CSE 254: MAP estimation via agreement on (hyper)trees: Message-passing and linear programming approaches

A presentation by Evan Ettinger

November 11, 2005
Outline

Introduction
   Motivation and Background

Tree-Reparameterizations
   Definitions and Upper Bounds

LP Approaches
   IP to LP to Lagrangian Dual

Message-passing Approaches
   Two Algorithms

Summary
### Outline

#### Introduction
- Motivation and Background

#### Tree-Reparameterizations
- Definitions and Upper Bounds

#### LP Approaches
- IP to LP to Lagrangian Dual

#### Message-passing Approaches
- Two Algorithms

#### Summary
Motivation

- Want: MAP estimation on pairwise MRFs – i.e. configurations in the set $\arg\max_{x \in \chi^n} p(x; \theta)$.
- We limit ourselves to \textit{pairwise MRFs} and \textit{discrete} sets of labels, $\chi$. 

MAP estimation via agreement on (hyper)trees
MRFs as Exponential Family

- Remember that we can write any MRF in the exponential family form as:

\[ p(x; \theta) \propto \exp\{\langle \theta, \phi(x) \rangle \} \equiv \exp\{ \sum_{\alpha} \theta_\alpha \phi_\alpha(x) \} \]

- Here \( \mathcal{I} \) indexes over the vertices, edges and labels.

- We will use the overcomplete representation

\[ \{ \delta_j(x_s) \mid j \in \chi_s \} \quad \text{for} \ s \in V \]
\[ \{ \delta_j(x_s)\delta_k(x_t) \mid (j, k) \in \chi_s \times \chi_t \} \quad \text{for} \ (s, t) \in E \]

- Consequence: There are many exponential parameters corresponding to a given distribution (i.e. \( p(x; \theta) = p(x; \hat{\theta}) \) for \( \theta \neq \hat{\theta} \))
Marginal Distributions

- Using this representation we can now write marginal probabilities as:

\[
\mu_{s:j} = E[\delta_j(x_s)] := \sum_{x \in \chi^n} p(x; \theta)\delta_j(x_s)
\]

\[
\mu_{st:jk} = E[\delta_j(x_s)\delta_k(x_t)] := \sum_{x \in \chi^n} p(x; \theta)\delta_j(x_s)\delta_k(x_t)
\]

- We define the *marginal polytope* as:

\[
MARG(G) := \{ \mu \in \mathbb{R}^d \mid \mu_{s:j} = E_p[\delta_j(x_s)], \mu_{st:jk} = E_p[\delta_j(x_s)\delta_k(x_t)] \text{ for some } p(\cdot) \}
\]
MAP Estimation

- We can represent the exponential parameters of our sufficient statistic \( \phi(x) = \{\delta_j(x_s), \delta_j(x_s)\delta_k(x_t)\} \) by:

\[
\bar{\theta}_s(x_s) := \sum_{j \in x_s} \bar{\theta}_{s:j} \delta_j(x_s)
\]

\[
\bar{\theta}_{st}(x_s, x_t) := \sum_{(j, k) \in x_s \times x_t} \bar{\theta}_{st:jk} \delta_j(x_s) \delta_k(x_t)
\]

- The MAP problem is then equivalent to:

\[
\Phi_{\infty}(\bar{\theta}) := \max_{x \in \mathcal{X}^n} \langle \bar{\theta}, \phi(x) \rangle \equiv \max_{x \in \mathcal{X}^n} \sum_{s \in V} \bar{\theta}_s(x_s) + \sum_{(s, t) \in E} \bar{\theta}_{st}(x_s, x_t)
\]
Outline

Introduction
Motivation and Background

Tree-Reparameterizations
Definitions and Upper Bounds

LP Approaches
IP to LP to Lagrangian Dual

Message-passing Approaches
Two Algorithms

Summary

MAP estimation via agreement on (hyper)trees
Consider \( \{\theta^i\} \) such that \( \sum_i p^i \theta^i = \bar{\theta} \) and \( \sum_i p^i = 1 \).

**Claim:** \( \Phi_{\infty}(\bar{\theta}) \leq \sum_i p^i \Phi_{\infty}(\theta^i) \)

Let \( x^* \) be our MAP solution for \( \Phi_{\infty}(\bar{\theta}) \)

\[
\Phi_{\infty}(\bar{\theta}) = \max_{x \in \chi^n} \langle \bar{\theta}, \phi(x) \rangle = \langle \bar{\theta}, \phi(x^*) \rangle = \sum_i p_i \langle \theta^i, \phi(x^*) \rangle \\
\leq \sum_i p_i \left( \max_{x \in \chi^n} \langle \theta^i, \phi(x) \rangle \right) \\
= \sum_i p_i \Phi_{\infty}(\theta^i)
\]

**Goals:**

1. Characterize \( \{\theta^i\} \) by tree-structured exponential parameters.
2. When is the upper bound tight?
3. When not tight, how can we make it as tight as possible?
Tree Reparameterizations

- Let \( \{T_i\} \) be a set of spanning trees on G, \( p \) be a probability distribution over this set.
- \( \bar{\theta} = \sum_i p(T_i)\theta(T_i) \) is a \( p \)-reparameterization.
- Consider \( p_e \), the probability a given edge \( e \in E \) appears in a spanning tree chosen at random under \( p \).
- We can always find \( \sum_i p(T_i)\theta(T_i) = \bar{\theta} \) as long as \( p_e > 0 \) for all \( e \in E \).

E.g. Above, if \( p(T_i) = 1/3 \) for all \( i \), then \( p_b = 1, p_e = 2/3, p_f = 1/3 \).
Consider a 4-node cycle and let \( x \in \{0, 1\}^4 \). Also let,

\[
p(x; \tilde{\theta}) \propto \exp\{x_1 x_2 + x_2 x_3 + x_3 x_4 + x_4 x_1\}
\]

So, \( \tilde{\theta} = [0 \ 0 \ 0 \ 0 \ 1 \ 1 \ 1 \ 1] \)

We will reparameterize \( \tilde{\theta} \) by a convex combination of exponential parameters of the following spanning trees:

MAP estimation via agreement on (hyper)trees
Example (cont.)

- Give each tree, $T_i$, equal weighting – $p(T_i) = 1/4$
- We construct each tree-structured exponential parameter as:
  \[
  \begin{align*}
  \theta(T_1) &= (4/3)[0 \ 0 \ 0 \ 0 \ 1 \ 1 \ 1 \ 0], \\
  \theta(T_2) &= (4/3)[0 \ 0 \ 0 \ 0 \ 1 \ 1 \ 0 \ 1], \\
  \theta(T_3) &= (4/3)[0 \ 0 \ 0 \ 0 \ 1 \ 0 \ 1 \ 1], \\
  \theta(T_4) &= (4/3)[0 \ 0 \ 0 \ 0 \ 0 \ 1 \ 1 \ 1] 
  \end{align*}
  \]
- This gives us $\bar{\theta} = \sum_i p(T_i)\theta(T_i)$. 
Tightness of Upper Bound

- For any tree $p$-reparameterization we have:

\[
\Phi_\infty(\bar{\theta}) \leq \sum_i p(T_i)\Phi_\infty(\theta(T_i)) = \sum_i p(T_i) \max_{x \in \chi^n} \{\langle \theta(T_i), \phi(x) \rangle \}
\]

- When does the above inequality meet with equality?

- Let’s define the set of optimal configurations for a given $\theta$:

\[
OPT(\theta) := \{ x \in \chi^n \mid \langle \theta, \phi(x') \rangle \leq \langle \theta, \phi(x) \rangle \ \text{for all} \ x' \in \chi^n \}
\]

**Lemma:** Let $\bar{\theta} = \sum_i p(T_i)\theta(T_i)$, then

\[
\bigcap_{T_i} OPT(\theta(T_i)) \subseteq OPT(\bar{\theta})
\]

Moreover, the bound is tight iff the intersection on the LHS is non-empty.
Proof

Let \( x^* \in OPT(\bar{\theta}) \) and consider the difference between the RHS and LHS:

\[
0 \leq \left[ \sum_i p(T_i)\Phi_\infty(\theta(T_i)) \right] - \Phi_\infty(\bar{\theta}) = \left[ \sum_i p(T_i)\Phi_\infty(\theta(T_i)) \right] - \langle \bar{\theta}, \phi(x^*) \rangle = \sum_i p(T_i) \left[ \Phi_\infty(\theta(T_i)) - \langle \theta(T_i), \phi(x^*) \rangle \right]
\]

This underscored term is non-negative, and equal to zero only when \( x^* \in OPT(\theta(T_i)) \).

So bound is tight iff \( x^* \in \bigcap_{T_i} OPT(\theta(T_i)) \).
Tree Reparameterization Wrap-up

- A set of spanning trees where $\bigcap_{T_i} OPT(\theta(T_i))$ is non-empty is said to follow the tree agreement.

- Suppose that we fix the spanning tree distribution $p$ and our target parameter $\bar{\theta}$.

- Goal: Optimize the upper bound as a function of $\theta(T_i)$

- Two Approaches:
  1. Direct Minimization through LP relaxation techniques
  2. Message-passing Approaches
Outline

Introduction
  Motivation and Background

Tree-Reparameterizations
  Definitions and Upper Bounds

LP Approaches
  IP to LP to Lagrangian Dual

Message-passing Approaches
  Two Algorithms

Summary
MAP as an Integer Program (IP)

- Fix our set of spanning trees \( \{ T_i \} \) and our distribution over them, \( p \).
- Main Problem: We would like to minimize the RHS of the following inequality w.r.t. \( \theta(T_i) \) to get a bound as tight as possible:

\[
\Phi_\infty(\tilde{\theta}) \leq \sum_i p(T_i)\Phi_\infty(\theta(T_i))
\]

- Consider the MAP optimization problem:

\[
\arg \max_{x \in X^n} \langle \tilde{\theta}, \phi(x) \rangle
\]

- We first relax the above integer program to an equivalent linear program...
LP Relaxation

- Remind yourself what the *marginal polytope* is:

\[
\text{MARG}(G) := \{ \mu \in \mathbb{R}^d \mid \mu_{s,j} = E_p[\delta_j(x_s)], \\
\mu_{st,jk} = E_p[\delta_j(x_s)\delta_k(x_t)] \text{ for some } p(\cdot) \} 
\]

- **Claim:** \( \max_{x \in \mathcal{X}^n} \langle \bar{\theta}, \phi(x) \rangle = \max_{\mu \in \text{MARG}(G)} \langle \bar{\theta}, \mu \rangle \)

- Let \( \mathcal{P} = \{ p(\cdot) \mid p(x) \geq 0, \sum_x p(x) = 1 \} \).

\[
\max_{x \in \mathcal{X}^n} \langle \bar{\theta}, \phi(x) \rangle = \max_{p \in \mathcal{P}} \left( \sum_{x \in \mathcal{X}^n} p(x) \langle \bar{\theta}, \phi(x) \rangle \right) 
\]

- We now transform the RHS of this equation...
LP Relaxation (cont.)

- From the last slide we had:

\[
\max_{p \in \mathcal{P}} \left( \sum_{x \in \chi^n} p(x) \langle \theta, \phi(x) \rangle \right) = \sum_{x \in \chi^n} p(x) \left( \sum_{s \in V} \bar{\theta}_s(x_s) + \sum_{(s,t) \in E} \bar{\theta}_{st}(x_s, x_t) \right)
\]

\[
= \sum_{s \in V} \sum_{j \in \chi_s} \bar{\theta}_{s;j} \mu_{s;j} + \sum_{(s,t) \in E} \sum_{(j,k) \in \chi_s \times \chi_t} \bar{\theta}_{st;jk} \mu_{st;jk}
\]

- So as \( p \) ranges over \( \mathcal{P} \) the marginals \( \mu \) range over \( \text{MARG}(G) \).
- Conclusion: \( \max_{x \in \chi^n} \langle \theta, \phi(x) \rangle = \max_{\mu \in \text{MARG}(G)} \langle \bar{\theta}, \mu \rangle \)
Lagrangian Dual

- We now have an LP formulation of MAP which is much easier to solve.
- Remember we have:
  \[ \Phi_{\infty}(\bar{\theta}) \leq \sum_i p(T_i)\Phi_{\infty}(\theta(T_i)) = \sum_i p(T_i) \max_{x \in \mathcal{X}^n} \langle \theta(T_i), \phi(x) \rangle \]

- We now would like to address the problem of:
  \[ \min_{\theta(T_i)} \sum_i p(T_i)\Phi_{\infty}(\theta(T_i)) \]
  \[ \text{s.t. } \sum_i p(T_i)\theta(T_i) = \bar{\theta} \]

- Attack problem via its Lagrangian dual, which yields the relaxed LP:
  \[ \Phi_{\infty}(\bar{\theta}) \leq \max_{\tau \in \text{LOCAL}(G)} \langle \tau, \bar{\theta} \rangle \]

\[ \text{LOCAL}(G) := \{ \tau \in \mathbb{R}_+^d \mid \sum_{j \in \mathcal{X}_s} \tau_{s,j} = 1 \ \forall s \in V, \sum_{k \in \mathcal{X}_t} \tau_{st,jk} = \tau_{s,j} \ \forall (s, t) \in E, j \in \mathcal{X}_s \} \]
LOCAL(G)

- We’ve relaxed our original constraint set from \( MARG(G) \) to \( LOCAL(G) \).
- The relaxation is exact for any problem on a tree-structured graph.
- \( LOCAL(G) \) is characterized by \( \mathcal{O}(mn + m^2|E|) \), \( m := \max_s |\chi_s| \) constraints.
- 2 optimum possibilities:
  1. Optimum is a vertex that is both in \( MARG(G) \) and \( LOCAL(G) \), then tree agreement holds.
  2. Optimum is a fractional vertex of \( LOCAL(G) \), then no tree agreement can occur.

MAP estimation via agreement on (hyper)trees
We’ve now reformulated the MAP estimation problem as an approximate LP.

We could easily use classical LP techniques (i.e. Simplex Method) to solve this problem.

However, we would like to exploit the graphical structure of our MRF.

Want: Message-passing algorithms to solve the same LP over \( \text{LOCAL}(G) \) – fixed points of iterative algorithm = optimum of LP relaxation.
Outline

Introduction
Motivation and Background

Tree-Reparameterizations
Definitions and Upper Bounds

LP Approaches
IP to LP to Lagrangian Dual

Message-passing Approaches
Two Algorithms

Summary
Preliminaries

- Let's define \textit{max-marginals}:

\begin{align*}
\nu_{s,j} & := \kappa_s \max_{\{x \mid x_s = j\}} p(x; \theta(T)) \\
\nu_{st,jk} & := \kappa_{st} \max_{\{x \mid (x_s, x_t) = (j, k)\}} p(x; \theta(T)) \\
\nu_s(x_s) & := \sum_{j \in \chi_s} \nu_{s,j} \delta_j(x_s) \\
\nu_{st}(x_s, x_t) & := \sum_{(j, k) \in \chi_s \times \chi_t} \nu_{st,jk} \delta_j(x_s) \delta_k(x_t)
\end{align*}

- For trees we can decompose the joint distributions into the junction tree decomposition:

\[
p(x; \theta(T)) \propto \prod_{s \in V} \nu_s(x_s) \prod_{(s, t) \in E(T)} \frac{\nu_{st}(x_s, x_t)}{\nu_s(x_s) \nu_t(x_t)}
\]
Trees and Max-marginals

- For every edge \((s, t) \in E(T)\)
  \[ \nu_s(x_s) = \kappa \max_{x'_t \in X_t} \nu_{st}(x_s, x'_t) \]

- \(x^*\) is in \(OPT(\theta(T))\) iff:
  \[ x^*_s \in \arg \max_{x_s} \nu_s(x_s) \ \forall \ s \]
  \[ (x^*_s, x^*_t) \in \arg \max_{x_s, x_t} \nu_{st}(x_s, x_t) \ \forall \ (s, t) \]

- All the above are \textit{local} conditions.

- So, to find a MAP configuration:
  1. root T at \(x_r\) and choose \(x^*_r \in \arg \max_{x_r} \nu_r(x_r)\).
  2. from parent s to child t choose
    \[ x^*_t \in \arg \max_{x_t} \nu_{st}(x^*_s, x_t) \]

MAP estimation via agreement on (hyper)trees
Algorithm Idea

- For graphs with cycles we develop the idea of *pseudo-max marginals*
  \[ \nu = \{\nu_s, \nu_{st}\} \]
- Denote \(\nu(T)\) as elements of \(\nu\) pertaining to tree \(T\).
- \(\nu(T)\) implicitly specifies a tree exponential parameter \(\theta(T)\) by junction tree decomposition
- Goal: Given \(p\) and \(\{T_i\}\), we would like to find a \(\nu\) that satisfies the following two properties:
  1. \(\nu\) implicitly specifies a \(p\)-reparameterization of \(\tilde{\theta} - \nu(T_i) \rightarrow \theta(T_i)\)
     and \(\tilde{\theta} = \sum_i p_i \theta(T_i)\).
  2. \(\nu(T_i)\) is exact max-marginal under \(p(x; \theta(T_i))\) \(\forall i - tree\ consistency\)
- We devise in iterative algorithm that after each iteration maintains (1) and upon convergence achieves (2).
Preliminary Idea

- Update pseudo-max marginal $\nu$ iteratively.
- Two strategies:
  1. Tree based updates – update only $\nu(T)$ a tree at a time.
  2. Edge based updates – update $\nu_{st}$ along with associated $\nu_s, \nu_t$ in parallel.
- The algorithm to follow uses (2) and is motivated by:

$$\kappa \nu_s(x_s) = \max_{x'_t \in \chi_t} \nu_{st}(x_s, x'_t)$$

- So $\nu$ is consistent on every tree $T_i$ iff the above holds for every $(s, t) \in E$. 
Algorithm 1: Edge-based updates

Algorithm 1 (Edge-based reparameterization updates).

1. Initialize the pseudo-max-marginals \( \nu_s^0, \nu_{st}^0 \) in terms of the original exponential parameter vector as follows:

\[
\begin{align*}
\nu_s^0(x_s) &= \kappa \exp(\bar{\theta}_s(x_s)) \\
\nu_{st}^0(x_s, x_t) &= \kappa \exp\left(\frac{1}{\rho_{st}} \bar{\theta}_{st}(x_s, x_t) + \bar{\theta}_t(x_t) + \bar{\theta}_s(x_s)\right)
\end{align*}
\]  
(44a)  
(44b)

2. For iterations \( n = 0, 1, 2, \ldots \), update the pseudo-max-marginals as follows:

\[
\begin{align*}
\nu_s^{n+1}(x_s) &= \kappa \nu_s^n(x_s) \prod_{t \in \Gamma(s)} \left[\max_{x'_t} \frac{\nu_{st}^n(x_s, x'_t)}{\nu_s^n(x_s)}\right]^{\rho_{st}} \\
\nu_{st}^{n+1}(x_s, x_t) &= \kappa \frac{\nu_{st}^n(x_s, x_t)}{\max_{x'_t} \nu_{st}^n(x_s, x'_t)} \frac{\nu_{st}^n(x'_s, x_t)}{\max_{x'_s} \nu_{st}^n(x'_s, x_t)} \nu_s^{n+1}(x_s) \nu_t^{n+1}(x_t)
\end{align*}
\]  
(45a)  
(45b)

\[
\text{Here } \rho_{st} \text{ is the edge-appearance probability in our } \{ T_i \}\].
\]
**p-reparameterization proof**

- **Note:** From junction-tree decomposition we can define:

\[
\theta^n_s(T)(x_s) = \log \nu^n_s(x_s) \quad \forall \ s \in V
\]

\[
\theta^n_{st}(T)(x_s, x_t) = \begin{cases} 
\log \frac{\nu^n_{st}(x_s, x_t)}{\nu^n_s(x_s)\nu^n_t(x_t)} & \text{if } (s, t) \in E(T) \\
0 & \text{otherwise}
\end{cases} \quad \forall (s, t) \in E
\]

- **Lemma:** At each iteration we satisfy the p-reparameterization requirement. (proof by induction)

  - **Base Case** (n=0):

\[
\sum_i p(T_i) \left[\sum_{s \in V} \theta^0_s(T) + \sum_{(s, t) \in E(T)} \theta^0_{st}(T)\right] = \sum_i p(T_i) \left[\sum_{s \in V} \bar{\theta}_s + \sum_{(s, t) \in E(T)} \frac{1}{\rho_{st}} \bar{\theta}_{st}\right] = \sum_{s \in V} \bar{\theta}_s + \sum_{(s, t) \in E} \bar{\theta}_{st}
\]
Induction Step

- Induction Step: $\sum_i p(T_i)\theta^{n+1}(T)(x) =$

$$= \sum_i p(T_i) \sum_{s \in V} \left( \log \nu_s^n(x_s) + \sum_{t \in \Gamma(s)} \rho_{st} \log \frac{\max_{x'_t} \nu_{st}^n(x_s, x'_t)}{\nu_s^n(x_s)} \right)$$

$$+ \sum_i p(T_i) \sum_{(s,t) \in E(T)} \log \frac{\nu_{st}^n(x_s, x_t)}{\max_{x'_t} \nu_{st}^n(x_s, x'_t) \max_{x'_s} \nu_{st}^n(x'_s, x_t)}$$

$$= \sum_{s \in V} \log \nu_s^n(x_s) + \sum_{(s,t) \in E} \log \frac{\nu_{st}^n(x_s, x_t)}{\nu_s^n(x_s) \nu_t^n(x_t)}$$

- This last line is equal to $\sum_i p(T_i)\theta^n(T)(x)$ up to an additive constant independent of $x$. 
Convergence Criteria Proof

- **Lemma**: A fixed point of the algorithm $\nu^*$ satisfies the tree consistency condition.

- Substitute $\nu^* = \nu^{n+1} = \nu^n$ into all the updates in the algorithm and we get:

$$\frac{\nu_{st}^*(x_s)}{\max_{x'_t} \nu_{st}^*(x_s, x'_t)} \frac{\nu_t^*(x_t)}{\max_{x'_s} \nu_{st}^*(x'_s, x'_t)} = \kappa \quad \forall (x_s, x_t)$$

- This implies $\nu$ is consistent on every tree $T_i$

- Reminder: consistent on every tree $\neq$ consistent on entire graph (i.e. unique MAP).
Two Algorithms

Algorithm 2: Message passing version

- We can create a message passing version of the same algorithm by utilizing the following mapping:

\[
\nu_s(x_s) := \exp(\bar{\theta}_s(x_s)) \prod_{v \in \Gamma(s)} [M_{vs}(x_s)]^{\rho_{vs}}
\]

\[
\nu_{st}(x_s, x_t) := \exp\left(\frac{1}{\rho_{st}} \bar{\theta}_{st}(x_s, x_t) + \bar{\theta}_s(x_s) + \bar{\theta}_t(x_t)\right) \frac{\prod_{v \in \Gamma(s) \setminus t} [M_{vs}(x_s)]^{\rho_{vs}}}{[M_{ts}(x_s)]^{(1-\rho_{ts})}} \frac{\prod_{v \in \Gamma(t) \setminus s} [M_{vt}(x_t)]^{\rho_{vt}}}{[M_{st}(x_t)]^{(1-\rho_{st})}}
\]

Algorithm 2 (Parallel tree-reweighted max-product).

1. Initialize the messages \( M^0 = \{M^0_{st}\} \) with arbitrary positive real numbers.

2. For iterations \( n = 0, 1, 2, \ldots \), update the messages as follows:

\[
M_{ts}^{n+1}(x_s) = \kappa \max_{x'_t \in \mathcal{X}_t} \left\{ \exp\left(\frac{1}{\rho_{st}} \bar{\theta}_{st}(x_s, x'_t) + \bar{\theta}_t(x'_t)\right) \frac{\prod_{v \in \Gamma(t) \setminus s} [M^n_{vt}(x'_t)]^{\rho_{vt}}}{[M^n_{st}(x'_t)]^{(1-\rho_{st})}} \right\}
\]  

\[ (48) \]
Analysis of Algorithms

- The equivalence between algorithms 1 & 2 follows from the equivalence between message-passing and reparameterization (Buhm’s talk).

- Key observations about Algorithm 2:
  1. if $\rho_{st} = 1$ for every edge, equivalent to loopy BP.
  2. Key difference from standard message passing is $M_{ts}$ depends on $M_{st}$.

- Not always guaranteed to converge.

- In practice it’s nice to use a dampening technique to encourage convergence.

- Connection to LP relaxation: Let $\lambda^* = \log M^*$ elementwise where $M^*$ is fixed point of algorithm 2. Then $\lambda^*$ is an optimal solution to the LP relaxation given earlier.
Outline

Introduction
Motivation and Background

Tree-Reparameterizations
Definitions and Upper Bounds

LP Approaches
IP to LP to Lagrangian Dual

Message-passing Approaches
Two Algorithms

Summary
Wrapping Things Up

- MAP can be attacked via tree-reparameterizations and LP relaxations.
- Message-passing can be viewed as an alternative way to solve the LP dual problem proposed.
- Authors claim they’ve successfully used these algorithms for a few applications. Results?
- When we do not converge, can we perform a further relaxation by clustering random variables like Kikuchi approximations?
Citations


