# Message passing for Graph-Structured Linear Programs: Proximal Projections and Rounding Schemes

Pradeep Ravikumar, *Alekh Agarwal*, Martin Wainwright

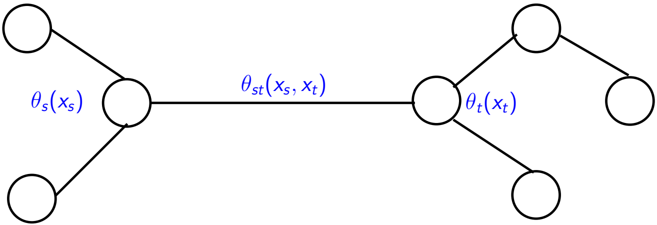
University of California Berkeley

# MAP estimation in MRFs

- Collection of random variables $(X_1, \ldots, X_p) \in \{1, \ldots, m\}^p$

$$\mathbb{P}(\mathbf{x}; \theta) \propto \exp\left\{ \sum_{s \in V} \underbrace{\theta_s(x_s)}_{\text{Node Potential}} + \sum_{(s,t) \in E} \underbrace{\theta_{st}(x_s, x_t)}_{\text{Edge Potential}} \right\}$$

- $x_{MAP} = \arg\max_{x \in \mathcal{X}^p} \left\{ \sum_{s \in V} \theta_s(x_s) + \sum_{(s,t) \in E} \theta_{st}(x_s, x_t) \right\}$



# Previous Work

- Max-Product - dynamic programming, exact for trees
- Graph Cuts (Boykov et al, 2001)
- TRW-Max Product (Wainwright et al, 2005; Kolmogorov et al, 2005, Globerson et al, 2007)
- Convex-free energy approximations (Weiss et al, 2007)
- Convex relaxations to LP, QP, SOCP, SDP etc. (Chekuri et al, 2005; Wainwright et al, 2005; Ravikumar et al, 2006; Kumar et al, 2006)
- Lagrangian relaxation and simulated annealing (Johnson et al, 2007)
- Dual decomposition and subgradient method (Komodakis et al, 2007)

# First-order LP Relaxation for MAP estimation

$$x_{MAP} = \arg\max_{x \in \mathcal{X}^p} \left\{ \sum_{s \in V} \theta_s(x_s) + \sum_{(s,t) \in E} \theta_{st}(x_s, x_t) \right\}$$

$$= \arg\max_{x \in \mathcal{X}^p} \left\{ \sum_{s \in V, j \in \mathcal{X}} \theta_{s;j} \mathbb{I}(x_s = j) + \sum_{(s,t) \in E, j,k \in \mathcal{X}} \theta_{st;jk} \mathbb{I}(x_s = j, x_t = k) \right\}$$

- Relax indicator variables to their expected values

$$\mu_{s;j} = P(x_s = j), \mu_{st;jk} = P(x_s = j, x_t = k).$$

# First-order LP Relaxation contd.

$$x_{MAP} = \arg \max_{\mu \geq 0,\ \mu \in \mathbb{L}(G)} \left\{ \sum_{s \in V, j \in \mathcal{X}} \theta_{s;j} \mu_{s;j} + \sum_{(s,t) \in E, j,k \in \mathcal{X}} \theta_{st;jk} \mu_{st;jk} \right\}$$

$$= \arg \max_{\mu \geq 0,\ \mu \in \mathbb{L}(G)} \langle \theta, \mu \rangle$$

$$\mathbb{L}(G) := \sum_j \mu_{s;j} = 1 \quad \forall s \in V \qquad \text{(Normalization)}$$

$$\sum_k \mu_{st;jk} = \mu_{s;j} \quad \forall (s,t) \in E, k \in \mathcal{X} \qquad \text{(Marginalization)}$$

- The constraint set $\mathbb{L}(G)$ is a (local) polytope.

# Solving the LP relaxation

- Can solve the LP using interior point, subgradient methods etc.
  - Too expensive/ slow for practical problems.
  - These do not fully exploit problem structure.
  - Hard to parallelize/distribute in general.
- TRW-MP (approximately) solves dual problem.
  - Hard to add in new constraints in TRW setup.

# MAP estimation desiderata

1. Exact solution to LP relaxation.
2. Time complexity competitive with Max-Product, TRW-MP.
3. Distributable, parallelizable updates.
4. Be easy to incorporate new constraints.

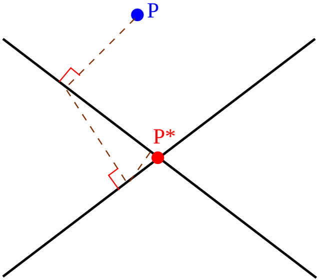
# Proximal Minimization

$$\mu^{n+1} = \arg \min_{\mu \in \mathbb{L}(G)} \left\{ -\theta^T \mu + \frac{1}{\omega^n} \mathrm{D}_f \left( \mu \| \mu^n \right) \right\}$$

- $\mathrm{D}_f(\cdot \| \cdot)$ Bregman divergence.
- Makes problem strictly convex.
- Similar to annealing; equivalent for certain choices of $D_f$.
- $\mathrm{D}_f \left( \mu^{n+1} \| \mu^n \right)$ goes to 0, so can use fixed $\omega^n$.

# Bregman Projections

- Projection of $\mu$ on $C$: $\hat{\mu} := \arg\min_{\mu \in C} \mathrm{D}_f\,(\mu \| \nu)$.
- If $C = \cap_i C_i$, $C_i$ convex




  - $\hat{\mu}$ obtained by iterating projections onto $C_i$'s.

# Proximal minimization via Bregman Projections

$$\text{LP}: \quad \arg\min_{\mu \in \mathbb{L}(G)} \langle -\theta, \mu \rangle$$

# Proximal minimization via Bregman Projections

$$\text{LP}: \quad \arg \min_{\mu \in \mathbb{L}(G)} \langle -\theta, \mu \rangle$$

$$\text{Proximal}: \quad \arg \min_{\mu \in \mathbb{L}(G)} \langle -\theta, \mu \rangle + \frac{1}{\omega^n} \mathrm{D}_f \left( \mu \| \mu^n \right)$$

# Proximal minimization via Bregman Projections

$$\text{LP}: \quad \arg\min_{\mu \in \mathbb{L}(G)} \langle -\theta, \mu \rangle$$

$$\text{Proximal}: \quad \arg\min_{\mu \in \mathbb{L}(G)} \langle -\theta, \mu \rangle + \frac{1}{\omega^n} \mathrm{D}_f\left(\mu \| \mu^n\right)$$

$$\text{Proximal Rewriting}: \quad \arg\min_{\mu \in \cap_i \mathbb{L}_i(G)} \mathrm{D}_f\left(\mu \| \tilde{\mu^n}\right)$$

# Proximal minimization via Bregman Projections

$$\text{LP}: \quad \arg\min_{\mu \in \mathbb{L}(G)} \langle -\theta, \mu \rangle$$

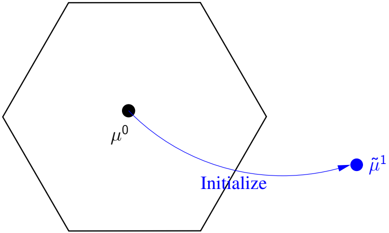
$$\text{Proximal}: \quad \arg\min_{\mu \in \mathbb{L}(G)} \langle -\theta, \mu \rangle + \frac{1}{\omega^n} \mathrm{D}_f \left( \mu \| \mu^n \right)$$

$$\text{Proximal Rewriting}: \quad \arg\min_{\mu \in \cap_i \mathbb{L}_i(G)} \mathrm{D}_f \left( \mu \| \tilde{\mu^n} \right)$$

1. Updated Parameters : $\tilde{\mu^n} = \mu^n \odot_f \theta$
   — $\ell_2$ Distance : $\tilde{\mu^n} = \mu^n + \theta$
   — KL Divergence : $\tilde{\mu^n} = \mu^n \exp(\theta)$
2. Intersection of local constraints : $\mathbb{L}(G) = \cap_i \mathbb{L}_i(G)$

# Outline of algorithm

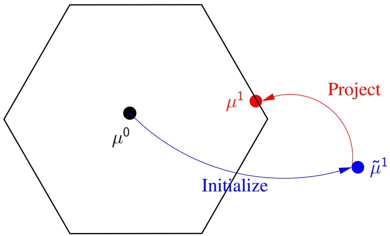
- Outer loop: $\mu^{n+1} = \arg\min_{\mu \in \mathbb{L}(G)} \mathrm{D}_f\left(\mu \| \tilde{\mu}^n\right)$




- Solve outer loop via cyclic projections:
  - Initialize $\mu^{n+1,0} = \mu^n \odot_f \theta$

# Outline of algorithm

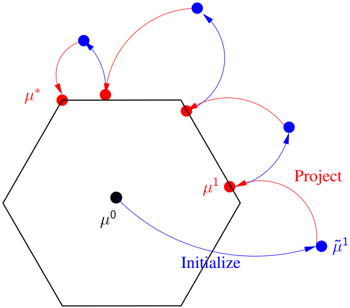
- Outer loop: $\mu^{n+1} = \arg\min_{\mu \in \mathbb{L}(G)} \mathrm{D}_f\left(\mu \| \mu^n\right)$




- Solve outer loop via cyclic projections:
  - Initialize $\mu^{n+1,0} = \mu^n \odot_f \theta$
  - $\mu^{n+1,\tau+1} = \text{Projection of } \mu^{n+1,\tau} \text{ onto current constraint } \mathbb{L}_{i(\tau)}(G)$

# Outline of algorithm

- Outer loop: $\mu^{n+1} = \arg\min_{\mu \in \mathbb{L}(G)} \mathrm{D}_f\left(\mu \| \mu^n\right)$




- Solve outer loop via cyclic projections:
  - Initialize $\mu^{n+1,0} = \mu^n \odot_f \theta$
  - $\mu^{n+1,\tau+1} = \text{Projection of } \mu^{n+1,\tau} \text{ onto current constraint } \mathbb{L}_{i(\tau)}(G)$

# Entropy Messages

Iterate $n = 1, 2, \ldots$

Initialize :

$$\mu_{st}^{(n,0)} = \mu_{st}^{(n-1)} \exp\left(\omega_n \ \theta_{st}\right)$$

$$\mu_{s}^{(n,0)} = \mu_{s}^{(n-1)} \exp\left(\omega_n \ \theta_{s}\right)$$

Iterate over edges $(s,t)$:

$$\mu_{st}^{n,\tau+1} \propto \mu_{st}^{n,\tau} \sqrt{\frac{\mu_s^{n,\tau}}{\sum_{x_t} \mu_{st}^{n,\tau}}}$$

$$\mu_{s}^{n,\tau+1} \propto \mu_{s}^{n,\tau} \sqrt{\frac{\sum_{x_t} \mu_{st}^{n,\tau}}{\mu_s^{n,\tau}}}$$

- Striking similarity with belief propagation.
- Similar derivation for other divergences.

# MAP estimation desiderata

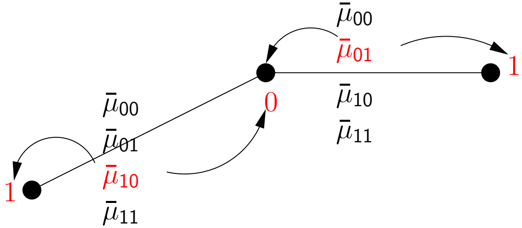
- $\surd$ Exact solution to LP relaxation.
- $\surd$ Time complexity competitive with Max-Product, TRW-MP (*with rounding*).
- $\surd$ Distributable, parallelizable updates.
- $\surd$ Easy to incorporate new constraints.

# Rounding Schemes

- Naive Rounding: $\hat{x}_s = \arg\min_{x_s} \mu_s^n(x_s)$

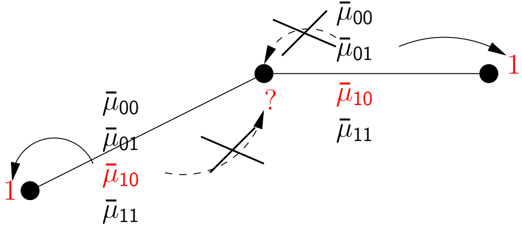
# Rounding Schemes

- Naive Rounding
- Edge Rounding: $(\hat{x}_s, \hat{x}_t) = \arg\min_{x_s, x_t} \bar{\mu}^n_{st}(x_s, x_t)$, *provided* they are consistent.
- $\bar{\mu}^n_{st}(x_s, x_t) = \mu^n_s(x_s)^{1/d_s} \mu^n_t(x_t)^{1/d_t} \mu^n_{st}(x_s, x_t)$



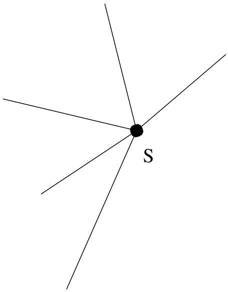
# Rounding Schemes

- Naive Rounding
- Edge Rounding: $(\hat{x}_s, \hat{x}_t) = \arg\min_{x_s, x_t} \bar{\mu}^n_{st}(x_s, x_t)$, *provided* they are consistent.



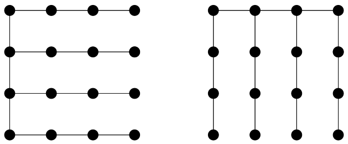
# Rounding Schemes

- Naive Rounding
- Edge Rounding
- Star Rounding: $\hat{x} \leftarrow$ optimal configuration of a reparameterization for each local neighborhood.



# Rounding Schemes

- Naive Rounding
- Edge Rounding
- Star Rounding
- Tree Rounding: $\hat{x} \leftarrow$ optimal configuration for each tree in set.



# Optimality Certificate

- Naive Rounding $\equiv$ no guarantees in general

# Optimality Certificate

- Naive Rounding $\equiv$ no guarantees in general
- **Theorem**: At any iteration, if any of {Edge,Star,Tree} rounding schemes find a consistent configuration $\hat{x}$, then $\hat{x}$ is the MAP configuration.

# Optimality Certificate

- Naive Rounding $\equiv$ no guarantees in general
- **Theorem**: At any iteration, if any of {Edge,Star,Tree} rounding schemes find a consistent configuration $\hat{x}$, then $\hat{x}$ is the MAP configuration.
- Usually takes 8-10 rounds of TRW worth time to converge via rounding for integral LP

# Rate of convergence

- Superlinear rate of convergence assuming exact solution to proximal subproblem

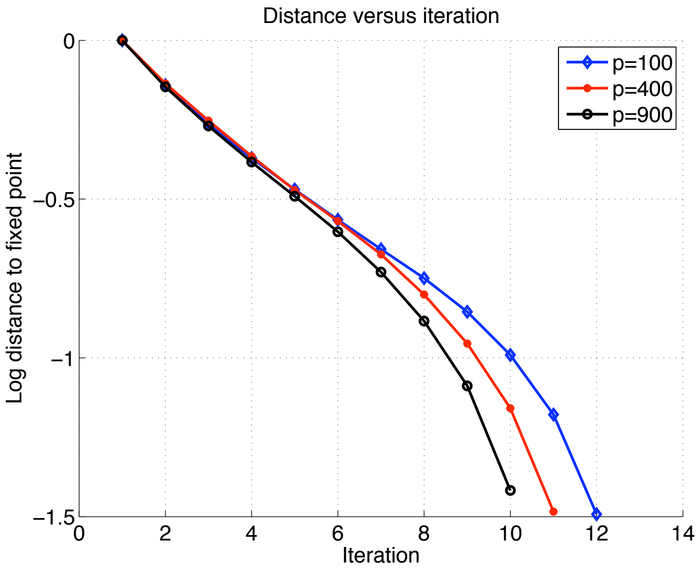
$$\lim_{n \to 0} \frac{\|\mu^n - \mu^*\|}{\|\mu^{n-1} - \mu^*\|} \to 0$$

- Overall linear rate guaranteed even with inexact inner solutions
- Much faster convergence for integral LPs due to rounding

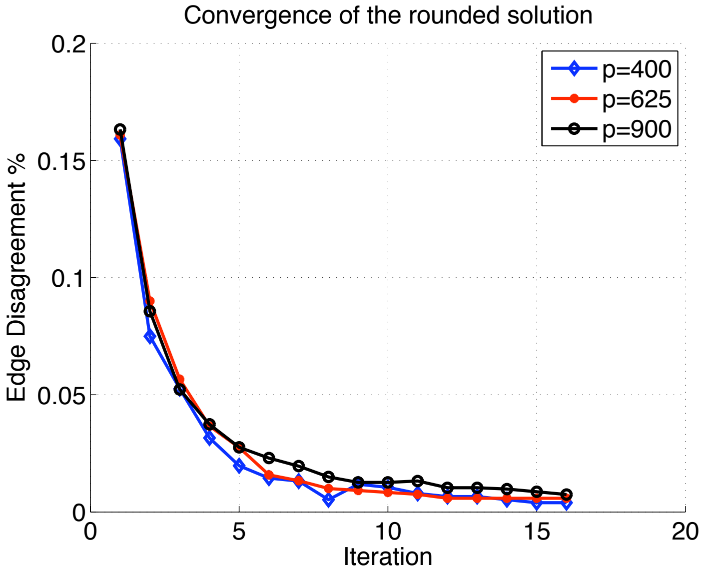
# Experiments

- Grid graphs between 100-900 nodes, 5 labels
- Potts potential with varying SNR
- Empirical verification of superlinear convergence
- Effectiveness of rounding schemes

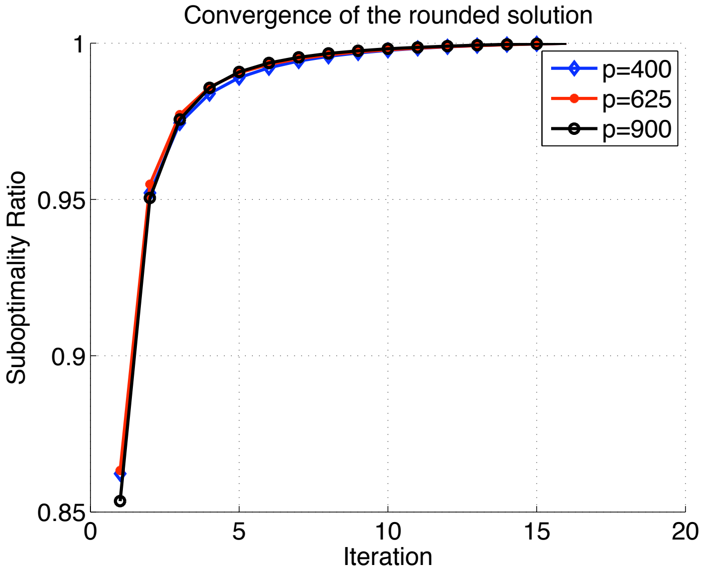
# Superlinear rate of convergence



# Convergence of rounded solution



# Convergence of rounded solution contd.



# Summary

- We propose new MAP algorithms that
  - use the proximal minimization framework,
  - with a graph structured Bregman divergence that exploits problem structure,
  - and solved by cyclic Bregman projections that exploit the intersection structure of the local constraint polytope.
- Simple message passing updates with a guarantee of convergence to LP optimum.
- Rounding schemes with extremely fast solutions for integral LPs.