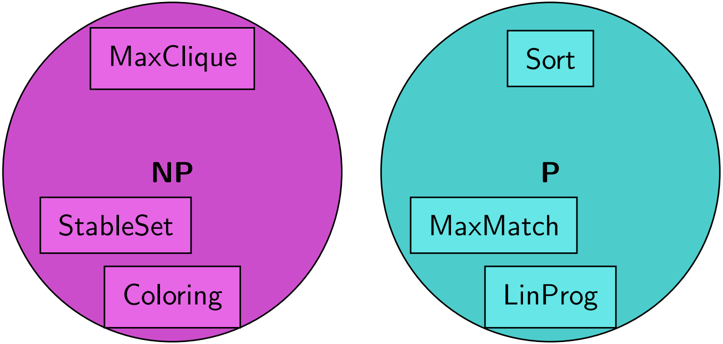
Graphical Modeling and Machine Learning with Perfect Graphs

Tony Jebara, Columbia University

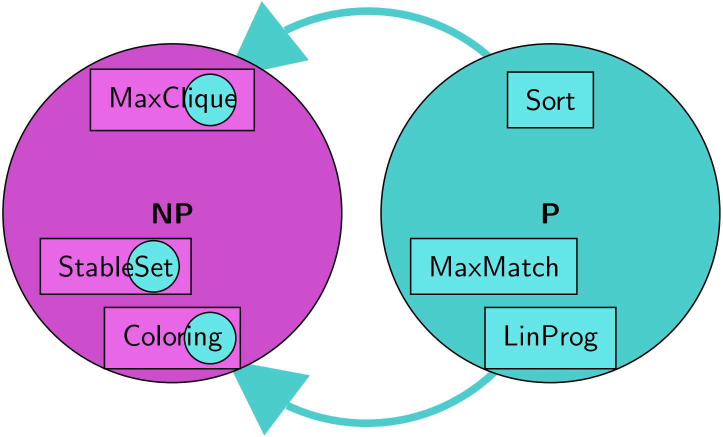
November 23, 2010

Tractability



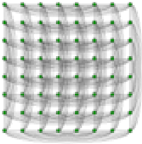
- Tractability: what problems are P or NP-hard?

Tractability



- Tractability: what problems are P or NP-hard?
- However, many instances of NP-hard problems are easy.
- On perfect graphs, all the above NP problems are in P

Origins of Perfect Graphs

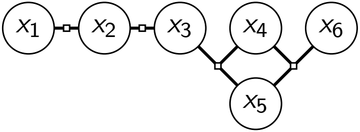
- In 1960, Berge introduces perfect graphs where every induced subgraph has clique# = coloring#
- Berge also poses two conjectures:
  - Weak: a graph is perfect iff its complement is perfect
  - Strong: a graph is perfect iff it is Berge
- Weak perfect graph theorem (Lovász '72)
- Link between perfection and integral LPs (Lovász '72)
- Strong perfect graph theorem (SPGT) open for 4+ decades

Progress on Perfect Graphs

- SPGT Proof (Chudnovsky, Robertson, Seymour, Thomas '03)
- Berge passes away shortly after hearing of the proof
- Many NP-hard and hard to approximate problems are P for perfect graphs
  - Graph coloring
  - Maximum clique
  - Maximum independent set
- Recognizing perfect graphs is $O(n^9)$ (Chudnovsky *et al.* '06)

Graphical Models



- MAP Estimation, Message Passing & Perfect graphs (J '09)
- Graphical model: a bipartite factor graph G representing a distribution $p(X)$ where $X = \{x_1, \ldots, x_n\}$ and $x_i \in \mathbb{Z}$
- $p(X)$ factorizes as product of $\{\psi_1, \ldots, \psi_C\}$ functions (squares) over $\{X_1, \ldots, X_C\}$ subsets of variables (circles)

$$p(x_1, \ldots, x_n) \quad = \quad \frac{1}{Z} \prod_{c \in C} \psi_c(X_c)$$

- E.g. $p(x_1, \ldots, x_6) = \psi(x_1, x_2)\psi(x_2, x_3)\psi(x_3, x_4, x_5)\psi(x_4, x_5, x_6)$

Maximum A Posteriori (MAP) Estimation

- A canonical problem, find most probable configuration

$$X^* = \operatorname{argmax} p(x_1, \ldots, x_n)$$

- Useful for image processing, protein folding, coding, etc.
- NP-hard for general graphs (Shimony '94)
- Efficient for trees and singly linked graphs (Pearl '88)
- Efficient for submodular $\psi_c(X_c)$ (Grieg '89, Kolmogorov '04)
- First order LP relaxations (Wainwright *et al.* '02)
- TRW max-product (Kolmogorov & Wainwright '06)
- Higher order LP relaxations (Sontag *et al.* '08)
- Fractional and integral LP rounding (Ravikumar *et al.* '08)
- Or alternative approach: max product and message passing...

Max Product Message Passing

Until converged:

1. For each x_i to each X_c: $m^{t+1}_{i \to c} = \prod_{d \in \mathrm{Ne}(i) \setminus c} m^t_{d \to i}$
2. For each X_c to each x_i: $m^{t+1}_{c \to i} = \max_{X_c \setminus x_i} \psi_c(X_c) \prod_{j \in c \setminus i} m^t_{j \to c}$

- A simple and fast algorithm that performs well in practice
- Exact for trees (Pearl '88)
- Converges for single-loop graphs (Weiss & Freeman '01)
- Convergence via Gibbs measure (Tatikonda & Jordan '02)
- Local optimality guarantees (Wainwright *et al.* '03)
- Exact for bipartite matchings (Bayati *et al.* '05)
- Exact for bipartite *b*-matchings (Huang and J '07)

Bipartite Matching

	Motorola	Apple	IBM
"laptop"	0\$	2\$	2\$
"server"	0\$	2\$	3\$
"phone"	2\$	3\$	0\$

$$\rightarrow C = \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ 1 & 0 & 0 \end{bmatrix}$$

- Given W, $\max_{C \in \mathbb{B}^{n\times n}} \sum_{ij} W_{ij} C_{ij}$ such that $\sum_i C_{ij} = \sum_j C_{ij} = 1$
- Known as the Hungarian marriage problem $O(n^3)$
- Graphical model is not associative and has many loops
- Max product takes $O(n^3)$ for matching (Bayati *et al.* '05)

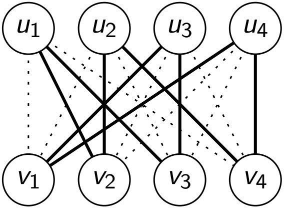
Bipartite Generalized Matching

	Motorola	Apple	IBM
"laptop"	0\$	2\$	2\$
"server"	0\$	2\$	3\$
"phone"	2\$	3\$	0\$

$$\rightarrow C = \begin{bmatrix} 0 & 1 & 1 \\ 1 & 0 & 1 \\ 1 & 1 & 0 \end{bmatrix}$$

- Given W, $\max_{C \in \mathbb{B}^{n \times n}} \sum_{ij} W_{ij} C_{ij}$ such that $\sum_i C_{ij} = \sum_j C_{ij} = b$
- Known as the b-matching problem (used in Google Adwords)
- Graphical model is not associative and has many loops
- Max product takes $O(bn^3)$ for exact MAP (Huang & J '07)

Bipartite Generalized Matching



- Define x_i (y_j) as set of neighbors of u_i (v_j)
- Then $p(X, Y) = \frac{1}{Z} \prod_i \prod_j \psi(x_i, y_j) \prod_k \phi(x_k)\phi(y_k)$ where $\phi(y_j) = \exp(\sum_{u_i \in y_j} W_{ij})$ and $\psi(x_i, y_j) = \neg(v_j \in x_i \oplus u_i \in y_j)$

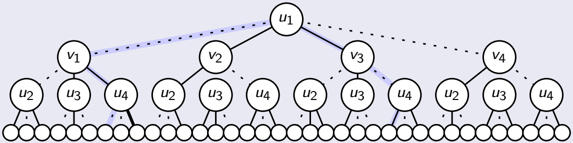
Bipartite Generalized Matching

Theorem (Huang & J 2007)

Max product on G converges in $O(bn^3)$ time.

Proof.

Form unwrapped tree T of depth $\Omega(n)$, maximizing belief at root of T is equivalent to maximizing belief at corresponding node in G



□

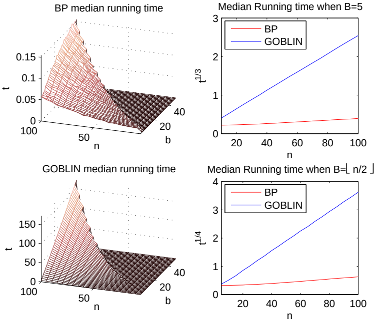
Theorem (Salez & Shah 2009)

Under mild assumptions, max product 1-matching is $O(n^2)$.

Bipartite Generalized Matching

- Code at http://www.cs.columbia.edu/~jebara/code

Generalized Matching



Applications:
unipartite matching
clustering (J & S 2006)
classification (H & J 2007)
collaborative filtering (H & J 2009)
semisupervised (J *et al.* 2009)
visualization (S & J 2009)

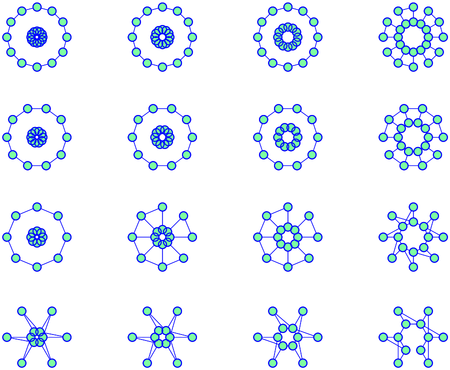
Appears to be $O(n^2)$ on dense graphs (Salez & Shah 2009)

Unipartite Generalized Matching

$$\begin{array}{c|c|c|c|c} & p_1 & p_2 & p_3 & p_4 \\ \hline p_1 & 0 & 2 & 1 & 2 \\ p_2 & 2 & 0 & 2 & 1 \\ p_3 & 1 & 2 & 0 & 2 \\ p_4 & 2 & 1 & 2 & 0 \end{array} \rightarrow C = \begin{bmatrix} 0 & 1 & 0 & 1 \\ 1 & 0 & 1 & 0 \\ 0 & 1 & 0 & 1 \\ 1 & 0 & 1 & 0 \end{bmatrix}$$

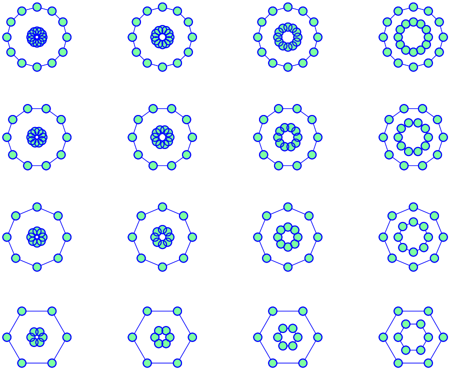
- $\max_{C \in \mathbb{B}^{n \times n}, C_{ii}=0} \sum_{ij} W_{ij} C_{ij}$ such that $\sum_i C_{ij} = b$, $C_{ij} = C_{ji}$
- Combinatorial unipartite matching is efficient (Edmonds '65)
- Makes an LP with exponentially many blossom inequalities
- Max product exact if blossomless LP integral (Sanghavi '08)
- Reuse b-matching code from bipartite case (Huang and J '07)

Unipartite Generalized Matching



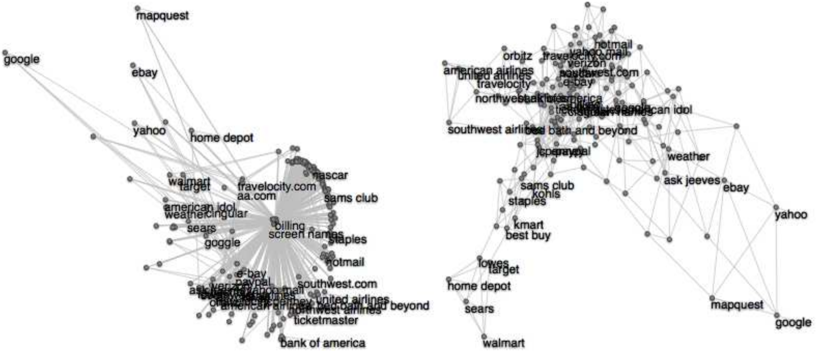
- Above is k-nearest neighbors with $k = 2$

Unipartite Generalized Matching



- Above is unipartite b-matching with $b = 2$

Unipartite Generalized Matching



- Left is k-nearest neighbors, right is unipartite b-matching.

Back to Perfect Graphs

- Max product and exact MAP depend on the LP's integrality
- Matchings have special integral LPs (Edmonds '65)
- How to generalize beyond matchings?
- Perfect graphs imply LP integrality (Lovász '72)

Lemma (Lovász '72)

For every non-negative vector $\vec{w} \in \mathbb{R}^N$, the linear program

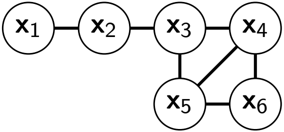
$$\beta = \max_{\vec{x} \in \mathbb{R}^N} \vec{w}^\top \vec{x} \text{ subject to } \vec{x} \geq 0 \text{ and } A\vec{x} \leq \vec{1}$$

recovers a vector $\vec{x}$ which is integral if and only if the (undominated) rows of A form the vertex versus maximal cliques incidence matrix of some perfect graph.

Back to Perfect Graphs

Lemma (Lovász '72)

$$\beta \;=\; \max_{\vec{x}\in\mathbb{R}^N} \vec{w}^\top \vec{x} \text{ subject to } \vec{x} \geq 0 \text{ and } A\vec{x} \leq \vec{1}$$



$$A \;=\; \begin{bmatrix} 1 & 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 1 & 1 & 0 \\ 0 & 0 & 0 & 1 & 1 & 1 \end{bmatrix}$$

nand Markov Random Fields

- Lovász's lemma is not solving $\max p(X)$ on G
- How to apply the lemma to any model G and space X?
- We have $p(x_1, \ldots, x_n) = \frac{1}{Z} \prod_{c \in C} \psi_c(X_c)$
- Without loss of generality assume $\psi_c(X_c) \leftarrow \frac{\psi_c(X_c)}{\min_{X_c} \psi_c(X_c)} + \epsilon$
- Consider procedure to G to $\mathcal{G}$ in NMRF form
- NMRF is a nand Markov random field over space $\mathbf{X}$
  - all variables are binary $\mathbf{X} = \{\mathbf{x}_1, \ldots, \mathbf{x}_N\}$
  - all potential functions are pairwise nand gates $\Phi(\mathbf{x}_i, \mathbf{x}_j) = \delta[\mathbf{x}_i + \mathbf{x}_j \leq 1]$

nand Markov Random Fields

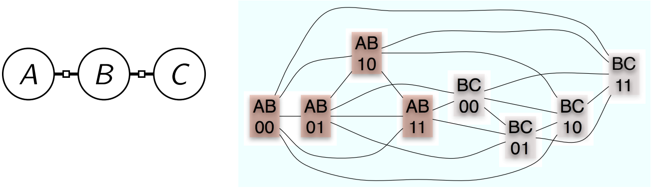


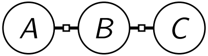
Figure: Binary graphical model G (left) and nand MRF $\mathcal{G}$ (right).

```
Initialize 𝒢 as the empty graph
For each clique c in graph G do
    For each configuration k ∈ X_c do
        add a corresponding binary node x_{c,k} to 𝒢
        for each x_{d,l} ∈ 𝒢 which is incompatible with x_{c,k}
            connect x_{c,k} and x_{d,l} with an edge
```

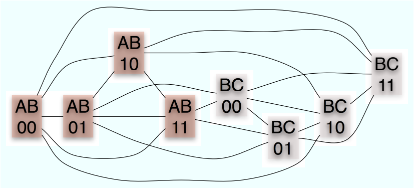
Figure: Algorithm to convert G into a NMRF $\mathcal{G}$

nand Markov Random Fields

G encoding $p(X)$



$\mathcal{G}$ encoding $\rho(\mathbf{X})$



$\prod_{c \in C} \psi_c(X_c)$

$\prod_{i \in \mathsf{V}(\mathcal{G})} e^{\mathbf{w}_i \mathbf{x}_i} \prod_{(i,j) \in \mathsf{E}(\mathcal{G})} (1 - \mathbf{x}_i \mathbf{x}_j)$

- $\mathcal{G}$ has N binary $\mathbf{x}_{c,k}$ with weights $\mathbf{w}_{c,k} = \log \psi_c(X_c = k)$
- Clique c is in configuration k in G $\iff$ $\mathbf{x}_{c,k} = 1$ in $\mathcal{G}$

Lemma (J '09)

The MAP estimate for $\rho(\mathbf{X})$ on $\mathcal{G}$ recovers MAP for $p(X)$ on G

Packing Linear Programs

- MAP on $\rho(\mathbf{X})$ is just Maximum Weight Stable Set (MWSS)
- MWSS is NP-hard in general but P if graph $\mathcal{G}$ is perfect
- Relaxed MAP on $\log \rho(\mathbf{X}) \equiv$ set packing linear program
- If graph $\mathcal{G}$ is perfect, LP is integral

Lemma (Lovász '72)

For every non-negative vector $\vec{w} \in \mathbb{R}^N$, the linear program

$$\beta = \max_{\vec{x} \in \mathbb{R}^N} \vec{w}^\top \vec{x} \text{ subject to } \vec{x} \geq 0 \text{ and } A\vec{x} \leq \vec{1}$$

recovers a vector $\vec{x}$ which is integral if and only if the (undominated) rows of A form the vertex versus maximal cliques incidence matrix of some perfect graph.

Packing Linear Programs

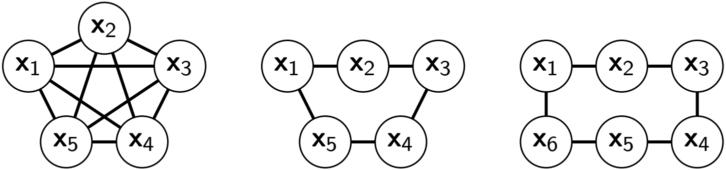
- For general graph G, MAP is NP-hard (Shimony '94) but...
- Convert G to $\mathcal{G}$ (polynomial time)
- If graph $\mathcal{G}$ is perfect (polynomial time)
  - Solve MAP via MWSS (polynomial time)
    - ...by finding cliques C and solving LP in $O(\sqrt{|C|}N^3)$
    - ...by Lovász theta function semidefinite program in $O(N^5)$

Theorem (J '09)

MAP estimation of any graphical model G with cliques $c \in C$ over variables $\{x_1, \ldots, x_n\}$ producing a nand Markov random with a perfect graph $\mathcal{G}$ is in P and requires no more than $O\left(\left(\sum_{c \in C}\left(\prod_{i \in c}|x_i|\right)\right)^5\right)$.

Perfect Graphs

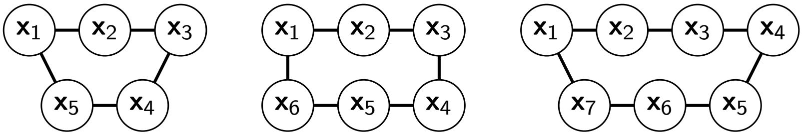
- To determine if $\mathcal{G}$ is perfect
  - Run algorithm on $\mathcal{G}$ in $O(N^9)$ (Chudnovsky *et al.* '05)
  - or use tools from perfect graph theory to prove perfection
- Clique number of a graph $\omega(\mathcal{G})$: size of its maximum clique
- Chromatic number of a graph $\chi(\mathcal{G})$: minimum number of colors such that no two adjacent vertices have the same color
- A perfect graph $\mathcal{G}$ is a graph where every induced subgraph $\mathcal{H} \subseteq \mathcal{G}$ has $\omega(\mathcal{H}) = \chi(\mathcal{H})$



Perfect — Not Perfect — Perfect

Strong Perfect Graph Theorem

- A graph is perfect iff it is Berge (Chudnovsky *et al.* '03)
- Berge graph: a graph that contains no odd hole and whose complement also contains no odd hole
- Hole: an induced subgraph of $\mathcal{G}$ which is a chordless cycle of length at least 5. An odd hole has odd cycle length.
- Complement: a graph $\bar{\mathcal{G}}$ with the same vertex set $V(\mathcal{G})$ as $\mathcal{G}$, where distinct vertices $\mathbf{u}, \mathbf{v} \in V(\mathcal{G})$ are adjacent in $\bar{\mathcal{G}}$ just when they are not adjacent in $\mathcal{G}$



odd hole even hole odd hole

Recognition using Strong Perfect Graph Theorem

- SPGT implies that a Berge graph is one of these primitives
  - bipartite graphs
  - complements of bipartite graphs
  - line graphs[1] of bipartite graphs
  - complements of line graphs of bipartite graphs
  - double split graphs
- or decomposes structurally (into graph primitives)
  - via a 2-join
  - via a 2-join in the complement
  - via an M-join
  - via a balanced skew partition

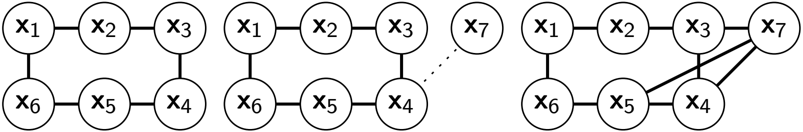
[1] $L(\mathcal{G})$: a line graph contains a vertex for each edge of $\mathcal{G}$ where vertices are adjacent iff they correspond to edges of $\mathcal{G}$ with a common end vertex.

Recognition using Perfect Graph Theory

- Many tools (joins, decompositions,...) to prove perfection

Lemma (Replication, Lovász '72)

Let $\mathcal{G}$ be a perfect graph and let $v \in V(\mathcal{G})$. Define a graph $\mathcal{G}'$ by adding a new vertex v' and joining it to v and all neighbors of v. Then $\mathcal{G}'$ is perfect.

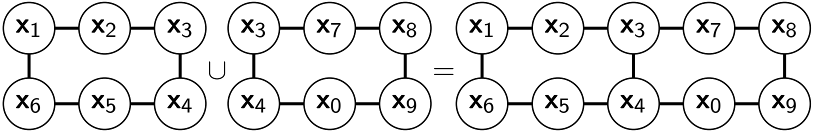


Recognition using Strong Perfect Graph Theorem

- Many tools (joins, decompositions,...) to prove perfection

Lemma (Gluing on Cliques, Skew Partition, Berge & Chvátal '84)

Let $\mathcal{G}$ be a perfect graph and let $\mathcal{G}'$ be a perfect graph. If $\mathcal{G} \cap \mathcal{G}'$ is a clique (clique cutset), then $\mathcal{G} \cup \mathcal{G}'$ is a perfect graph.



Proving Exact MAP for Tree Graphs

Theorem (J '09)

Let G be a tree, the NMRF $\mathcal{G}$ obtained from G is a perfect graph.

Proof.

Consider star graph with internal node v with $|v|$ configurations.
Obtain $\mathcal{G}$ by creating one configuration for non internal nodes.
Resulting graph is a complete $|v|$-partite graph which is perfect.
Add configurations for non-internal nodes using replication lemma.
Resulting $\mathcal{G}_{star}$ is perfect.
Obtain a tree by induction, add stars $\mathcal{G}_{star}$ and $\mathcal{G}_{star'}$.
Intersection is clique cutset so result is perfect.
Continue gluing all stars in G. □

Proving Exact MAP for Bipartite Matchings

Theorem (J '09)

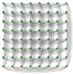
The maximum weight bipartite matching graphical model

$$p(X) = \prod_{i=1}^{n} \delta\left[\sum_{j=1}^{n} x_{ij} \leq 1\right] \delta\left[\sum_{j=1}^{n} x_{ji} \leq 1\right] \prod_{k=1}^{n} e^{f_{ik} x_{ik}}$$

with $f_{ij} \geq 0$ *has integral LP and yields exact MAP estimates.*

Proof.

The graphical model is in NMRF form so G and $\mathcal{G}$ are equivalent.
$\mathcal{G}$ is the line graph of a (complete) bipartite graph (Rook's graph).
Therefore, $\mathcal{G}$ is perfect, the LP is integral and recovers MAP. $\square$



Proving Exact MAP for Unipartite Matchings

Theorem (J '09)

The unipartite matching graphical model $G = (V, E)$ with $f_{ij} \geq 0$

$$p(X) = \prod_{i \in V} \delta\left[\sum_{j \in Ne(i)}^{n} x_{ij} \leq 1\right] \prod_{ij \in E} e^{f_{ij} x_{ij}}$$

has integral LP and produces the exact MAP estimate if G is a perfect graph.

Proof.

The graphical model is in NMRF form and graphs G and $\mathcal{G}$ are equivalent. The set packing LP relaxation is integral and recovers the MAP estimate if $\mathcal{G}$ is a perfect graph. $\square$

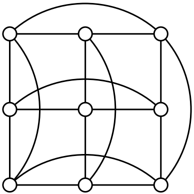
Proving Exact MAP for Associative MRFs

Theorem (Greig, Porteous, and Seheult (1989))

MAP for associative Markov random fields is in P via mincut.

Proof via Perfect Graphs.

The resulting NMRF $\mathcal{G}$ from G is bipartite and hence perfect. □



Convergent Message Passing (Globerson & Jaakkola '07)

- LPs are slow. Try convergent message passing on $\mathcal{G}$ instead.

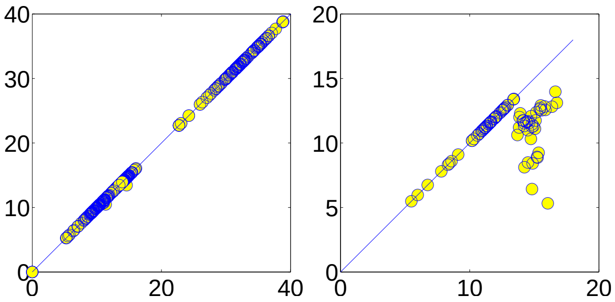
Input: NMRF $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ and weights w_i for $i \in \mathcal{V}$.
Find all maximal cliques $\mathcal{C}$ in $\mathcal{G}$
Until converged do

$$\lambda_{i,\mathbf{c}} = \frac{1-|\mathbf{c}|}{|\mathbf{c}|} \sum_{\mathbf{c}' \in C \setminus \mathbf{c} : i \in \mathbf{c}'} \lambda_{i,\mathbf{c}'} + \frac{1}{|\mathbf{c}|} \frac{w_i}{\sum_{\mathbf{c} \in C} [i \in \mathbf{c}]}$$
$$- \frac{1}{|\mathbf{c}|} \max\left[0, \max_{i' \in \mathbf{c} \setminus i} \left[\frac{w_{i'}}{\sum_{\mathbf{c} \in C} [i' \in \mathbf{c}]} + \sum_{\mathbf{c}' \in C \setminus \mathbf{c} : i' \in \mathbf{c}'} \lambda_{i',\mathbf{c}'} \right]\right]$$

Theorem (J '09)

Convergent message passing on perfect NMRFs solves MAP.

MAP Experiments for Unipartite Matching



(a) Perfect Graphs (b) Random Graphs

Figure: Scores for the exact MAP estimate (horizontal axis) and message passing estimate (vertical axis) for random graphs and weights. Figure (a) shows scores for four types of basic Berge graphs while (b) shows scores for arbitrary graphs. Minor score discrepancies on Berge graphs arose due to numerical issues and early stopping.

Conclusions

- Perfect graph theory is fascinating, many recent breakthroughs
- A crucial tool for exploring LP integrality and MAP estimation
- Solve MAP on NMRF $\mathcal{G}$ as Max Weight Stable Set problem
- If graph $\mathcal{G}$ is perfect, MWSS is polynomial
    - ...via Lovász theta function semidefinite program
    - ...via linear programming or message passing
    - ...future work: weighted SAT solvers when not perfect
- Exact MAP and message passing applies to
    - Trees and singly-linked graphs
    - Single loop graphs
    - Matchings
    - Generalized matchings
    - *and now* Perfect graphs

Further Reading and Thanks

- MAP Estimation, Message Passing, and Perfect Graphs, T. Jebara. *Uncertainty in Artificial Intelligence*, June 2009.
- Graphical Models, Exponential Families and Variational Inference, M.J. Wainwright and M.I. Jordan. *Foundations and Trends in Machine Learning*, Vol 1, Nos 1-2, 2008.
- Loopy Belief Propagation for Bipartite Maximum Weight b-Matching, B. Huang and T. Jebara. *Artificial Intelligence and Statistics*, March 2007.
- Thanks to Maria Chudnovsky, Delbert Dueck and Bert Huang.

Pruning NMRFs

- Three pruning procedures on $\mathcal{G}$ to set up a perfect MWSS
- Easy to propagate the solution to final MAP answer
- **Remove**: For each $c \in C$, remove from the MWSS problem the set of nodes $\{\mathbf{x}_{c,k}\}$ such that $f_{c,k} = \min_{\kappa} f_{c,\kappa}$
- **Merge**: For unconnected $\mathbf{x}_{c,k}$ and $\mathbf{x}_{d,l}$ where $\text{Ne}(\mathbf{x}_{c,k}) = \text{Ne}(\mathbf{x}_{d,l})$, keep only $\mathbf{x}_{c,k}$ with weight $f_{c,k} \leftarrow f_{c,k} + f_{d,l}$
- **Anti-Merge**: For connected $\mathbf{x}_{c,k}$ and $\mathbf{x}_{d,l}$ where $\text{Ne}(\mathbf{x}_{c,k}) = \text{Ne}(\mathbf{x}_{d,l})$, keep only $\mathbf{x}_{c,k}$ with weight $f_{c,k} \leftarrow \max(f_{c,k}, f_{d,l})$