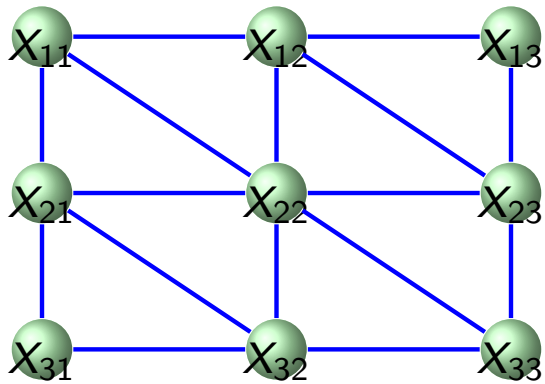
MAP Inference in Undirected Graphical Models: Understanding Polytope Relaxations

Mark Rowland
(joint work with Adrian Weller and David Sontag (NYU))

November 28, 2016

Undirected Graphical Models

A collection of random variables with a joint distribution that factors over an associated graph.



The graph G is a collection of vertices V and edges E.

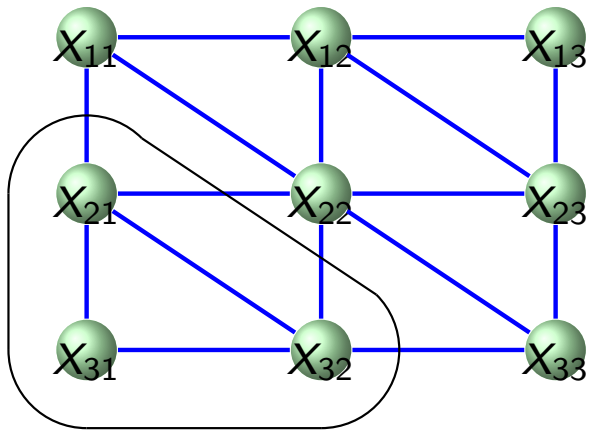
A clique of a graph is a subset $C \subseteq V$ so that any two vertices in C have an edge between them. Write $\mathcal{C}$ for the set of all cliques in G.

The probability distribution of a graphical model with underlying graph G can be written

$$p(x_1, \ldots, x_n) \propto \prod_{C \in \mathcal{C}} \phi_C(x_C) = \exp\left(\sum_{C \in \mathcal{C}} \psi_C(x_C)\right)$$

Undirected Graphical Models

A collection of random variables with a joint distribution that factors over an associated graph.



The graph G is a collection of vertices V and edges E.

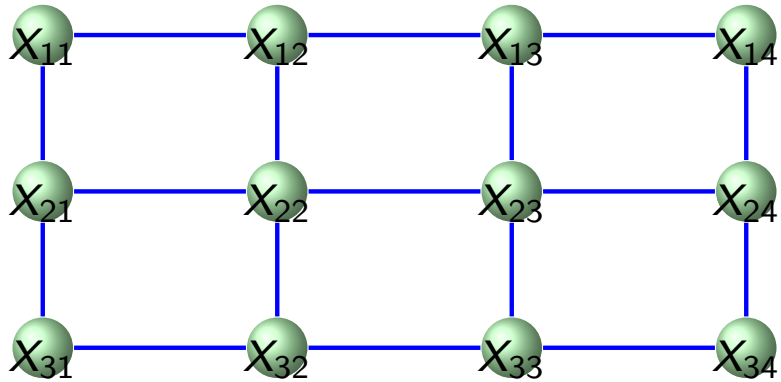
A clique of a graph is a subset $C \subseteq V$ so that any two vertices in C have an edge between them. Write $\mathcal{C}$ for the set of all cliques in G.

The probability distribution of a graphical model with underlying graph G can be written

$$p(x_1, \ldots, x_n) \propto \prod_{C \in \mathcal{C}} \phi_C(x_C) = \exp\left(\sum_{C \in \mathcal{C}} \psi_C(x_C)\right)$$

Applications: Image Denoising

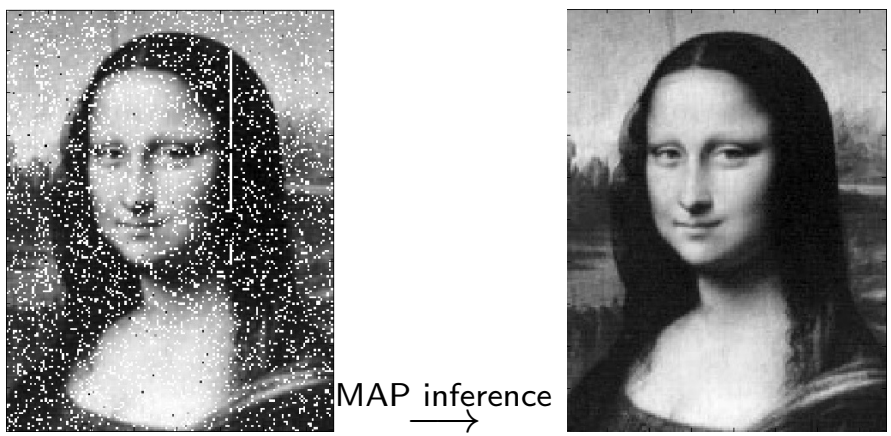
Random variables (X_{ij}) describing pixel intensities



Combine a contiguous prior: $p((x_{ij})) \propto \exp\left(\sum_{(ij)\sim(kl)} W|x_{ij} - x_{kl}|\right)$

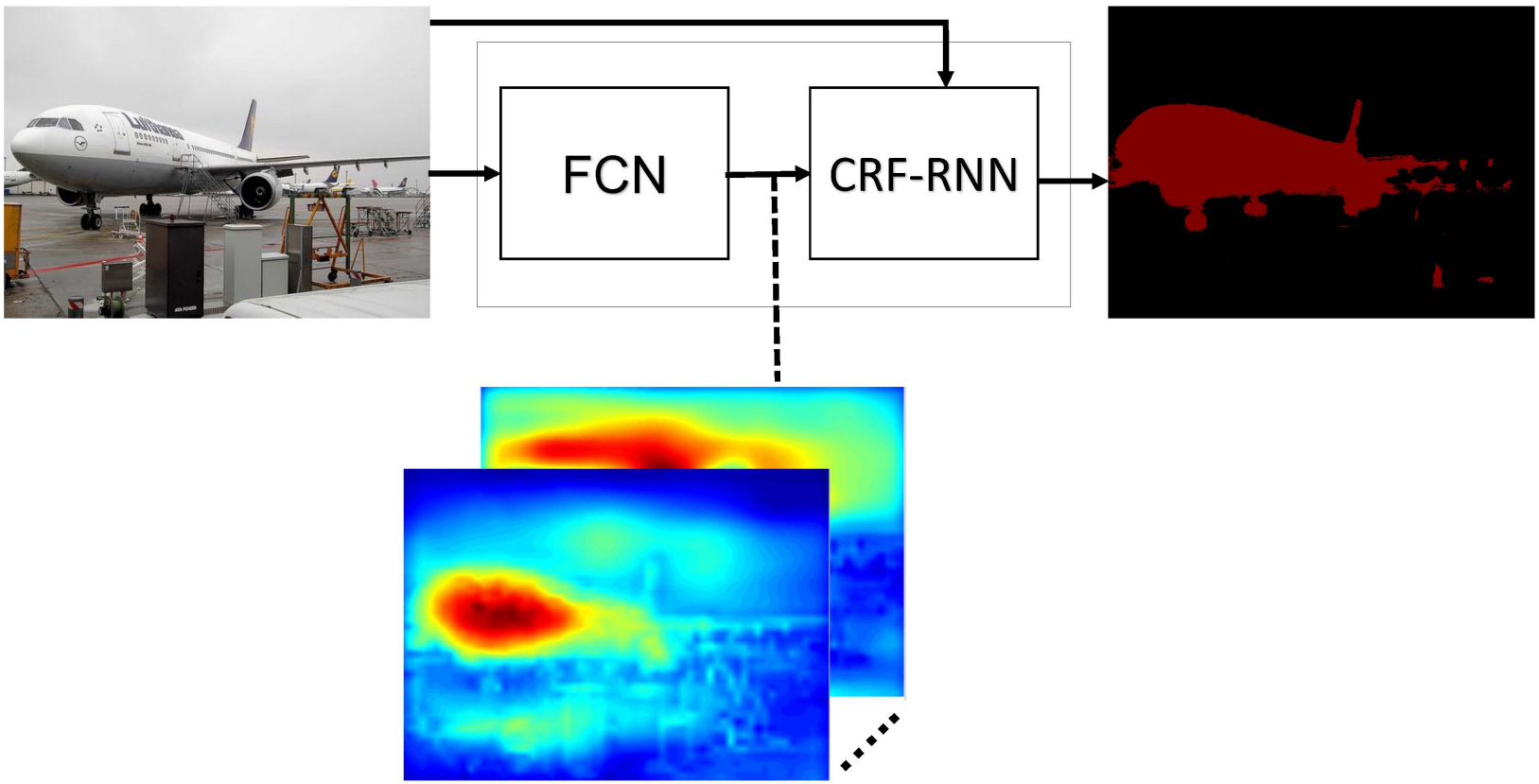
With a likelihood term from an observed image: $p((y_{ij})|(x_{ij})) \propto \exp\left(\sum_{(ij)} |y_{ij} - x_{ij}|\right)$

To obtain a posterior distribution over the latent, non-noisy image. Finding most likely state under posterior:



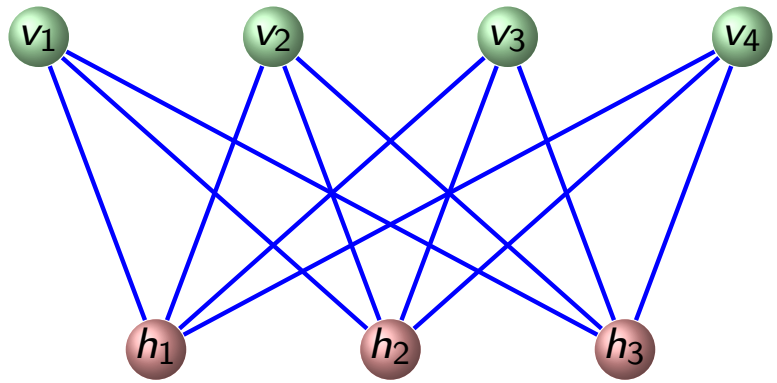
Application: Image Segmentation

CNN for pixel-level labelling, and UGM for image-level segmentation.



[Conditional Random Fields as Recurrent Neural Networks, Zheng et al. (2015)]

Application: Restricted Bolztmann Machines



$$p(v_1, \ldots, v_n, h_1, \ldots, h_m) \propto \exp\left(\sum_{i=1}^{n} a_i v_i + \sum_{j=1}^{m} b_j h_j + \sum_{i=1}^{n}\sum_{j=1}^{m} v_i W_{ij} h_j\right)$$

- Bipartite graph structure - visible and hidden units.
- Learn model parameters so that hidden units capture latent features of observed data

Inference

Given an undirected graphical model

$$p(x_1, \ldots, x_n) \propto \exp\left(\sum_{C \in \mathcal{C}} \psi_C(x_C)\right)$$

we might want to:

- Find the marginal, or conditional, distribution of a subset $S \subset V$ of variables

$$p(x_S) \propto \sum_{i \in V \setminus S} \sum_{x_i \in \mathcal{X}_i} p(x_1, \ldots, x_n) \qquad p(x_S | x_T) \propto \sum_{i \in V \setminus (S \cup T)} \sum_{x_i \in \mathcal{X}_i} p(x_1, \ldots, x_n)$$

- Find the normalising constant

$$Z = \sum_{i \in V} \sum_{x_i \in \mathcal{X}_i} \exp\left(\sum_{C \in \mathcal{C}} \psi_C(x_C)\right)$$

- Find a most likely state (MAP inference)

$$x \in \operatorname*{arg\,max}_{y \in \prod_{i \in V} \mathcal{X}_i} p(y_1, \ldots, y_n)$$

The first two tasks are essentially integration, whilst MAP inference is an optimisation problem.

Binary Pairwise Graphical Models

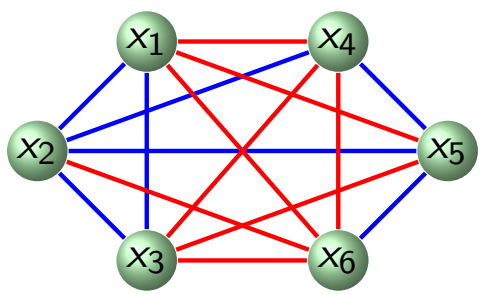
Particular case where

- Each random variable X_i takes values in the set $\{0, 1\}$ (binary)
- The set of cliques $\mathcal{C}$ consists only of single variables and pairs of variables (pairwise)

This leads to a probability distribution of the form

$$p(x_1, \ldots, x_n) \propto \exp\left(\sum_{i \in V} \theta_i x_i + \sum_{ij \in E} W_{ij} x_i x_j\right)$$

- θ_i encodes singleton preferences.
- W_{ij} encodes edge interactions.
  - $W_{ij} > 0$ - attractive.
  - $W_{ij} < 0$ - repulsive.



MAP Inference in Binary Pairwise Models

$$x \in \underset{y \in \prod_{i \in V} \mathcal{X}_i}{\arg\max}\ p(y_1, \ldots, y_n)$$

In a binary pairwise model, we can write

$$p(x_1, \ldots, x_n) \propto \exp\left(\sum_{i \in V} \theta_i x_i + \sum_{ij \in E} W_{ij} x_i x_j\right)$$

So the MAP inference problem becomes

$$\max_{x \in \{0,1\}^V} \left[\sum_{i \in V} \theta_i x_i + \sum_{ij \in E} W_{ij} x_i x_j\right]$$

- Quadratic integer program
- Max-Cut reduces in polynomial time to this QIP, so it is NP-hard
- Brute force solution for 400×800 B+W image: $> 10^{96}$ states to check

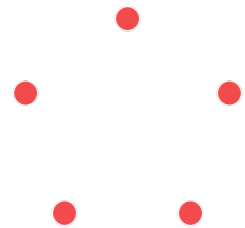
Methods for MAP inference:

- Junction tree algorithm
- Max-product belief propagation (e.g. Viterbi algorithm for HMMs)
- Linear programming

Formulating a linear program via polytope relaxations

Combinatorial optimisation problem

$$\max_{\substack{x\in\{0,1\}^{V\cup E}\\ x_{ij}=x_i x_j \forall ij\in E}} \left[\sum_{i\in V}\theta_i x_i + \sum_{ij\in E} W_{ij}x_{ij}\right]$$



Equivalent linear program

$$\max_{q\in\mathbb{M}} \left[\sum_{i\in V}\theta_i q_i + \sum_{ij\in E} W_{ij}q_{ij}\right]$$

Marginal polytope $\mathbb{M}$: enforce global consistency on (pseudo)marginals $(q_i)_{i\in V}$ and $(q_{ij})_{ij\in E}$

$(q_1,\ldots,q_n,q_{12},\ldots,q_{n-1n})\in\mathbb{M}$ if and only if there is a distribution μ on $X_1,\ldots,X_n$ so that $q_i=\mathbb{P}_\mu(X_i=1)\forall i$, and $q_{ij}=\mathbb{P}_\mu(X_i=X_j=1)\forall ij$.

Relaxed linear program

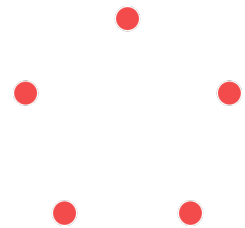
$$\max_{q\in\mathbb{L}_k} \left[\sum_{i\in V}\theta_i q_i + \sum_{ij\in E} W_{ij}q_{ij}\right]$$

Sherali-Adams polytope $\mathbb{L}_k$: enforce consistency over each cluster of k variables on (pseudo)marginals $(q_i)_{i\in V}$ and $(q_{ij})_{ij\in E}$

Formulating a linear program via polytope relaxations

Combinatorial optimisation problem

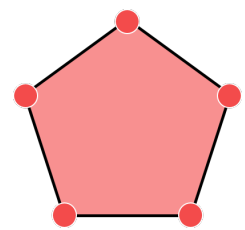
$$\max_{\substack{x\in\{0,1\}^{V\cup E}\\ x_{ij}=x_i x_j \forall ij\in E}} \left[\sum_{i\in V}\theta_i x_i + \sum_{ij\in E} W_{ij}x_{ij}\right]$$



Equivalent linear program

$$\max_{q\in\mathbb{M}} \left[\sum_{i\in V}\theta_i q_i + \sum_{ij\in E} W_{ij}q_{ij}\right]$$

Marginal polytope $\mathbb{M}$: enforce global consistency on (pseudo)marginals $(q_i)_{i\in V}$ and $(q_{ij})_{ij\in E}$



$(q_1,\ldots,q_n,q_{12},\ldots,q_{n-1n})\in\mathbb{M}$ if and only if there is a distribution μ on $X_1,\ldots,X_n$ so that $q_i=\mathbb{P}_\mu(X_i=1)\forall i$, and $q_{ij}=\mathbb{P}_\mu(X_i=X_j=1)\forall ij$.

Relaxed linear program

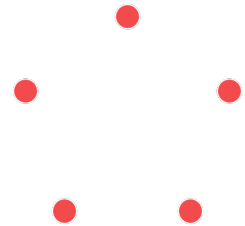
$$\max_{q\in\mathbb{L}_k} \left[\sum_{i\in V}\theta_i q_i + \sum_{ij\in E} W_{ij}q_{ij}\right]$$

Sherali-Adams polytope $\mathbb{L}_k$: enforce consistency over each cluster of k variables on (pseudo)marginals $(q_i)_{i\in V}$ and $(q_{ij})_{ij\in E}$

Formulating a linear program via polytope relaxations

Combinatorial optimisation problem

$$\max_{\substack{x\in\{0,1\}^{V\cup E}\\ x_{ij}=x_i x_j \forall ij\in E}} \left[\sum_{i\in V}\theta_i x_i + \sum_{ij\in E} W_{ij}x_{ij}\right]$$

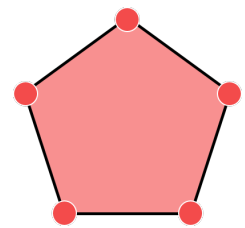


Equivalent linear program

$$\max_{q\in\mathbb{M}} \left[\sum_{i\in V}\theta_i q_i + \sum_{ij\in E} W_{ij}q_{ij}\right]$$

Marginal polytope $\mathbb{M}$: enforce global consistency on (pseudo)marginals $(q_i)_{i\in V}$ and $(q_{ij})_{ij\in E}$

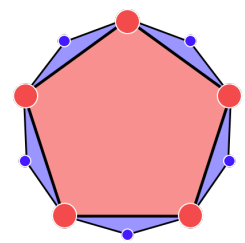
$(q_1,\ldots,q_n,q_{12},\ldots,q_{n-1n})\in\mathbb{M}$ if and only if there is a distribution μ on $X_1,\ldots,X_n$ so that $q_i=\mathbb{P}_\mu(X_i=1)\forall i$, and $q_{ij}=\mathbb{P}_\mu(X_i=X_j=1)\forall ij$.



Relaxed linear program

$$\max_{q\in\mathbb{L}_k} \left[\sum_{i\in V}\theta_i q_i + \sum_{ij\in E} W_{ij}q_{ij}\right]$$

Sherali-Adams polytope $\mathbb{L}_k$: enforce consistency over each cluster of k variables on (pseudo)marginals $(q_i)_{i\in V}$ and $(q_{ij})_{ij\in E}$

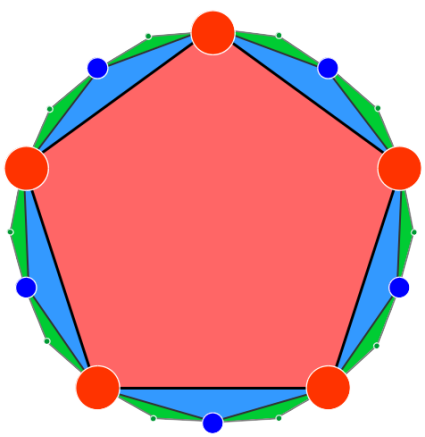


The Sherali-Adams hierarchy of polytope relaxations

← Computationally cheaper

More often exact →

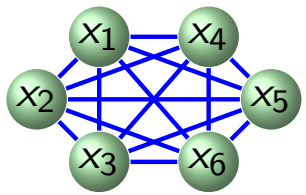
$\mathbb{L}_n = \mathbb{M}$	Enforces global consistency	$\max_{q \in \mathbb{L}_n} \left[\sum_{i \in V} \theta_i q_i + \sum_{ij \in E} W_{ij} q_{ij} \right]$
$\vdots$	$\vdots$	$\vdots$
$\mathbb{L}_3$	Enforces triplet consistency	$\max_{q \in \mathbb{L}_3} \left[\sum_{i \in V} \theta_i q_i + \sum_{ij \in E} W_{ij} q_{ij} \right]$
$\mathbb{L}_2$	Enforces pairwise consistency	$\max_{q \in \mathbb{L}_2} \left[\sum_{i \in V} \theta_i q_i + \sum_{ij \in E} W_{ij} q_{ij} \right]$

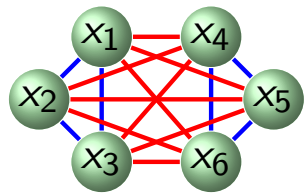


Results

Already known:

- $\mathbb{L}_2$ is exact for tree-structured graphical models.
- $\mathbb{L}_2$ is exact for attractive (and more generally, balanced) models.

attractive: 

balanced: 

We prove:

- $\mathbb{L}_3$ is exact for almost attractive and almost balanced models.
- There are certain "model pasting" operations that preserve exactness of relaxations, allowing us to extend these results further.

almost attractive:

almost balanced:

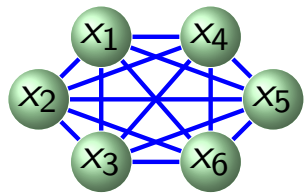
Future research:

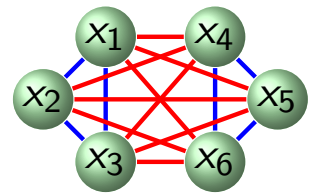
- Extend proof techniques to deal with general polytope relaxations $\mathbb{L}_k$, multi-label models, and higher-order interactions.

Results

Already known:

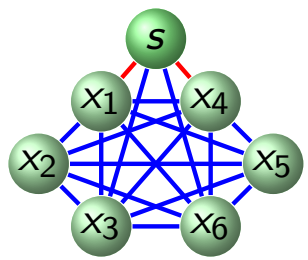
- $\mathbb{L}_2$ is exact for tree-structured graphical models.
- $\mathbb{L}_2$ is exact for attractive (and more generally, balanced) models.

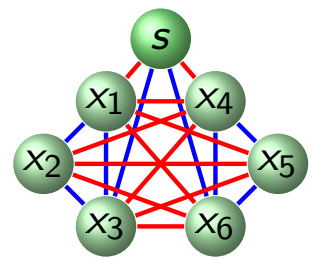
attractive: 

balanced: 

We prove:

- $\mathbb{L}_3$ is exact for almost attractive and almost balanced models.
- There are certain "model pasting" operations that preserve exactness of relaxations, allowing us to extend these results further.

almost attractive: 

almost balanced: 

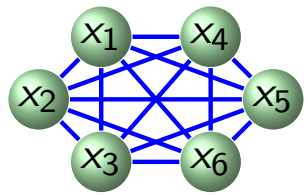
Future research:

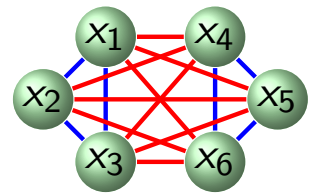
- Extend proof techniques to deal with general polytope relaxations $\mathbb{L}_k$, multi-label models, and higher-order interactions.

Results

Already known:

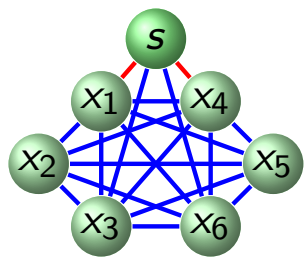
- $\mathbb{L}_2$ is exact for tree-structured graphical models.
- $\mathbb{L}_2$ is exact for attractive (and more generally, balanced) models.

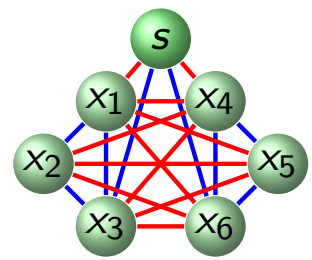
attractive: 

balanced: 

We prove:

- $\mathbb{L}_3$ is exact for almost attractive and almost balanced models.
- There are certain "model pasting" operations that preserve exactness of relaxations, allowing us to extend these results further.

almost attractive: 

almost balanced: 

Future research:

- Extend proof techniques to deal with general polytope relaxations $\mathbb{L}_k$, multi-label models, and higher-order interactions.

Thank you!