
Min-Max Problems on Factor-Graphs

Siamak Ravanbakhsh*

Christopher Srinivasa[‡]

Brendan Frey[‡]

Russell Greiner*

MRAVANBA@UALBERTA.CA

CHRIS@PSI.UTORONTO.CA

FREY@PSI.UTORONTO.CA

RGREINER@UALBERTA.CA

* Computing Science Dept., University of Alberta, Edmonton, AB T6G 2E8 Canada

[‡] PSI Group, University of Toronto, ON M5S 3G4 Canada

Abstract

We study the min-max problem in factor graphs, which seeks the assignment that minimizes the maximum value over all factors. We reduce this problem to both min-sum and sum-product inference, and focus on the later. In this approach the min-max inference problem is reduced to a sequence of Constraint Satisfaction Problems (CSP), which allows us to solve the problem by sampling from a uniform distribution over the set of solutions. We demonstrate how this scheme provides a message passing solution to several NP-hard combinatorial problems, such as min-max clustering (*a.k.a.* K -clustering), asymmetric K -center clustering problem, K -packing and the bottleneck traveling salesman problem. Furthermore, we theoretically relate the min-max reductions and several NP hard decision problems such as clique cover, set-cover, maximum clique and Hamiltonian cycle, therefore also providing message passing solutions for these problems. Experimental results suggest that message passing often provides near optimal min-max solutions for moderate size instances.

1. Introduction

In recent years, message passing methods have achieved a remarkable success in solving different classes of optimization problems, including maximization (*e.g.*, Frey & Dueck 2007; Bayati et al. 2005), integration (*e.g.*, Huang & Jebara 2009) and constraint satisfaction problems (*e.g.*, Mezard et al. 2002). When formulated as a graphical model, these problems correspond to different modes of inference: (a) solving a CSP corresponds to sampling

from a uniform distribution over satisfying assignments, while (b) counting and integration usually correspond to estimation of the partition function and (c) maximization corresponds to Maximum a Posteriori (MAP) inference. Here we introduce and study a new class of inference over graphical models – *i.e.*, (d) the min-max inference problem, where the objective is to find an assignment to minimize the maximum value over a set of functions.

The min-max objective appears in various fields, particularly in building robust models under uncertain and adversarial settings. In the context of probabilistic graphical models, several different min-max objectives have been previously studied (*e.g.*, Kearns et al. 2001; Ibrahimi et al. 2011). In combinatorial optimization, min-max may refer to the relation between maximization and minimization in dual combinatorial objectives and their corresponding linear programs (*e.g.*, Schrijver 1983), or it may refer to min-max settings due to uncertainty in the problem specification (*e.g.*, Averbakh 2001; Aissi et al. 2009).

Our setting is closely related to a third class of min-max combinatorial problems that are known as *bottleneck* problems. Instances of these problems include bottleneck traveling salesman problem (Parker & Rardin 1984), min-max clustering (Gonzalez 1985), k -center problem (Dyer & Frieze 1985; Khuller & Sussmann 2000) and bottleneck assignment problem (Gross 1959).

Edmonds & Fulkerson (1970) introduce a bottleneck framework with a duality theorem that relates the min-max objective in one problem instance to a max-min objective in a dual problem. An intuitive example is the duality between the min-max cut separating nodes a and b – the cut with the minimum of the maximum weight – and min-max path between a and b , which is the path with the minimum of the maximum weight (Fulkerson 1966). Hochbaum & Shmoys (1986) leverage triangle inequality in metric spaces to find constant factor approximations to several NP-hard min-max problems under a unified framework.

The common theme in a majority of heuristics for min-max or bottleneck problems is the relation of the min-max objective with a CSP (e.g., Hochbaum & Shmoys 1986; Panigrahy & Vishwanathan 1998). In this paper, we establish a similar relation within the context of factor-graphs, by reducing the min-max inference problem on the original factor-graph to that of sampling (i.e., solving a CSP) on the reduced factor-graph. We also consider an alternative approach where the factor-graph is transformed such that the min-sum objective produces the same optimal result as min-max objective on the original factor-graph. Although this reduction is theoretically appealing, in its simple form it suffers from numerical problems and is not further pursued here.

Section 2 formalizes min-max problem in probabilistic graphical models and provides an inference procedure by reduction to a sequence of CSPs on the factor graph. Section 3 reviews Perturbed Belief Propagation equations (Ravnbakhsh & Greiner 2014) and several forms of high-order factors that allow efficient sum-product inference. Finally Section 4 uses these factors to build efficient algorithms for several important min-max problems with general distance matrices. These applications include problems, such as K-packing, that were not previously studied within the context of min-max or bottleneck problems.

2. Factor Graphs and CSP Reductions

Let $x = \{x_1, \dots, x_n\}$, where $x \in \mathcal{X} \triangleq \mathcal{X}_1 \times \dots \times \mathcal{X}_n$ denote a set of discrete variables. Each factor $f_I(x_I) : \mathcal{X}_I \rightarrow \mathcal{Y}_I \subset \mathbb{R}$ is a real valued function with range \mathcal{Y}_I , over a subset of variables –i.e., $I \subseteq \{1, \dots, n\}$ is a set of indices. Given the set of factors \mathcal{F} , the min-max objective is

$$x^* = \arg_x \min_{I \in \mathcal{F}} \max f_I(x_I) \quad (1)$$

This model can be conveniently represented as a bipartite graph, known as *factor graph* (Kschischang & Frey 2001), that includes two sets of nodes: variable nodes x_i , and factor nodes f_I . A variable node i (note that we will often identify a variable x_i with its index “ i ”) is connected to a factor node I if and only if $i \in I$. We will use ∂ to denote the neighbors of a variable or factor node in the factor graph— that is $\partial I = \{i \text{ s.t. } i \in I\}$ (which is the set I) and $\partial i = \{I \text{ s.t. } i \in I\}$.

Let $\mathcal{Y} = \bigcup_I \mathcal{Y}_I$ denote the union over the range of all factors. The min-max value belongs to this set $\max_{I \in \mathcal{F}} f_I(x_I^*) \in \mathcal{Y}$. In fact for any assignment x , $\max_{I \in \mathcal{F}} f_I(x_I) \in \mathcal{Y}$.

Example Bottleneck Assignment Problem: given a matrix $D \in \mathbb{R}^{N \times N}$, select a subset of entries of size N that includes exactly one entry from each row and each column, whose largest value is as small as possible. As an applica-

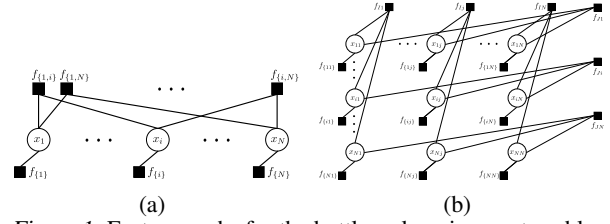


Figure 1. Factor-graphs for the bottleneck assignment problem

tion assume the entry $D_{i,j}$ is the time required by worker i to finish task j . The min-max assignment minimizes the maximum time required by any worker to finish his assignment. This problem is also known as bottleneck bipartite matching and belongs to class \mathcal{P} (e.g., Garfinkel 1971). Here we show two factor-graph representations of this problem.

Categorical variable representation: Consider a factor-graph with $x = \{x_1, \dots, x_N\}$, where each variable $x_i \in \{1, \dots, N\}$ indicates the column of the selected entry in row i of D . For example $x_1 = 5$ indicates the fifth column of the first row is selected (see Figure 1(a)). Define the following factors: (a) local factors $f_{\{i\}}(x_{\{i\}}) = D_{i,x_i}$ and (b) pairwise factors $f_{\{i,j\}}(x_{\{i,j\}}) = \infty \mathbb{I}(x_i = x_j) - \infty \mathbb{I}(x_i \neq x_j)$ that enforce the constraint $x_i \neq x_j$. Here $\mathbb{I}(\cdot)$ is the indicator function –i.e., $\mathbb{I}(True) = 1$ and $\mathbb{I}(False) = 0$. Also by convention $\infty \cdot 0 \triangleq 0$. Note that if $x_i = x_j$, $f_{\{i,j\}}(x_{\{i,j\}}) = \infty$, making this choice unsuitable in the min-max solution. On the other hand with $x_i \neq x_j$, $f_{\{i,j\}}(x_{\{i,j\}}) = -\infty$ and this factor has no impact on the min-max value.

Binary variable representation: Consider a factor-graph where $x = [x_{1-1}, \dots, x_{1-N}, x_{2-1}, \dots, x_{2-N}, \dots, x_{N-1}, \dots, x_{N-N}] \in \{0, 1\}^{N \times N}$ indicates whether each entry is selected $x_{i-j} = 1$ or not $x_{i-j} = 0$ (Figure 1(b)). Here the factors $f_I(x_I) = -\infty \mathbb{I}(\sum_{i \in \partial I} x_i = 1) + \infty \mathbb{I}(\sum_{i \in \partial I} x_i \neq 1)$ ensures that only one variable in each row and column is selected and local factors $f_{i-j}(x_{i-j}) = x_{i-j} D_{i,j} - \infty(1 - x_{i-j})$ have any effect only if $x_{i-j} = 1$.

The min-max assignment in both of these factor-graphs as defined in eq. (1) gives a solution to the bottleneck assignment problem.

For any $y \in \mathcal{Y}$ in the range of factor values, we *reduce* the original min-max problem to a CSP using the following reduction. For any $y \in \mathcal{Y}$, μ_y -**reduction** of the min-max problem eq. (1), is given by

$$\mu_y(x) \triangleq \frac{1}{Z_y} \prod_I \mathbb{I}(f_I(x_I) \leq y) \quad (2)$$

where

$$Z_y \triangleq \sum_{\mathcal{X}} \prod_I \mathbb{I}(f_I(x_I) \leq y) \quad (3)$$

is the normalizing constant and $\mathbb{I}(\cdot)$ is the indicator function.¹ This distribution defines a CSP over \mathcal{X} , where $\mu_y(x) > 0$ iff x is a satisfying assignment. Moreover, Z_y gives the number of satisfying assignments.

We will use $f_I^y(x_I) \triangleq \mathbb{I}(f_I(x_I) \leq y)$ to refer to reduced factors. The following theorem is the basis of our approach in solving min-max problems.

Theorem 2.1² *Let x^* denote the min-max solution and y^* be its corresponding value –i.e., $y^* = \max_I f_I(x_I^*)$. Then $\mu_y(x)$ is satisfiable for all $y \geq y^*$ (in particular $\mu_{y^*}(x^*) > 0$) and unsatisfiable for all $y < y^*$.*

This theorem enables us to find a min-max assignment by solving a sequence of CSPs. Let $y^{(1)} \leq \dots \leq y^{(N)}$ be an ordering of $y \in \mathcal{Y}$. Starting from $y = y^{(N/2)}$, if μ_y is satisfiable then $y^* \leq y$. On the other hand, if μ_y is not satisfiable, $y^* > y$. Using binary search, we need to solve $\log(|\mathcal{Y}|)$ CSPs to find the min-max solution. Moreover at any time-step during the search, we have both upper and lower bounds on the optimal solution. That is $\underline{y} < y^* \leq \bar{y}$, where \underline{y} is the latest unsatisfiable and \bar{y} is the latest satisfiable reduction.

Example Bottleneck Assignment Problem: Here we define the μ_y -reduction of the binary-valued factor-graph for this problem by reducing the constraint factors to $f^y(x_I) = \mathbb{I}(\sum_{i \in \partial I} x_i = 1)$ and the local factors to $f_{\{i-j\}}^y(x_{i-j}) = x_{i-j} \mathbb{I}(D_{i,j} \leq y)$. The μ_y -reduction can be seen as defining a uniform distribution over the all valid assignments (i.e., each row and each column has a single entry) where none of the N selected entries are larger than y .

2.1. Reduction to Min-Sum

Kabadi & Punnen (2004) introduce a simple method to transform instances of bottleneck TSP to TSP. Here we show how this results extends to min-max problems over factor-graphs.

Lemma 2.2 *Any two sets of factors, $\{f_I\}_{I \in \mathcal{F}}$ and $\{f'_I\}_{I \in \mathcal{F}}$, have identical min-max solution*

$$\arg_x \min_I \max f_I(x_I) = \arg_x \min_I \max f'_I(x_I)$$

if $\forall I, J \in \mathcal{F}, x_I \in \mathcal{X}_I, x_J \in \mathcal{X}_J$

$$f_I(x_I) < f_J(x_J) \Leftrightarrow f'_I(x_I) < f'_J(x_J)$$

This lemma simply states that what matters in the min-max solution is the *relative ordering* in the factor-values.

Let $y^{(1)} \leq \dots \leq y^{(N)}$ be an ordering of elements in \mathcal{Y} , and let $r(f_I(x_I))$ denote the rank of $y_I = f_I(x_I)$ in this ordering. Define the min-sum reduction of $\{f_I\}_{I \in \mathcal{F}}$ as

$$f'_I(x_I) = 2^{r(f_I(x_I))} \quad \forall I \in \mathcal{F}$$

Theorem 2.3

$$\arg_x \min \sum_I f'_I(x_I) = \arg_x \min \max_I f_I(x_I)$$

where $\{f'_I\}_I$ is the min-sum reduction of $\{f_I\}_I$.

Although this allows us to use min-sum message passing to solve min-max problems, the values in the range of factors grow exponentially fast, resulting in numerical problems.

3. Solving CSP-reductions

Previously in solving CSP-reductions, we assumed an ideal CSP solver. However, finding an assignment x such that $\mu_y(x) > 0$ or otherwise showing that no such assignment exists is in general NP-hard (Cooper 1990). However, message passing methods have been successfully used to provide state-of-the-art results in solving difficult CSPs. We use Perturbed Belief Propagation (PBP Ravanbakhsh & Greiner 2014) for this purpose. By using an incomplete solver (Kautz et al. 2009), we lose the lower-bound \underline{y} on the optimal min-max solution, as PBP may not find a solution even if the instance is satisfiable.³ However the following theorem states that, as we increase y from the min-max value y^* , the number of satisfying assignments to μ_y -reduction increases, making it potentially easier to solve.

Proposition 3.1

$$y_1 < y_2 \Rightarrow Z_{y_1} \leq Z_{y_2} \quad \forall y_1, y_2 \in \mathbb{R}$$

where Z_y is defined in eq. (3).

This means that the sub-optimality of our solution is related to our ability to solve CSP-reductions – that is, as the gap $y - y^*$ increases, the μ_y -reduction potentially becomes easier to solve.

PBP is a message passing method that interpolates between Belief Propagation (BP) and Gibbs Sampling. At each iteration, PBP sends a message from variables to factors ($\nu_{i \rightarrow I}$) and vice versa ($\nu_{I \rightarrow i}$). The factor to variable message is given by

$$\nu_{I \rightarrow i}(x_i) \propto \sum_{x_{I \setminus i} \in \mathcal{X}_{\partial I \setminus i}} f_I^y(x_i, x_{I \setminus i}) \prod_{j \in \partial I \setminus i} \nu_{j \rightarrow I}(x_j) \quad (4)$$

¹ To always have a well-defined probability, we define $\frac{0}{0} \triangleq 0$.

² All proofs appear in Appendix A.

³ To maintain the lower-bound one should be able to correctly assert unsatisfiability.

where the summation is over all the variables in I except for x_i . The variable to factor message for PBP is slightly different from BP; it is a linear combination of the BP message update and an indicator function, defined based on a sample from the current estimate of marginal $\hat{\mu}(x_i)$:

$$\nu_{i \rightarrow I}(x_i) \propto (1 - \gamma) \prod_{J \in \partial i \setminus I} \nu_{J \rightarrow i}(x_i) + \gamma \mathbb{I}(\hat{x}_i = x_i) \quad (5)$$

$$\text{where } \hat{x}_i \sim \hat{\mu}(x_i) \propto \prod_{J \in \partial i} \nu_{J \rightarrow i}(x_i) \quad (6)$$

where for $\gamma = 0$ we have BP updates and for $\gamma = 1$, we have Gibbs Sampling. PBP starts at $\gamma = 0$ and linearly increases γ at each iteration, ending at $\gamma = 1$ at its final iteration. At any iteration, PBP may encounter a contradiction where the product of incoming messages to node i is zero for all $x_i \in \mathcal{X}_i$. This could mean that either the problem is unsatisfiable or PBP is not able to find a solution. However if it reaches the final iteration, PBP produces a sample from $\mu_y(x)$, which is a solution to the corresponding CSP. The number of iterations T is the only parameter of PBP and increasing T , increases the chance of finding a solution (Only downside is time complexity; nb., no chance of a false positive.)

3.1. Computational Complexity

PBP's time complexity per iteration is identical to that of BP—*i.e.*,

$$\mathcal{O} \left(\sum_I (|\partial I| |\mathcal{X}_I|) + \sum_i (|\partial i| |\mathcal{X}_i|) \right) \quad (7)$$

where the first summation accounts for all factor-to-variable messages (eq. (4))⁴ and the second one accounts for all variable-to-factor messages (eq. (5)).

To perform binary search over \mathcal{Y} we need to sort \mathcal{Y} , which requires $\mathcal{O}(|\mathcal{Y}| \log(|Y|))$. However, since $|\mathcal{Y}_i| \leq |\mathcal{X}_i|$ and $|\mathcal{Y}| \leq \sum_I |\mathcal{Y}_i|$, the cost of sorting is already contained in the first term of eq. (7), and may be ignored in asymptotic complexity.

The only remaining factor is that of binary search itself, which is $\mathcal{O}(\log(|\mathcal{Y}|)) = \mathcal{O}(\log(\sum_I (|\mathcal{X}_I|)))$ —*i.e.*, at most logarithmic in the cost of PBP's iteration (*i.e.*, first term in eq. (7)). Also note that the factors added as constraints only take two values of $\pm\infty$, and have no effect in the cost of binary search.

As this analysis suggests, the dominant cost is that of sending factor-to-variable messages, where a factor may depend

⁴The $|\partial I|$ is accounting for the number of messages that are sent from each factor. However if the messages are calculated synchronously this factor disappears. Although more expensive, in practice, asynchronous message updates performs better.

on a large number of variables. The next section shows that many interesting factors are sparse, which allows efficient calculation of messages.

3.2. High-Order Factors

The factor-graph formulation of many interesting min-max problems involves sparse high-order factors. In all such factors, we are able to significantly reduce the $\mathcal{O}(|\mathcal{X}_I|)$ time complexity of calculating factor-to-variable messages. Efficient message passing over such factors is studied by several works in the context of sum-product and max-product inference (*e.g.*, Potetz & Lee 2008; Gupta et al. 2007; Tarlow et al. 2010; Tarlow et al. 2012). The simplest form of sparse factor in our formulation is the so-called Potts factor, $f_{\{i,j\}}^y(x_i, x_j) = \mathbb{I}(x_i = x_j) \phi(x_i)$. This factor assumes the same domain for all the variables ($\mathcal{X}_i = \mathcal{X}_j \forall i, j$) and its tabular form is non-zero only across the diagonal. It is easy to see that this allows the marginalization of eq. (4) to be performed in $\mathcal{O}(|\mathcal{X}_i|)$ rather than $\mathcal{O}(|\mathcal{X}_i| |\mathcal{X}_j|)$. Another factor of similar form is the inverse Potts factor, $f_{\{i,j\}}^y(x_i, x_j) = \mathbb{I}(x_i \neq x_j)$, which ensures $x_i \neq x_j$. In fact any pair-wise factor that is a constant plus a band-limited matrix allows $\mathcal{O}(|\mathcal{X}_i|)$ inference (*e.g.*, see Section 4.4).

In Section 4, we use cardinality factors, where $\mathcal{X}_i = \{0, 1\}$ and the factor is defined based on the number of non-zero values—*i.e.*, $f_K^y(x_K) = f_K^y(\sum_{i \in K} x_i)$. The μ_y -reduction of the factors we use in the binary representation of the bottleneck assignment problem is in this category. Gail et al. (1981) propose a simple $\mathcal{O}(|\partial I| K)$ method for $f_K^y(x_K) = \mathbb{I}(\sum_{i \in K} x_i = K)$. We refer to this factor as K-of-N factor and use similar algorithms for at-least-K-of-N $f_K^y(x_K) = \mathbb{I}(\sum_{i \in K} x_i \geq K)$ and at-most-K-of-N $f_K^y(x_K) = \mathbb{I}(\sum_{i \in K} x_i \leq K)$ factors (see Appendix B). An alternative for more general forms of high order factors is the clique potential of Potetz & Lee (2008). For large K , more efficient methods evaluate the sum of pairs of variables using auxiliary variables forming a binary tree and use Fast Fourier Transform to reduce the complexity of K-of-N factors to $\mathcal{O}(N \log(N)^2)$ (see Tarlow et al. (2012) and its references).

4. Applications

Here we introduce the factor-graph formulation for several NP-hard min-max problems. Interestingly the CSP-reduction for each case is an important NP-hard problem. Table 1 shows the relationship between the min-max and the corresponding CSP and the factor α in the constant factor approximation available for each case. For example, $\alpha = 2$ means the results reported by some algorithm is guaranteed to be within factor 2 of the optimal min-max value \hat{y}^* when the distances satisfy the triangle inequality.

Table 1. Min-max combinatorial problems and the corresponding CSP-reductions. The last column reports the best α -approximations when triangle inequality holds. * indicates best possible approximation.

min-max problem	μ_y -reduction	msg-passing cost	α
min-max clustering	clique cover problem	$\mathcal{O}(N^2 K \log(N))$	2^* (Gonzalez 1985)
K-packing	max-clique problem	$\mathcal{O}(N^2 K \log(N))$	N/A
(weighted) K-center problem	dominating set problem	$\mathcal{O}(N^3 \log(N))$ or $\mathcal{O}(NR^2 \log(N))$	$\min(3, 1 + \frac{\max_i w_i}{\min_i w_i})$ (Dyer & Frieze 1985)
asymmetric K-center problem	set cover problem	$\mathcal{O}(N^3 \log(N))$ or $\mathcal{O}(NR^2 \log(N))$	$\log(N)^*$ (Panigrahy & Vishwanathan 1998; Chuzhoy et al. 2005)
bottleneck TSP	Hamiltonian cycle problem	$\mathcal{O}(N^3 \log(N))$	2^* (Parker & Rardin 1984)
bottleneck Asymmetric TSP	directed Hamiltonian cycle	$\mathcal{O}(N^3 \log(N))$	$\log(n) / \log(\log(n))$ (An et al. 2010)

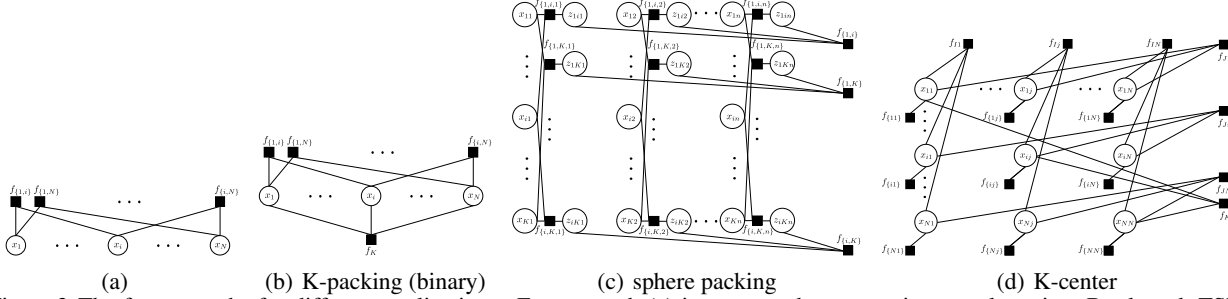


Figure 2. The factor-graphs for different applications. Factor-graph (a) is common between min-max clustering, Bottleneck TSP and K-packing (categorical). However the definition of factors is different in each case.

ity. This table also includes the complexity of the message passing procedure (assuming asynchronous message updates) in finding the min-max solution. See Appendix C for details.

4.1. Min-Max Clustering

Given a symmetric matrix of pairwise distances $D \in \mathbb{R}^{N \times N}$ between N data-points, and a number of clusters K , min-max clustering seeks a partitioning of data-points that minimizes the maximum distance between all the pairs in the same partition.

Let $x = \{x_1, \dots, x_N\}$ with $x_i \in \mathcal{X}_i = \{1, \dots, K\}$ be the set of variables, where $x_i = k$ means, point i belongs to cluster k . The Potts factor $f_{\{i,j\}}(x_i, x_j) = \mathbb{I}(x_i = x_j)D_{i,j} - \infty \mathbb{I}(x_i \neq x_j)$ between any two points is equal to $D_{i,j}$ if points i and j belong the same cluster and $-\infty$ otherwise (Figure 2(a)). When applied to this factor graph, the min-max solution x^* of eq. (1) defines the clustering of points that minimizes the maximum inter-cluster distances.

Now we investigate properties of μ_y -reduction for this factor-graph. The y -neighborhood graph for distance matrix $D \in \mathbb{R}^{N \times N}$ is defined as graph $\mathcal{G}(D, y) = (\mathcal{V}, \mathcal{E})$, where $\mathcal{V} = \{1, \dots, N\}$ and $\mathcal{E} = \{(i, j) \mid D_{i,j} \leq y\}$. Note than this definition is also valid for an asymmetric adjacency matrix D . In such cases, the y -neighborhood graph is a directed-graph.

The K -clique-cover $\mathcal{C} = \{\mathcal{C}_1, \dots, \mathcal{C}_K\}$ for a graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ is a partitioning of \mathcal{V} to at most K partitions such that $\forall i, j, k \quad i, j \in \mathcal{C}_k \Rightarrow (i, j) \in \mathcal{E}$.

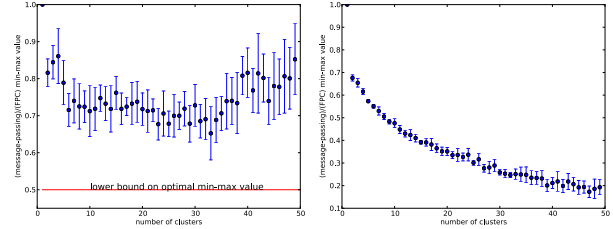


Figure 3. Min-max clustering of 100 points with varying numbers of clusters (x-axis). Each point is an average over 10 random instances. The y-axis is the ratio of the min-max value obtained by message passing ($T = 50$ iterations for PBP) over the min-max value of FPC. (left) Clustering of random points in 2D Euclidean space. The red line is the lower bound on the optimal result based on the worst case guarantee of FPC. (right) Using symmetric random distance matrix where $D_{i,j} = D_{j,i} \sim U(0, 1)$.

Proposition 4.1 The μ_y -reduction of the min-max clustering factor-graph defines a uniform distribution over the K -clique-covers of $\mathcal{G}(D, y)$.

Figure 3 compares the performance of min-max clustering using message passing to that of Furthest Point Clustering (FPC) of Gonzalez (1985) which is 2-optimal when the triangle inequality holds. Note that message passing solutions are superior even when using Euclidean distance.

4.2. K-Packing

Given a symmetric distance matrix $D \in \mathbb{R}^{N \times N}$ between N data-points and a number of code-words K , the K -packing problem is to choose a subset of K points such that the minimum distance $D_{i,j}$ between any two code-words is max-

imized. Here we introduce two different factor-graph formulations for this problem.

4.2.1. FIRST FORMULATION: BINARY VARIABLES

Let binary variables $x = \{x_1, \dots, x_N\} \in \{0, 1\}^N$, indicate a subset of variables of size K that are selected as code-words (Figure 2(b)). Use the factor $f_K(x) = \infty \mathbb{I}(\sum_i x_i \neq K) - \infty \mathbb{I}(\sum_i x_i = K)$ (here $K = \{1, \dots, N\}$) to ensure this constraint. The μ_y -reduction of this factor for any $-\infty < y < +\infty$ is a K-of-N factor as defined in Section 3.2. Furthermore, for any two variables x_i and x_j , define factor $f_{x_i, x_j}(x_i, x_j) = -D_{i,j}x_i x_j - \infty(1 - x_i x_j)$. This factor is effective only if both points are selected as code-words. The use of $-D_{i,j}$ is to convert the initial max-min objective to min-max.

4.2.2. SECOND FORMULATION: CATEGORICAL VARIABLES

Define the K-packing factor-graph as follows: Let $x = \{x_1, \dots, x_K\}$ be the set of variables where $x_i \in \mathcal{X}_i = \{1, \dots, N\}$ (Figure 2(a)). For every two distinct points $1 \leq i < j \leq K$ define the factor $f_{\{i,j\}}(x_i, x_j) = -D_{x_i, x_j} \mathbb{I}(x_i \neq x_j) + \infty \mathbb{I}(x_i = x_j)$. Here each variable represents a code-word and the last term of each factor ensures that code-words are distinct.

Proposition 4.2 *The μ_y -reduction of the K-packing factor-graph for the distance matrix $D \in \mathbb{R}^{N \times N}$ defines a uniform distribution over the cliques of $\mathcal{G}(-D, -y)$ that are larger than K .*

The μ_y -reduction of our second formulation is similar to the factor-graph used by Ramezanpour & Zecchina (2012) to find non-linear binary codes. The authors consider the Hamming distance between all binary vectors of length n (i.e., $N = 2^n$) to obtain binary codes with known minimum distance y . As we saw, this method is $\mathcal{O}(N^2 K^2) = \mathcal{O}(2^{2n} K^2)$ –i.e., grows exponentially in the number of bits n . In the following section, we introduce a factor-graph formulation specific to categorical variables with Hamming distance that have message passing complexity $\mathcal{O}(n^2 K^2 y)$ –i.e., not exponential in n . Using this formulation we find optimal binary codes where both n and y are large.

4.2.3. SPHERE PACKING WITH HAMMING DISTANCE

Our factor-graph defines a distribution over the K binary vectors of length n such that the distance between every pair of binary vectors is at least y .⁵ Finding so-called “nonlinear binary codes” is a fundamental problem in information theory and a variety of methods have

⁵ For convenience we restrict this construction to the case of binary vectors. A similar procedure may be used to find maximally distanced ternary and q -ary vectors, for arbitrary q .

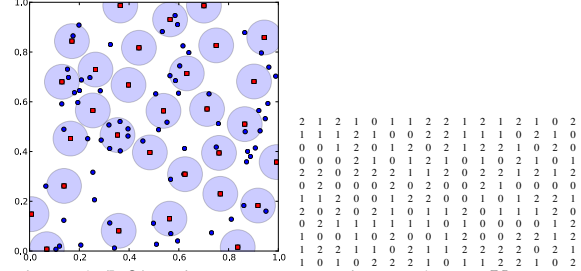


Figure 4. (left) Using message passing to choose $K = 30$ out of $N = 100$ random points in the Euclidean plane to maximize the minimum pairwise distance (with $T = 500$ iterations for PBP). Touching circles show the minimum distance. (right) Example of an optimal ternary code ($n = 16, y = 11, K = 12$), found using the K-packing factor-graph of Section 4.2.3. Here each of $K = 12$ lines contains one code-word of length 16, and every pair of code-words are different in at least $y = 11$ digits.

been used to find better codes, trying either to maximize the number of keywords K or their minimum distance y (e.g., see Litsyn et al. 1999 and its references). Let $x = \{x_{1-1}, \dots, x_{1-n}, x_{2-1}, \dots, x_{2-n}, \dots, x_{K-n}\}$ be the set of our binary variables, where $x_i = \{x_{i-1}, \dots, x_{i-n}\}$ represents the i^{th} binary vector or code-word. Additionally for each $1 \leq i < j \leq K$, define an auxiliary binary vector $z_{i,j} = \{z_{i,j,1}, \dots, z_{i,j,n}\}$ of length n (Figure 2(c)).

For each distinct pair of binary vectors x_i and x_j , and a particular bit $1 \leq k \leq n$, the auxiliary variable $z_{i,j,k} = 1$ iff $x_{i-k} \neq x_{j-k}$. Then we define an at-least- y -of- n factor over $z_{i,j}$ for every pair of code-words, to ensure that they differ in at least y bits.

In more details, define the following factors on x and z :

(a) *z-factors*: For every $1 \leq i < j \leq K$ and $1 \leq k \leq n$, define a factor to ensure that $z_{i,j,k} = 1$ iff $x_{i-k} \neq x_{j-k}$:

$$f(x_{i-k}, x_{j-k}, z_{i,j,k}) = \mathbb{I}(x_{i-k} \neq x_{j-k}) \mathbb{I}(z_{i,j,k} = 1) + \mathbb{I}(x_{i-k} = x_{j-k}) \mathbb{I}(z_{i,j,k} = 0).$$

This factor depends on three binary variables, therefore we can explicitly define its value for each of $2^3 = 8$ possible inputs.

(b) *distance-factors*: For each $z_{i,j}$ define at-least- y -of- n factor:

$$f_K(z_{i,j}) = \mathbb{I}\left(\sum_{1 \leq k \leq n} z_{i,j,k} \geq y\right)$$

Table 2 reports some optimal codes including codes with large number of bits n , recovered using this factor-graph. Here Perturbed BP used $T = 1000$ iterations.

4.3. (Asymmetric) K-center Problem

Given a pairwise distance matrix $D \in \mathbb{R}^{N \times N}$, the K-center problem seeks a partitioning of nodes, with one center per

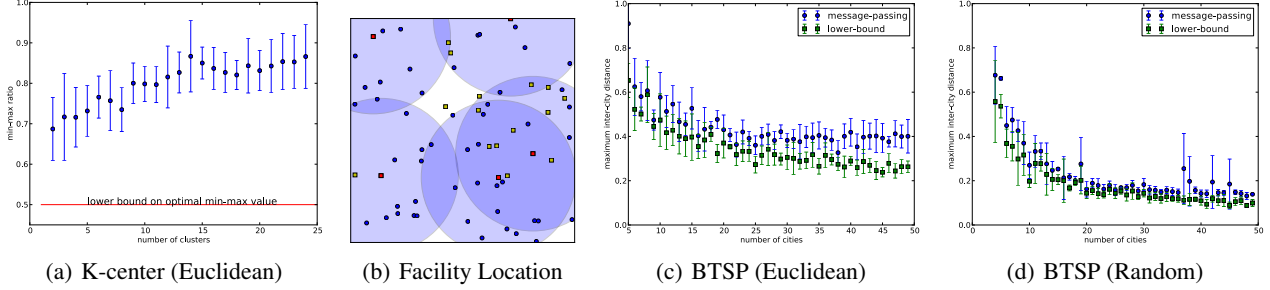


Figure 5. (a) K-center clustering of 50 random points in a 2D plane with various numbers of clusters (x-axis). The y-axis is the ratio of the min-max value obtained by message passing ($T = 500$ for PBP) over the min-max value of 2-approximation of Dyer & Frieze (1985). (b) Min-max K-facility location formulated as an asymmetric K-center problem and solved using message passing. Squares indicate 20 potential facility locations and circles indicate 50 customers. The task is to select 5 facilities (red squares) to minimize the maximum distance from any customer to a facility. The radius of circles is the min-max value. (c,d) The min-max solution for Bottleneck TSP with different number of cities (x-axis) for 2D Euclidean space as well as asymmetric random distance matrices ($T = 5000$ for PBP). The error-bars in all figures show one standard deviation over 10 random instances.

Table 2. Some optimal binary codes from Litsyn et al. 1999 recovered by K-packing factor-graph in the order of increasing y . n is the length of the code, K is the number of code-words and y is the minimum distance between code-words.

n	K	y	n	K	y	n	K	y	n	K	y
8	4	5	11	4	7	14	4	9	16	6	9
17	4	11	19	6	11	20	8	11	20	4	13
23	6	13	24	8	13	23	4	15	26	6	15
27	8	15	28	10	15	28	5	16	26	4	17
29	6	17	29	4	19	33	6	19	34	8	19
36	12	19	32	4	21	36	6	21	38	8	21
39	10	21	35	4	23	39	6	23	41	8	23
39	4	25	43	6	23	46	10	25	47	12	25
41	4	27	46	6	27	48	8	27	50	10	27
44	4	29	49	6	29	52	8	29	53	10	29

partition such that the maximum distance from any node to the center of its partition is minimized. This problem is known to be NP-hard, even for Euclidean distance matrices (Masuyama et al. 1981). Frey & Dueck (2007) use max-product message passing to solve the min-sum variation of this problem –a.k.a. K-median problem. A binary variable factor-graph for the same problem is introduced in Givoni & Frey (2009). Here we introduce a binary variable model for the asymmetric K-center problem. Let $x = \{x_{1-1}, \dots, x_{1-N}, x_{2-1}, \dots, x_{2-N}, \dots, x_{N-N}\}$ denote N^2 binary variables, where $x_{i-j} = 1$ indicates that point i participates in the partition that has j as its center. Now define the following factors:

- A. *local factors*: $\forall i \neq j \quad f_{\{i-j\}}(x_{i-j}) = D_{i,j}x_{i-j} - \infty(1 - x_{i-j})$.
- B. *uniqueness factors*: every point is associated with exactly one center (which can be itself). For every i consider $I = \{i-j \mid 1 \leq j \leq N\}$ and define $f_I(x_I) = \infty\mathbb{I}(\sum_{i-j \in I} x_{i-j} \neq 1) - \infty\mathbb{I}(\sum_{i-j \in I} x_{i-j} = 1)$.
- C. *consistency factors*: when j is selected as a center by any node i , node j also recognizes itself as a center.

$$\forall j, i \neq j \text{ define } f(x_{\{j-j, i-j\}}) = \infty\mathbb{I}(x_{j-j} = 0 \wedge x_{i-j} = 1) - \infty\mathbb{I}(x_{j-j} = 1 \vee x_{i-j} = 0).$$

- D. *K-of-N factor*: only K nodes are selected as centers. Letting $\mathcal{K} = \{i-i \mid 1 \leq i \leq N\}$, define $f_{\mathcal{K}}(x_{\mathcal{K}}) = \infty\mathbb{I}(\sum_{i-i \in \mathcal{K}} x_{i-i} \neq K) - \infty\mathbb{I}(\sum_{i-i \in \mathcal{K}} x_{i-i} = K)$.

For variants of this problem such as the capacitated K-center, additional constraints on the maximum/minimum points in each group may be added as the at-least/at-most K-of-N factors.

We can significantly reduce the number of variables and the complexity (which is $\mathcal{O}(N^3 \log(N))$) by bounding the distance to the center of the cluster \bar{y} . Given an upper bound \bar{y} , we may remove all the variables x_{i-j} for $D_{i,j} > \bar{y}$ from the factor-graph. Assuming that at most R nodes are at distance $D_{i-j} \leq \bar{y}$ from every node j , the complexity of min-max inference drops to $\mathcal{O}(NR^2 \log(N))$.

Figure 5(a) compares the performance of message-passing and the 2-approximation of Dyer & Frieze (1985) when triangle inequality holds. The min-max facility location problem can also be formulated as an asymmetric K-center problem where the distance to all customers is ∞ and the distance from a facility to another facility is $-\infty$ (Figure 5(b)).

The following proposition establishes the relation between the K-center factor-graph above and dominating set problem as its CSP reduction. The *K-dominating set* of graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ is a subset of nodes $\mathcal{D} \subseteq \mathcal{V}$ of size $|\mathcal{D}| = K$ such that any node in $\mathcal{V} \setminus \mathcal{D}$ is adjacent to at least one member of \mathcal{D} –i.e., $\forall i \in \mathcal{V} \setminus \mathcal{D} \quad \exists j \in \mathcal{D} \text{ s.t. } (i, j) \in \mathcal{E}$.

Proposition 4.3 For symmetric distance matrix $D \in \mathbb{R}^{N \times N}$, the μ_y -reduction of the K-center factor-graph

above, is non-zero (i.e., $\mu_y(x) > 0$) iff x defines a K -dominating set for $\mathcal{G}(D, y)$.

Note that in this proposition (in contrast with Propositions 4.1 and 4.2) the relation between the assignments x and K -dominating sets of $\mathcal{G}(D, y)$ is not one-to-one as several assignments may correspond to the same dominating set. Here we establish a similar relation between asymmetric K -center factor-graph and set-cover problem.

Given universe set \mathcal{V} and a set $\mathcal{S} = \{\mathcal{V}_1, \dots, \mathcal{V}_M\}$ s.t. $\mathcal{V}_m \subseteq \mathcal{V}$, we say $\mathcal{C} \subseteq \mathcal{S}$ covers \mathcal{V} iff each member of \mathcal{V} is present in at least one member of \mathcal{C} -i.e., $\bigcup_{\mathcal{V}_m \in \mathcal{C}} \mathcal{V}_m = \mathcal{V}$. Now we consider a natural set-cover problem induced by any directed-graph. Given a directed-graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$, for each node $i \in \mathcal{V}$, define a subset $\mathcal{V}_i = \{j \in \mathcal{V} \mid (j, i) \in \mathcal{E}\}$ as the set of all nodes that are connected to i . Let $\mathcal{S} = \{\mathcal{V}_1, \dots, \mathcal{V}_N\}$ denote all such subsets. An *induced K -set-cover* of \mathcal{G} is a set $\mathcal{C} \subseteq \mathcal{S}$ of size K that covers \mathcal{V} .

Proposition 4.4 *For a given asymmetric distance matrix $D \in \mathbb{R}^{N \times N}$, the μ_y -reduction of the K -center factor-graph as defined above, is non-zero (i.e., $\mu_y(x) > 0$) iff x defines an induced K -set-cover for $\mathcal{G}(D, y)$.*

4.4. (Asymmetric) Bottleneck Traveling Salesman Problem

Given a distance matrix $D \in \mathbb{R}^{N \times N}$, the task in the Bottleneck Traveling Salesman Problem (BTSP) is to find a tour of all N points such that the maximum distance between two consecutive cities in the tour is minimized (Kabadi & Punnen 2004). Any constant-factor approximation for arbitrary instances of this problem is \mathcal{NP} -hard (Parker & Rardin 1984).

Let $x = \{x_1, \dots, x_N\}$ denote the set of variables where $x_i \in \mathcal{X}_i = \{0, \dots, N-1\}$ represents the time-step at which node i is visited. Also, we assume modular arithmetic (module N) on members of \mathcal{X}_i -e.g., $N \equiv 0 \pmod N$ and $1-2 \equiv N-1 \pmod N$. For each pair x_i and x_j of variables, define the factor (Figure 2(a))

$$f_{\{i,j\}}(x_i, x_j) = \infty \mathbb{I}(x_i = x_j) - \infty \mathbb{I}(|x_i - x_j| > 1) \quad (8) \\ + D_{i,j} \mathbb{I}(x_i = x_j - 1) + D_{j,i} \mathbb{I}(x_i = x_j + 1)$$

where the first term ensures $x_i \neq x_j$ and the second term means this factor has no effect on the min-max value when node i and j are not consecutively visited in a path. The third and fourth terms express the distance between i and j depending on the order of visit. Figure 6 shows the tabular form of this factor. In Appendix B.2 we show an $\mathcal{O}(N)$ procedure to marginalize this type of factor.

Here we relate the min-max factor-graph above to a uniform distribution over Hamiltonian cycles.

∞	$D_{i,j}$	$-\infty$	\dots	$-\infty$	$-\infty$	$D_{j,i}$
$D_{j,i}$	∞	$D_{i,j}$	\dots	$-\infty$	$-\infty$	$-\infty$
$-\infty$	$D_{j,i}$	∞	\dots	$-\infty$	$-\infty$	$-\infty$
\vdots	\vdots	\vdots	\ddots	\vdots	\vdots	\vdots
$-\infty$	$-\infty$	$-\infty$	\dots	∞	$D_{i,j}$	$-\infty$
$-\infty$	$-\infty$	$-\infty$	\dots	$D_{j,i}$	∞	$D_{i,j}$
$D_{i,j}$	$-\infty$	$-\infty$	\dots	$-\infty$	$D_{j,i}$	∞

Figure 6. The tabular form of $f_{\{i,j\}}(x_i, x_j)$ used for the bottleneck TSP.

Proposition 4.5 *For any distance matrix $D \in \mathbb{R}^{N \times N}$, the μ_y -reduction of the BTSP factor-graph (shown above), defines a uniform distribution over the (directed) Hamiltonian cycles of $\mathcal{G}(D, y)$.*

Figure 5(c,d) reports the performance of message passing (over 10 instances) as well as a lower-bound on the optimal min-max value for tours of different length (N). Here we report the results for random points in 2D Euclidean space as well as asymmetric random distance matrices. For symmetric case, the lower-bound is the maximum over j of the distance of two closest neighbors to each node j . For the asymmetric random distance matrices, the maximum is over all the minimum length incoming edges and minimum length outgoing edges for each node.⁶

5. Conclusion

This paper introduces the problem of min-max inference in factor-graphs and provides a general methodology to solve such problems. We use Factor-graphs to represent several important combinatorial problems such as min-max clustering, K -clustering, bottleneck TSP and K -packing and use message passing to find near optimal solutions to each problem. In doing so, we also suggest a message passing solution to several \mathcal{NP} -hard decision problems including the clique-cover, max-clique, dominating-set, set-cover and Hamiltonian path problem. For each problem we also analyzed the complexity of message passing and established its practicality using several relevant experiments.

References

- Aissi, Hassene, Bazgan, Cristina, and Vanderpooten, Daniel. Min-max and min-max regret versions of combinatorial optimization problems: A survey. *European Journal of Operational Research*, 197(2):427–438, 2009.
- An, Hyung-Chan, Kleinberg, Robert D, and Shmoys, David B. Approximation algorithms for the bottleneck asymmetric traveling salesman problem. In *Approximation, Randomization, and Combinatorial Optimization. Algorithms and Techniques*, pp. 1–11. Springer, 2010.

⁶If for one node the minimum length in-coming and out-going edges point to the same city, the second minimum length incoming and out-going edges are also considered in calculating a tighter bound.

- Averbakh, Igor. On the complexity of a class of combinatorial optimization problems with uncertainty. *Mathematical Programming*, 90(2):263–272, 2001.
- Bayati, Mohsen, Shah, Devavrat, and Sharma, Mayank. Maximum weight matching via max-product belief propagation. In *ISIT*, pp. 1763–1767. IEEE, 2005.
- Chuzhoy, Julia, Guha, Sudipto, Halperin, Eran, Khanna, Sanjeev, Kortsarz, Guy, Krauthgamer, Robert, and Naor, Joseph Seffi. Asymmetric k-center is $\log^* n$ -hard to approximate. *Journal of the ACM (JACM)*, 52(4):538–551, 2005.
- Cooper, Gregory F. The computational complexity of probabilistic inference using bayesian belief networks. *Artificial intelligence*, 42(2):393–405, 1990.
- Dyer, Martin E and Frieze, Alan M. A simple heuristic for the p-centre problem. *Operations Research Letters*, 3(6):285–288, 1985.
- Edmonds, Jack and Fulkerson, Delbert Ray. Bottleneck extrema. *Journal of Combinatorial Theory*, 8(3):299–306, 1970.
- Frey, Brendan J and Dueck, Delbert. Clustering by passing messages between data points. *science*, 315(5814):972–976, 2007.
- Fulkerson, DR. Flow networks and combinatorial operations research. *The American Mathematical Monthly*, 73(2):115–138, 1966.
- Gail, Mitchell H, Lubin, Jay H, and Rubinstein, Lawrence V. Likelihood calculations for matched case-control studies and survival studies with tied death times. *Biometrika*, 68(3):703–707, 1981.
- Garfinkel, Robert S. An improved algorithm for the bottleneck assignment problem. *Operations Research*, pp. 1747–1751, 1971.
- Givoni, Inmar E and Frey, Brendan J. A binary variable model for affinity propagation. *Neural computation*, 21(6):1589–1600, 2009.
- Gonzalez, Teofilo F. Clustering to minimize the maximum inter-cluster distance. *Theoretical Computer Science*, 38:293–306, 1985.
- Gross, O. The bottleneck assignment problem. Technical report, DTIC Document, 1959.
- Gupta, Rahul, Diwan, Ajit A, and Sarawagi, Sunita. Efficient inference with cardinality-based clique potentials. In *Proceedings of the 24th international conference on Machine learning*, pp. 329–336. ACM, 2007.
- Hochbaum, Dorit S and Shmoys, David B. A unified approach to approximation algorithms for bottleneck problems. *Journal of the ACM (JACM)*, 33(3):533–550, 1986.
- Huang, Bert and Jebara, Tony. Approximating the permanent with belief propagation. *arXiv preprint arXiv:0908.1769*, 2009.
- Ibrahimi, Morteza, Javanmard, Adel, Kanoria, Yashodhan, and Montanari, Andrea. Robust max-product belief propagation. In *Signals, Systems and Computers (ASILOMAR), 2011 Conference Record of the Forty Fifth Asilomar Conference on*, pp. 43–49. IEEE, 2011.
- Kabadi, Santosh N and Punnen, Abraham P. The bottleneck tsp. In *The Traveling Salesman Problem and Its Variations*, pp. 697–735. Springer, 2004.
- Kautz, Henry A, Sabharwal, Ashish, and Selman, Bart. Incomplete algorithms., 2009.
- Kearns, Michael, Littman, Michael L, and Singh, Satinder. Graphical models for game theory. In *Proceedings of the Seventeenth conference on UAI*, pp. 253–260. Morgan Kaufmann Publishers Inc., 2001.
- Khuller, Samir and Sussmann, Yoram J. The capacitated k-center problem. *SIAM Journal on Discrete Mathematics*, 13(3):403–418, 2000.
- Kschischang, FR and Frey, BJ. Factor graphs and the sum-product algorithm. *Information Theory, IEEE*, 2001.
- Litsyn, Simon, M., Rains E., and A., Sloane N. J. The table of nonlinear binary codes. <http://www.eng.tau.ac.il/~litsyn/tableand/>, 1999. [Online; accessed Jan 30 2014].
- Masuyama, Shigeru, Ibaraki, Toshihide, and Hasegawa, Toshiharu. The computational complexity of the m-center problems on the plane. *IEICE TRANSACTIONS (1976-1990)*, 64(2):57–64, 1981.
- Mezard, M., Parisi, G., and Zecchina, R. Analytic and algorithmic solution of random satisfiability problems. *Science*, 297(5582):812–815, August 2002. ISSN 0036-8075, 1095-9203. doi: 10.1126/science.1073287.
- Panigrahy, Rina and Vishwanathan, Sundar. An $o(\log^* n)$ approximation algorithm for the asymmetric p-center problem. *Journal of Algorithms*, 27(2):259–268, 1998.
- Parker, R Gary and Rardin, Ronald L. Guaranteed performance heuristics for the bottleneck travelling salesman problem. *Operations Research Letters*, 2(6):269–272, 1984.
- Potetz, Brian and Lee, Tai Sing. Efficient belief propagation for higher-order cliques using linear constraint nodes. *Computer Vision and Image Understanding*, 112(1):39–54, 2008.
- Ramezanpour, Abolfazl and Zecchina, Riccardo. Cavity approach to sphere packing in hamming space. *Physical Review E*, 85(2):021106, 2012.
- Ravanbakhsh, Siamak and Greiner, Russell. Perturbed message passing for constraint satisfaction problems. *arXiv preprint arXiv:1401.6686*, 2014.
- Schrijver, Alexander. *Min-max results in combinatorial optimization*. Springer, 1983.
- Tarlow, Daniel, Givoni, Inmar E, and Zemel, Richard S. Hop-map: Efficient message passing with high order potentials. In *International Conference on Artificial Intelligence and Statistics*, pp. 812–819, 2010.
- Tarlow, Daniel, Swersky, Kevin, Zemel, Richard S, Adams, Ryan Prescott, and Frey, Brendan J. Fast exact inference for recursive cardinality models. *arXiv preprint arXiv:1210.4899*, 2012.