Approximate Inference in Graphical Models using LP Relaxations

David Sontag

Based on joint work with Tommi Jaakkola, Amir Globerson, Talya Meltzer, and Yair Weiss

Massachusetts Institute of Technology
MAP in Undirected Graphical Models

Real-world problems:

Protein backbone → Protein design

Side-chains → Stereo vision

 MAP in Undirected Graphical Models

\[
\Pr(x; \theta) \propto \exp \left( \sum_{(i,j) \in E} \theta_{ij}(x_i, x_j) \right)
\]

Find most likely assignment:

\[
x_{\text{map}} = \arg \max_x \sum_{(i,j) \in E} \theta_{ij}(x_i, x_j)
\]
How to solve MAP?

- MAP is known to be NP-hard (e.g., MAP on binary MRFs is equivalent to Max-Cut)
- Real-world MAP problems are not necessarily as hard as theoretical worst case
How to solve MAP?

- New toolkit: Message-passing algorithms based on linear programming relaxations
  (Schlensinger ’76, Kolmogorov & Wainwright ‘05, Vontobel & Koetter ‘06, Johnson et al. ’07, Komodakis et al. ‘07, Globerson & Jaakkola ’08…)

- Solves exactly when LP relaxation is tight: trees, binary submodular MRFs, and matchings

- In practice, we seldom have these structures

- By tightening the relaxation (problem specific), we can solve hard real-world problems, exactly
We can formulate the MAP problem as a linear program

$$\max_x \sum_{(i,j) \in E} \theta_{ij}(x_i, x_j) = \max_x \sum_{\mu} \sum_{(i,j) \in E} \delta(x_i, x_j) \theta_{ij}(x_i, x_j)$$

where the variables $\mu_{ij}$ are defined over edges.

The marginal polytope constrains the $\mu_{ij}$ to be marginals of some distribution:

$$\mathcal{M}(G) = \{ \mu \mid \exists \Pr(\mathbf{x}; \mathbf{\theta}) \text{ s.t. } \mu_{ij}(x_i, x_j) = \Pr(x_i, x_j; \mathbf{\theta}) \}$$

Very many constraints!
Relaxing the MAP LP

\[
\max_x \sum_{(i,j) \in E} \theta_{ij}(x_i, x_j) = \max_{\mu \in \mathcal{M}(G)} \sum_{(i,j) \in E} \sum_{x_i, x_j} \mu_{ij}(x_i, x_j) \theta_{ij}(x_i, x_j)
\]
Relaxing the MAP LP

\[
\max_x \sum_{(i,j) \in E} \theta_{ij}(x_i, x_j) \leq \max_{\mu \in S} \sum_{(i,j) \in E} \sum_{x_i, x_j} \mu_{ij}(x_i, x_j) \theta_{ij}(x_i, x_j)
\]

Such that \( \mathcal{M}(G) \subseteq S \)

Simplest outer bound:

\[
\sum_{x_i, x_j} \mu_{ij}(x_i, x_j) = 1
\]
Tightening the LP

\[
\max_x \sum_{(i,j) \in E} \theta_{ij}(x_i, x_j) \leq \max_{\mu \in S} \sum_{(i,j) \in E} \sum_{x_i, x_j} \mu_{ij}(x_i, x_j) \theta_{ij}(x_i, x_j)
\]

Such that \( M(G) \subseteq S \)

\[
\sum_{x_2} \mu_{12}(x_1, x_2) = \sum_{x_4} \mu_{14}(x_1, x_4)
\]  

\textit{Partial pairwise consistency}
Tightening the LP

\[
\max_x \sum_{(i,j) \in E} \theta_{ij}(x_i, x_j) \leq \max_{\mu \in S} \sum_{(i,j) \in E} \sum_{x_i, x_j} \mu_{ij}(x_i, x_j) \theta_{ij}(x_i, x_j)
\]

Such that \( \mathcal{M}(G) \subseteq S \)

Objective

\[
\sum_{x_1} \mu_{14}(x_1, x_4) = \sum_{x_5} \mu_{45}(x_4, x_5)
\]

\( \text{Partial pairwise consistency} \)
Tightening the LP

\[
\max_x \sum_{(i,j) \in E} \theta_{ij}(x_i, x_j) \leq \max_{\mu \in S} \sum_{(i,j) \in E} \sum_{x_i, x_j} \mu_{ij}(x_i, x_j) \theta_{ij}(x_i, x_j)
\]

Such that \( \mathcal{M}(G) \subseteq S \)

Pairwise consistency

\[
\sum_{x_i} \mu_{ij}(x_i, x_j) = \sum_{x_k} \mu_{ij}(x_j, x_k)
\]
Tightening the LP

\[
\max_x \sum_{(i,j) \in E} \theta_{ij}(x_i, x_j) \leq \max_{\mu \in S} \sum_{(i,j) \in E} \sum_{x_i, x_j} \mu_{ij}(x_i, x_j) \theta_{ij}(x_i, x_j)
\]

Such that \( \mathcal{M}(G) \subseteq S \)

\[ \sum_{x_k} \mu_{ijk}(x_i, x_j, x_k) = \mu_{ij}(x_i, x_j) \]

\( \text{Triplet consistency} \)
Tightening the LP

\[
\max_x \sum_{(i,j) \in E} \theta_{ij}(x_i, x_j) \leq \max_{\mu \in S} \sum_{(i,j) \in E} \sum_{x_i, x_j} \mu_{ij}(x_i, x_j) \theta_{ij}(x_i, x_j)
\]

Such that \( M(G) \subseteq S \)

\[
\sum_{x_k, x_l} \mu_{ijkl}(x_i, x_j, x_k, x_l) = \mu_{ij}(x_i, x_j)
\]

\{ Quadruplet consistency \}
Tightening the LP

\[
\max_x \sum_{(i,j) \in E} \theta_{ij}(x_i, x_j) \leq \max_{\mu \in S} \sum_{(i,j) \in E} \mu_{ij}(x_i, x_j) \theta_{ij}(x_i, x_j)
\]

Such that \( \mathcal{M}(G) \subseteq S \)

Great! But...

- Can we efficiently solve the LP?
- What clusters to add?
- How do we avoid re-solving?

Might be “lucky” and solve earlier
Our solution

- Can we efficiently solve the LP?
  - We work in one of the dual LPs (Globerson & Jaakkola ‘07)
  - Dual can be solved by an efficient message-passing algorithm
  - Corresponds to coordinate-descent algorithm

- What cluster to add next?
  - We propose a greedy bound minimization algorithm
  - Add clusters with guaranteed improvement – upper bound gets tighter

- How do we avoid re-solving?
  - “Warm start” of new messages using the old messages
Dual algorithm

1. Run message-passing
2. Decode assignment from messages
3. Choose a cluster to add to relaxation
4. Warm start: initialize new cluster messages

Is gap (dual obj – assignment val) small?  No.

Same objective value

Messages

Objective

Integer solution

Iteration

Dual

Done!

MAP
Dual algorithm
What cluster to add next?

\[
\sum_{e \in c} \max_{x_e} b_e(x_e) - \max_{x_c} \left[ \sum_{e \in c} b_e(x_e) \right]
\]
What cluster to add next?

\[ \sum_{e \in c} \max_{x_e} b_e(x_e) - \max_{x_c} \left[ \sum_{e \in c} b_e(x_e) \right] \]

\[ \max_{x_1, x_2, x_3} [b_{12}(x_1, x_2) + b_{23}(x_2, x_3) + b_{13}(x_1, x_3)] \]

\[
\begin{align*}
\max_{x_1, x_2} b_{12}(x_1, x_2) \\
\max_{x_1, x_3} b_{13}(x_1, x_3) \\
\max_{x_2, x_3} b_{23}(x_2, x_3)
\end{align*}
\]
What cluster to add next?

\[
\sum_{e \in c} \max_{x_e} b_e(x_e) - \max_{x_c} \left[ \sum_{e \in c} b_e(x_e) \right]
\]

\[3 \times 99 \quad \quad 2 \times 99 - 10\]

\[x_1 = 1 \quad \quad x_2 = 0 \quad \quad x_3 = 1\]

\[b_{ij}(x_i, x_j) = 99 \quad \text{if } x_i \neq x_j\]
\[b_{ij}(x_i, x_j) = -10 \quad \text{otherwise}\]

If dual decreases, there was frustration.
Coarsened cluster consistency

- Each new cluster requires adding a large number of LP variables $\mu_{ijk}(x_i, x_j, x_k)$ and constraints.
- Is it possible to use just a subset of these constraints?
- We give a new class of sparse cluster constraints, enforcing consistency on coarsened variables.

(Sontag, Globerson, Jaakkola, NIPS ‘08)
Experiments: Protein design

- Given protein’s 3D shape, choose amino-acids giving the most stable structure

  (MRFs from Yanover, Meltzer, Weiss ‘06)

- Each state corresponds to a choice of amino-acid and side-chain angle

- MRFs have 41-180 variables, each variable with 95-158 states

- Hard to solve
  - Very large treewidth
  - Many small cycles (20,000 triangles) and frustration
Primal LP, pairwise, is large

CPLEX can only run on 3: must move to dual!

(Yanover, Meltzer, Weiss, JMLR '06)
Protein design results

- Pairwise constraints solve only 2 of the 97 proteins
- Iteratively tightening relaxation with triplets, we **exactly solve 96** of the 97 proteins (!!!)
- Using the coarsened clusters, average time to solve 15 largest proteins is 1.5 hours
- Bound criterion finds the right constraints: Only 5 to 735 triplets needed to be added per problem
Coarsening clusters really helps
Related Work

- Similar ideas can be done directly in the primal
  - Selection criteria of constraint violation instead of bound minimization
  - (Sontag & Jaakkola ’08)

- Can also be applied to marginals
  - Guidance by bound on partition function rather than MAP value
  - Similar to region-pursuit algorithm for generalized BP (Welling UAI ’04)
Conclusions & Future Work

- New toolkit of message-passing algorithms based on dual LP relaxations
  + Iterative tightening of LP relaxation
  = Ability to solve interesting real world-problems

- More generally, when can we expect these MAP inference techniques to be successful?

- How should we do learning with approximate inference – in particular, with LP relaxations?