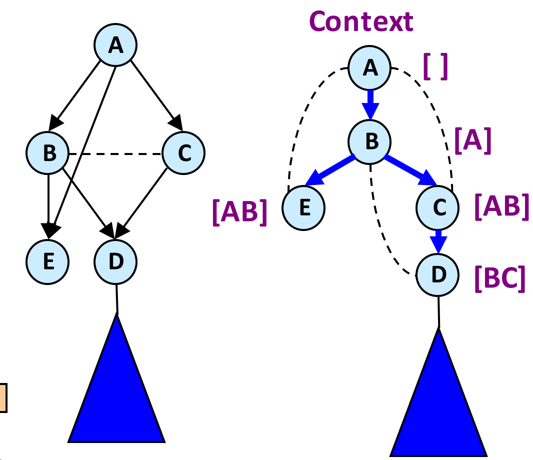
A	B=0	B=1
0	.4	.6
1	.1	.9

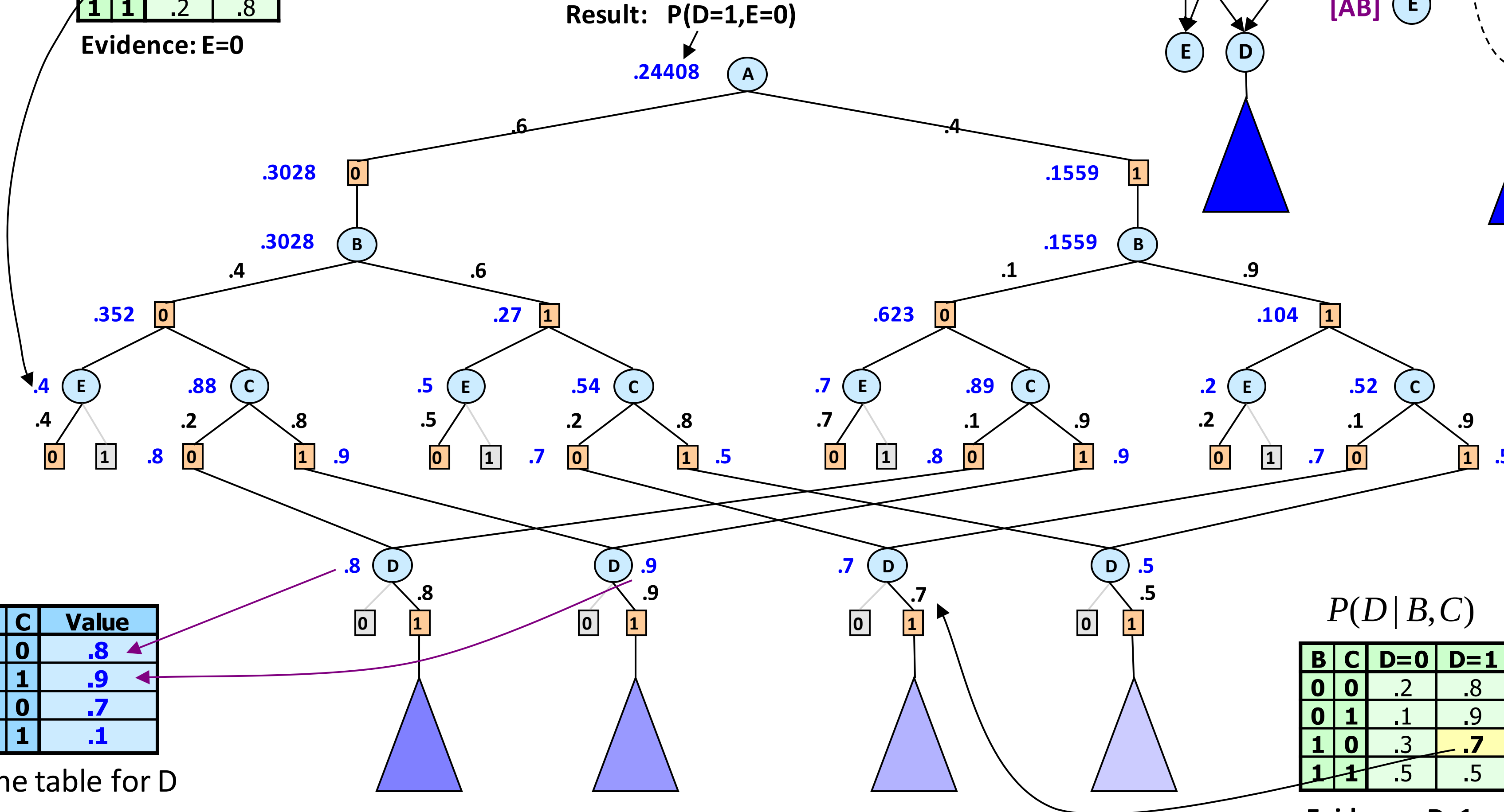
$P(C \mid A)$

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

$P(A)$

A	P(A)
0	.6
1	.4





B	C	Value
0	0	.8
0	1	.9
1	0	.7
1	1	.1

Cache table for D

$P(D \mid B, C)$

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

Potential search spaces

$$F^* = \min_x \sum_\alpha f_\alpha(x_\alpha)$$

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

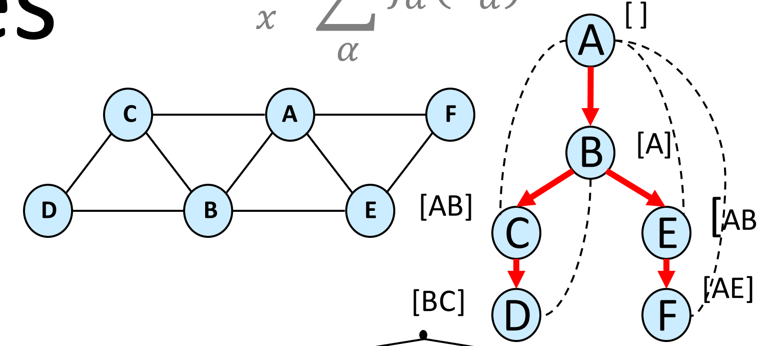
B	C	f_5
0	0	0
0	1	1
1	0	2
1	1	4

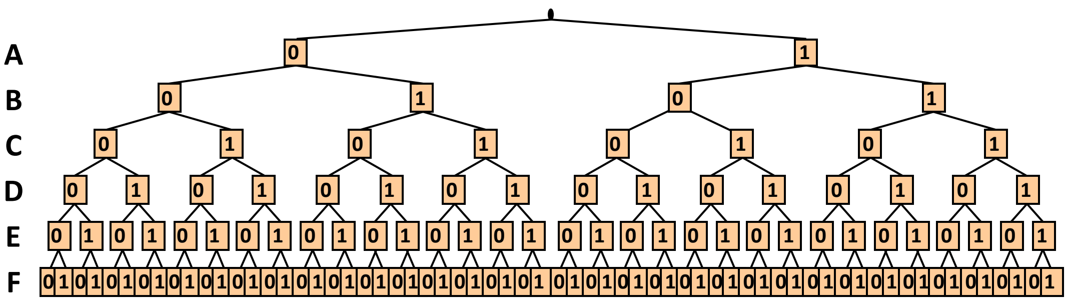
B	D	f_6
0	0	4
0	1	2
1	0	1
1	1	0

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

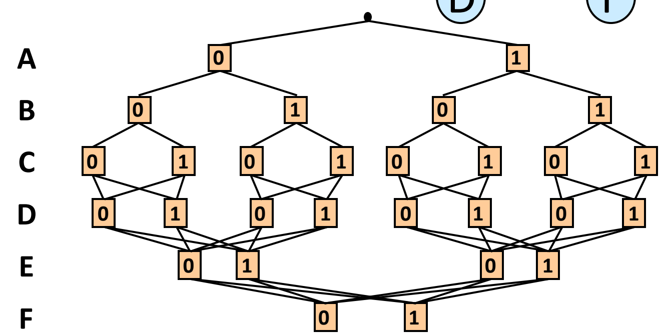
E	F	f_9
0	0	1
0	1	0
1	0	0
1	1	2





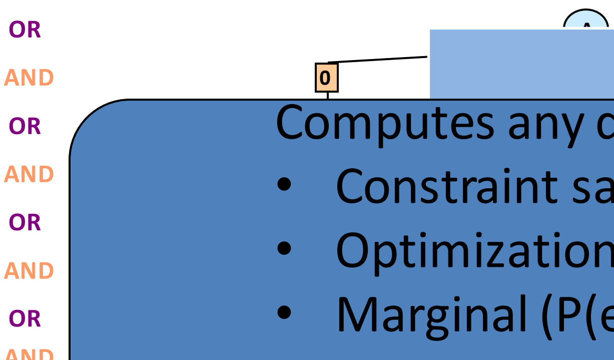
Full OR search tree

126 nodes



Context minimal OR search graph

28 nodes

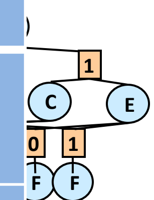


34 AND nodes

Computes any query:

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

	OR tree	AND/OR	OR graph	AND/OR graph
			$O(n\,k^{pw^*})$	$O(n\,k^{w^*})$
			$O(n\,k^{pw^*})$	$O(n\,k^{w^*})$

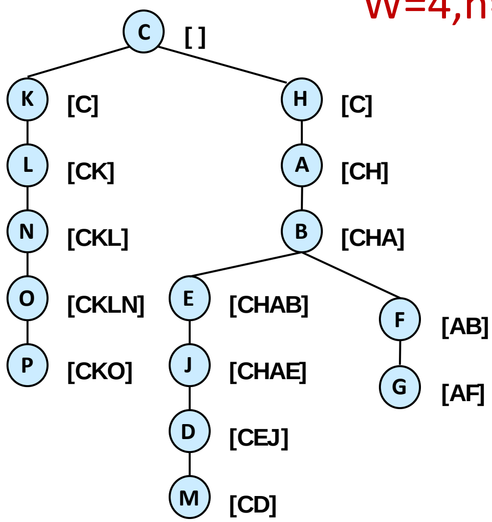


18 AND nodes

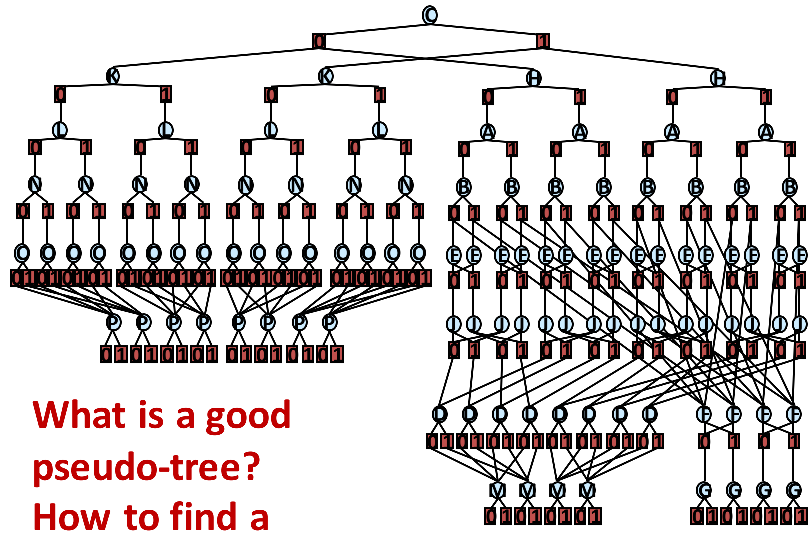
Any query is best computed
Over the c-minimal AO search space

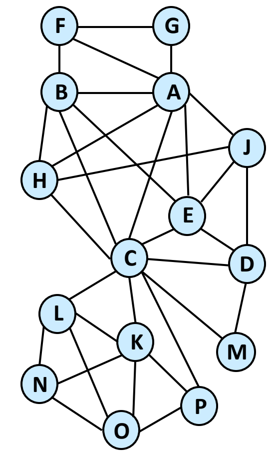
The Impact of the Pseudo-Tree

W=4,h=8



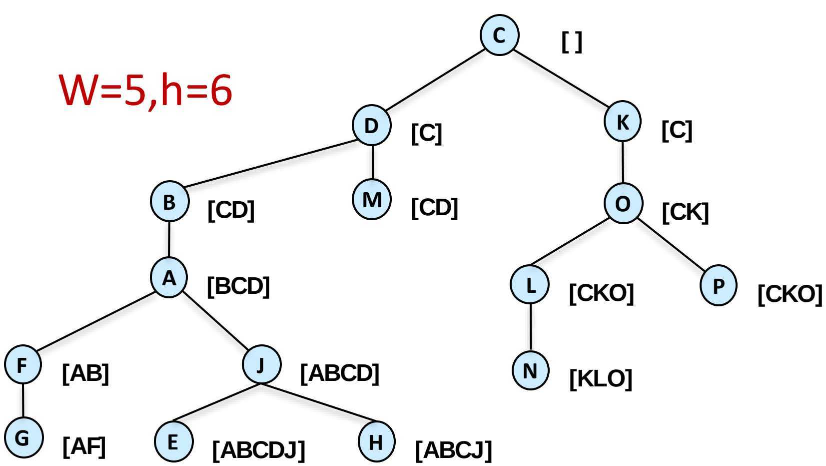
(C K H A B E J L N O D P M F G)

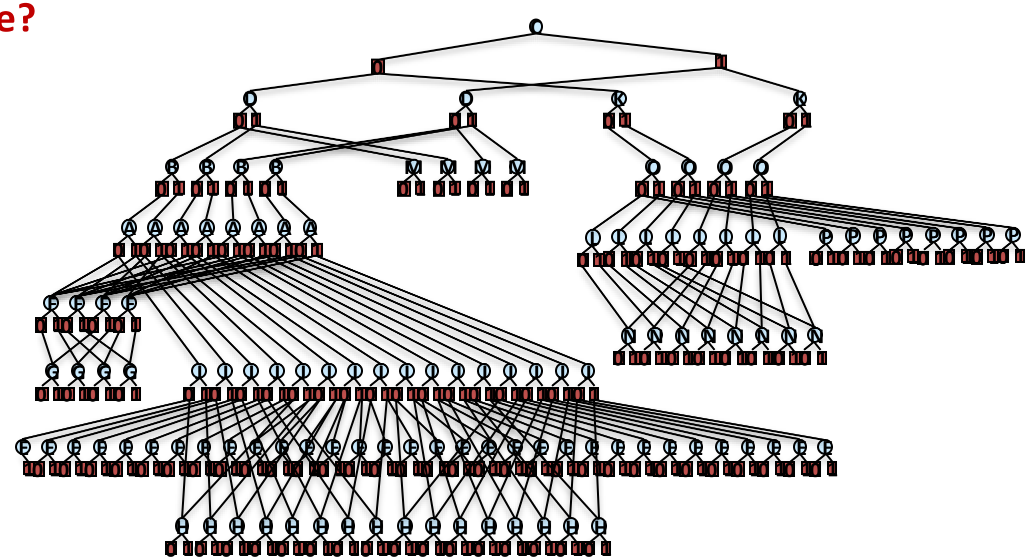




What is a good pseudo-tree? How to find a good one?

W=5,h=6

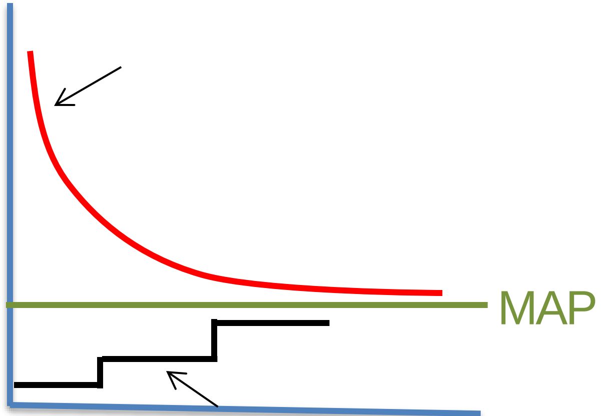




(C D K B A O M L N P J H E F G)

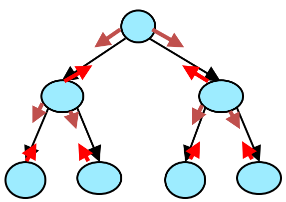
Outline

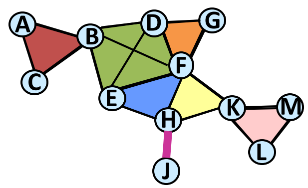
- Graphical models, Queries, Inference vs search
- AND/OR search spaces
- Bounded Inference: a) mini-bucket, b) cost-shifting
- Generating heuristics using mini-bucket elimination
- AND/OR Heuristic Search for Map and Marginal Map
- Conclusion

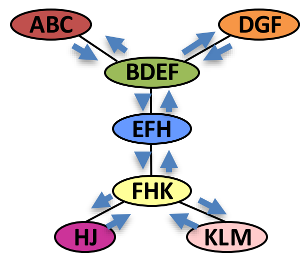


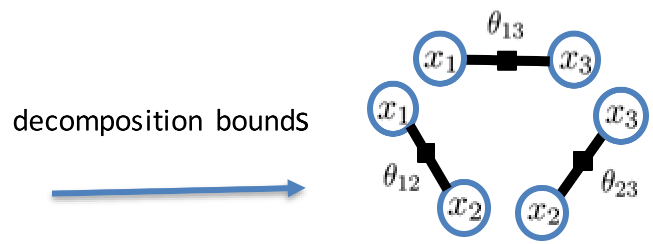
Search Collaborates with Inference

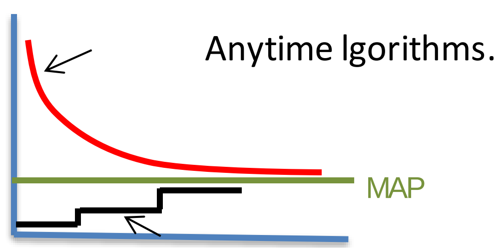
Decomopsition Bounds











Mini-Bucket Elimination (MAP)

Split a bucket into mini-buckets —> bound complexity

bucket (X) =

$$\{ \underbrace{f_1,\ f_2,\ \ldots\ f_r,\ f_{r+1},\ \ldots\ f_n}_{} \}$$

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

$$\{ f_1,\ \ldots\ f_r \}$$

$$\lambda_{X,1}(\cdot) = \max_x \prod_{i=1}^{r} f_i(x, \ldots)$$

$$\{ f_{r+1},\ \ldots\ f_n \}$$

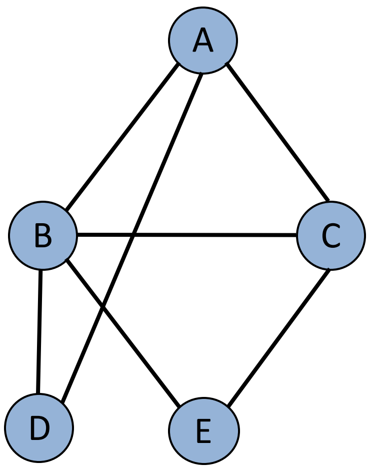
$$\lambda_{X,2}(\cdot) = \max_x \prod_{i=r+1}^{n} f_i(x, \ldots)$$

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

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

Mini-Bucket Elimination (MAP)

mini-buckets [Dechter & Rish 2003]



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

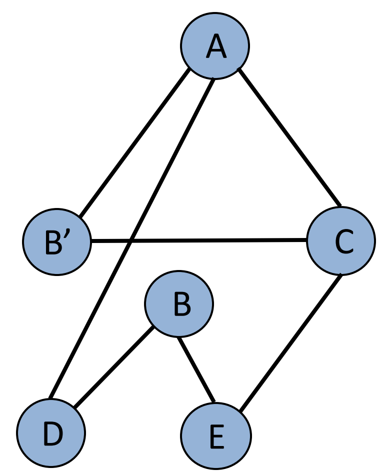
bucket C: $f(c,a)\ f(c,e)\ \lambda_{B\to C}(a,c)$

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

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

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

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 (MAP)

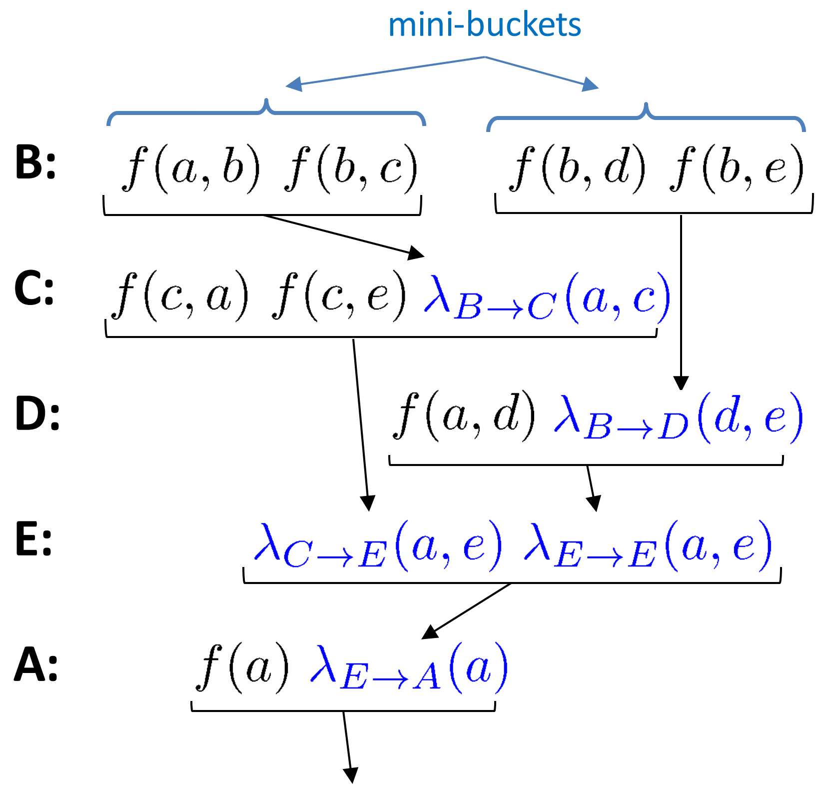
$$\mathbf{b}^* = \arg\max_b f(a^*, b) \cdot f(b, c^*) \cdot f(b, d^*) \cdot f(b, e^*)$$

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

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

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

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



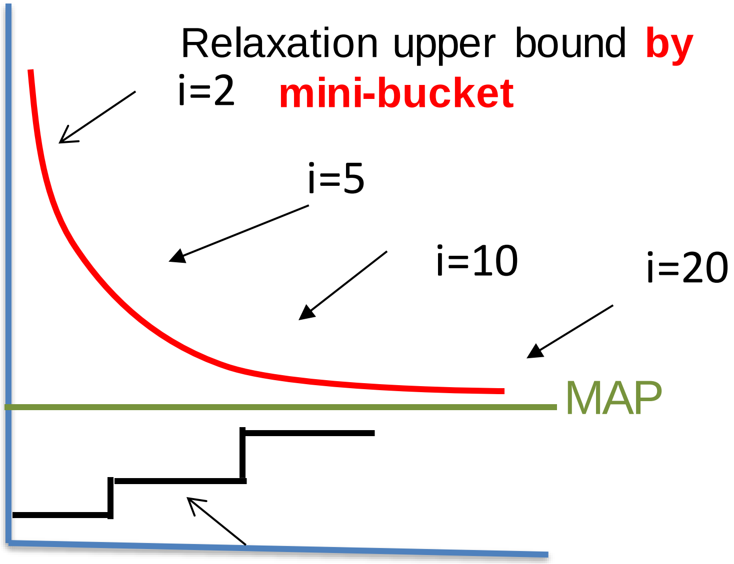
U = upper bound

Greedy configuration = lower bound

Dechter, Flairs-2018

Properties of Mini-Bucket Eliminaton

- 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

Likelihood Queries: Bounded Inference

$$Z = \sum_{x_A, x_B} \prod_{x_\alpha} \varphi_\alpha$$

Marginal Map : Bounded Inference

$$\boldsymbol{x}_B^* = \arg\max_{\boldsymbol{x}_B} \sum_{\boldsymbol{x}_A} \prod_{\alpha} \psi(\boldsymbol{x}_\alpha)$$

Decomposition Bounds for Sum

- Generalize technique to sum via Holder's inequality:

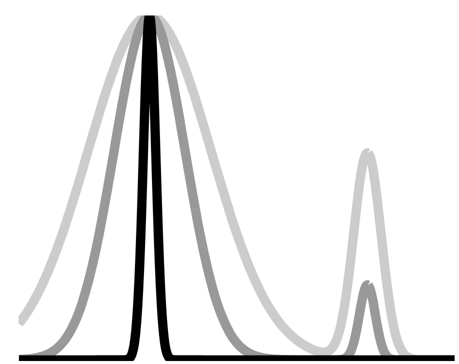
$$F(x) = f_1(x) \cdot f_2(x)$$

$$\sum_x f_1(x) \cdot f_2(x) \quad \leq \quad \Big[\sum_x f_1(x)^{\frac{1}{w_1}} \Big]^{w_1} \cdot \Big[\sum_x f_2(x)^{\frac{1}{w_2}} \Big]^{w_2} \qquad w_1 + w_2 = 1$$

- Define the weighted (or powered) sum:

$$\sum_{x_1}^{w_1} f(x_1) = \Big[\sum_{x_1} f(x_1)^{\frac{1}{w_1}} \Big]^{w_1}$$

  – "Temperature" interpolates between sum & max:



$$\lim_{w \to 0^+} \sum_x^{w} f(x) = \max_x f(x)$$

  – Different weights do not commute:

$$\sum_{x_1}^{w_1} \sum_{x_2}^{w_2} f(x_1, x_2) \neq \sum_{x_2}^{w_2} \sum_{x_1}^{w_1} f(x_1, x_2)$$

Weighted Mini-bucket (for Sum)

[Liu & Ihler 2011]

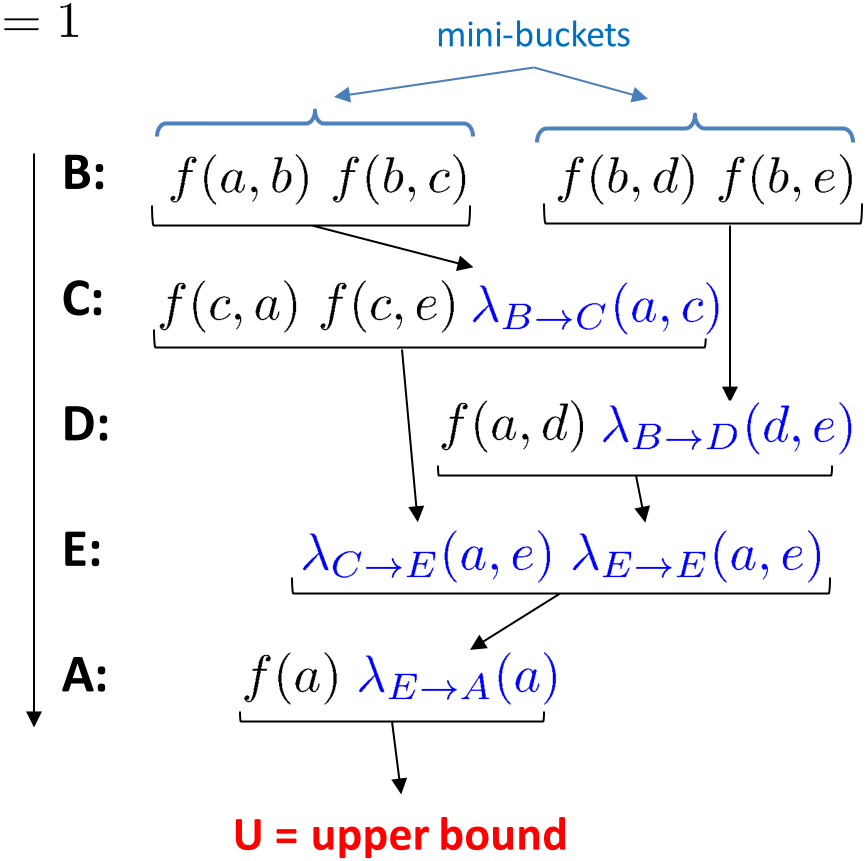
$$\lambda_{B\to C} = \sum_b^{w_{B1}} f(a,b)\cdot f(b,c)$$

$$w_{B1} + w_{B2} = 1$$

$$\lambda_{B\to D} = \sum_b^{w_{B2}} f(b,d)\cdot f(b,e)$$

$$\lambda_{C\to E} = \sum_c f(c,a)\cdot f(c,e)\cdot \lambda_{B\to C}$$

⋮

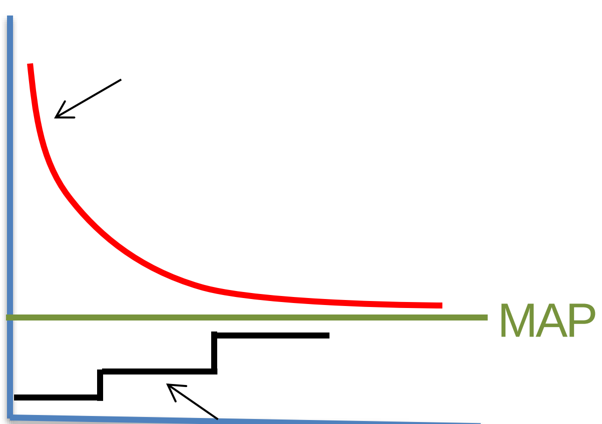


Compute downward messages
using weighted sum

Upper bound if all weights positive
(corresponding lower bound if only one positive, rest negative)

Outline

- Graphical models, Queries, Inference vs search
- AND/OR search spaces
- Bounded Inference: a) weighted mini-bucket, b) cost shifting
- Generating heuristics using mini-bucket elimination
- AND/OR Heuristic Search for Map and Marginal Map
- Conclusion



Tightening the Bound

(Reparameterization, or cost-shifting)

$+\lambda(B)$

A	B	f(A,B)
b	b	6 + 3
b	g	0 - 1
g	b	0 + 3
g	g	6 - 1

+

$-\lambda(B)$

B	C	f(B,C)
b	b	6 - 3
b	g	0 - 3
g	b	0 + 1
g	g	6 + 1

A	B	C	f(A,B,C)
b	b	b	12
b	b	g	6
b	g	b	0
b	g	g	6
g	b	b	6
g	b	g	0
g	g	b	6
g	g	g	12

= 0 + 6

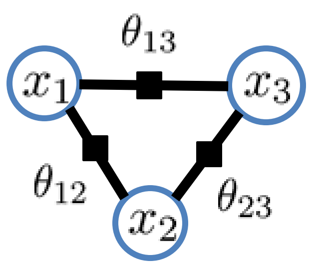
B	λ(B)
b	3
g	-1

Modify the individual functions

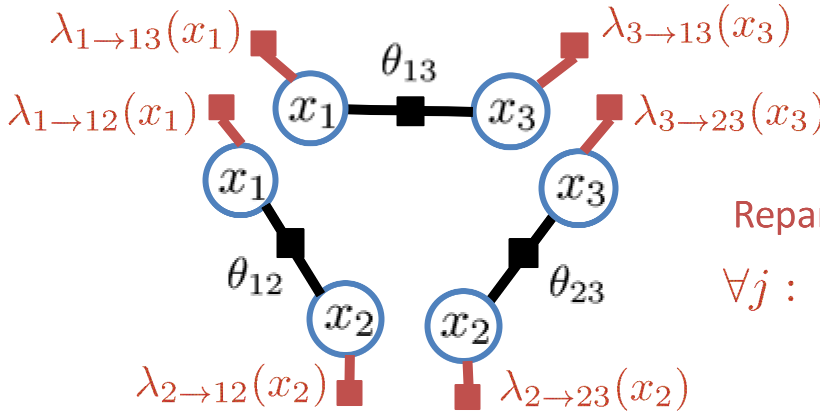
- but -

keep the sum or product of functions unchanged

Tightening the Bound



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



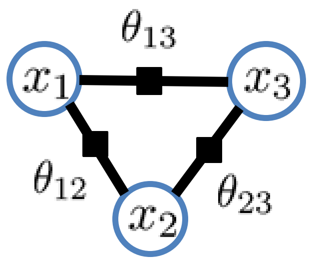
Reparameterization:

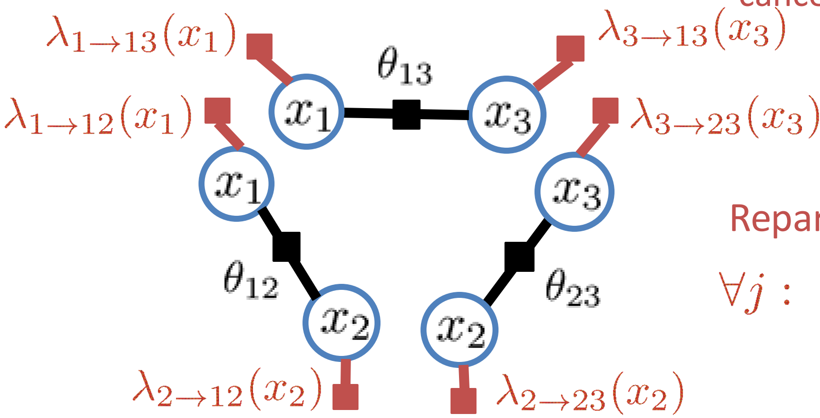
$$\forall j : \sum_{\alpha \ni j} \lambda_{j\to\alpha}(x_j) = 0$$

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

- 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:

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

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

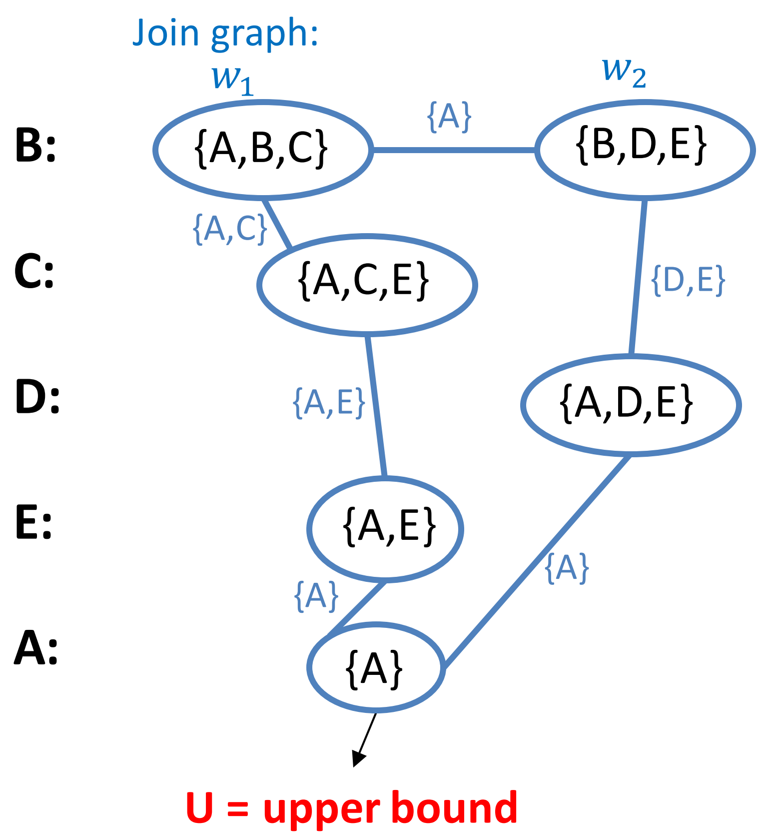
- Many names for the same class of bounds
  - Dual decomposition [Komodakis et al. 2007]
  - TRW, MPLP [Wainwright et al. 2005; Globerson & Jaakkola 2007]
  - Soft arc consistency [Cooper & Schiex 2004]
  - Max-sum diffusion [Warner 2007]

Can use any decomposition updates , yields message passing, subgradient, coordinated decen, etc.

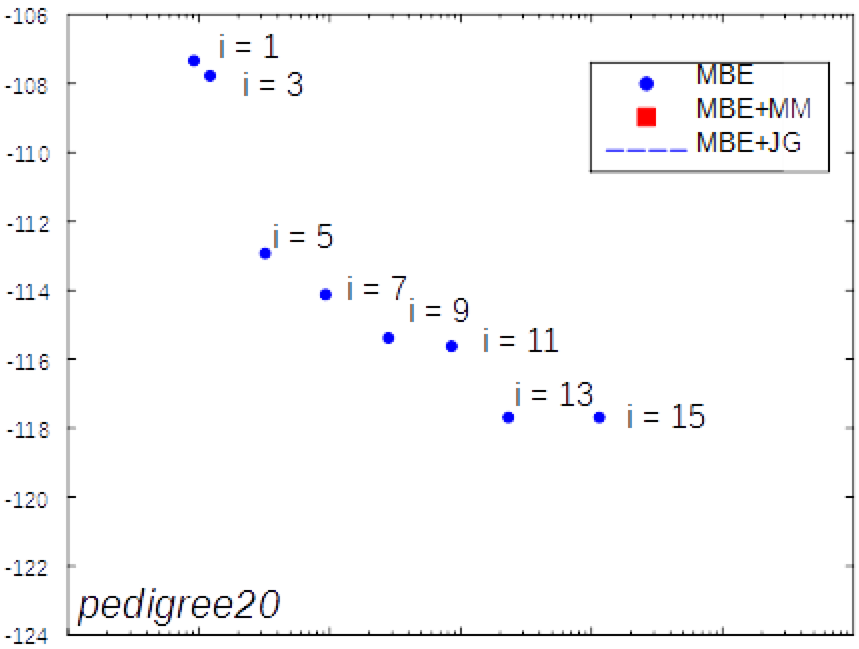
Weighted Mini-Bucket with Moment Matching (WMB-MM)

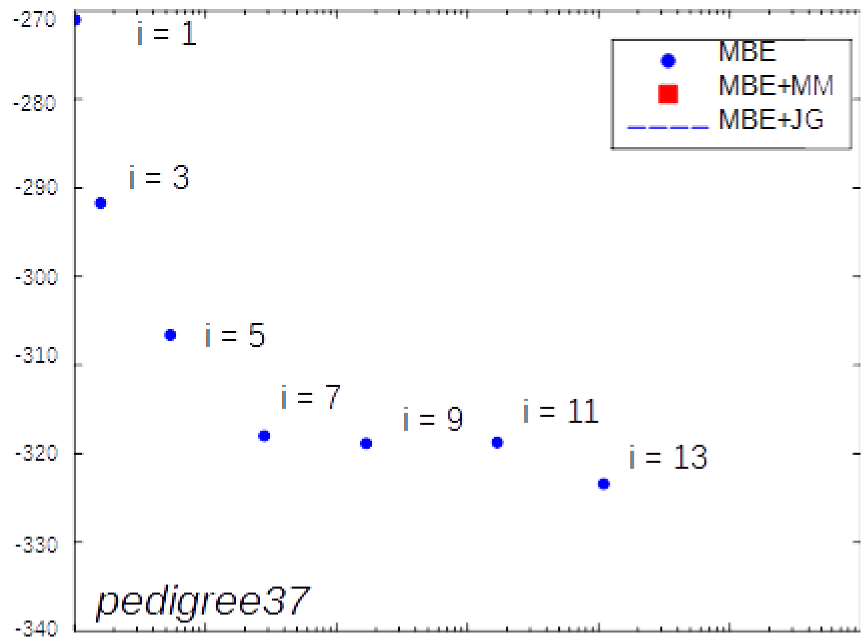
[Ihler et al. 2012]

- Downward pass as cost shifting
- Can also do cost shifting within mini-buckets:
  "Join graph" message passing
- "Moment-matching" version:
  One message exchange within each bucket, during downward sweep
- Can optimize the bound over:
  Cost-shifting
  Weights



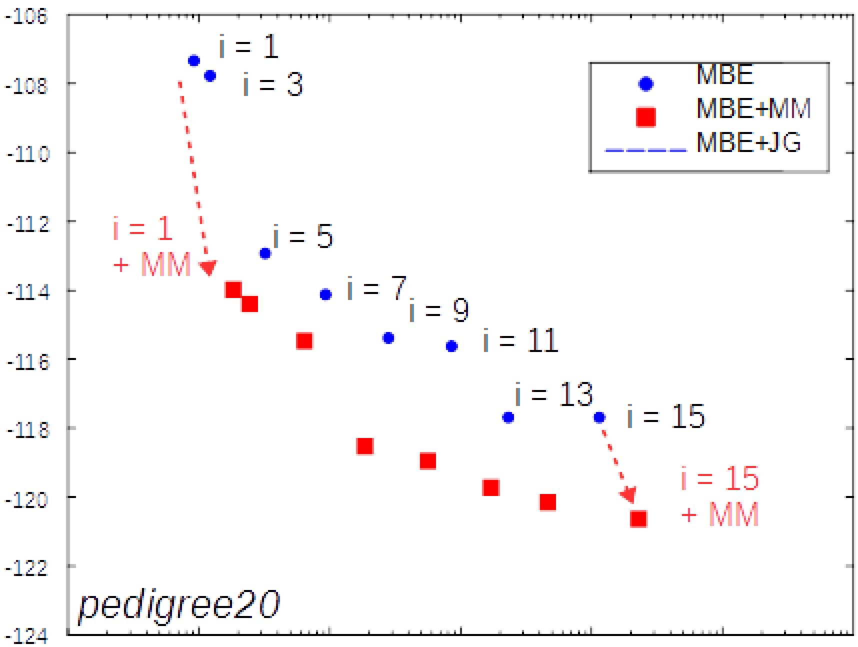
Anytime Approximation (MAP)

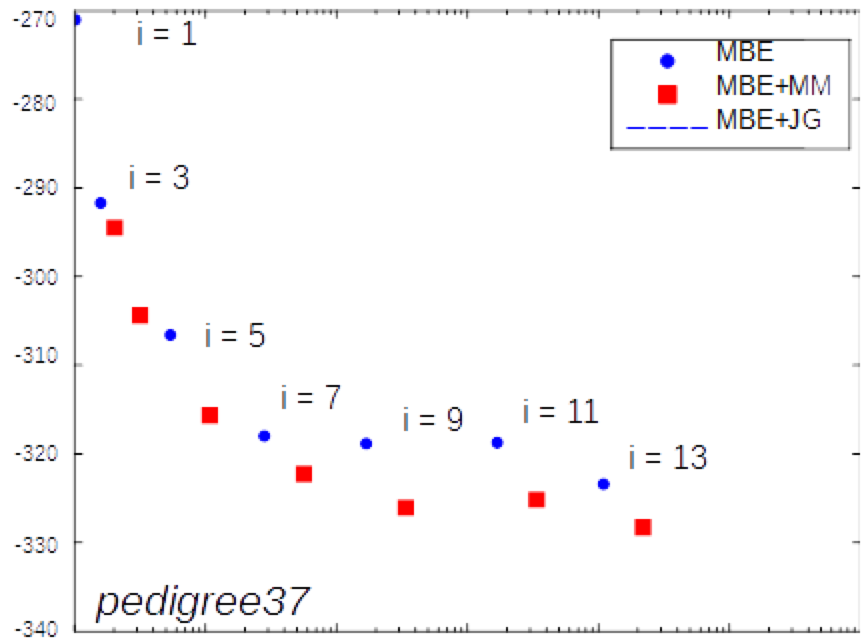




- Can tighten the bound in various ways
  - Cost-shifting (improve consistency between cliques)
  - Increase i-bound (higher order consistency)
- Simple moment-matching step improves bound significantly

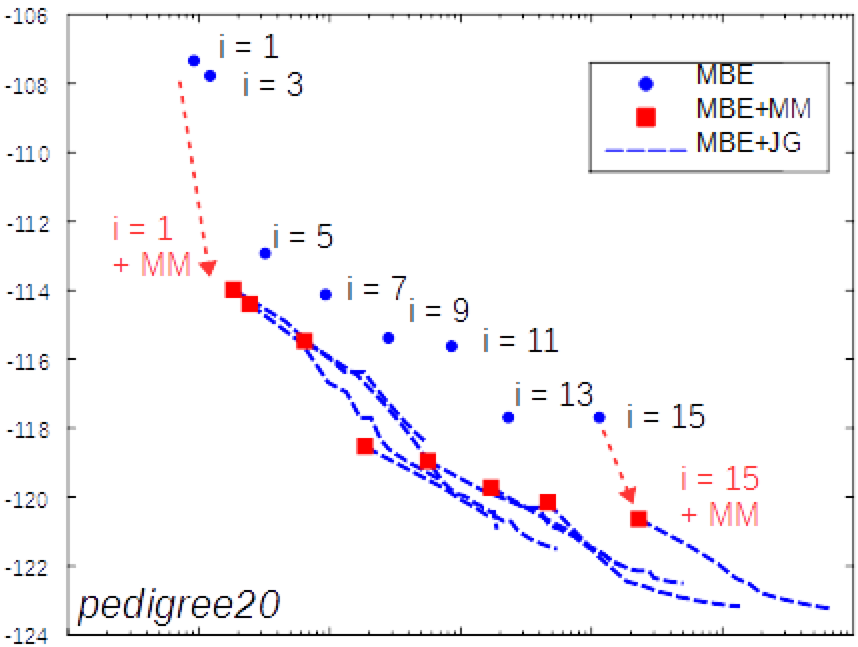
Anytime Approximation (MAP)

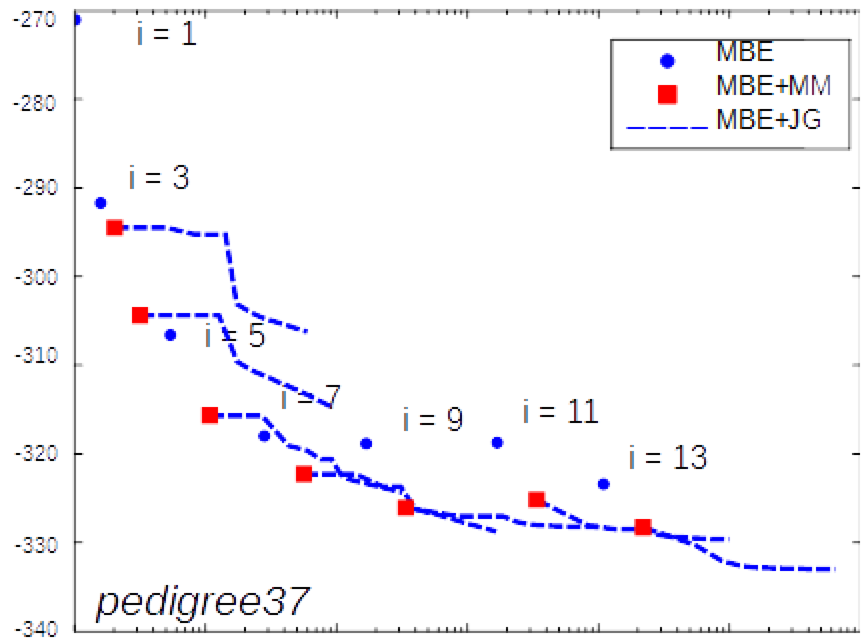




- Can tighten the bound in various ways
  - Cost-shifting (improve consistency between cliques)
  - Increase i-bound (higher order consistency)
- Simple moment-matching step improves bound significantly

Anytime Approximation (MAP)

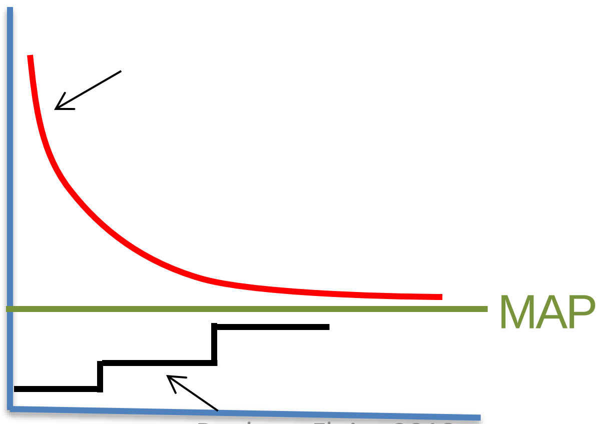




- Can tighten the bound in various ways
  - Cost-shifting (improve consistency between cliques)
  - Increase i-bound (higher order consistency)
- Simple moment-matching step improves bound significantly

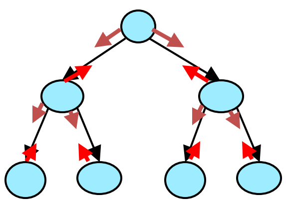
Outline

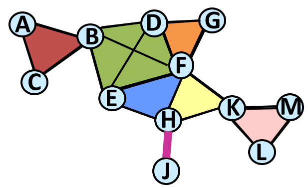
- Graphical models, Queries, Inference vs search
- AND/OR search spaces
- Bounded Inference: a) mini-bucket, b) cost-shifting
- Generating heuristics using mini-bucket elimination
- AND/OR Heuristic Search for Map and Marginal Map
- Conclusion

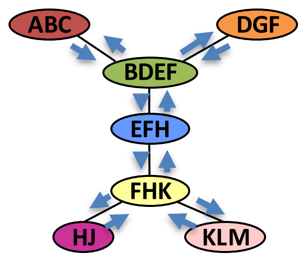


Search Collaborates with Inference

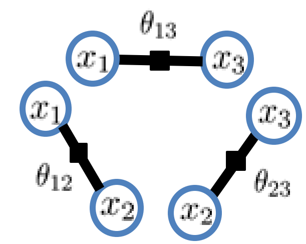
- Inference: message-passing on cluster-tree





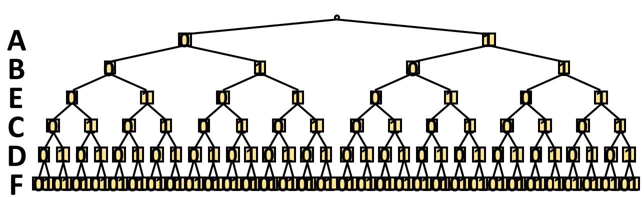


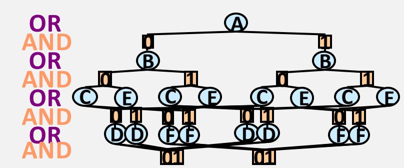
decomposition bounds



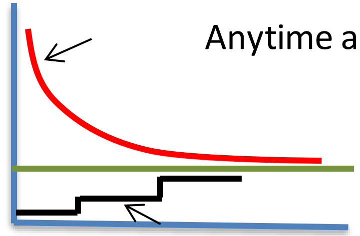
Heuristics
To guide AND/OR
search

- Search → exploiting structure





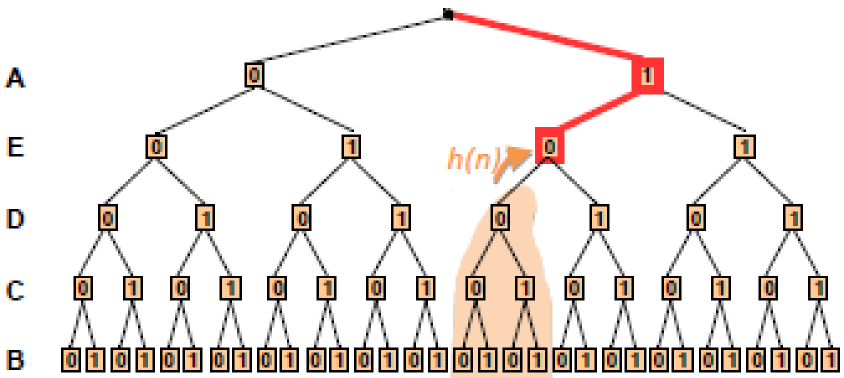
Context minimal AND/OR search graph

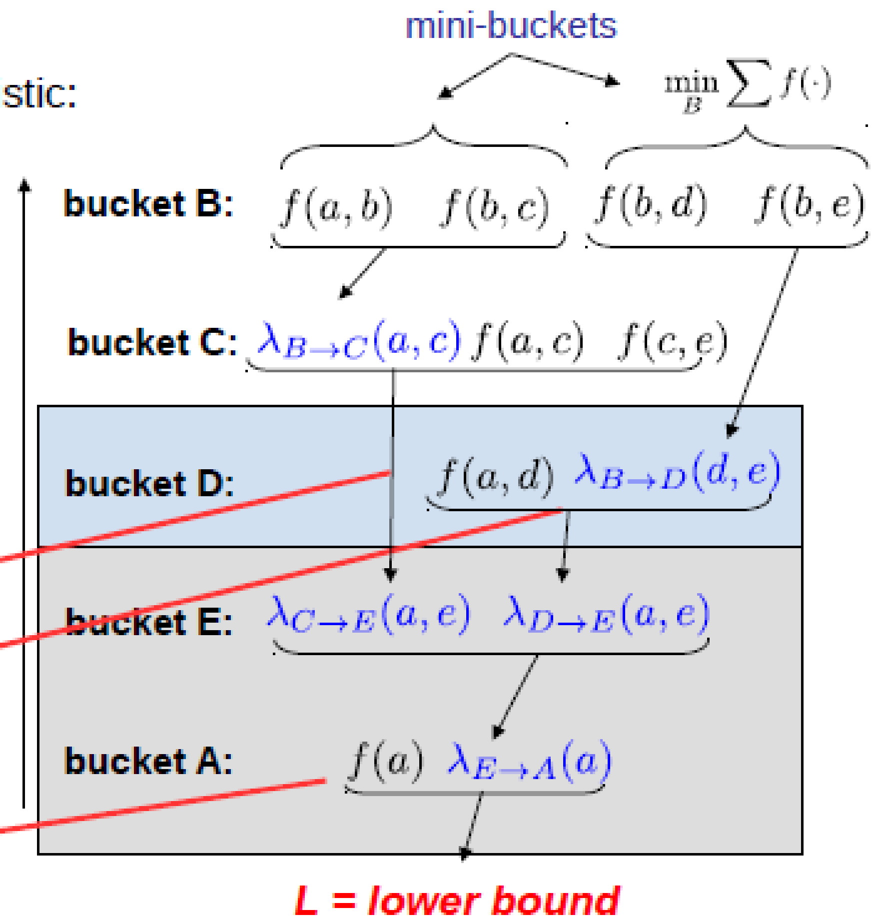


Anytime algorithms.

Static Mini-Bucket Heuristics

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





cost to go:

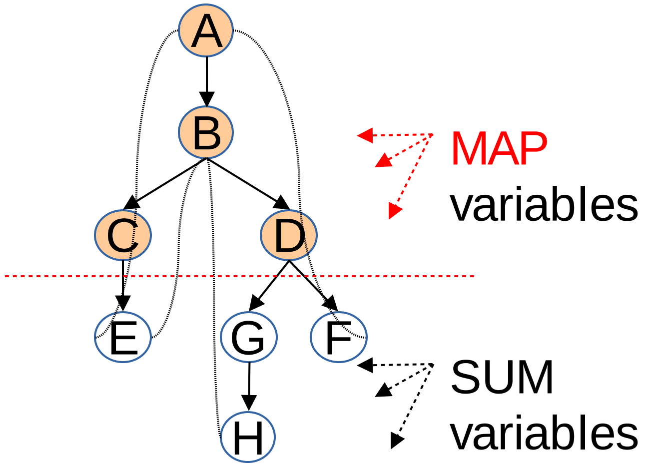
$$\tilde{h}(\hat{a},\hat{e},D) = \lambda_{C\to E}(\hat{a},\hat{e}) + f(\hat{a},D) + \lambda_{B\to D}(D,\hat{e})$$

(admissible: $\tilde{h}(\hat{a},\hat{e},D) \le h^*(\hat{a},\hat{e},D)$)

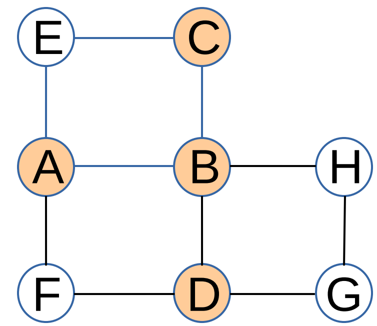
cost so far:

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

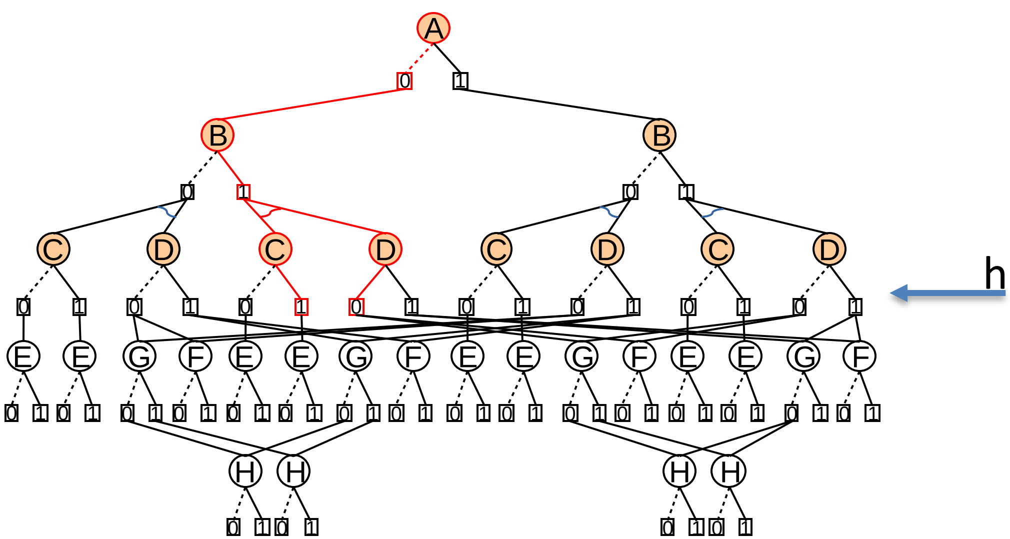
AND/OR Search for Marginal MAP

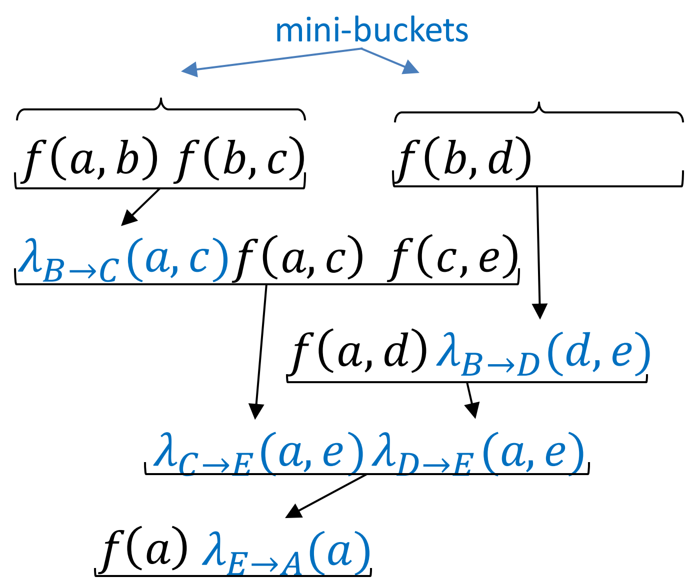


constrained pseudo tree



primal graph





[Marinescu, Dechter and Ihler, 2014]

AO search for MAP winning UAI Probabilistic Inference Competitions

MPE/MAP

- **2006** 

(aolib)
- **2008** 

(aolib)
- **2011** 

(daoopt)
- **2014** 

(daoopt)

MMAP

- **2014** 

(daoopt) 

(merlin)

Anytime AND/OR solvers for MMAP

- **Weighted Heuristic:** [Lee et. al. AAAI-2016]
  - Weighted Restarting AOBF (WAOBF)
  - Weighted Restarting RBFAOO (WRBFAOO)
  - Weighted Repairing AOBF (WRAOBF)

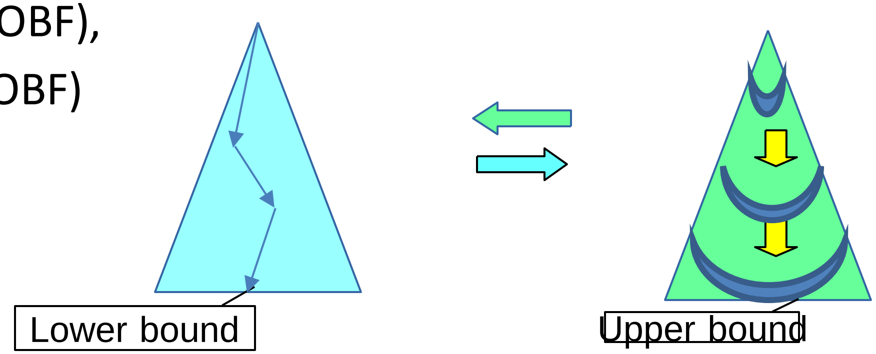
Weighted A* search [Pohl 1970]

- non-admissible heuristic
- Evaluation function:

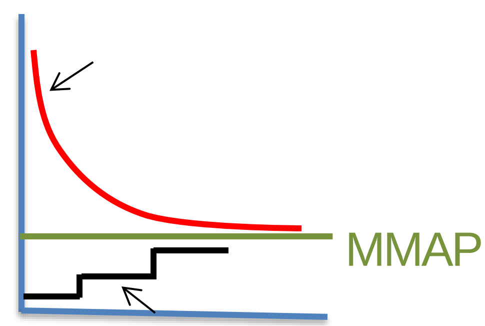
$$f(n)=g(n)+w \cdot h(n)$$

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

- **Interleaving Best and depth-first search**: (Marinescu et. al AAAI-2017)
  - Look-ahead (LAOBF),
  - alternating (AAOBF)



Goal: anytime bounds
And anytime solution



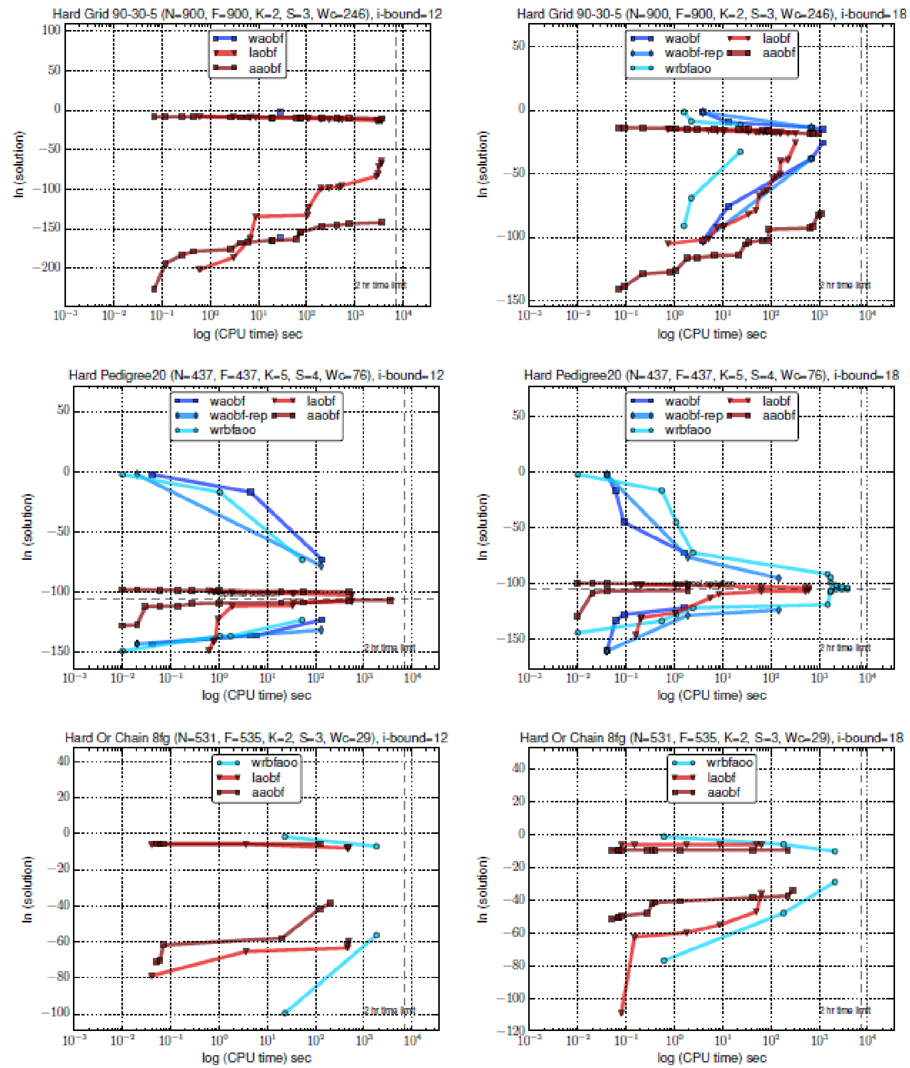
Anytime Bounding of Marginal MAP

(UAI'14, IJCAI'15, AAAI'16, AAAI'17, (Marinescu, Lee, Ihler, Dechter)

- 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.

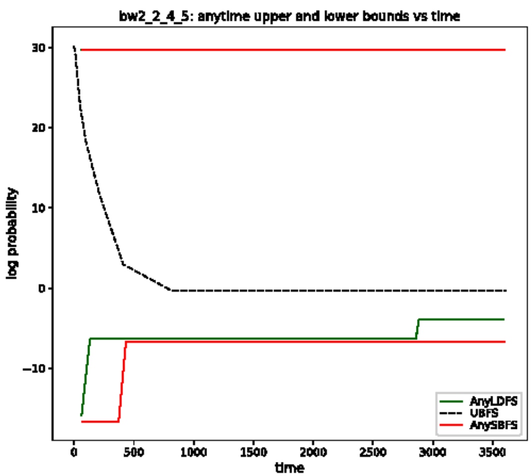


New Generation Algorithms (Approximate Summation)

[Lou, Dechter, Ihler, AAAI-2018: "Anytime Anyspace AND/OR Best-first Search for Bounding Marginal MAP"]
[Lou, Dechter, Ihler, UAI-2018: "Finite Sample Bounds for Marginal MAP", UAI 2018]
[Marinescu, Ihler, Dechter: IJCAI-2018 "Stochastic Anytime Search for Bounding Marginal MAP"]

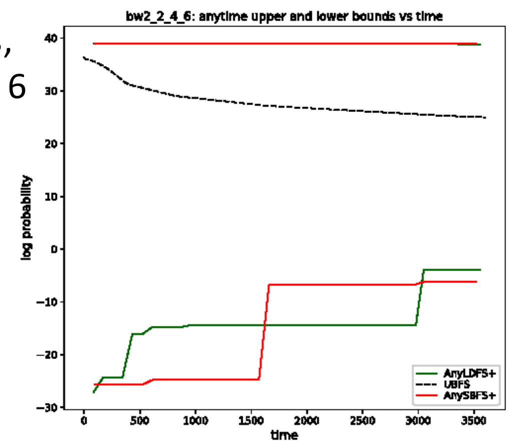
(n,k,c,w*,h)

(314,3,317,56,248)



2 blocks,
T=5 and 6

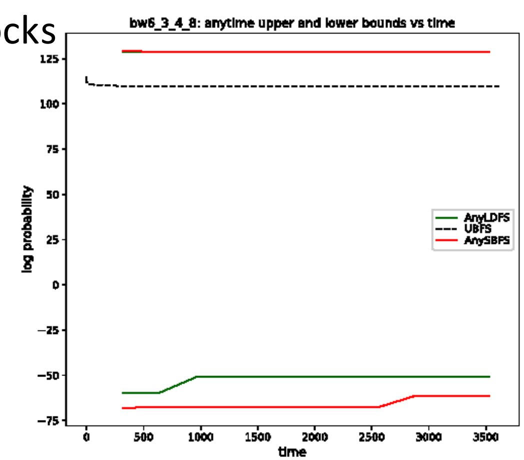
(375,3,378,64,302)



Algorithms:
UBFS
ANYLDFS
AnySBFS

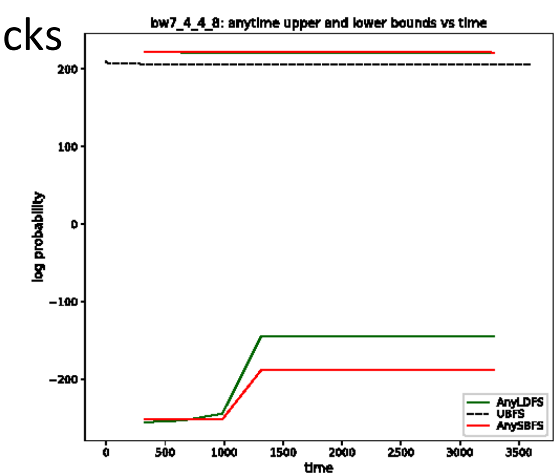
(1134,3,1044,173,908)

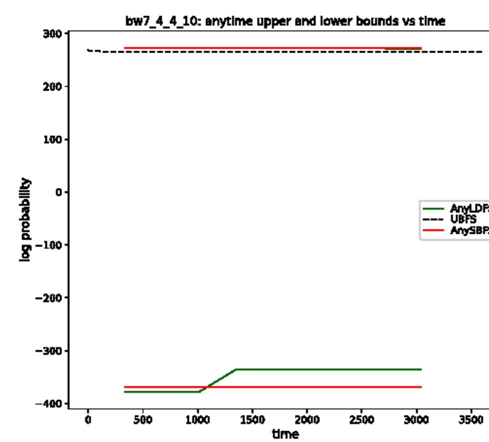
6 blocks
T=8



(2161,3,2168,302,1484

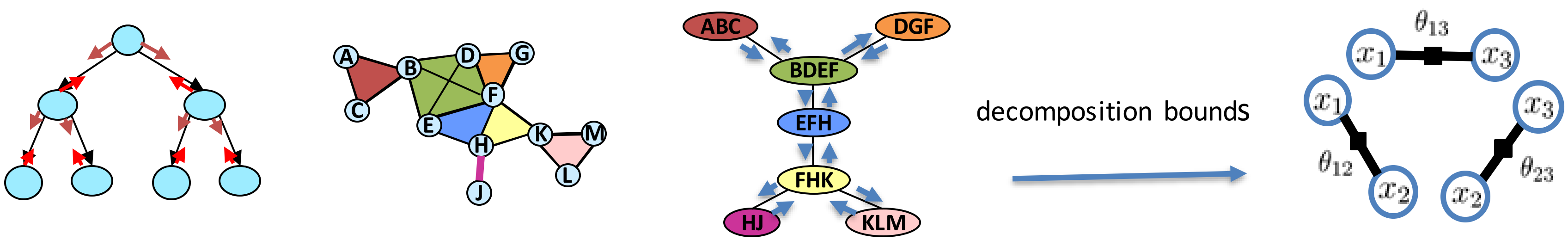
7 blocks
T=8





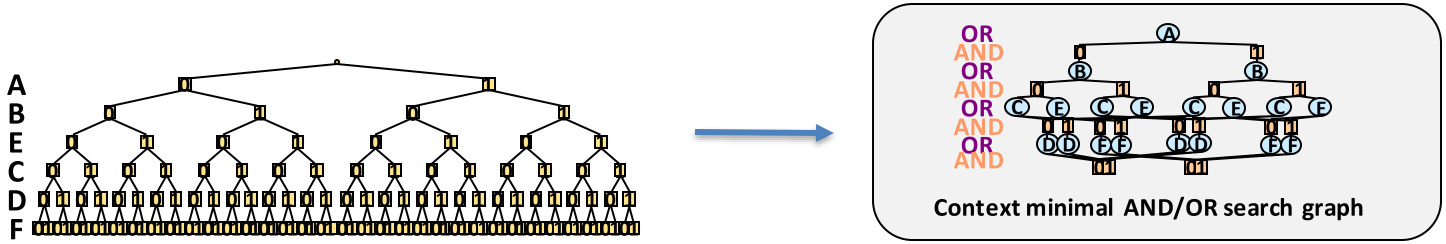
Conclusion:
Search Collaborates with Inference

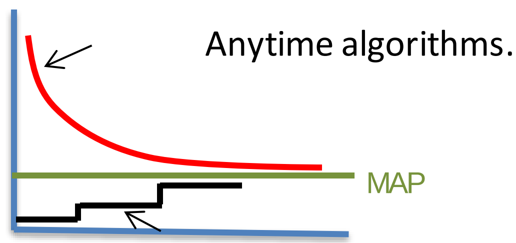
- Inference: message-passing on cluster-tree



Heuristics
To guide AND/OR
search

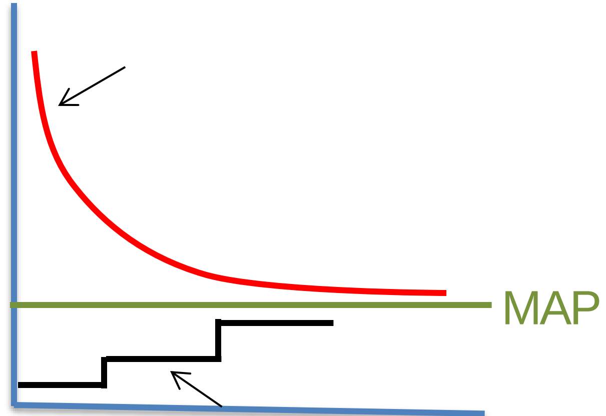
- Search → exploiting structure





Outline/Conclusions

- Graphical models, Queries, Inference vs search
- AND/OR search spaces
- Bounded Inference: a) mini-bucket, b) cost-shifting
- Generating heuristics using mini-bucket elimination
- AND/OR Heuristic Search for Map and Marginal Map
- Conclusion



Thank You !

For publication see:
http://www.ics.uci.edu/~dechter/publications.html








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