Graphical Models and CSP

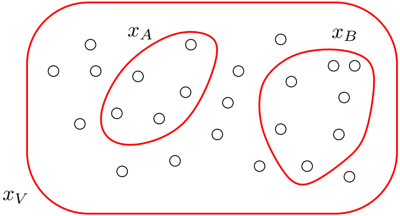
Tomáš Werner

Center for Machine Perception
Faculty of Electrical Engineering
Czech Technical University in Prague

The 25th International Conference on Principles and Practice of Constraint Programming
Stamford, October 1 2019

It is Good to Have a Joint!

Variables $V = \{1, \dots, n\}$. Variable $i \in V$ takes values $x_i \in D_i$ from a finite domain D_i.
Joint probability distribution $p(x_V) = p(x_1, \dots, x_n)$ captures all info about our system!



For $A \subseteq V$, denote $x_A = (x_i)_{i \in V} \in D_A = \prod_{i \in A} D_i$.

- Given $A \subseteq V$, compute marginals $p(x_A) = \sum_{x_{V \setminus A}} p(x_V)$
- Given $A, B \subseteq V$ and x_B, compute conditional $p(x_A \,|\, x_B)$
- Find the mode (= maximum) of $p(x_V)$ or of its marginal or conditional
- Sample from $p(x_V)$ or from its marginal or conditional
- Given $A, B \subseteq V$ and x_B, infer the 'most likely' configuration x_A
- Having a family $\{\, p_\theta(x_V) \mid \theta \in \Theta \,\}$, learn θ from training data

Undirected Graphical Model = Gibbs Distribution

Gibbs distribution with hypergraph $H \subseteq 2^V$ and potentials $\psi\colon D_A \to \mathbb{R}_+$:

$$p(x_V) = \frac{1}{Z} \prod_{A \in H} \psi_A(x_A) \qquad \text{where} \quad Z = \sum_{x_V \in D_V} \prod_{A \in H} \psi_A(x_A)$$

Examples:

- $V = \{1,2,3,4\}$, $H = \{\{2,3,4\},\{1,2\},\{3,4\},\{3\}\}$:

$$p(x_1,x_2,x_3,x_4) \propto \psi_{234}(x_2,x_3,x_4)\,\psi_{12}(x_1,x_2)\,\psi_{34}(x_3,x_4)\,\psi_3(x_3)$$

- Distribution of arity 2 ('pairwise') with graph $E \subseteq \binom{V}{2}$:

$$p(x_V) \propto \prod_{i \in V} \psi_i(x_i) \prod_{\{i,j\} \in E} \psi_{ij}(x_i,x_j)$$

- Ising ($D_i = \{0,1\}$) and Potts ($D_i = \{1,\dots,k\}$) model:

$$p(x_V) \propto \prod_{\{i,j\} \in E} c_{ij} [\![x_i = x_j]\!]$$

Maximum Entropy Property

Recall: The marginal distribution of $p(x_V)$ on variables $A \subseteq V$ is

$$p(x_A) = \sum_{x_{V \setminus A}} p(x_V)$$

Fact: Gibbs distribution $p(x_V)$ has maximum entropy among all distributions with given marginals over H:

$$p(x_A) = p_A(x_A), \qquad A \in H,\ x_A \in D_A$$

(Potentials ψ_A appear as Lagrange multipliers.)

Computing potentials ψ_A from marginals p_A ('moment matching'):

- ψ_A are unique (up to reparameterizations)
- not possible in closed form
- iterative algorithms: Iterative Proportional Fitting (IPF)

Application: ML learning of potentials from an i.i.d. sample from $p(x_V)$.

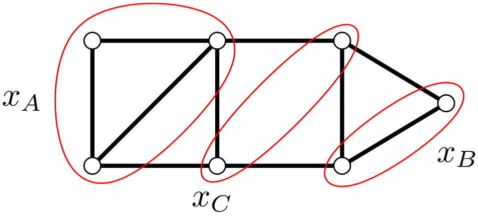
- Given an undirected graph $E \subseteq \binom{V}{2}$
- A distribution $p(x_V)$ has a (global) Markov property if

$$C \text{ separates } A \text{ and } B \text{ in } E \quad \Longrightarrow \quad p(x_A, x_B \,|\, x_C) = p(x_A \,|\, x_C) p(x_B \,|\, x_C)$$

for all $A, B, C \subseteq V$.

Hammersley-Clifford: For every positive distribution $p(x_V)$, TFAE:

- $p(x_V)$ satisfies Markov property (= is a MRF) w.r.t. E.
- $p(x_V)$ is a Gibbs distribution with H being the (maximal) cliques of E.



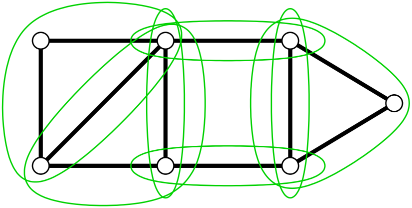
- Given an undirected graph $E \subseteq \binom{V}{2}$
- A distribution $p(x_V)$ has a (global) Markov property if

$$C \text{ separates } A \text{ and } B \text{ in } E \quad \Longrightarrow \quad p(x_A, x_B \,|\, x_C) = p(x_A \,|\, x_C) p(x_B \,|\, x_C)$$

for all $A, B, C \subseteq V$.

Hammersley-Clifford: For every positive distribution $p(x_V)$, TFAE:

- $p(x_V)$ satisfies Markov property (= is a MRF) w.r.t. E.
- $p(x_V)$ is a Gibbs distribution with H being the (maximal) cliques of E.



Let $A \cup B = V$ and $A \cap B = \emptyset$. Inference: Given x_B, infer x_A.

1. Form the posterior $p(x_A \,|\, x_B)$ (= a Gibbs distribution over a smaller hypergraph)
2. Find the 'most likely' assignment x_A:

$$x_A^* \in \operatorname*{argmin}_{x_A} \sum_{y_A} p(y_A \,|\, x_B)\, \ell(x_A, y_A)$$

Two natural loss functions:

- $\ell(x_A, y_A) = -[\![x_A = y_A]\!] \implies$ maximum aposteriori (MAP) inference:

$$x_A^* \in \operatorname*{argmax}_{x_A} \sum_{y_A} p(y_A \,|\, x_B)\, [\![x_A = y_A]\!] = \operatorname*{argmax}_{x_A} p(x_A \,|\, x_B)$$

- $\ell(x_A, y_A) = -\sum_{i \in A} [\![x_i = y_i]\!] \implies$ maximum posterior marginal inference:

$$x_A^* \in \operatorname*{argmax}_{x_A} \sum_{i \in A} \underbrace{\sum_{y_A} p(y_A \,|\, x_B)\, [\![x_i = y_i]\!]}_{p(x_i \,|\, x_B)}$$

$$x_i^* \in \operatorname*{argmax}_{x_i} p(x_i \,|\, x_B) \quad \forall i \in A$$

Let $A \cup B = V$ and $A \cap B = \emptyset$. Inference: Given x_B, infer x_A.

1. Form the posterior $p(x_A \,|\, x_B)$ (= a Gibbs distribution over a smaller hypergraph)
2. Find the 'most likely' assignment x_A:

$$x_A^* \in \operatorname*{argmin}_{x_A} \sum_{y_A} p(y_A \qquad) \, \ell(x_A, y_A)$$

Two natural loss functions:

- $\ell(x_A, y_A) = -[\![x_A = y_A]\!] \quad \Longrightarrow$ maximum aposteriori (MAP) inference:

$$x_A^* \in \operatorname*{argmax}_{x_A} \sum_{y_A} p(y_A \qquad) \, [\![x_A = y_A]\!] = \operatorname*{argmax}_{x_A} p(x_A \qquad)$$

- $\ell(x_A, y_A) = -\sum_{i \in A} [\![x_i = y_i]\!] \quad \Longrightarrow$ maximum posterior marginal inference:

$$x_A^* \in \operatorname*{argmax}_{x_A} \sum_{i \in A} \underbrace{\sum_{y_A} p(y_A \qquad) \, [\![x_i = y_i]\!]}_{p(x_i \qquad)}$$

$$x_i^* \in \operatorname*{argmax}_{x_i} p(x_i \qquad) \quad \forall i \in A$$

Gibbs Distribution in Exponential Form

$$p(x_V) \propto \prod_{A \in H} \psi_A(x_A) = \mathrm{e}^{F(x_V)} \qquad \text{where} \quad F(x_V) = \sum_{A \in H} f_A(x_A)$$

and $f_A \colon D_A \to \overline{\mathbb{R}} = \mathbb{R} \cup \{-\infty\}$ are given by $f_A(x_A) = \log \psi_A(x_A)$.

VCSP (a.k.a. discrete energy minimization):

$$\operatorname*{argmax}_{x_V} p(x_V) = \operatorname*{argmax}_{x_V} F(x_V)$$

- Sometimes also temperature $t > 0$:

  $$p_t(x_V) \propto \mathrm{e}^{F(x_V)/t}$$

  Statistical physics: $F(x_V)$ is (up to constant) negative energy of the system at thermal equilibrium at microstate x_V.

- Zero temperature limit: $\lim_{t \to 0+} p_t(x_V) > 0 \iff x_V \in \operatorname*{argmax}_{x_V} F(x_V)$

GM in Computer Vision before 2000

Multidiscplinary topic: statistics, statistical physics, machine learning, computer vision, optimization, AI, OR, signal processing, control, ...

In computer vision (+ machine learning?):

- iterated conditional modes (ICM) method for MAP inference: very poor
- Introducing MRFs to computer vision, annealed Gibbs sampler [Geman and Geman, 1984]
- mean field (from statistical physics) [Mezard and Montanari, 2009], [Wainwright and Jordan, 2008]
- Belief propagation/revision (= sum-product/max-product algorithm), junction tree alg. [Pearl, 1988]:
  - exactly computing (max-)marginals for bounded treewidth
  - for any commutative semiring [Aji and McEliece, 2000]
- Loopy belief propagation [Pearl, 1988], [Murphy et al., 1999]:
  - empirically, BP often approximates marginals even on cyclic graphs

Want a sequence of samples from $p(x_V)$.

- Simple MCMC: Choose $i \in V$ and sample new x_i from $p(x_i \mid x_{V \setminus \{i\}})$.
- For Gibbs distribution, known as Gibbs sampler.

Drawbacks:

- Often mixes slowly (infinitely slowly for crisp constraints).
- The samples are very dependent.

Towards curing both problems: perturb-and-MAP sampling [Hazan and Jaakkola, 2012]

- Perturb parameters of p randomly (in a clever way...).
- Find a maximizer x_V of p.

Let $A, B \in H$ and $B \subseteq A$. For any function $\lambda\colon D_B \to \mathbb{R}$,

$$f_A(x_A) + f_B(x_B) = \underbrace{f_A(x_A) + \lambda(x_B)}_{f'_A(x_A)} + \underbrace{f_B(x_B) - \lambda(x_B)}_{f'_B(x_B)}.$$

Hence, replacing (f_A, f_B) with (f'_A, f'_B) preserves the function $F(x_V) = \sum_{A \in H} f_A(x_A)$.

- This is a reparameterization of a single pair (f_A, f_B).
- More complex reparameterizations of $F(x_V)$: compose reparameterizations for different pairs ($\implies$ linear transformation of the weight vector f).

For functions of arity 2,

$$F(x_V) = \sum_{i \in V} f_i(x_i) + \sum_{\{i,j\} \in E} f_{ij}(x_i, x_j),$$

reparameterization of a pair (f_i, f_{ij}) reads

$$f_{ij}(x_i, x_j) + f_i(x_i) = \underbrace{f_{ij}(x_i, x_j) + \lambda(x_i)}_{f'_{ij}(x_i, x_j)} + \underbrace{f_i(x_i) - \lambda(x_i)}_{f'_i(x_i)}.$$

Let E be a tree. Iteration of Belief Revision: Reparameterize (f_i, f_{ij}) to enforce

$$\max_{x_j}(f_{ij}(x_i, x_j) + f_j(x_j)) = 0$$

That is: Find unary function $\lambda\colon D_i \to \mathbb{R}$ such that $\max_{x_j}(f_{ij}(x_i, x_j) + \lambda(x_i) + f_j(x_j)) = 0$. Hence $\lambda(x_i) = -\max_{x_j}(f_{ij}(x_i, x_j) + f_j(x_j))$.

After two passes (from/to a root) it holds globally. This exposes max-marginals:

$$\max_{x_{V \setminus i}} F(x_V) = f_i(x_i), \qquad \max_{x_{V \setminus \{i,j\}}} F(x_V) = f_i(x_i) + f_{ij}(x_i, x_j) + f_j(x_j)$$

Works for any commutative semiring, not only $(\overline{\mathbb{R}}, \max, +)$ [Aji and McEliece, 2000]:

- semiring $(\mathbb{R}_+, +, \times)$: 'sum-product algorithm', belief propagation
- semiring $(\mathbb{R}_+, \max, \times)$: 'max-product algorithm', belief revision

Loopy BP: If E has loops, repeatedly enforcing

$$\max_{x_j}(f_{ij}(x_i, x_j) + f_j(x_j)) = \text{const}_{ij}$$

often converges. Then, max-marginals are exposed (up to constants) in every subtree!
$\implies$ approximate max-marginals

Graph Cut Revolution in Vision (2000-2005)

- reduction for VCSPs with $D_i = \{0, 1\}$ and $f_{ij}(x_i, x_j) = [\![x_i = x_j]\!]$ [Greig et al., 1989]
- reduction for $D_i = \{1, \dots, k\}$ and $f_{ij}(x_i, x_j) = g(|x_i - x_j|)$ with convex g [Ishikawa, 2003]
- reinventing submodularity:
  - LP relaxation exact for supermodular VCSPs of arity ≤ 2 [Schlesinger and Flach, 2000]
  - supermodular VCSPs with $D_i = \{0, 1\}$ and arity ≤ 3 [Kolmogorov and Zabih, 2002]

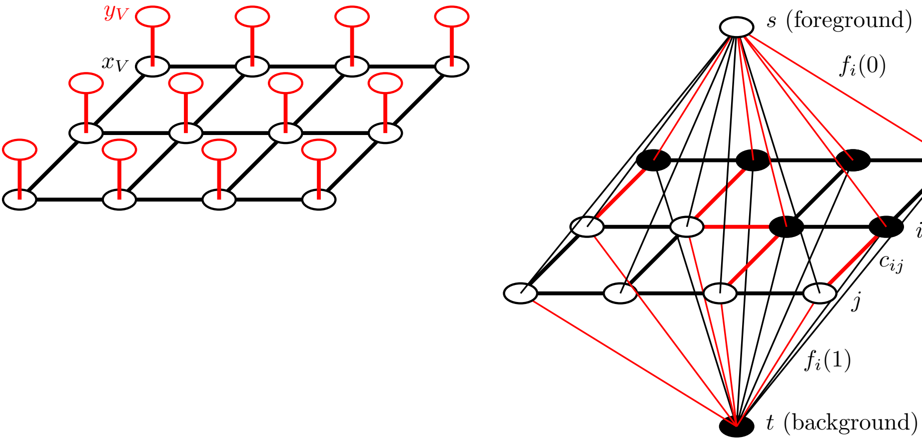
  Recall: For totally ordered domains D_i, function f_A is supermodular if

  $$f_A(\min\{x_A, y_A\}) + f_A(\max\{x_A, y_A\}) \geq f_A(x_A) + f_A(y_A) \quad \forall x_A, y_A \in D_A$$

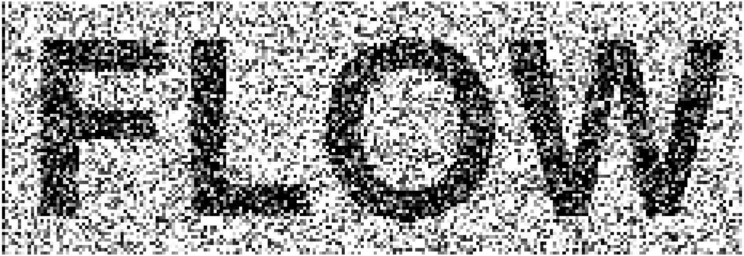
- very-large-neighborhood search with graph cuts (α-expansion, $\alpha\beta$-swap, ...) [Boykov et al., 2001]
- max-flow implementation efficient for vision problems: [Boykov and Kolmogorov, 2004]
- multilabel (variable with any finite domain) supermodular finite-valued [Schlesinger and Flach, 2006]
- permuted submodular [Schlesinger, 2007]
- persistency by roof duality [Hammer et al., 1984, Boros and Hammer, 2002], [Rother et al., 2007]

Observing image values $y_V \in \{0,\ldots,255\}^V$, infer segmentation $x_V \in \{0,1\}^V$:

$$\max_{x_V \in \{0,1\}^V} \sum_{i \in V} \log p(y_i \,|\, x_i) + \sum_{\{i,j\} \in E} c_{ij} [\![x_i = x_j]\!] \qquad (c_{ij} \geq 0)$$



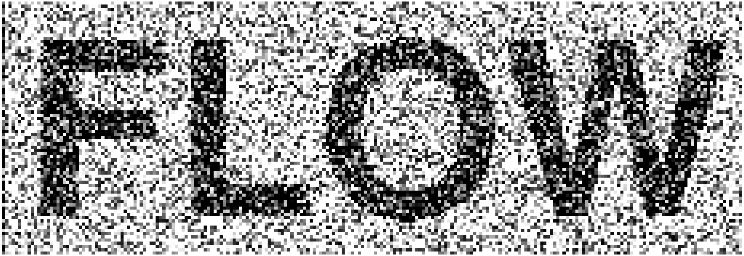
Example



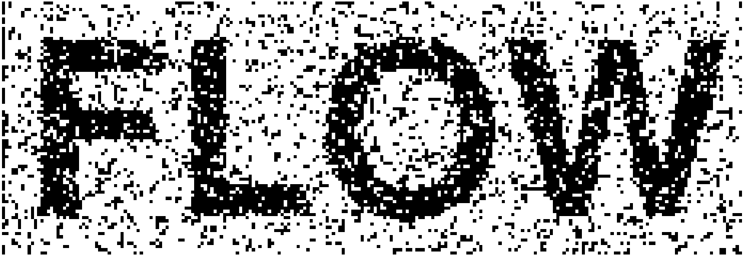
input image

segmentation, $c_{ij} = c = 0$

Example

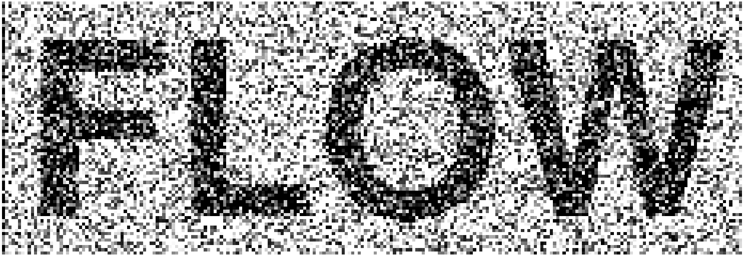


input image



segmentation, $c_{ij} = c = 20$

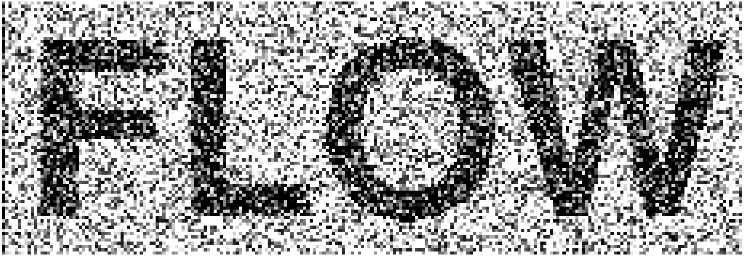
Example



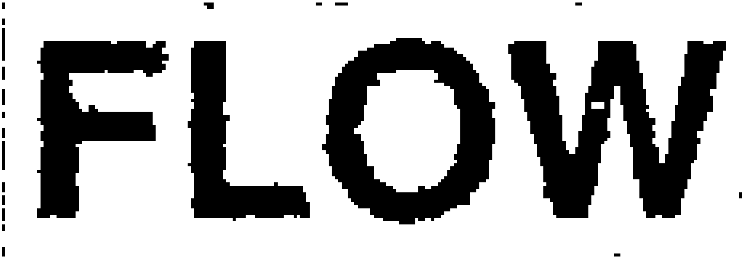
input image

segmentation, $c_{ij} = c = 30$

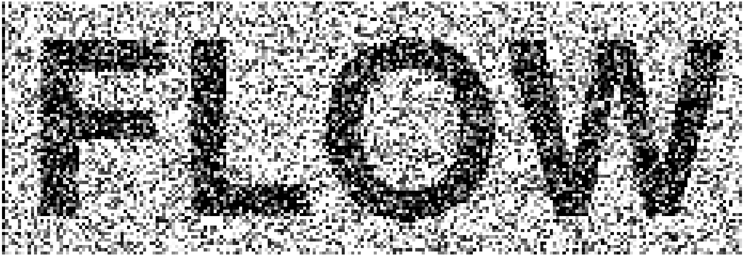
Example



input image

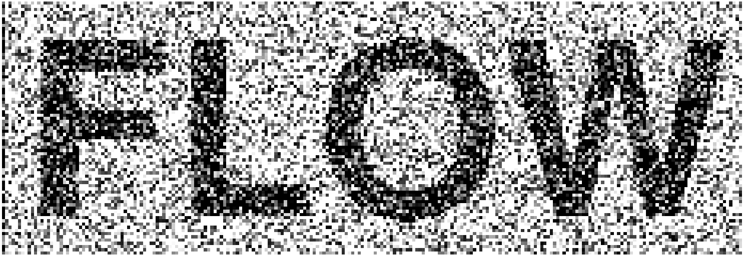


segmentation, $c_{ij} = c = 40$



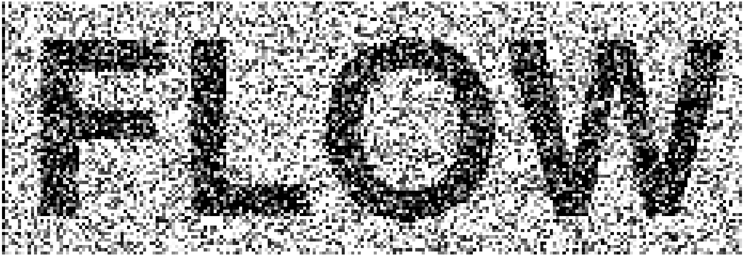
input image

segmentation, $c_{ij} = c = 50$



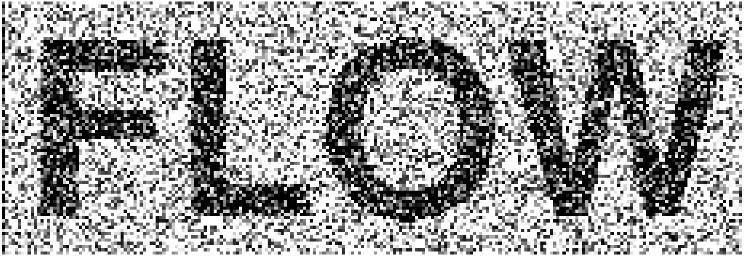
input image

segmentation, $c_{ij} = c = 60$



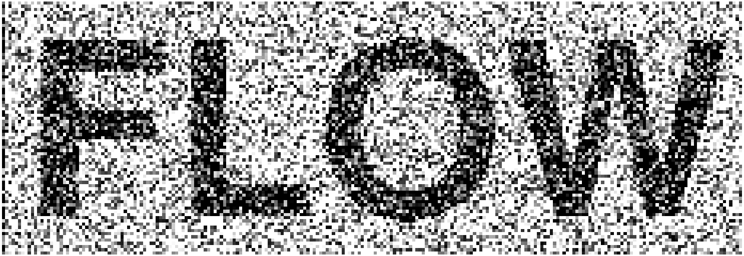
input image

segmentation, $c_{ij} = c = 62$



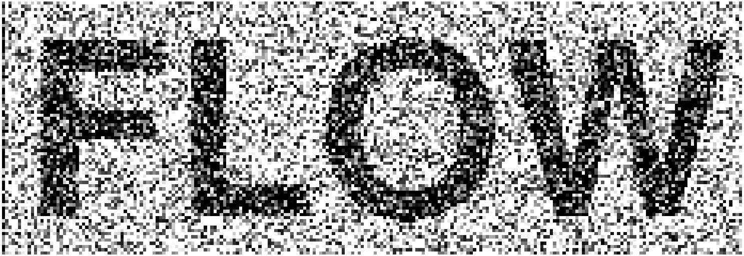
input image

segmentation, $c_{ij} = c = 64$



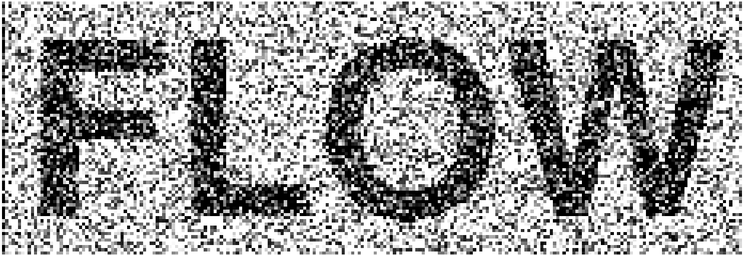
input image

segmentation, $c_{ij} = c = 65$



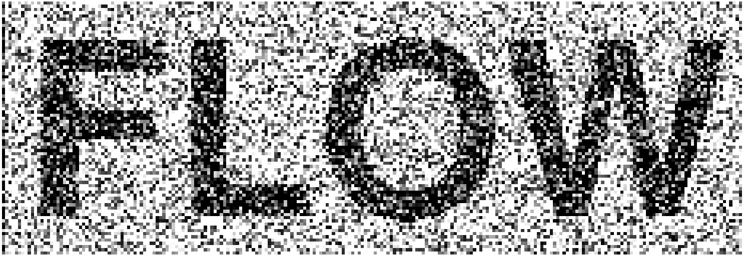
input image

segmentation, $c_{ij} = c = 66$



input image

segmentation, $c_{ij} = c = 67$



input image

segmentation, $c_{ij} = c = 68$

GrabCut [Rother et al., 2004]

Practical image segmentation:

- Powerfull pixel-wise color model $p_\theta(y_i \,|\, x_i)$ (mixture of Gaussians, histogram, ...).
- Estimate simultaneously labeling x_V and parameters θ: alternating maximization

Let $D = \{1, \dots, k\}$. Want to solve

$$\max_{x_V \in D^V} F(x_V) = \sum_{i \in V} f_i(x_i) + \sum_{\{i,j\} \in E} f_{ij}(x_i, x_j)$$

Given a labelling $x_V \in D^V$ and a label $\alpha \in D$, do α-expansion:

$$\max_{y_V \in \{0,1\}^V} F(x_V \sqcap y_V) \qquad \text{where} \quad (x_V \sqcap y_V)_i = \begin{cases} x_i & \text{if } y_i = 0 \\ \alpha & \text{if } y_i = 1 \end{cases}$$

Fact: This problem is submodular if f_{ij} are metric:

- $f(x, y) = f(y, x) \geq 0$
- $f(x, y) = 0 \implies x = y$
- $f(x, y) + f(y, z) \leq f(x, z)$

Examples:

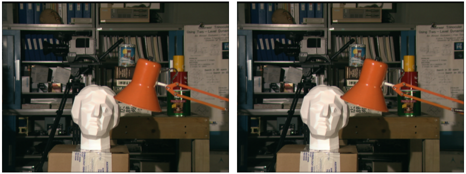
- $f(x, y) = [\![x = y]\!]$ (uniform/Potts metric)
- $f(x, y) = -\min\{K, |x - y|\}$ (truncated linear metric)

VLNS: Choose $\alpha \in D$ and do α-expansion till convergence:

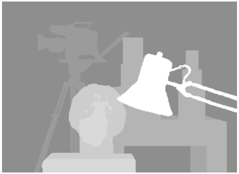
- constant approximation ratio
- [Boykov et al., 2001] and many follow-ups!

Examples from Vision [courtesy of Y.Boykov and co-authors]

Stereo correspondence



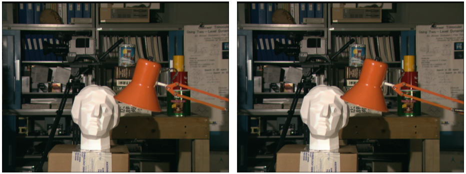
input



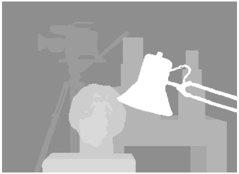
ground truth disparity

Examples from Vision [courtesy of Y.Boykov and co-authors]

Stereo correspondence



input

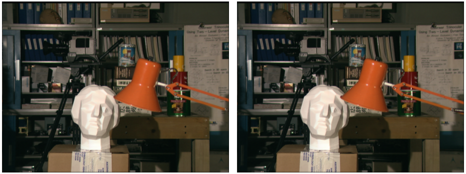


ground truth disparity

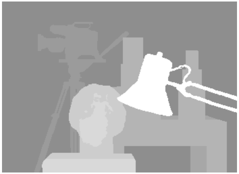
α-expansions

Examples from Vision [courtesy of Y.Boykov and co-authors]

Stereo correspondence



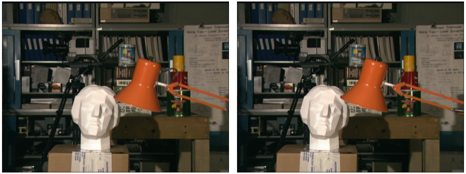
input



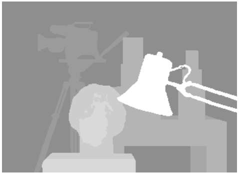
ground truth disparity

α-expansions

Stereo correspondence



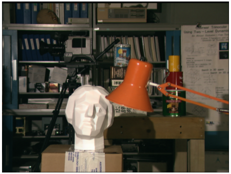
input



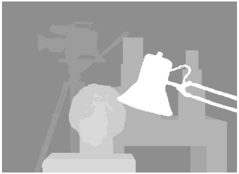
ground truth disparity

α-expansions

Stereo correspondence

input

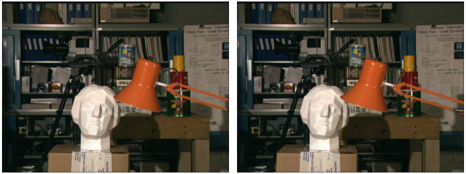


ground truth disparity

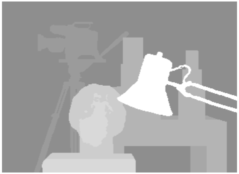
α-expansions

Examples from Vision [courtesy of Y.Boykov and co-authors]

Stereo correspondence



input

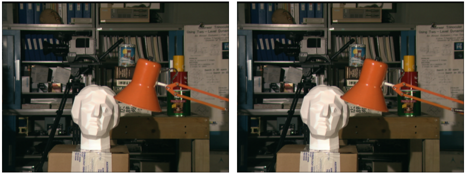


ground truth disparity

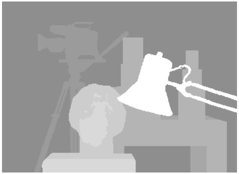
α-expansions

Examples from Vision [courtesy of Y.Boykov and co-authors]

Stereo correspondence



input

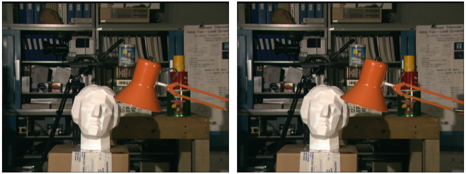


ground truth disparity

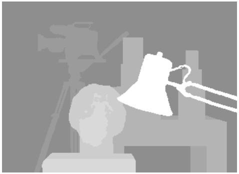
α-expansions

Examples from Vision [courtesy of Y.Boykov and co-authors]

Stereo correspondence



input

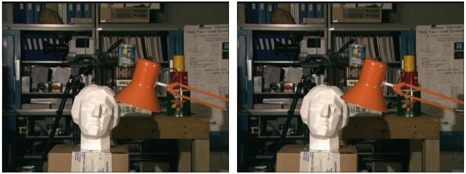


ground truth disparity

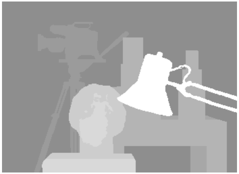
α-expansions

Examples from Vision [courtesy of Y.Boykov and co-authors]

Stereo correspondence



input

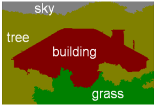


ground truth disparity

α-expansions

Semantic segmentation

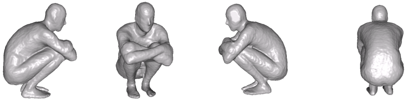
input



segmentation

Volumetric reconstructions from images:

input



output

Kiev (Ukraine), 1970's (reviewed by [Werner, 2007]):

- [Shlezinger, 1976]: binary CSP and VCSP ('2-dimensional grammars'), LP relaxation, reparameterizations
- [Kovalevsky and Koval, 1975]: max-sum diffusion
- [Koval and Schlesinger, 1976]: algorithm similar to VAC algorithm [Cooper et al., 2010]

In optimization:

- [Koster et al., 1998]: LP relaxation for binary VCSP, cycle-based cutting planes
- [Chekuri et al., 2001]: LP relaxation for metric binary VCSP

In machine learning and computer vision:

- [Wainwright and Jordan, 2008] + earlier works since 2002: marginal polytope, dual = combination of spanning (hyper-)trees, tree-reweighted message passing (TRW)
- [Kolmogorov, 2006]: sequential version of TRW (TRW-S) converges, fixed points are not global optima for dual LP
- [Johnson et al., 2007], [Komodakis et al., 2007]: dual decomposition
- higer-level LP relaxations, cutting planes: [Sontag and Jaakkola, 2007], [Johnson et al., 2007], [Komodakis and Paragios, 2008], [Sontag et al., 2008], [Werner, 2010]

VCSP as a linear program:

$$\max_{x_V \in D_V} F(x_V) = \max_{p \in \Delta_V} \langle F, p \rangle \qquad \text{where} \quad \Delta_V = \Big\{\, p\colon D_V \to \mathbb{R}_+,\ \sum_{x_V} p(x_V) = 1 \,\Big\}$$

Substitute for $F(x_V)$ and split the sum over x_V:

$$\langle F, p \rangle = \sum_{x_V} \sum_{A \in H} f_A(x_A) p(x_V) = \sum_{A \in H} \sum_{x_A} f_A(x_A) \underbrace{\sum_{x_{V \setminus A}} p(x_V)}_{p(x_A)} = \sum_{A \in H} \langle f_A, p_A \rangle$$

where $p_A(x_A) = p(x_A)$ are marginals of p on every $A \in H$.

Here is the resulting LP ...

$$\begin{aligned} \max \quad & \sum_{A \in H} \langle f_A, p_A \rangle \\ \text{subject to} \quad & p(x_A) = p_A(x_A), \qquad A \in H,\ x_A \in D_A \\ & p \in \Delta_V \end{aligned}$$

VCSP as a linear program:

$$\max_{x_V \in D_V} F(x_V) = \max_{p \in \Delta_V} \langle F, p \rangle \qquad \text{where} \quad \Delta_V = \Big\{ p\colon D_V \to \mathbb{R}_+,\ \sum_{x_V} p(x_V) = 1 \Big\}$$

Substitute for $F(x_V)$ and split the sum over x_V:

$$\langle F, p \rangle = \sum_{x_V} \sum_{A \in H} f_A(x_A) p(x_V) = \sum_{A \in H} \sum_{x_A} f_A(x_A) \underbrace{\sum_{x_{V \setminus A}} p(x_V)}_{p(x_A)} = \sum_{A \in H} \langle f_A, p_A \rangle$$

where $p_A(x_A) = p(x_A)$ are marginals of p on every $A \in H$.

... add (redundant) marginal distributions p_A for remaining $A \subseteq V$:

$$\begin{aligned} \max \quad & \sum_{A \in H} \langle f_A, p_A \rangle \\ \text{subject to} \quad & p(x_A) = p_A(x_A), \qquad A \subseteq V,\ x_A \in D_A \\ & p \in \Delta_V \end{aligned}$$

VCSP as a linear program:

$$\max_{x_V \in D_V} F(x_V) = \max_{p \in \Delta_V} \langle F, p \rangle \qquad \text{where} \quad \Delta_V = \Big\{ p\colon D_V \to \mathbb{R}_+,\ \sum_{x_V} p(x_V) = 1 \Big\}$$

Substitute for $F(x_V)$ and split the sum over x_V:

$$\langle F, p \rangle = \sum_{x_V} \sum_{A \in H} f_A(x_A) p(x_V) = \sum_{A \in H} \sum_{x_A} f_A(x_A) \underbrace{\sum_{x_{V \setminus A}} p(x_V)}_{p(x_A)} = \sum_{A \in H} \langle f_A, p_A \rangle$$

where $p_A(x_A) = p(x_A)$ are marginals of p on every $A \in H$.

... add (redundant) marginal equalities for remaining $B \subseteq A \subseteq V$:

$$\begin{aligned} \max \quad & \sum_{A \in H} \langle f_A, p_A \rangle \\ \text{subject to} \quad & p_A(x_B) = p_B(x_B), \qquad && B \subseteq A \subseteq V,\ x_B \in D_B \\ & p_A \in \Delta_A, && A \subseteq V \end{aligned}$$

Hierarchy of LP Relaxations of VCSP

VCSP as a linear program:

$$\max_{x_V \in D_V} F(x_V) = \max_{p \in \Delta_V} \langle F, p \rangle \qquad \text{where} \quad \Delta_V = \Big\{ p\colon D_V \to \mathbb{R}_+,\ \sum_{x_V} p(x_V) = 1 \Big\}$$

Substitute for $F(x_V)$ and split the sum over x_V:

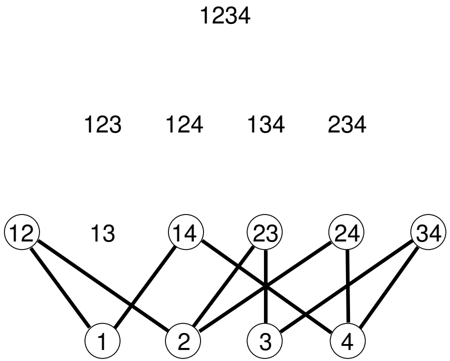
$$\langle F, p \rangle = \sum_{x_V} \sum_{A \in H} f_A(x_A) p(x_V) = \sum_{A \in H} \sum_{x_A} f_A(x_A) \underbrace{\sum_{x_{V \setminus A}} p(x_V)}_{p(x_A)} = \sum_{A \in H} \langle f_A, p_A \rangle$$

where $p_A(x_A) = p(x_A)$ are marginals of p on every $A \in H$.

... impose marginalization equalities only for some pairs $J \subseteq \{ (A, B) \mid B \subseteq A \subseteq V \}$:

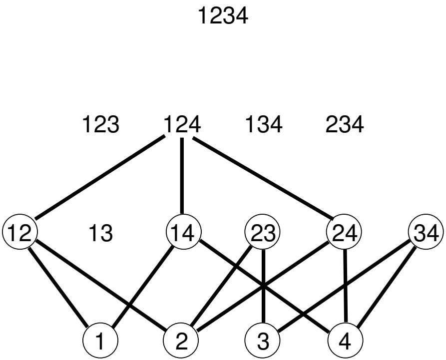
$$\begin{aligned} \max \quad & \sum_{A \in H} \langle f_A, p_A \rangle \\ \text{subject to} \quad & p_A(x_B) = p_B(x_B), && (A, B) \in J,\ x_B \in D_B \\ & p_A \in \Delta_A, && A \subseteq V \end{aligned}$$

- Nodes are all subsets $A \subseteq V = \{1, 2, 3, 4\}$.
- Subsets $A \in H$ are circled.
- Edges form the coupling graph $J \subseteq \{ (A, B) \mid B \subseteq A \subseteq V \}$.



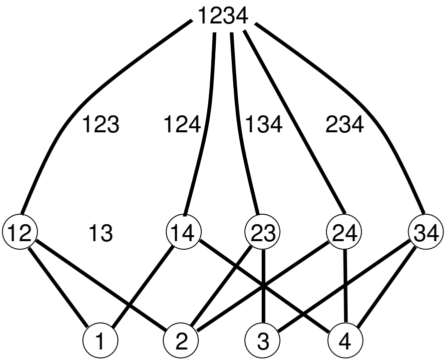
local ($=$ within H) relaxation

- Nodes are all subsets $A \subseteq V = \{1, 2, 3, 4\}$.
- Subsets $A \in H$ are circled.
- Edges form the coupling graph $J \subseteq \{\, (A, B) \mid B \subseteq A \subseteq V \,\}$.



non-local relaxation

- Nodes are all subsets $A \subseteq V = \{1, 2, 3, 4\}$.
- Subsets $A \in H$ are circled.
- Edges form the coupling graph $J \subseteq \{\, (A, B) \mid B \subseteq A \subseteq V \,\}$.



exact

Recall a reparameterization of a pair (f_A, f_B) (where $A, B \in H$ and $B \subseteq A$):

$$f_A(x_A) + f_B(x_B) = \underbrace{f_A(x_A) + \lambda(x_B)}_{f'_A(x_A)} + \underbrace{f_B(x_B) - \lambda(x_B)}_{f'_B(x_B)}.$$

Replacing (f_A, f_B) with (f'_A, f'_B) preserves the function $F(x_V) = \sum_{A \in H} f_A(x_A)$.

Add zero constraints $f_A = 0$ for all subsets $A \subseteq V$, $A \notin H$:

$$F(x_V) = \sum_{A \in H} f_A(x_A) = \sum_{A \subseteq V} f_A(x_A)$$

- Now we can reparameterize any pair (f_A, f_B) such that $B \subseteq A \subseteq V$.
- Reparameterizations permitted by J: $(A, B) \in J$.

$$\max_{x_V} \sum_{A \subseteq V} f_A(x_A) \leq \sum_{A \subseteq V} \max_{x_A} f_A(x_A)$$

where equality holds iff there is $x_V \in D_V$ such that

$$x_A \in \operatorname*{argmax}_{y_A} f_A(y_A) \qquad \forall A \subseteq V\ .$$

This is the CSP $\bar{f}$ formed by locally maximal tuples:

$$\bar{f}_A(x_A) = \begin{cases} 1 & \text{if } x_A \in \operatorname*{argmax}_{y_A} f_A(y_A) \\ 0 & \text{otherwise} \end{cases}$$

Dual LP: Minimize the upper bound by reparameterizations permitted by J.

Max-sum Diffusion

Minimizing the upper bound by reparameterizations = unconstrained minimization of convex piecewise affine function.

Max-sum diffusion [Kovalevsky and Koval, 1975], [Werner, 2007], [Werner, 2010]:
Iteration: Choose a pair $(A, B) \in J$ and reparameterize (f_A, f_B) to enforce

$$\max_{x_{A \setminus B}} f_A(x_A) = f_B(x_B), \qquad x_B \in D_B.$$

That is: find $\lambda \colon D_B \to \mathbb{R}$ such that

$$\max_{x_{A \setminus B}} (f_A(x_A) + \lambda(x_B)) = \max_{x_{A \setminus B}} f_A(x_A) + \lambda(x_B) = f_B(x_B) - \lambda(x_B),$$

hence $\lambda(x_B) = \frac{1}{2}\big(f_B(x_B) - \max_{x_{A \setminus B}} f_A(x_A)\big)$.

- Monotonically decreases the upper bound.
- = version of block-coordinate descent to solve the dual LP relaxation.
- Empirically, always converges to a fixed point (proof unknown!).
- Can be formulated for other commutative semirings [Werner, 2015].

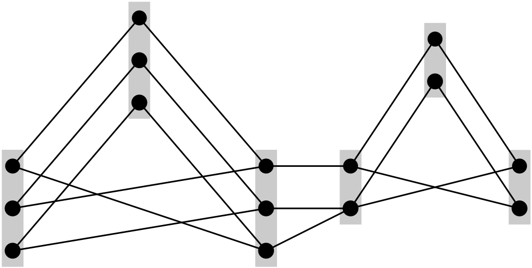
For non-smooth convex functions, block-coordinate descent can get stuck in a local minimum.

Example: For the (convex) function

$$f(x, y) = \max\{x - 2y, y - 2x\},$$

point $(x, y) = (0, 0)$ is minimal separately for each x and y but not globally.

Example: Arc-consistent CSP for which the upper bound is not minimal:



Recall the CSP $\bar{f}$ formed by locally maximal tuples:

$$\bar{f}_A(x_A) = \begin{cases} 1 & \text{if } x_A \in \operatorname*{argmax}_{y_A} f_A(y_A) \\ 0 & \text{otherwise} \end{cases}$$

Clearly,

$$\max_{x_{A \setminus B}} f_A(x_A) = f_B(x_B) \quad \Longrightarrow \quad \max_{x_{A \setminus B}} \bar{f}_A(x_A) = \bar{f}_B(x_B)$$

Thus, at a fixed point of max-sum diffusion, the CSP satisfies J-consistency:

$$\max_{x_{A \setminus B}} \bar{f}_A(x_A) = \bar{f}_B(x_B), \qquad (A, B) \in J,\ x_B \in D_B.$$

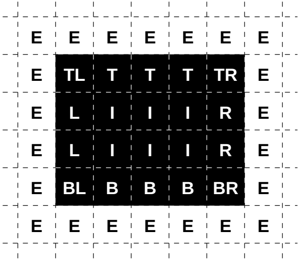
Special cases of J-consistency:

- GAC: $J = \{\, (A, \{i\}) \mid i \in V,\ A \in H \,\}$ (exact for permuted supermodular VCSPs)
- PWC: $J = \{\, (A, B) \mid A, B \in H \,\}$ and H is closed under intersection $(A, B \in H \implies A \cap B \in H)$
- k-consistency: PWC and $\binom{V}{k} \subseteq H$ (can be added incrementally)

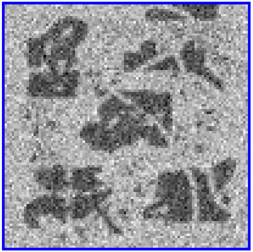
Example: Syntactic image analysis

Find the image containing non-overlapping rectangles, nearest to input image.

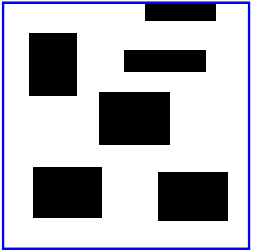
- V = pixels, E = edges between neighboring pixels
- $D_i = \{$ E, I, L, R, T, B, TL, TR, BL, BR $\}$ are syntactic parts of a rectangle.
- $f_i(x_i)$ = agreement of intensity of label x_i and intensity of input pixel i.
- $f_{ij}(x_i, x_j)$ equals 0 if parts x_i and x_j ever neighbor, and $-\infty$ otherwise.



hidden states = syntactic parts
observed states = {black,white}



input



output

Example: Binary VCSP with a Global (Cardinality) Constraint [Werner, 2010]

$$\max_{x_V \in \{0,1\}^V} F(x_V) = \sum_{i \in V} p(y_i \,|\, x_i) + \sum_{\{i,j\} \in E} c[\![x_i = x_j]\!] + \delta^m(x_V)$$

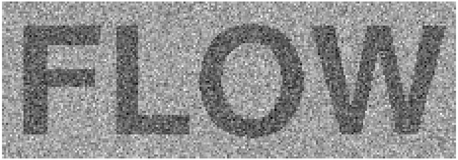
where

$$\delta^m(x_V) = \begin{cases} 0 & \text{if } \sum_{i \in V} = m \\ \infty & \text{otherwise} \end{cases}$$

To enforce GAC by max-sum diffusion, we need an oracle to calculate

$$\max_{x_{V \setminus \{j\}}} \Big(\delta^m(x_V) + \sum_{i \in V} \lambda_i(x_i) \Big)$$

for any $j \in V$ and any unary functions $\lambda_i \colon D_i \to \mathbb{R}$.



$m = 2000$

$$\max_{x_V \in \{0,1\}^V} F(x_V) = \sum_{i \in V} p(y_i \,|\, x_i) + \sum_{\{i,j\} \in E} c[\![x_i = x_j]\!] + \delta^m(x_V)$$

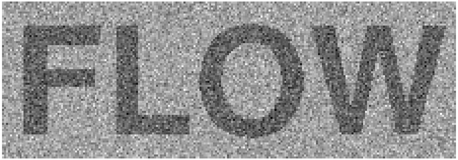
where

$$\delta^m(x_V) = \begin{cases} 0 & \text{if } \sum_{i \in V} = m \\ \infty & \text{otherwise} \end{cases}$$

To enforce GAC by max-sum diffusion, we need an oracle to calculate

$$\max_{x_{V \setminus \{j\}}} \left(\delta^m(x_V) + \sum_{i \in V} \lambda_i(x_i) \right)$$

for any $j \in V$ and any unary functions $\lambda_i \colon D_i \to \mathbb{R}$.



$m = 3000$

Example: Binary VCSP with a Global (Cardinality) Constraint [Werner, 2010]

$$\max_{x_V \in \{0,1\}^V} F(x_V) = \sum_{i \in V} p(y_i \,|\, x_i) + \sum_{\{i,j\} \in E} c[\![x_i = x_j]\!] + \delta^m(x_V)$$

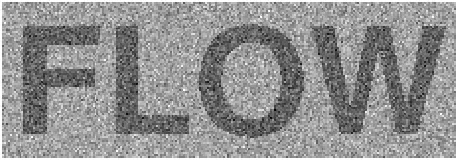
where

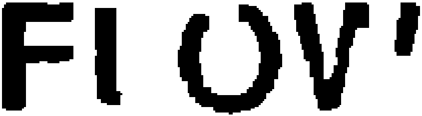
$$\delta^m(x_V) = \begin{cases} 0 & \text{if } \sum_{i \in V} = m \\ \infty & \text{otherwise} \end{cases}$$

To enforce GAC by max-sum diffusion, we need an oracle to calculate

$$\max_{x_{V \setminus \{j\}}} \left(\delta^m(x_V) + \sum_{i \in V} \lambda_i(x_i) \right)$$

for any $j \in V$ and any unary functions $\lambda_i \colon D_i \to \mathbb{R}$.





$m = 4000$

Example: Binary VCSP with a Global (Cardinality) Constraint [Werner, 2010]

$$\max_{x_V \in \{0,1\}^V} F(x_V) = \sum_{i \in V} p(y_i \,|\, x_i) + \sum_{\{i,j\} \in E} c[\![x_i = x_j]\!] + \delta^m(x_V)$$

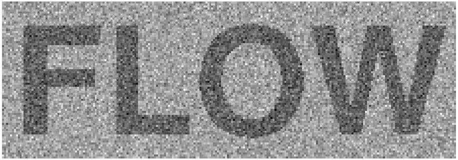
where

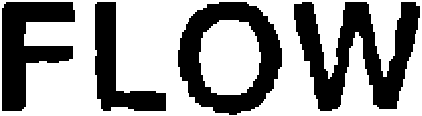
$$\delta^m(x_V) = \begin{cases} 0 & \text{if } \sum_{i \in V} = m \\ \infty & \text{otherwise} \end{cases}$$

To enforce GAC by max-sum diffusion, we need an oracle to calculate

$$\max_{x_{V \setminus \{j\}}} \Big(\delta^m(x_V) + \sum_{i \in V} \lambda_i(x_i) \Big)$$

for any $j \in V$ and any unary functions $\lambda_i \colon D_i \to \mathbb{R}$.





$m = 5000$

$$\max_{x_V \in \{0,1\}^V} F(x_V) = \sum_{i \in V} p(y_i \,|\, x_i) + \sum_{\{i,j\} \in E} c[\![x_i = x_j]\!] + \delta^m(x_V)$$

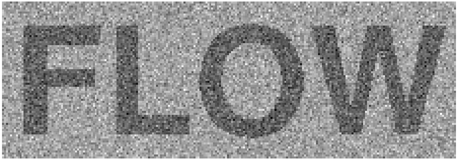
where

$$\delta^m(x_V) = \begin{cases} 0 & \text{if } \sum_{i \in V} = m \\ \infty & \text{otherwise} \end{cases}$$

To enforce GAC by max-sum diffusion, we need an oracle to calculate

$$\max_{x_{V \setminus \{j\}}} \Big(\delta^m(x_V) + \sum_{i \in V} \lambda_i(x_i) \Big)$$

for any $j \in V$ and any unary functions $\lambda_i \colon D_i \to \mathbb{R}$.



$m = 5368$

Example: Binary VCSP with a Global (Cardinality) Constraint [Werner, 2010]

$$\max_{x_V \in \{0,1\}^V} F(x_V) = \sum_{i \in V} p(y_i \,|\, x_i) + \sum_{\{i,j\} \in E} c[\![x_i = x_j]\!] + \delta^m(x_V)$$

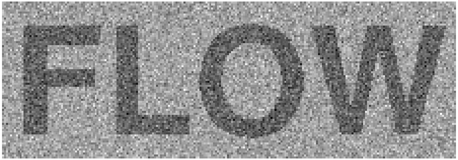
where

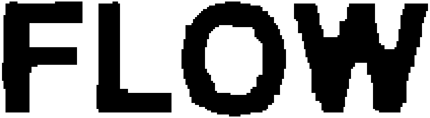
$$\delta^m(x_V) = \begin{cases} 0 & \text{if } \sum_{i \in V} = m \\ \infty & \text{otherwise} \end{cases}$$

To enforce GAC by max-sum diffusion, we need an oracle to calculate

$$\max_{x_{V \setminus \{j\}}} \Big(\delta^m(x_V) + \sum_{i \in V} \lambda_i(x_i) \Big)$$

for any $j \in V$ and any unary functions $\lambda_i \colon D_i \to \mathbb{R}$.





$m = 6000$

Example: Binary VCSP with a Global (Cardinality) Constraint [Werner, 2010]

$$\max_{x_V \in \{0,1\}^V} F(x_V) = \sum_{i \in V} p(y_i \,|\, x_i) + \sum_{\{i,j\} \in E} c[\![x_i = x_j]\!] + \delta^m(x_V)$$

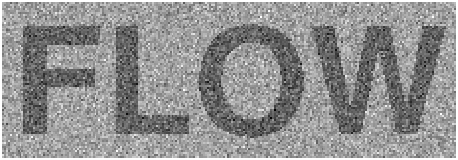
where

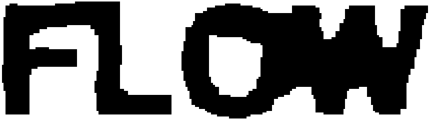
$$\delta^m(x_V) = \begin{cases} 0 & \text{if } \sum_{i \in V} = m \\ \infty & \text{otherwise} \end{cases}$$

To enforce GAC by max-sum diffusion, we need an oracle to calculate

$$\max_{x_{V \setminus \{j\}}} \Big(\delta^m(x_V) + \sum_{i \in V} \lambda_i(x_i) \Big)$$

for any $j \in V$ and any unary functions $\lambda_i \colon D_i \to \mathbb{R}$.





$m = 7000$

$$\max_{x_V \in \{0,1\}^V} F(x_V) = \sum_{i \in V} p(y_i \mid x_i) + \sum_{\{i,j\} \in E} c[\![x_i = x_j]\!] + \delta^m(x_V)$$

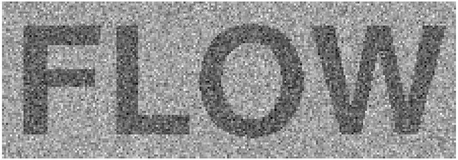
where

$$\delta^m(x_V) = \begin{cases} 0 & \text{if } \sum_{i \in V} = m \\ \infty & \text{otherwise} \end{cases}$$

To enforce GAC by max-sum diffusion, we need an oracle to calculate

$$\max_{x_{V \setminus \{j\}}} \Big(\delta^m(x_V) + \sum_{i \in V} \lambda_i(x_i) \Big)$$

for any $j \in V$ and any unary functions $\lambda_i \colon D_i \to \mathbb{R}$.



$m = 8000$

$$\max_{x_V \in \{0,1\}^V} F(x_V) = \sum_{i \in V} p(y_i \,|\, x_i) + \sum_{\{i,j\} \in E} c[\![x_i = x_j]\!] + \delta^m(x_V)$$

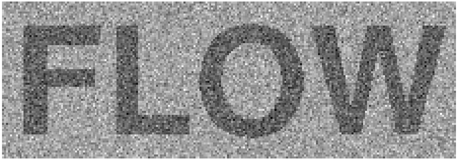
where

$$\delta^m(x_V) = \begin{cases} 0 & \text{if } \sum_{i \in V} = m \\ \infty & \text{otherwise} \end{cases}$$

To enforce GAC by max-sum diffusion, we need an oracle to calculate

$$\max_{x_{V \setminus \{j\}}} \left(\delta^m(x_V) + \sum_{i \in V} \lambda_i(x_i) \right)$$

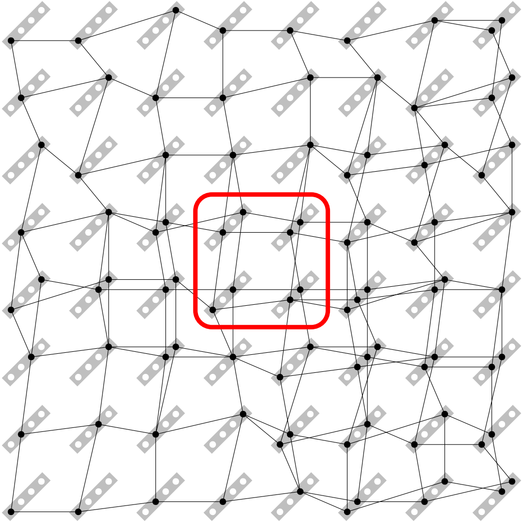
for any $j \in V$ and any unary functions $\lambda_i \colon D_i \to \mathbb{R}$.



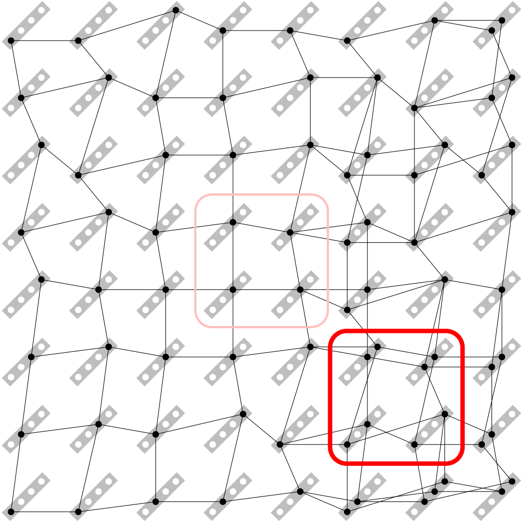
$m = 9000$

Dual Cutting Plane Algorithm

Find an infeasible sub-CSP and add the zero constraint over its scope:

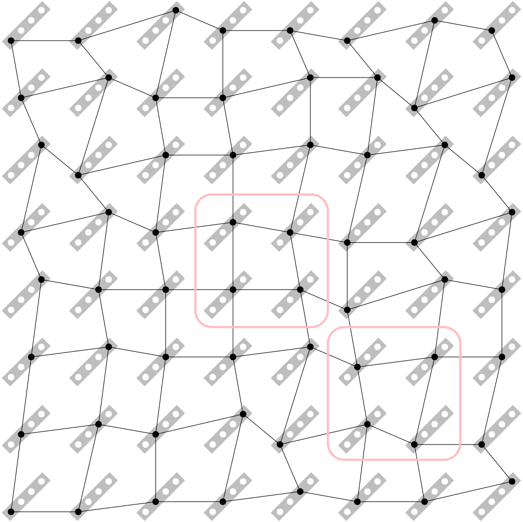


Find an infeasible sub-CSP and add the zero constraint over its scope:



Dual Cutting Plane Algorithm

Find an infeasible sub-CSP and add the zero constraint over its scope:



- MPLP [Globerson and Jaakkola, 2008]:
  Choose $A \in V$ and reparameterize (f_A, f_B) such that $(A, B) \in J$, to enforce
  $$\frac{1}{n_A} \max_{x_{A \setminus B}} \Big(f_A(x_A) + \sum_{(A,C) \in J} f_C(x_C) \Big) = f_B(x_B)$$
  where n_A is the number of pairs $(A, B) \in J$.

  Compare fixed point conds. for binary problems ($A = \{i, j\}$ and $B = \{i\}$):
  $$\begin{aligned} \max_{x_j} f_{ij}(x_i, x_j) &= f_i(x_i) && \text{(max-sum diffusion)} \\ \max_{x_j} (f_{ij}(x_i, x_j) + f_j(x_j)) &= f_i(x_i) && \text{(MPLP)} \\ \max_{x_j} (f_{ij}(x_i, x_j) + f_j(x_j)) &= \mathrm{const}_{ij} && \text{(loopy BP)} \end{aligned}$$

- TRW-S [Kolmogorov, 2006]: chain-wise iterations (takes finite time for a tree)
- SRMP [Kolmogorov, 2015]: generalizes TRW-S for VCSPs of any arity

VAC algorithm [Cooper et al., 2010] (and [Koval and Schlesinger, 1976]): similar nature

Let

$$F(x_V) = \sum_k F^k(x_V) \qquad \text{where} \quad F^k(x_V) = \sum_{A \in H^k} f_A^k(x_A)$$

Then

$$\max_{x_V} F(x_V) = \max_{x_V} \sum_k F^k(x_V) \leq \sum_k \max_{x_V} F^k(x_V)$$

Minimize RHS over functions f_A^k subject to $F(x_V) = \sum_k F^k(x_V)$.

- Yields a hierarchy of relaxations, equivalent to the above LP hierarchy.

Block-coordinate descent iteration:

- Choose $A \in H$ and minimize over f_A^k, $H^k \ni A$.
- Sufficient for optimum: max-marginals

$$\max_{x_{V \setminus A}} F^k(x_A)$$

must be the same for all k ('max-marginal averaging').

Find an efficient algorithm for (exactly) solving LP relaxation of VCSP!

Known for VCSP with $|D_i| = 2$ and binary constraints: its LP relaxation reduces in linear time to max-flow [Edmonds and Pulleyblank], [Boros and Hammer, 2002]

Universality of LP Relaxation of (V)CSP

Find an efficient algorithm for (exactly) solving LP relaxation of VCSP!

Known for VCSP with $|D_i| = 2$ and binary constraints: its LP relaxation reduces in linear time to max-flow [Edmonds and Pulleyblank], [Boros and Hammer, 2002]

Impossible for more general VCSP:

- [Průša and Werner, 2015]: The general LP reduces in linear time to LP relaxation of VCSP with $|D_i| = 3$ and binary constraints
  $\Longrightarrow$ Finding a feasible solution to LP relaxation of CSP is as hard as solving any system of linear inequalities
- [Průša and Werner, 2019]: The general LP reduces in linear time to LP relaxation of:
  set cover/packing,
  facility location,
  maximum satisfiability,
  maximum independent set,
  multiway cut

Open problem: Is enforcing VAC in VCSP easier than solving the general LP?

[Shekhovtsov et al., 2015, Shekhovtsov, 2016, Shekhovtsov et al., 2018]
[slides on persistency are courtesy of A.Shekhovtsov, P.Swoboda, B.Savchynskyy]

Integer (0-1) LP : $\max\{ \langle c, x \rangle \mid Ax = b,\ x \in \{0,1\}^n \}$

Its LP relaxation: $\max\{ \langle c, x \rangle \mid Ax = b,\ x \in [0,1]^n \}$

$$x^* = (0, 1, 1, 0, \tfrac{1}{2}, \tfrac{1}{2}, \tfrac{1}{3}, 1, 0, \dots)$$

1. Is the integer part of an optimal solution to LP optimal for ILP?
2. Is any part of the integer part of an optimal solution to LP optimal for ILP?
3. Find optimal LP solution with largest integer part optimal to ILP!

Examples

Vertex packing:

- [Balinski, 1965], [Lorentzen, 1966]: All vertices of the LP feasible set are half-integral.
- [Edmonds and Pulleyblank] LP reduces to max-flow
- [Nemhauser and Trotter, 1975]: weak persistency: variables with binary values in an optimal LP solution attain the same values in an optimal ILP solution.
- [Picard and Queyranne, 1977]: There is unique largest set of variables that are integer in an optimal solution to LP.

Quadratic pseudo-boolean optimization (QPBO) [Hammer et al., 1984, Boros et al., 1991]:

- generalizes vertex packing
- All vertices of the LP feasible set are half-integral.
- LP reduces to max-flow
- weak persistency: variables with binary values in an optimal LP solution attain the same values in an optimal ILP solution.
- strong persistency: variables with binary values in all optimal LP solutions attain the same values in all optimal ILP solutions.

Improving Maps

Let $P\colon D_V \to D_V$ be induced by idempotent maps $P_i\colon D_i \to D_i$ component-wise.

- P is improving if
$$F(P(x_V)) \geq F(x_V) \qquad \forall x_V \in D_V.$$
Existence of an improving map $\implies$ some domains can be reduced!
NP-complete to decide if a given P is improving.

- P is locally improving if
$$f_A(P(x_A)) \geq f_A(x_A) \qquad \forall A \in H,\ x_A \in D_A$$
Sufficient for being improving, easy to decide, but weak.

- P is relaxed-improving if
$$f'_A(P(x_A)) \geq f'_A(x_A) \qquad \forall A \in H,\ x_A \in D_A$$
for some reparameterization f'_A of functions f_A.
Sufficient for being improving, tractable to decide (via LP relaxation), stronger!

Even possible maximum persistency [Shekhovtsov et al., 2018]:

$$\min_{P \in \mathcal{P}} \sum_{i \in V} |P_i(D_i)| \quad \text{subject to that } P \text{ is relaxed-improving}$$

Generality of Relaxed-Improving Maps

Relaxed-improving condition holds for all of these previous methods:

- arity ≤ 2:
  - simple DEE [Goldstein, 1994]
  - MQPBO [Kohli et al., 2008]
  - [Kovtun, 2003] one-against-all
  - [Kovtun, 2011] iterative
  - [Swoboda et al., 2014]
- any arity, boolean variables:
  - roof duality / QPBO [Hammer et al., 1984]
  - reductions: HOCR [Ishikawa, 2011], [Fix et al., 2011]
  - bisubmodular relaxations [Kolmogorov, 2010]
  - generalized roof duality [Kahl and Strandmark, 2011]
  - persistency by [Adams et al., 1998]

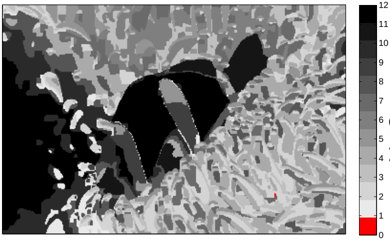
OpenGM Benchmark: Easy Examples

Some problems are easy (TRWS finds optimal solution or near)

Object Segmentation

100%
1.7s

Color Segmentation



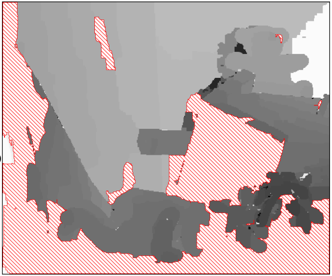
99.99%
2.8s

OpenGM Benchmark: Hard Examples

Some are harder: Stereo

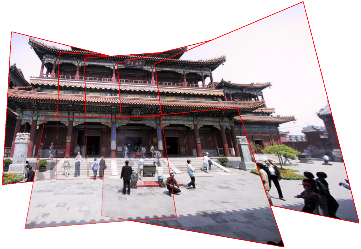
TRW-S
62s

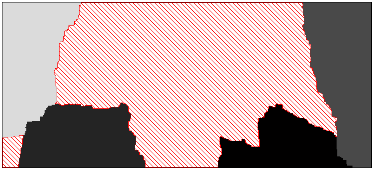
+180s
75%

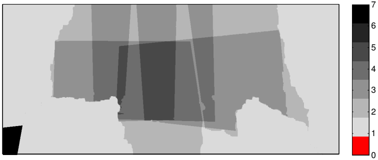


OpenGM Benchmark: Very Hard Examples

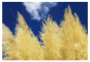
Panorama Stitching







Panorama Stitching with Constraints



ColorSegmentation (N8)
J. Lellmann et.al.
converted by J. Lellmann and J.H. Kappes

Object Segmentation
K. Alahari et.al.
converted by J.H. Kappes

MRF Stereo
R. Szeliski et.al.
converted by J.H. Kappes

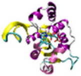
ColorSegmentation
K. Alahari et.al.
converted by J.H. Kappes

MRF Photomontage
R. Szeliski et.al.
converted by J.H. Kappes

MRF Inpainting
R. Szeliski et.al.
converted by J.H. Kappes

Chinese Characters
S. Nowozin et.al.
converted by S. Nowozin and J. H. Kappes

Scene Decomposition
Gould et.al.
converted by S. Nowozin and J. H. Kappes



Protein Folding
Yanover et. al.
converted by Joerg Kappes

Brain 3mm
J. H. Kappes et.al.
converted by J. H. Kappes

Geometric Surface Labeling (3)
Gallagher et.al.
converted by D. Batra and J. H. Kappes

Problem family	#I	#L	#V	MQPBO		MQPBO-10		Kovtun		[29]-TRWS		Our-TRWS	
mrf-stereo	3	16-60	> 100000	†		†		†		2.5h	13%	117s	**73.56%**
mrf-photomontage	2	5-7	≤ 514080	93s	22%	866s	16%	†		3.7h	16%	483s	**41.98%**
color-seg	3	3-4	≤ 424720	22s	11%	87s	16%	**0.3s**	98%	1.3h	**>99%**	61.8s	**99.95%**
color-seg-n4	9	3-12	≤ 86400	22s	8%	398s	14%	**0.2s**	67%	321s	90%	4.9s	**99.26%**
ProteinFolding	21	≤ 483	≤ 1972	685s	2%	2705s	2%	†		48s	18%	9.2s	**55.70%**
object-seg	5	4-8	68160	3.2s	0.01%	†		**0.1s**	93.86%	138s	98.19%	2.2s	**100%**

Max-cut relaxation:

$$\begin{aligned} \max \quad & \sum_{i,j=1}^{n} c_{ij} \langle x_i, x_j \rangle = \langle C, XX^T \rangle \\ \text{subject to} \quad & \langle x_i, x_i \rangle = 1, \qquad i = 1, \ldots, n \\ & x_i \in \mathbb{R}^k, \qquad i = 1, \ldots, n \end{aligned}$$

$$\begin{aligned} \max \quad & \langle C, Y \rangle \\ \text{subject to} \quad & Y_{ii} = 1, \qquad i = 1, \ldots, n \\ & \operatorname{rank} Y \leq k \\ & Y \in \mathbb{R}^{n \times n} \end{aligned}$$

- $k = 1$: original (non-relaxed) max-cut problem
- $k = n$: SDP relaxation [Goemans and Williamson, 1995]
- Low-rank methods: Optimize directly over vectors x_i!
  [Burer and Monteiro, 2003], [Boumal et al., 2016]
  Extension for boolean VCSP [Erdogdu et al., 2017] (optimized by ADMM-like algs.)

References I

[Adams et al., 1998] Adams, W. P., Lassiter, J. B., and Sherali, H. D. (1998).
Persistency in 0-1 polynomial programming.
Mathematics of Operations Research, 23(2):359–389.

[Aji and McEliece, 2000] Aji, S. M. and McEliece, R. J. (2000).
The generalized distributive law.
IEEE Trans. on Information Theory, 46(2):325–343.

[Balinski, 1965] Balinski, M. L. (1965).
Integer programming: methods, uses, computation.
Management Science, 12(3):253–313.

[Boros and Hammer, 2002] Boros, E. and Hammer, P. L. (2002).
Pseudo-Boolean optimization.
Discrete Applied Mathematics, 123(1-3):155–225.

[Boros et al., 1991] Boros, E., Hammer, P. L., and Sun, X. (1991).
Network flows and minimization of quadratic pseudo-Boolean functions.
Technical Report RRR 17-1991, RUTCOR.

[Boumal et al., 2016] Boumal, N., Voroninski, V., and Bandeira, A. (2016).
The non-convex burer-monteiro approach works on smooth semidefinite programs.
In Adv. in Neural Information Processing Systems, pages 2757–2765.

[Boykov and Kolmogorov, 2004] Boykov, Y. and Kolmogorov, V. (2004).
An experimental comparison of min-cut/max-flow algorithms for energy minimization in vision.
IEEE Trans. Pattern Analysis and Machine Intelligence, 26(9):1124–1137.

[Boykov et al., 2001] Boykov, Y., Veksler, O., and Zabih, R. (2001).
Fast approximate energy minimization via graph cuts.
IEEE Trans. Pattern Analysis and Machine Intelligence, 23(11):1222–1239.

[Burer and Monteiro, 2003] Burer, S. and Monteiro, R. D. C. (2003).
A nonlinear programming algorithm for solving semidefinite programs via low-rank factorization.
Math. Program., 95(2):329–357.

References II

[Chekuri et al., 2001] Chekuri, C., Khanna, S., Naor, J., and Zosin, L. (2001).
Approximation algorithms for the metric labeling problem via a new linear programming formulation.
In Symposium on Discrete Algorithms, pages 109–118.

[Cooper et al., 2010] Cooper, M. C., de Givry, S., Sanchez, M., Schiex, T., Zytnicki, M., and Werner, T. (2010).
Soft arc consistency revisited.
Artificial Intelligence, 174(7-8):449–478.

[Erdogdu et al., 2017] Erdogdu, M. A., Deshpande, Y., and Montanari, A. (2017).
Inference in graphical models via semidefinite programming hierarchies.
In Adv. in Neural Information Processing Systems, pages 417–425.

[Fix et al., 2011] Fix, A., Gruber, A., Boros, E., and Zabih, R. (2011).
A graph cut algorithm for higher-order Markov random fields.
In ICCV, pages 1020 –1027.

[Geman and Geman, 1984] Geman, S. and Geman, D. (1984).
Stochastic relaxation, Gibbs distributions and Bayesian restoration of images.
IEEE Trans. Pattern Analysis Machine Intelligence, 6:721–741.

[Globerson and Jaakkola, 2008] Globerson, A. and Jaakkola, T. (2008).
Fixing max-product: Convergent message passing algorithms for MAP LP-relaxations.
In Neural Information Processing Systems, pages 553–560.

[Goemans and Williamson, 1995] Goemans, M. X. and Williamson, D. P. (1995).
Improved approximation algorithms for maximum cut and satisfiability problems using semidefinite programming.
J. ACM, 42(6):1115–1145.

[Goldstein, 1994] Goldstein, R. F. (1994).
Efficient rotamer elimination applied to protein side-chains and related spin glasses.
Biophysical Journal, 66(5):1335–1340.

[Greig et al., 1989] Greig, D., Porteous, B., and Seheult, A. (1989).
Exact maximum a posteriori estimation for binary images.
J. Royal Statistical Society B, (51):271–279.

References III

[Hammer et al., 1984] Hammer, P., Hansen, P., and Simeone, B. (1984).
Roof duality, complementation and persistency in quadratic 0-1 optimization.
Mathematical Programming, pages 121–155.

[Hazan and Jaakkola, 2012] Hazan, T. and Jaakkola, T. S. (2012).
On the partition function and random maximum a-posteriori perturbations.
In 29th International Conference on Machine Learning.

[Ishikawa, 2003] Ishikawa, H. (2003).
Exact optimization for Markov random fields with convex priors.
IEEE Trans. Pattern Analysis and Machine Intelligence, 25(10):1333–1336.

[Ishikawa, 2011] Ishikawa, H. (2011).
Transformation of general binary MRF minimization to the first-order case.
IEEE Trans. Pattern Anal. Mach. Intell., 33(6):1234–1249.

[Johnson et al., 2007] Johnson, J. K., Malioutov, D. M., and Willsky, A. S. (2007).
Lagrangian relaxation for MAP estimation in graphical models.
CoRR, abs/0710.0013.

[Kahl and Strandmark, 2011] Kahl, F. and Strandmark, P. (2011).
Generalized roof duality for pseudo-Boolean optimization.
In ICCV, pages 255–262.

[Kohli et al., 2008] Kohli, P., Shekhovtsov, A., Rother, C., Kolmogorov, V., and Torr, P. (2008).
On partial optimality in multi-label MRFs.
In ICML, pages 480–487.

[Kolmogorov, 2006] Kolmogorov, V. (2006).
Convergent tree-reweighted message passing for energy minimization.
IEEE Trans. Pattern Analysis and Machine Intelligence, 28(10):1568–1583.

[Kolmogorov, 2010] Kolmogorov, V. (2010).
Generalized roof duality and bisubmodular functions.
In NIPS, pages 1144–1152.

[Kolmogorov, 2015] Kolmogorov, V. (2015).
A new look at reweighted message passing.
IEEE Trans. on Pattern Analysis and Machine Intelligence, 37(5).

[Kolmogorov and Zabih, 2002] Kolmogorov, V. and Zabih, R. (2002).
What energy functions can be minimized via graph cuts?
In Eur. Conf. on Computer Vision, pages 65–81. Springer-Verlag.

[Komodakis and Paragios, 2008] Komodakis, N. and Paragios, N. (2008).
Beyond loose LP-relaxations: Optimizing MRFs by repairing cycles.
In Eur. Conf. on Computer Vision.

[Komodakis et al., 2007] Komodakis, N., Paragios, N., and Tziritas, G. (2007).
MRF optimization via dual decomposition: Message-passing revisited.
In Intl. Conf. on Computer Vision.

[Koster et al., 1998] Koster, A., van Hoesel, C. P. M., and Kolen, A. W. J. (1998).
The partial constraint satisfaction problem: Facets and lifting theorems.
Operations Research Letters, 23(3-5):89–97.

[Koval and Schlesinger, 1976] Koval, V. K. and Schlesinger, M. I. (1976).
Dvumernoe programmirovanie v zadachakh analiza izobrazheniy (Two-dimensional programming in image analysis problems).
Automatics and Telemechanics, 8:149–168.

[Kovalevsky and Koval, 1975] Kovalevsky, V. A. and Koval, V. K. (approx. 1975).
A diffusion algorithm for decreasing the energy of the max-sum labeling problem.
Glushkov Institute of Cybernetics, Kiev, USSR. Unpublished.

[Kovtun, 2003] Kovtun, I. (2003).
Partial optimal labelling search for a NP-hard subclass of (max,+) problems.
In Conf. German Assoc. for Pattern Recognition (DAGM), pages 402–409.

[Kovtun, 2011] Kovtun, I. (2011).
Sufficient condition for partial optimality for (max, +) labeling problems and its usage.
Control Systems and Computers, (2).
Special issue.

[Mezard and Montanari, 2009] Mezard, M. and Montanari, A. (2009).
Information, Physics, and Computation.
Oxford University Press.

[Murphy et al., 1999] Murphy, K. P., Weiss, Y., and Jordan, M. I. (1999).
Loopy belief propagation for approximate inference: An empirical study.
In The 15th Conference on Uncertainty in Artificial Intelligence, pages 467–475. Morgan Kaufmann.

[Nemhauser and Trotter, 1975] Nemhauser, G. L. and Trotter, Jr., L. E. (1975).
Vertex packings: Structural properties and algorithms.
Mathematical Programming, 8:232–248.

[Pearl, 1988] Pearl, J. (1988).
Probabilistic reasoning in intelligent systems: Networks of plausible inference.
Morgan Kaufmann, San Francisco.

[Picard and Queyranne, 1977] Picard, J.-C. and Queyranne, M. (1977).
On the integer-valued variables in the linear vertex packing problem.
Mathematical Programming, 12(1):97–101.

[Průša and Werner, 2015] Průša, D. and Werner, T. (2015).
Universality of the local marginal polytope.
IEEE Trans. on Pattern Analysis and Machine Intelligence, 37(4):898–904.

[Průša and Werner, 2019] Průša, D. and Werner, T. (2019).
Solving lp relaxations of some np-hard problems is as hard as solving any linear program.
SIAM J. Optimization, 29(7):1745–1771.

[Rother et al., 2004] Rother, C., Kolmogorov, V., and Blake, A. (2004).
"GrabCut": interactive foreground extraction using iterated graph cuts.
In SIGGRAPH, pages 309–314. ACM Press.

[Rother et al., 2007] Rother, C., Kolmogorov, V., Lempitsky, V. S., and Szummer, M. (2007).
Optimizing binary MRFs via extended roof duality.
In Conf. Computer Vision and Pattern Recognition, Minneapolis, USA.

References VI

[Schlesinger, 2007] Schlesinger, D. (2007).
Exact solution of permuted submodular MinSum problems.
In Conf. Energy Minimization Methods in Computer Vision and Pattern Recognition (EMMCVPR), Ezhou, China, pages 28–38. Springer.

[Schlesinger and Flach, 2006] Schlesinger, D. and Flach, B. (2006).
Transforming an arbitrary MinSum problem into a binary one.
Technical Report TUD-FI06-01, Dresden University of Technology, Germany.

[Schlesinger and Flach, 2000] Schlesinger, M. I. and Flach, B. (2000).
Some solvable subclasses of structural recognition problems.
In Czech Pattern Recognition Workshop, pages 55–62. Czech Pattern Recognition Society.

[Shekhovtsov, 2016] Shekhovtsov, A. (2016).
Higher order maximum persistency and comparison theorems.
Computer Vision and Image Understanding, 143:54–79.

[Shekhovtsov et al., 2015] Shekhovtsov, A., Swoboda, P., and Savchynskyy, B. (2015).
Maximum persistency via iterative relaxed inference with graphical models.
In IEEE Conference on Computer Vision and Pattern Recognition, pages 521–529.

[Shekhovtsov et al., 2018] Shekhovtsov, A., Swoboda, P., and Savchynskyy, B. (2018).
Maximum persistency via iterative relaxed inference in graphical models.
IEEE Trans. Pattern Anal. Mach. Intell., 40(7):1668–1682.

[Shlezinger, 1976] Shlezinger, M. I. (1976).
Syntactic analysis of two-dimensional visual signals in noisy conditions.
Cybernetics and Systems Analysis, 12(4):612–628.
Translation from Russian.

[Sontag and Jaakkola, 2007] Sontag, D. and Jaakkola, T. (2007).
New outer bounds on the marginal polytope.
In Neural Information Processing Systems, pages 1393–1400.

[Sontag et al., 2008] Sontag, D., Meltzer, T., Globerson, A., Jaakkola, T., and Weiss, Y. (2008).
Tightening LP relaxations for MAP using message passing.
In Conf. Uncertainty in Artificial Intelligence, pages 503–510.

References VII

[Swoboda et al., 2014] Swoboda, P., Savchynskyy, B., Kappes, J. H., and Schnörr, C. (2014).
Partial optimality by pruning for MAP-inference with general graphical models.
In CVPR, page 8.

[Wainwright and Jordan, 2008] Wainwright, M. J. and Jordan, M. I. (2008).
Graphical models, exponential families, and variational inference.
Foundations and Trends in Machine Learning, 1(1-2):1–305.

[Werner, 2007] Werner, T. (2007).
A linear programming approach to max-sum problem: A review.
IEEE Trans. Pattern Analysis and Machine Intelligence, 29(7):1165–1179.

[Werner, 2010] Werner, T. (2010).
Revisiting the linear programming relaxation approach to Gibbs energy minimization and weighted constraint satisfaction.
IEEE Trans. Pattern Analysis and Machine Intelligence, 32(8):1474–1488.

[Werner, 2015] Werner, T. (2015).
Marginal consistency: Upper-bounding partition functions over commutative semirings.
IEEE Trans. on Pattern Analysis and Machine Intelligence, 37(7):1455–1468.