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

A* search

Anytime algorithms.

Dechter, Flairs 2018
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
    \[
    x^* = \arg\max_x \prod_\alpha f_\alpha(x_\alpha) \quad f(x^*) = \max_x \prod_{\alpha} f_\alpha(x_\alpha)
    \]

[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
    \[
p(x_i) = \frac{1}{Z} \sum_{x \backslash x_i} \prod_{\alpha} f_\alpha(x_\alpha) \quad \text{and} \quad Z = \sum_{x} \prod_{\alpha} f_\alpha(x_\alpha)
    \]
    “partition function”

Observation $y$

Marginals $p(x_i \mid y)$

Observation $y$

Marginals $p(x_i \mid y)$

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, ...)
    \[
    f(x^*_M) = \max_{x_M} \sum_{x_S} \prod_{\alpha} f_{\alpha}(x_{\alpha})
    \]

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:
  
  \[ A \neq B, \ A \neq D, \ D \neq E, \ \text{etc.} \]

Propositional Satisfiability

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

Semantics: set of all solutions

Primary task: find a solution

Dechter, Flairs-2018
Bayesian Networks (Pearl 1988)

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

- Posterior marginals, probability of evidence, MPE

\[ P( D= 0) = \sum_{S,L,B,X} P(S) \cdot P(C|S) \cdot P(B|S) \cdot P(X|C,S) \cdot P(D|C,B) \]

\[ \text{MAP}(P) = max_{S,L,B,X} P(S) \cdot P(C|S) \cdot P(B|S) \cdot P(X|C,S) \cdot P(D|C,B) \]

BN = (G, \Theta)

Combination: Product
Marginalization: sum/max
Monitoring Intensive-Care Patients

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

[Beinlich et al., 1989]
Chief Complaint: Sore Throat

Diseases

Observations

Unobservable nodes

KI

Dechter, Flairs-2018
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

Dechter, Flairs-2018
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:**

<p>| | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Max-Inference (MAP)</td>
<td>( f(x^*) = \max_x \prod_{\alpha} f_{\alpha}(x_{\alpha}) )</td>
</tr>
<tr>
<td>Sum-Inference (P($))</td>
<td>( Z = \sum_x \prod_{\alpha} f_{\alpha}(x_{\alpha}) )</td>
</tr>
<tr>
<td>Mixed-Inference</td>
<td>( f(x^*<em>M) = \max</em>{x_M} \sum_{x_S} \prod_{\alpha} f_{\alpha}(x_{\alpha}) )</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th></th>
<th>Conditional Probability Table (CPT)</th>
</tr>
</thead>
<tbody>
<tr>
<td>A</td>
<td>C</td>
</tr>
<tr>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>1</td>
<td>1</td>
</tr>
</tbody>
</table>

**Relation**

\( f_i := (F = A + C) \)

**Primal graph (interaction graph)**

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

Dechter, Flairs-2018
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<sub>pp</sub> complete
  Not necessarily easy on trees
Search Collaborates with Inference

• Inference: message-passing on cluster-tree

• Search $\Rightarrow$ exploiting structure

Anytime algorithms.
More on Inference: Bucket Elimination
Query 1: Belief updating: $P(X|\text{evidence})=\ ?$

```
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)

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

$P(a \mid e=0)$

$P(a)$

$\lambda_{E \rightarrow A}(a)$

$\lambda_{D \rightarrow A}(a, e)$

$\lambda_{C \rightarrow D}(a, d, e)$

$\lambda_{B \rightarrow C}(a, d, c, e)$

$\sum \prod_b$ Elimination operator

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

bucket C: $P(c \mid a)$

bucket D: $e=0$

bucket E: $e=0$

bucket A: $P(a)$

$P(e=0)$

$W^* = 4$

"induced width" (max clique size)

Complexity time and space $O(nk^{W^*+1})$
Query 2: Finding MAP by Bucket Elimination

\[ MPE = \max_{a,e,d,c,b} P(a)P(c|a)P(b|a) \]

Algorithm BE-mpe (Dechter 1996, Bertele and Briand, 1977)

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

Dechter, Flairs-2018
Query 2: Decoding the MAP-Tuple

5. \( b' = \arg \max_a P(b | a') \times P(d' | b, a') \times P(e' | b, c') \)

4. \( c' = \arg \max_a P(c | a') \times h^B(a', d', c, e') \)

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: \( P(b|a) \quad P(d|b,a) \quad P(e|b,c) \)

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

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

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

A: \( P(a) \quad h^E(a) \)

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 \(d\)

\(r = \text{number of functions}\)

The effect of the ordering:

“Moral” graph

\[
\begin{align*}
&\text{Finding the smallest induced width is hard!}
\end{align*}
\]
Query 3: Finding the Marginal Map (MMAP)

Bucket Elimination

\[ X_M = \{A, D, E\} \]
\[ X_S = \{B, C\} \]
\[ \max_{X_M} \sum_{X_S} P(X) \]

MAP* is the marginal MAP value

[Dechter, 1999]
Bucket Elimination for MMAP

Bucket Elimination

\[ X_M = \{ A, D, E \} \]
\[ X_S = \{ B, C \} \]
\[ \max_{X_M} \sum_{X_S} P(X) \]

constrained elimination order

\[ w^* = 4 \]

exact

\[ \max_X \sum_Y \phi \leq \sum_Y \max_X \phi \]

unconstrained elimination order

\[ w^* = 2 \]

Complexity exponential in the constrained induced-width

Dechter, Flairs-2018
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

Dechter, Flairs-2018
AND/OR vs. OR

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

Dechter, Flairs-2018
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

(b) DFS tree
height=3

(c) pseudo-tree
height=2

(d) Chain
height=6

\[ h \leq w \ast \log n \]
Cost of a Solution Tree

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)

\[
P(E | A, B)\quad P(B | A)\quad P(C | A)\quad P(A)
\]

<table>
<thead>
<tr>
<th>A</th>
<th>B</th>
<th>E=0</th>
<th>E=1</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0</td>
<td>.4</td>
<td>.6</td>
</tr>
<tr>
<td>0</td>
<td>1</td>
<td>.5</td>
<td>.5</td>
</tr>
<tr>
<td>1</td>
<td>0</td>
<td>.7</td>
<td>.3</td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>.2</td>
<td>.8</td>
</tr>
</tbody>
</table>

Evidence: E=0

\[
P(D=1, E=0) = ?
\]

\[
\prod_{n' \in \text{children}(n)} v(n')
\]

\[
\sum_{n \in \text{children}(n)} w(n, n') v(n')
\]

Value of node = updated belief for sub-problem below

**AND node: product**

**OR node: Marginalization by summation**

Dechter, Flairs-2018
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

Dechter, Flairs-2018
Merging Based on Context

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

\[ \text{context}(X) = \text{ancestors of } X \text{ in the pseudo tree that are connected to } X, \text{ or to descendants of } X \]
Answering Queries: Sum-Product (Belief Updating)

\[ P(E | A, B) \]
\[
\begin{array}{ccc}
A & B & E=0 & E=1 \\
0 & 0 & .4 & .6 \\
0 & 1 & .5 & .5 \\
1 & 0 & .7 & .3 \\
1 & 1 & .2 & .8 \\
\end{array}
\]

Evidence: E=0

\[ P(B | A) \]
\[
\begin{array}{ccc}
A & B=0 & B=1 \\
0 & .4 & .6 \\
1 & .1 & .9 \\
\end{array}
\]

\[ P(C | A) \]
\[
\begin{array}{ccc}
A & C=0 & C=1 \\
0 & .2 & .8 \\
1 & .7 & .3 \\
\end{array}
\]

\[ P(A) \]
\[
\begin{array}{c}
A & P(A) \\
0 & .6 \\
1 & .4 \\
\end{array}
\]

Result: \( P(D=1, E=0) \)

Cache table for D

Context

Evidence: D=1

\[ P(D | B, C) \]

\[ \text{Dechter, Flairs-2018} \]
Potential search spaces

- **Full OR search tree**: 126 nodes
- **Context minimal OR search graph**: 28 nodes
- **Full AND/OR search tree**: 54 AND nodes
- **Context minimal AND/OR search graph**: 18 AND nodes

Computes any query:
- Constraint satisfaction
- Optimization (MAP)
- Marginal (P(e))
- **Marginal map** (P(e))

Any query is best computed over the c-minimal AO search space.
The Impact of the Pseudo-Tree

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

W=4, h=8

W=5, h=6

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

(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

Decomposition Bounds

Anytime Algorithms.
Mini-Bucket Elimination (MAP)

Split a bucket into mini-buckets $\rightarrow$ bound complexity

bucket ($X$) =

$$
\begin{align*}
&\left\{ f_1, f_2, \ldots f_r, f_{r+1}, \ldots f_n \right\} \\
\lambda_X(\cdot) = \max_x \prod_{i=1}^{n} f_i(x, \ldots) \\
\lambda_{X,1}(\cdot) = \max_x \prod_{i=1}^{r} f_i(x, \ldots) \\
\lambda_{X,2}(\cdot) = \max_x \prod_{i=r+1}^{n} f_i(x, \ldots)
\end{align*}
$$

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

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

Dechter, Flairs-2018
Mini-Bucket Elimination (MAP)

\[ \lambda_{B\rightarrow C}(a, c) = \max_b f(a, b) \cdot f(b, c) \]
\[ \lambda_{B\rightarrow D}(d, e) = \max_b f(b, d) \cdot f(b, e) \]
\[ \lambda_{C\rightarrow E}(a, e) = \max_c \ldots \]

\[ U = \text{upper bound} \]

[Dechter & Rish 2003]
Mini-Bucket Decoding (MAP)

\[ b^* = \arg \max_b f(a^*, b) \cdot f(b, c^*) \cdot f(b, d^*) \cdot f(b, e^*) \]

\[ c^* = \arg \max_c f(c, a^*) \cdot f(c, e^*) \cdot \lambda_{B \rightarrow C}(a^*, c) \]

\[ d^* = \arg \max_d f(a^*, d) \cdot \lambda_{B \rightarrow D}(d, e^*) \]

\[ e^* = \arg \max_e \lambda_{C \rightarrow E}(a^*, e) \cdot \lambda_{D \rightarrow E}(a^*, e) \]

\[ a^* = \arg \max_a f(a) \cdot \lambda_{E \rightarrow A}(a) \]

Greedy configuration = lower bound

U = upper bound
Properties of Mini-Bucket Elimination

- Bounding from above and below
  - Relaxation upper bound by mini-bucket
  - $i = 2$, $i = 5$, $i = 10$, $i = 20$
  - MAP
  - 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

Dechter, Flairs-2018
Likelihood Queries: Bounded Inference

\[ Z = \sum \prod_{x_A, x_B, x_\alpha} \phi_\alpha \]

Marginal Map: Bounded Inference

\[ \mathbf{x}_B^* = \arg \max_{\mathbf{x}_B} \sum_{\mathbf{x}_A} \prod_{\alpha} \psi(\mathbf{x}_\alpha) \]
Decomposition Bounds for Sum

• Generalize technique to sum via Holder’s inequality:

\[ F(x) = f_1(x) \cdot f_2(x) \]

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

• Define the weighted (or powered) sum:

\[ \sum_{x_1}^{w_1} f(x_1) = \left[ \sum_{x_1} f(x_1)^{\frac{1}{w_1}} \right]^{w_1} \]

  – “Temperature” interpolates between sum & max:

  – Different weights do not commute:

\[ \lim_{w \to 0^+} \sum_{x} f(x) = \max_{x} f(x) \]

\[ \sum_{x_1}^{w_1} \sum_{x_2}^{w_2} f(x_1, x_2) \neq \sum_{x_2}^{w_2} \sum_{x_1}^{w_1} f(x_1, x_2) \]
Weighted Mini-bucket (for Sum)

\[
\lambda_{B \rightarrow C} = \sum_b w_{B1} f(a, b) \cdot f(b, c)
\]

\[
\lambda_{B \rightarrow D} = \sum_b w_{B2} f(b, d) \cdot f(b, e)
\]

\[
\lambda_{C \rightarrow E} = \sum_c f(c, a) \cdot f(c, e) \cdot \lambda_{B \rightarrow C}
\]

\[
w_{B1} + w_{B2} = 1
\]

Compute downward messages using weighted sum

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

[Dechter, Flairs-2018]

[Liu & Ihler 2011]
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)

Modify the individual functions
- but -
keep the sum or product of functions unchanged
Tightening the Bound

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

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

Reparameterization:
\[
\forall j : \sum_{\alpha \ni j} \lambda_{j \to \alpha}(x_j) = 0
\]
Tightening the Bound

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

Reparameterization:
\[ \forall j : \sum_{\alpha \ni j} \lambda_{j \rightarrow \alpha}(x_j) = 0 \]

\[
\log f(x^*) = \max_x \sum_{\alpha} \theta_{\alpha}(x_{\alpha}) \leq \min_{\{\lambda_{i \rightarrow \alpha}\}} \sum_{\alpha} \max_{x_{\alpha}} \left[ \theta_{\alpha}(x_{\alpha}) + \sum_{i \in \alpha} \lambda_{i \rightarrow \alpha}(x_i) \right]
\]

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.

Dechter, Flairs-2018
Weighted Mini-Bucket with Moment Matching (WMB-MM)

- 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

Join graph:

- B: \{A,B,C\} → \{A\} → \{B,D,E\} (w_1)
- C: \{A,C,E\} → \{A\} → \{D,E\}
- D: \{A,E\} → \{A\} → \{A,D,E\}
- E: \{A,E\} → \{A\}
- A: \{A\}

U = upper bound

[Ihler et al. 2012]
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

Dechter, Flairs-2018
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

Dechter, Flairs-2018
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

- Search $\rightarrow$ exploiting structure

Anytime algorithms.

Heuristics
To guide AND/OR search

Context minimal AND/OR search graph

decomposition bounds

MAP
Static Mini-Bucket Heuristics

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

\[
\tilde{h}(\hat{a}, \hat{e}, D) = \lambda_{C \rightarrow E}(\hat{a}, \hat{e}) + f(\hat{a}, D) + \lambda_{B \rightarrow D}(D, \hat{e})
\]
(admissible: \( \tilde{h}(\hat{a}, \hat{e}, D) \leq h^*(\hat{a}, \hat{e}, D) \))

cost so far:
\[
g(\hat{a}, \hat{e}) = f(A = \hat{a})
\]

Dechter, Flairs-2018
AND/OR Search for Marginal MAP

MAP variables
SUM variables
constrained pseudo tree

primal graph

\[ f(a, b) \quad f(b, c) \quad f(b, d) \]

\[ \lambda_{B \rightarrow C}(a, c) \quad f(a, c) \quad f(c, e) \]

\[ \lambda_{B \rightarrow D}(d, e) \]

\[ \lambda_{C \rightarrow E}(a, e) \quad \lambda_{D \rightarrow E}(a, e) \]

\[ f(a) \quad \lambda_{E \rightarrow A}(a) \]

[Marinescu, Dechter and Ihler, 2014]
AO search for MAP winning
UAI Probabilistic Inference Competitions

- 2006: (aolib)
- 2008: (aolib)
- 2011: (daoopt)
- 2014: (daoopt)

MPE/MAP

MMAP

Dechter, Flairs-2018
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)

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

**Weighted A* search** [Pohl 1970]
- non-admissible heuristic
- Evaluation function:
  \[ f(n) = g(n) + w \cdot h(n) \]
- Guaranteed \( w \)-optimal solution, cost \( C \leq w \cdot C^* \)

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”]
[Marinescu, Ihler, Dechter: IJCAI-2018 “Stochastic Anytime Search for Bounding Marginal MAP”]

(314,3,317,56,248) (n,k,c,w*,h) (375,3,378,64,302)

2 blocks, T=5 and 6

Algorithms:
UBFS
ANYLDFS
AnySBFS

(1134,3,1044,173,908) (2161,3,2168,302,1484)

6 blocks T=8

7 blocks T=8

6 blocks T=8
Conclusion:
Search Collaborates with Inference

- Inference: message-passing on cluster-tree

- 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