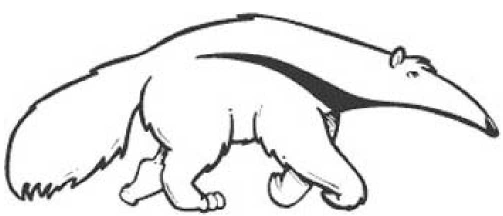
Variational algorithms for marginal MAP

Alexander Ihler

UC Irvine

CIOG Workshop

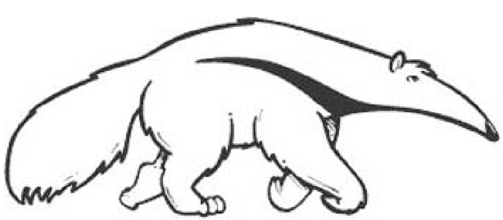
November 2011



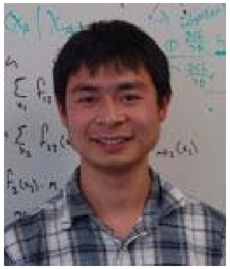
Variational algorithms for marginal MAP

Alexander Ihler
UC Irvine

CIOG Workshop
November 2011



Work with Qiang Liu



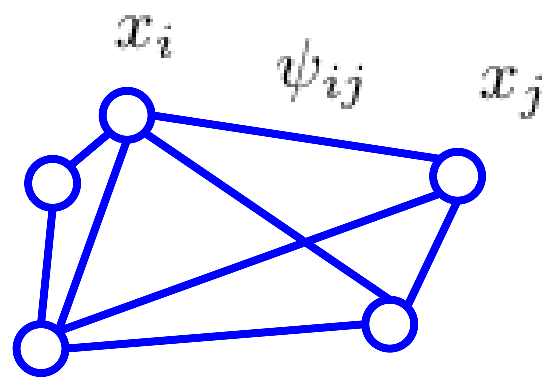
BREN:ICS
INFORMATION AND COMPUTER SCIENCES

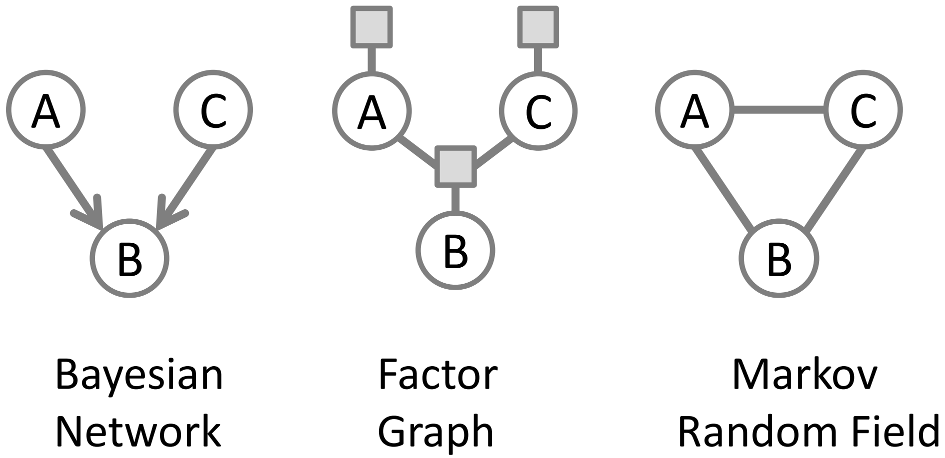
UNIVERSITY of CALIFORNIA IRVINE

Graphical models

$$p(\underline{x}) = p(x_1, x_2, \ldots, x_N)$$
$$= \frac{1}{Z} \prod_{\alpha} \psi_\alpha(x_\alpha)$$

where $Z = \sum_{\boldsymbol{x}} \prod_{\alpha \in \mathcal{I}} \psi(\boldsymbol{x}_\alpha)$ (partition function)





Types of queries

- Maximum a posterior (MAP) query
  - wCSPs, minimum energy configurations

$$\mathbf{x}^* = \arg\max_{\mathbf{x}} \prod_{\alpha} \psi_\alpha(x_\alpha)$$

Types of queries

- Maximum a posterior (MAP) query
  - wCSPs, minimum energy configurations

$$\mathbf{x}^* = \arg\max_{\mathbf{x}} \prod_{\alpha} \psi_\alpha(x_\alpha)$$

- Summation queries
  - Partition function, #CSP

$$Z = \sum_{\mathbf{x}} \prod_{\alpha} \psi_\alpha(x_\alpha)$$

Types of queries

- Maximum a posterior (MAP) query
  - wCSPs, minimum energy configurations

$$\mathbf{x}^* = \arg\max_{\mathbf{x}} \prod_\alpha \psi_\alpha(x_\alpha)$$

- Summation queries
  - Partition function, #CSP

$$Z = \sum_{\mathbf{x}} \prod_\alpha \psi_\alpha(x_\alpha)$$

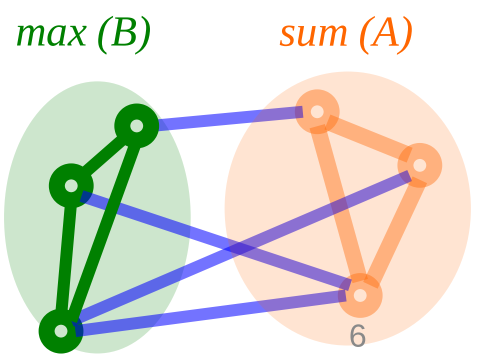
- Mixed queries
  - Combine more than one elimination operator

- Marginal-MAP

$$\mathbf{x}_B^* = \arg\max_{\mathbf{x}_B} \sum_{\mathbf{x}_A} \prod_\alpha \psi_\alpha(x_\alpha)$$

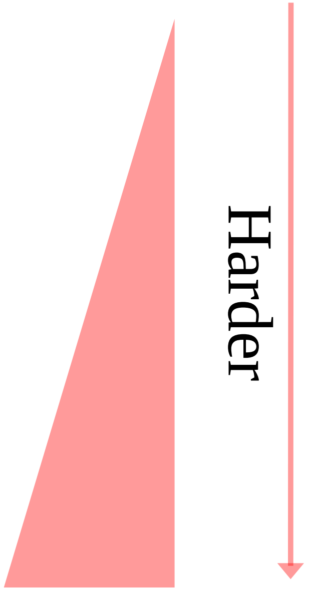
where $A \cup B = V$

- and many others…



Three type of queries

Max-Inference	$\max_{\mathbf{x}} \prod_{\alpha} \psi_\alpha(x_\alpha)$
Sum-Inference	$\sum_{\mathbf{x}} \prod_{\alpha} \psi_\alpha(x_\alpha)$
Mixed-Inference	$\max_{\mathbf{x}_B} \sum_{\mathbf{x}_A} \prod_{\alpha} \psi_\alpha(x_\alpha)$



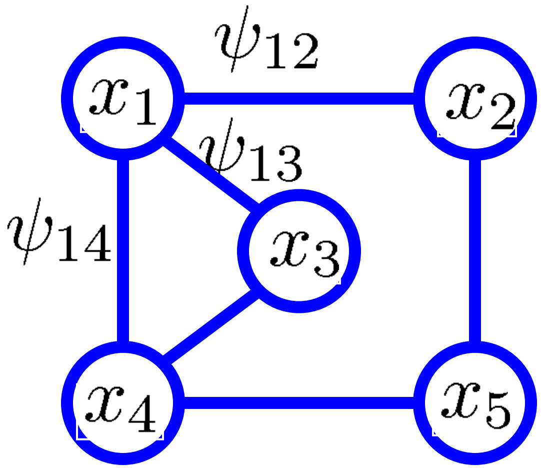
- **NP-hard**: exponentially many terms
- We will focus on **approximation** algorithms

Variable Elimination (exact sum)

$$p(\boldsymbol{x}) = \frac{1}{Z}\,\psi_{12}(x_1, x_2)\psi_{13}(x_1, x_3)\cdots$$

$$\sum_{x_1 \ldots x_5} \psi_{12}\psi_{13}\psi_{14}\psi_{25}\psi_{34}\psi_{45}$$

$$= \sum_{x_2 \ldots x_5} \psi_{25}\psi_{34}\psi_{45} \sum_{x_1} \psi_{12}\psi_{13}\psi_{14}$$



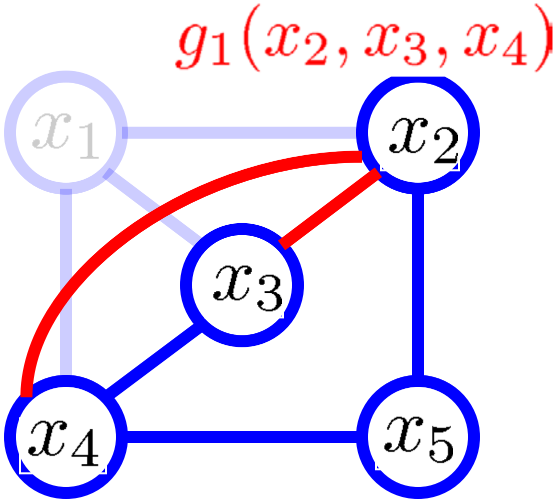
$$\begin{aligned}
B_1 &: \{\psi_{12},\ \psi_{13},\ \psi_{14}\ \} \\
B_2 &: \{\psi_{25}\ \} \\
B_3 &: \{\psi_{34}\ \} \\
B_4 &: \{\psi_{45}\ \} \\
B_5 &: \{\quad \}
\end{aligned}$$

Variable Elimination (exact sum)

$$\sum_{x_1 \dots x_5} \psi_{12}\psi_{13}\psi_{14}\psi_{25}\psi_{34}\psi_{45}$$

$$= \sum_{x_2 \dots x_5} \psi_{25}\psi_{34}\psi_{45} \sum_{x_1} \psi_{12}\psi_{13}\psi_{14}$$

$$= \sum_{x_3 \dots x_5} \psi_{34}\psi_{45} \sum_{x_2} \psi_{25} \; g_1(x_2, x_3, x_4)$$



$B_1 : \{\psi_{12}, \; \psi_{13}, \; \psi_{14} \}$

$B_2 : \{\psi_{25}, g_1 \}$

$B_3 : \{\psi_{34} \}$

$B_4 : \{\psi_{45} \}$

$B_5 : \{ \; \}$

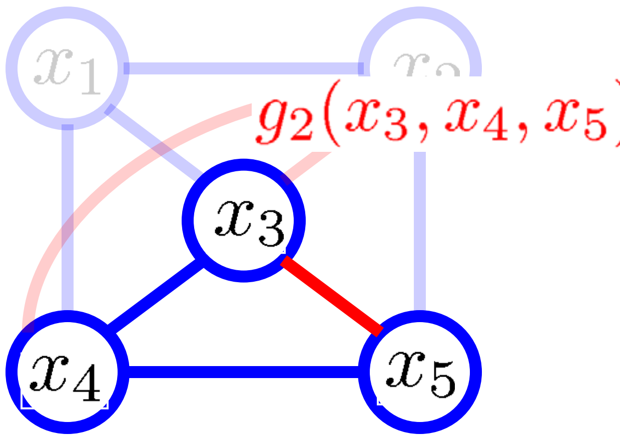
Variable Elimination (exact sum)

$$\sum_{x_1 \ldots x_5} \psi_{12}\psi_{13}\psi_{14}\psi_{25}\psi_{34}\psi_{45}$$

$$= \sum_{x_2 \ldots x_5} \psi_{25}\psi_{34}\psi_{45} \sum_{x_1} \psi_{12}\psi_{13}\psi_{14}$$

$$= \sum_{x_3 \ldots x_5} \psi_{34}\psi_{45} \sum_{x_2} \psi_{25} \; g_1(x_2, x_3, x_4)$$

$$= \sum_{x_3 \ldots x_5} \psi_{34}\psi_{45} \; g_2(x_3, x_4, x_5)$$



$B_1 : \{\psi_{12}, \; \psi_{13}, \; \psi_{14} \}$

$B_2 : \{\psi_{25}, g_1 \}$

$B_3 : \{\psi_{34}, g_2 \}$

$B_4 : \{\psi_{45} \}$

$B_5 : \{ \; \}$

Variable Elimination (exact sum)

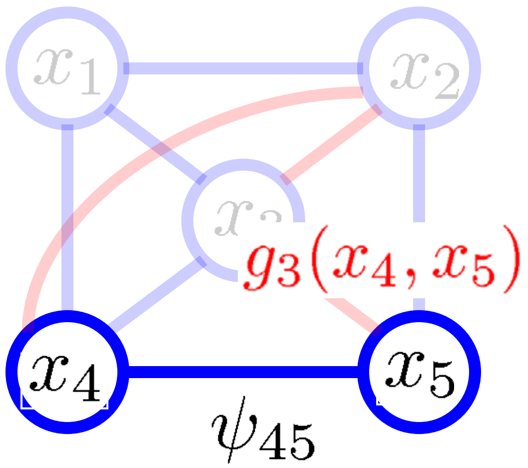
$$\sum_{x_1 \dots x_5} \psi_{12}\psi_{13}\psi_{14}\psi_{25}\psi_{34}\psi_{45}$$

$$= \sum_{x_2 \dots x_5} \psi_{25}\psi_{34}\psi_{45} \sum_{x_1} \psi_{12}\psi_{13}\psi_{14}$$

$$= \sum_{x_3 \dots x_5} \psi_{34}\psi_{45} \sum_{x_2} \psi_{25} \; g_1(x_2, x_3, x_4)$$

$$= \sum_{x_3 \dots x_5} \psi_{34}\psi_{45} \; g_2(x_3, x_4, x_5)$$

$$= \sum_{x_4, x_5} \psi_{45} \; g_3(x_4, x_5)$$



$B_1 : \{\psi_{12}, \; \psi_{13}, \; \psi_{14} \}$

$B_2 : \{\psi_{25}, g_1 \}$

$B_3 : \{\psi_{34}, g_2 \}$

$B_4 : \{\psi_{45}, g_3 \}$

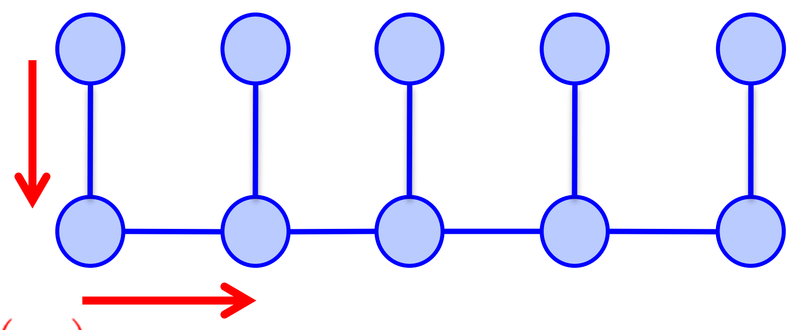
$B_5 : \{ \; \}$

Cost: exponential in the tree-width

Variable Elimination (exact sum / max)

- Interpretation as message-passing on trees
- Algorithm similar for max (dynamic programming)

$$g_1(x_2) = \sum_{x_1} \psi_{1,2}(x_1, x_2)$$

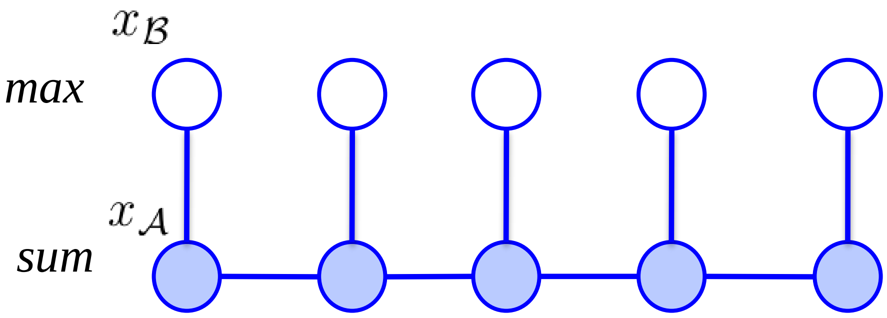


$$g_2(x_4) = \sum_{x_2} \psi_{2,4}(x_2, x_4) g_1(x_2)$$

Variable Elimination (exact mixed)

- Interpretation as message-passing on trees
- Algorithm similar for max (dynamic programming)

- Mixed-inference is harder!
  - Elimination orders are restricted: $\sum \max \neq \max \sum$

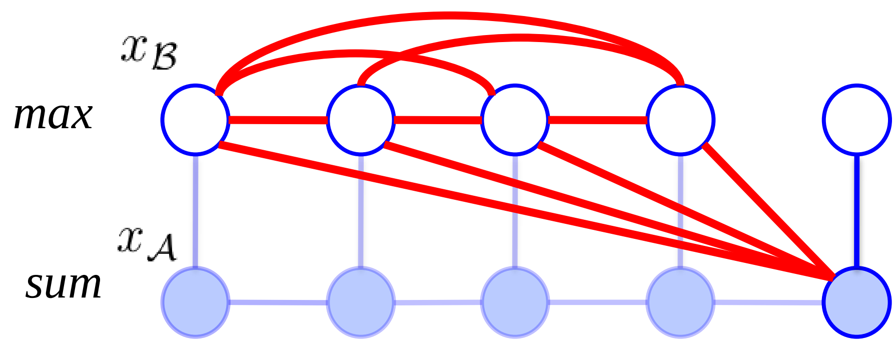


Example from D. Koller and N. Friedman (2009)

Variable Elimination (exact max /mixed)

- Interpretation as message-passing on trees
- Algorithm similar for max (dynamic programming)

- Mixed-inference is harder!
  - Elimination orders are restricted: $\sum \max \neq \max \sum$



Example from D. Koller and N. Friedman (2009)

Variational approaches

- Replace "elimination" with optimization over distributions

$$p(x^*) = \max_{q \in \mathbb{P}} \; \mathbb{E}_q[\log p(x)]$$

(maximum: q = $\mathbf{1}(x^*)$)

$\mathbb{P}$: set of joint distributions over x

Equivalently n terms of $b \in \mathcal{M}$, the "marginal polytope"

Variational approaches

- Replace "elimination" with optimization over distributions

$$p(x^*) = \max_{q \in \mathbb{P}} \mathbb{E}_q[\log p(x)]$$ (maximum: $q = \mathbf{1}(x^*)$)

$\mathbb{P}$: set of joint distributions over x

Equivalently n terms of $b \in \mathcal{M}$, the "marginal polytope"

$$\log Z = \max_{q \in \mathbb{P}} \mathbb{E}_q[\log \psi(x)] + H(x\,;\,q)$$ (maximum: $q = p$)

Variational approaches

- Replace "elimination" with optimization over distributions

$$p(x^*) = \max_{q \in \mathbb{P}} \; \mathbb{E}_q[\log p(x)]$$ (maximum: $q = \mathbf{1}(x^*)$)

$\mathbb{P}$: set of joint distributions over x

Equivalently n terms of $b \in \mathcal{M}$, the "marginal polytope"

$$\log Z = \max_{q \in \mathbb{P}} \; \mathbb{E}_q[\log \psi(x)] + H(x\,;\,q)$$ (maximum: $q = p$)

- Proof: $$D(q \| p) = \sum_x q(x) \log \left[\frac{q(x)}{\frac{1}{Z}\psi(x)} \right]$$ (Kullback–Leibler divergence)

$$= -H(x\,;\,q) - \mathbb{E}_q[\log \psi] + \log Z$$

$$\Rightarrow \log Z \geq \mathbb{E}_q[\log \psi] + H(x\,;\,q)$$ equal iff $p = q$

Variational approximations

- Replace $q \in P$ and $H(q)$ with simpler approximations

$$p(x^*) = \max_{q \in \mathbb{P}} \ \mathbb{E}_q[\log \psi(x)]$$

$$\log Z = \max_{q \in \mathbb{P}} \ \mathbb{E}_q[\log \psi(x)] + H(x\,;\,q)$$

Variational approximations

- Replace $q \in P$ and $H(q)$ with simpler approximations

$$p(x^*) = \max_{q \in \mathbb{P}} \; \mathbb{E}_q[\log \psi(x)]$$

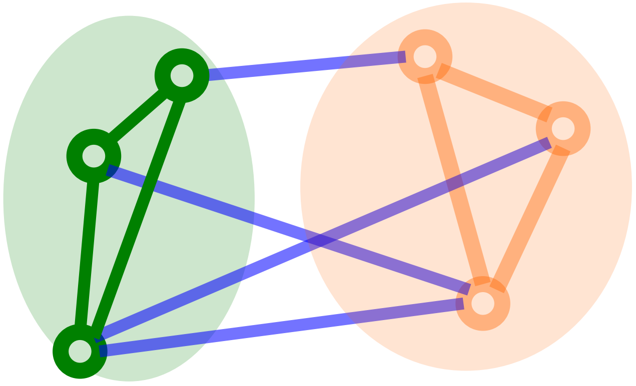
$$\log Z = \max_{q \in \mathbb{P}} \; \mathbb{E}_q[\log \psi(x)] + H(x\,;\,q)$$

- Algorithms & their properties:

	Method	distributions	entropy	value
Max:	Linear programming	$q \in \mathbb{L} \supseteq \mathbb{P}$	n/a	$\hat{p}_{lp} \geq p(x^*)$
Sum:	Mean field	$\{q = \prod q_i(x_i)\} \subseteq \mathbb{P}$	exact	$Z_{mf} \leq Z$
	Belief propagation	$q \in \mathbb{L} \supseteq \mathbb{P}$	$H_\beta \approx H(q)$	$Z_\beta \approx Z$
	Tree-reweighted	$q \in \mathbb{L} \supseteq \mathbb{P}$	$H_{tr} \geq H(q)$	$Z_{tr} \geq Z$

Variational methods for marginal MAP

$$(x^*), p^* \;=\; (\arg)\max_{x_B} \sum_{x_A} \psi(x)$$



"max part" $\log p^* = \max_{x_B} \log Z_A(x_B)$

"sum part" $Z_A(x_B) = \sum_{x_A} \psi(x)$

Apply the same approach to each part:

$$\begin{aligned} \log p^* &\ge \mathbb{E}_q[\log Z_A(x_B)] \\ &\ge \mathbb{E}_q[\log \psi(x)] + H(x_A | x_B \,;\, q) \end{aligned}$$

Variational methods for marginal MAP

$$p^* = \max_{x_B} \sum_{x_A} \psi(x) \quad \xrightarrow{\textit{dual}} \quad \max_{q \in \mathbb{P}} \; \mathbb{E}_q(\log \psi) + H(x_A | x_B \,;\, q)$$

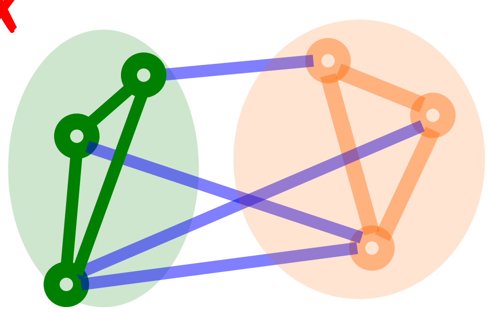
where $H(x_A|x_B) = H(x) - H(x_B)$

(Truncate the entropies of the max nodes)

- General framework for approximate algorithms
  - Truncated Bethe approximation
  - Truncated TRW approximation
  - Truncated mean field approximation

…

$H(x) = H(x_B) + H(x_A \mid x_B)$



Truncated free energy approximations

$$p^* = \max_{x_B} \sum_{x_A} \psi(x)$$

dual $\Longrightarrow$

$$\max_{q \in \mathbb{P}} \; \mathbb{E}_q(\log \psi) + H(x_A | x_B \,;\, q)$$

where $H(x_A|x_B) = H(x) - H(x_B)$

(Truncate the entropies of the max nodes)

- Truncated Bethe approximation

$$\hat{Z}_{bethe}(\tau, \log \psi) = \mathbb{E}_\tau[\log \psi] + \sum_{i \in A} H_i \; - \sum_{(ij) \in E_A \cup \partial_{AB}} I_{ij}.$$

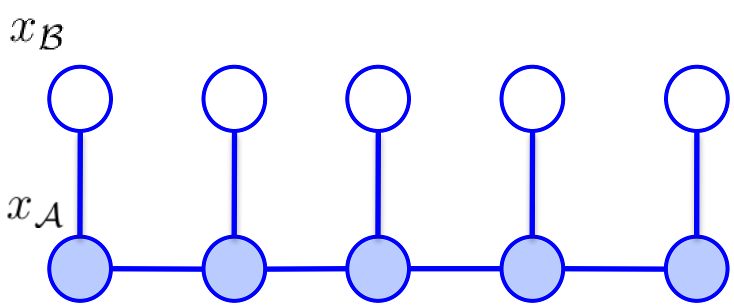
- Truncated tree-reweighted approximation

$$\hat{Z}_{trw}(\tau, \log \psi) = \mathbb{E}_\tau[\log \psi] + \sum_{i \in A} H_i \; - \sum_{(ij) \in E_A \cup \partial_{AB}} \rho_{ij} I_{ij}.$$

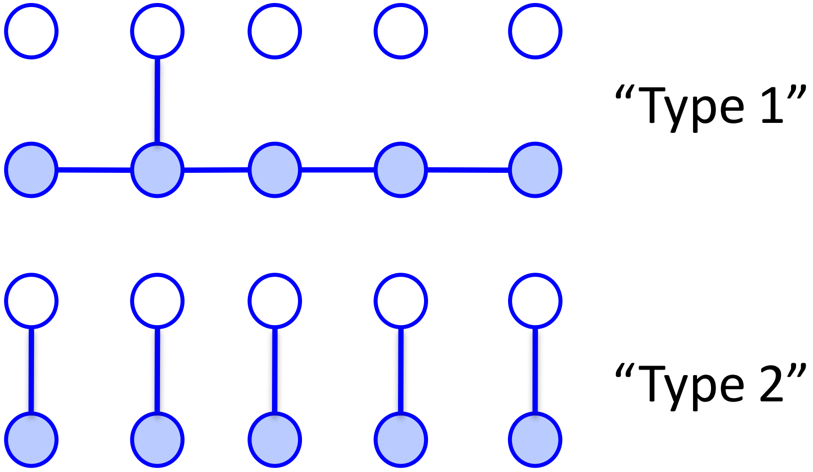
A-B trees

- In sum or max-inference, trees are "tractable" subproblems
- In mixed inference, they may not be

- A-B trees
  - Extend the notion to mixed inference
  - Graph structure that remains a tree during elimination



Example from D. Koller and N. Friedman (2009)



Designing message passing algorithms

- Can write as "generic" weighted objective

$$\hat{Z}(\tau, w, \log\psi) = \mathbb{E}_\tau[\log\psi] + \sum_{i\in A} w_i H_i \;-\; \sum_{(ij)\in E_A \cup \partial_{AB}} w_{ij} I_{ij}.$$

- Derive messages

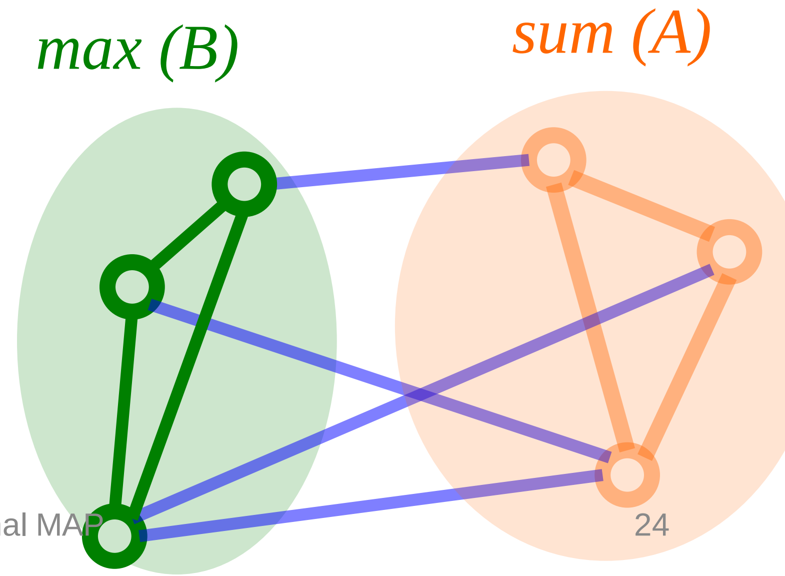
  (minor generalization of TRW)

- Take limit as some weights $\rightarrow$ 0

  (minor generalization of Weiss et al. 2007)

- Can opt to take limit "directly"

  on message update equations

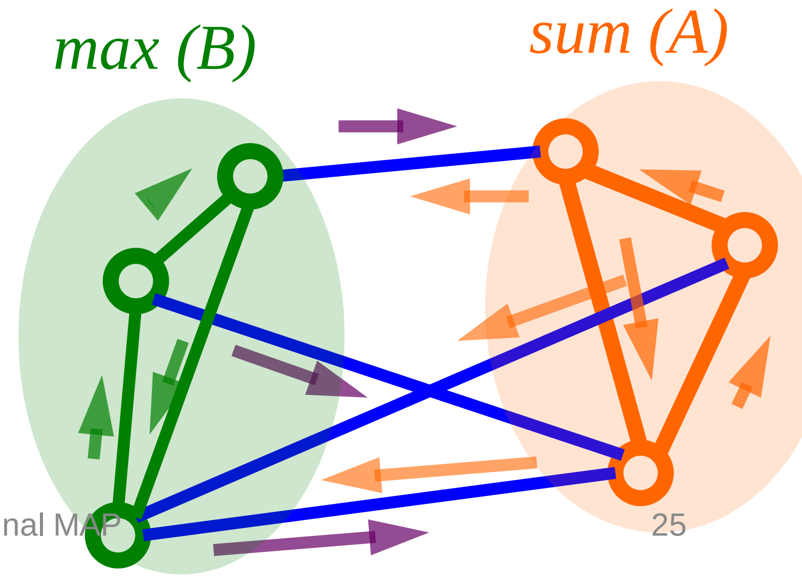


"Mixed" product message passing

A → A ∪ B $\quad m_{i\to j} \leftarrow \Big[\sum_{x_i} (\psi_i m_{\sim i}) \big(\frac{\psi_{ij}}{m_{j\to i}}\big)^{1/\rho_{ij}}\Big]^{\rho_{ij}}$ Sum- product

B → B $\quad m_{i\to j} \leftarrow \max_{x_i} (\psi_i m_{\sim i})^{\rho_{ij}} \big(\frac{\psi_{ij}}{m_{j\to i}}\big)$ Max- product

B → A $\quad m_{i\to j} \leftarrow \Big[\sum_{x_i \in \arg\max\{\psi_i m_{\sim i}\}} \big(\frac{\psi_{ij}}{m_{j\to i}}\big)^{1/\rho_{ij}}\Big]^{\rho_{ij}}$ Match max and sum



“Mixed” product message passing

Satisfies a reparameterization property,

$$p(x) \propto \prod_i b_i(x_i) \prod_{i,j} \frac{b_{ij}(x_i, x_j)}{b_i(x_i)\, b_j(x_j)}$$

where

$i \in A$ $$\sum_{x_i} b_{ij}(x_i, x_j) = b_j(x_j),$$ Sum- product

$i, j \in B \to B$ $$\max_{x_i} b_{ij}(x_i, x_j) = b_j(x_j),$$ Max- product

$i \in B$, $j \in A$ $$\sum_{x_i \in \arg\max b_i} b_{ij}(x_i, x_j) = b_j(x_j),$$ Match max and sum

Can use this to show local optimality properties similar to max-product

Variational methods for marginal MAP

$$p^* = \max_{x_B} \sum_{x_A} \psi(x)$$

dual $\Longrightarrow$

$$\max_{q \in \mathbb{P}} \; \mathbb{E}_q(\log \psi) + H(x_A | x_B \, ; \, q)$$

where $H(x_A | x_B) = H(x) - H(x_B)$

(Truncate the entropies of the max nodes)

"Double-loop" algorithms (CCCP & similar):

- Example: Truncated Bethe approximation

Solve summation problem: $\log Z = \max_{q \in \mathbb{P}} \; \mathbb{E}_q[\log \psi(x)] + \hat{H}_\beta(x)$

Variational methods for marginal MAP

$$p^* = \max_{x_B} \sum_{x_A} \psi(x) \quad \xrightarrow{\textit{dual}} \quad \max_{q \in \mathbb{P}} \; \mathbb{E}_q(\log \psi) + H(x_A | x_B \, ; \, q)$$

where $H(x_A|x_B) = H(x) - H(x_B)$

(Truncate the entropies of the max nodes)

"Double-loop" algorithms (CCCP & similar):

- Example: Truncated Bethe approximation

Solve summation problem: $\log Z = \max_{q \in \mathbb{P}} \; \mathbb{E}_q[\log \psi(x)] + \hat{H}_\beta(x)$

Remove excess entropy:

$$\begin{aligned}\mathbb{E}[\log \psi(x)] \longleftarrow & \mathbb{E}[\log \psi] - \hat{H}_\beta(x_B) \\ = & \mathbb{E}[\log \psi(x) + \log q(x_B)]\end{aligned}$$

Variational methods for marginal MAP

$$p^* = \max_{x_B} \sum_{x_A} \psi(x)$$

dual $\Rightarrow$

$$\max_{q \in \mathbb{P}} \; \mathbb{E}_q(\log \psi) + H(x_A | x_B \, ; \, q)$$

where $H(x_A|x_B) = H(x) - H(x_B)$

(Truncate the entropies of the max nodes)

"Double-loop" algorithms (CCCP & similar):

- Example: Truncated Bethe approximation

Solve summation problem: $\log Z = \max_{q \in \mathbb{P}} \; \mathbb{E}_q[\log \psi(x)] + \hat{H}_\beta(x)$

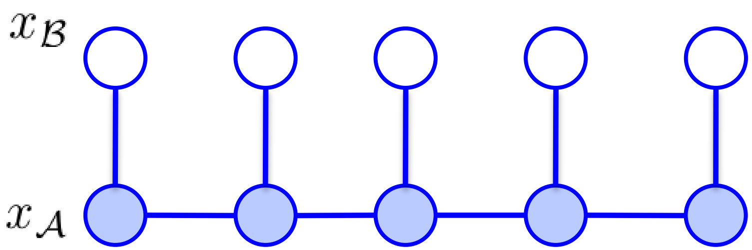
Remove excess entropy: $\mathbb{E}[\log \psi(x)] \longleftarrow \mathbb{E}[\log \psi] - \hat{H}_\beta(x_B)$

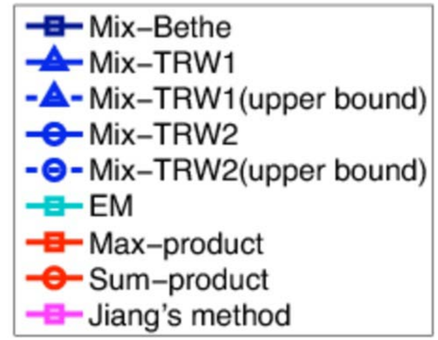
$$= \mathbb{E}[\log \psi(x) + \log q(x_B)]$$

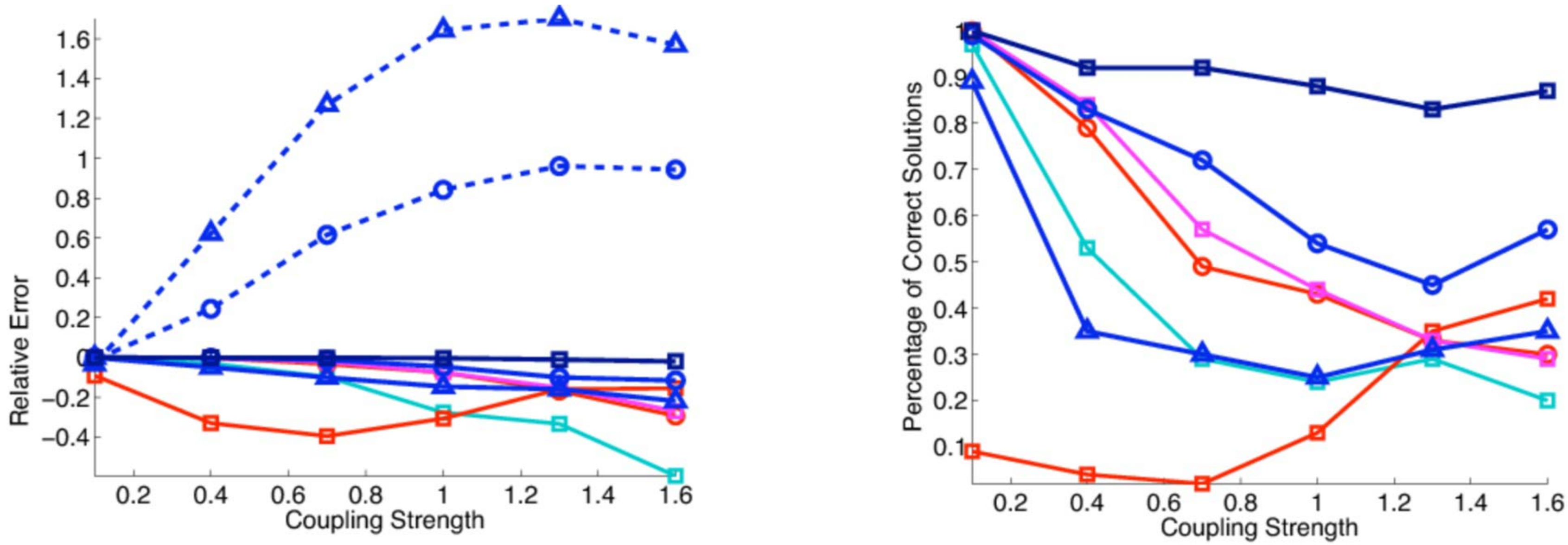
Iterate:

(a bit like annealing – makes the function "sharper")

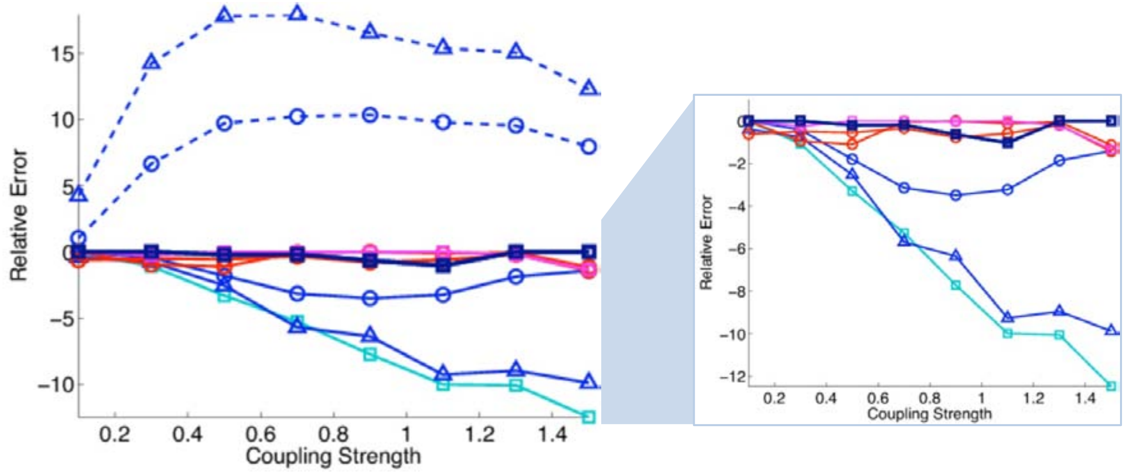
Experiments: trees



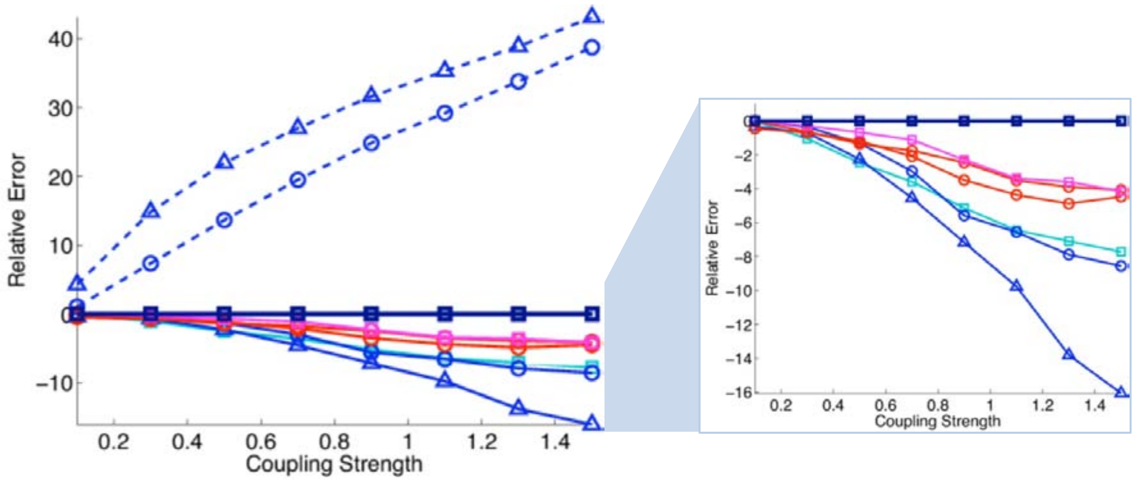




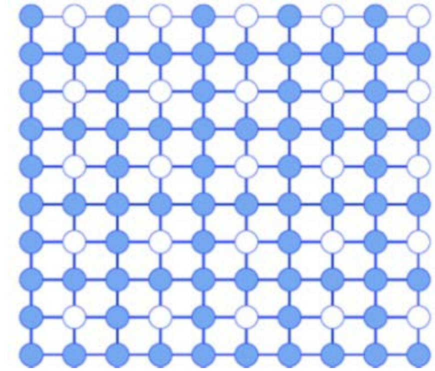
Experiments: cycles

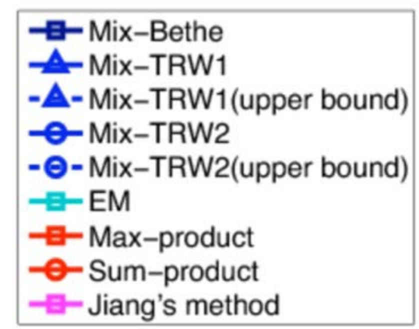


Attractive



Mixed





Conclusions

- Consider “mixed” inference tasks (marginal MAP)
- Derive a variational framework
- Develop analogues of Bethe, TRW, etc.
  - Approximations and bounds
- Develop algorithms
  - Message passing & double-loop methods

- Directions
  - Extend to more general mixed problems
  - Algorithmic improvements

Conclusions

Thanks!

- Consider “mixed” inference tasks (marginal MAP)
- Derive a variational framework
- Develop analogues of Bethe, TRW, etc.
  – Approximations and bounds
- Develop algorithms
  – Message passing & double-loop methods

- Directions
  – Extend to more general mixed problems
  – Algorithmic improvements