Classes of Linear Programs Solvable by Coordinate-Wise Minimization*

Tomáš Dlask[0000−0002−1944−6569] (✉) and Tomáš Werner[0000−0002−6161−7157]

Faculty of Electrical Engineering, Czech Technical University in Prague
{dlaskto2,werner}@fel.cvut.cz

Abstract Coordinate-wise minimization is a simple popular method for large-scale optimization. Unfortunately, for general (non-differentiable and/or constrained) convex problems, its fixed points may not be global minima. We present two classes of linear programs (LPs) that coordinate-wise minimization solves exactly. We show that these classes subsume the dual LP relaxations of several well-known combinatorial optimization problems and the method finds a global minimum with sufficient accuracy in reasonable runtimes. Moreover, we experimentally show that the method frequently yields good suboptima or even optima for sparse LPs where optimality is not guaranteed in theory. Though the presented problems can be solved by more efficient methods, our results are theoretically non-trivial and can lead to new large-scale optimization algorithms in the future.

Keywords: Coordinate-wise minimization, Linear programming, LP relaxation

1 Introduction

Coordinate-wise minimization, or *coordinate descent*, is an iterative optimization method, which in every iteration optimizes only over a single chosen variable while keeping the remaining variables fixed. Due to its simplicity, this method is popular among practitioners in large-scale optimization in areas such as machine learning or computer vision, see e.g. [55]. A natural extension of the method is *block-coordinate minimization*, where every iteration minimizes the objective over a block of variables. In this paper, we focus on coordinate minimization with exact updates, where in each iteration a global minimum over the chosen variable is found, applied to convex optimization problems.

For general convex optimization problems, the method need not converge and/or its fixed points need not be global minima. A simple example is the unconstrained minimization of the function $f(x, y) = \max\{x - 2y, y - 2x\}$, which is unbounded but any point with $x = y$ is a coordinate-wise local minimum. Despite this drawback, (block-)coordinate minimization can be very successful for some large-scale convex non-differentiable problems. The prominent example is the class of *convergent message passing* methods for solving dual linear programming relaxation of maximum a posteriori (MAP) inference in graphical models, which can be seen as various forms of (block-)coordinate descent applied to various forms of the dual. In the typical case, the dual LP relaxation boils down to the unconstrained minimization of a convex piece-wise affine (hence non-differentiable) function. These methods include, e.g., max-sum diffusion [36, 46, 52], TRW-S [33], MPLP [25], SRMP [34], MPLP++ [48], and SPAM [49]. They do not guarantee global optimality but for large sparse instances from computer vision the achieved coordinate-wise local optima are very good and the methods are significantly faster

*This work has been supported by the Czech Science Foundation (grant 19-09967S), the OP VVV project CZ.02.1.01/0.0/0.0/16_019/0000765, and the Grant Agency of the Czech Technical University in Prague (grant SGS19/170/OHK3/3T/13).

than competing approaches [47, 31], including popular first-order primal-dual methods such as ADMM [5] or [10].

This is a motivation to look for other classes of convex optimization problems for which (block-)coordinate descent would work well or, alternatively, to extend convergent message passing methods to a wider class of convex problems than the dual LP relaxation of MAP inference. A step in this direction is the work [54], where it was observed that if the minimizer of the problem over the current variable block is not unique, one should choose a minimizer that lies in the *relative interior* of the set of block-optimizers. It is shown that any update satisfying this rule is, in a precise sense, not worse than any other exact update. Message-passing methods such as max-sum diffusion and TRW-S satisfy this rule [53].

To be precise, suppose we minimize a convex function $f\colon X \to \mathbb{R}$ on a closed convex set $X \subseteq \mathbb{R}^n$. We assume that f is bounded from below on X. For brevity of formulation, we rephrase this as the minimization of the extended-valued function $\bar{f}\colon \mathbb{R}^n \to \mathbb{R} \cup \{\infty\}$ such that $\bar{f}(x) = f(x)$ for $x \in X$ and $\bar{f}(x) = \infty$ for $x \notin X$. One iteration of coordinate minimization with the relative interior rule [54] chooses a variable index $i \in [n] = \{1, \dots, n\}$ and replaces an estimate $x^k = (x_1^k, \dots, x_n^k) \in X$ with a new estimate $x^{k+1} = (x_1^{k+1}, \dots, x_n^{k+1}) \in X$ such that[1]

$$\begin{aligned} x_i^{k+1} &\in \operatorname{ri} \operatorname*{argmin}_{y \in \mathbb{R}} \bar{f}(x_1^k, \dots, x_{i-1}^k, y, x_{i+1}^k, \dots, x_n^k), \\ x_j^{k+1} &= x_j^k \quad \forall j \neq i, \end{aligned}$$

where $\operatorname{ri} Y$ denotes the relative interior of a convex set Y. As this is a univariate convex problem, the set $Y = \operatorname{argmin}_{y \in \mathbb{R}} \bar{f}(x_1^k, \dots, x_{i-1}^k, y, x_{i+1}^k, \dots, x_n^k)$ is either a singleton or an interval. In the latter case, the relative interior rule requires that we choose x_i^{k+1} from the interior of this interval. A point $x = (x_1, \dots, x_n) \in X$ that satisfies

$$x_i \in \operatorname{ri} \operatorname*{argmin}_{y \in \mathbb{R}} \bar{f}(x_1, \dots, x_{i-1}, y, x_{i+1}, \dots, x_n)$$

for all $i \in [n]$ is called a (coordinate-wise) *interior local minimum* (ILM) of function f on set X.

Some classes of convex problems are solved by coordinate-wise minimization exactly. E.g., for unconstrained minimization of a differentiable convex function, it is easy to see that any fixed point of the method is a global minimum; moreover, it has been proved that if the function has unique univariate minima, then any limit point is a global minimum [4, §2.7]. The same properties hold for convex functions whose non-differentiable part is separable [50]. These classical results need not assume the relative interior rule [54]. Therefore, it is natural to ask if the relative interior rule can widen the class of convex optimization problems that are exactly solved by coordinate-wise minimization. Leaving convergence aside[2], more precisely we can ask for which problems interior local minima are global minima. A succinct characterization of this class is currently out of reach. Two subclasses of this class are known [33, 46, 52]: the dual LP relaxation of MAP inference with pairwise potential functions and two labels, or with submodular potential functions. In case of LPs, we recently showed that every interior local minimum is a global optimum if and only if a certain propagation rule is able to decide feasibility of an associated system of linear inequalities [17].

1.1 Contribution and Organization

In this paper, we are going to restrict ourselves to linear programs (where f is linear and X is a convex polyhedron) and extend our previous results from [16] where a class of LPs with this property was presented. Also, we showed that this class subsumes the dual LP relaxations of a number of combinatorial optimization problems and experimentally observed coordinate-wise minimization to converge in reasonable time for large instances.

[1] In [54], the iteration is formulated in a more abstract (coordinate-free) notation. Since we focus only on coordinate-wise minimization here, we use a more concrete notation.

[2] We do not discuss convergence in this paper and assume that the method converges to an interior local minimum. This is supported by experiments, e.g., max-sum diffusion and TRW-S have this property. More on convergence can be found in [54].

Next, we extend [16] by the following material. Using our recent result [17], we significantly simplify the proof of the main theorem from [16] and additionally identify a second class of optimally solvable problems, namely, problems with acyclic structure. Additionally, utilizing a completely different proof technique allowed us to use clearer notation of the considered problem. We also present a new formulation of the maximum flow problem where every interior local optimum is a global optimum and coordinate-wise updates have a natural interpretation. We also include additional insights in multiple places, such as applicability of coordinate-wise minimization to a certain formulation of roof dual in pseudoboolean optimization. Newly, we experiment with linear programs with two non-zeros per column (assignment problem, shortest paths, and the LP relaxation of maximum weight matching).

This article is organized as follows: in §2, we point the attention of the reader to the fact that seemingly 'equivalent' reformulations of problems affect the quality of the fixed points of coordinate-wise minimization which explains why only certain favourable forms of LPs are considered. §3 formally states and proves our theoretical results which are then practically verified in §4. Finally, §5 reports our experimental results on problems where optimality of coordinate-wise minimization is not guaranteed in theory, but is frequently attained in practice.

Concerning the limitations of our approach, we note that there exist more efficient algorithms for solving the LPs from §4 to which our optimality theorem applies, such as reduction to max-flow. These parts thus only serve to verify our theoretical results and show the performance of coordinate-wise optimization when compared to an off-the-shelf LP solver. Regarding §5, we are aware of the fact that coordinate-wise optimization is not likely to replace the well-established polynomial-time algorithms for shortest paths, maximum weight matching, or assignment problem, however, we may draw an important conclusion from our results: even though a class of LPs may not in theory be solvable to optimality by coordinate-wise optimization, fixed points of this method may be close to global optima of the LP or even attain them in practice. Such approach may be useful for problems where no efficient method exists. As an example, coordinate-wise minimization applied on the dual LP relaxation of Max-SAT with clauses of length 3 or more, which is as hard to solve as any LP [44], frequently attained points not far from global optima even though this is not guaranteed in theory.

2 Reformulations of Problems

Before presenting our main result, we make an important remark: while a convex optimization problem can be reformulated in many ways to an 'equivalent' problem which has the same global minima, not all of these transformations are equivalent with respect to coordinate-wise minimization, in particular, not all preserve interior local minima.

Example 1. *One example is dualization. If coordinate-wise minimization achieves good local (or even global) minima on a convex problem, it can get stuck in very poor local minima if applied to its dual. Indeed, trying to apply (block-)coordinate minimization to the* primal *LP relaxation of MAP inference (linear optimization over the local marginal polytope) has been futile so far.* ■

Example 2. *Consider the linear program* $\min\{x_1+x_2 \mid x_1, x_2 \geq 0\}$*, which has one interior local minimum with respect to individual coordinates that coincides with the unique global optimum. But if one adds a redundant constraint, namely* $x_1 = x_2$*, then any feasible point will become an interior local minimum w.r.t. individual coordinates because the redundant constraint blocks changing the variable* x_i *without changing* x_{3-i} *for both* $i \in \{1, 2\}$*.* ■

Example 3. *Let* $m, n, p \in \mathbb{N}$*. Let* $a_{ij} \in \mathbb{R}^p$ *and* $b_{ij} \in \mathbb{R}$ *be given for all* $i \in [m]$*,* $j \in [n]$*. The linear program*[3]

$$\min\Big\{\sum_{j=1}^{n} z_j \;\Big|\; z \in \mathbb{R}^n,\ x \in \mathbb{R}^p,\ z_j \geq a_{ij}^T x + b_{ij}\ \forall i \in [m],\ j \in [n]\Big\} \tag{1}$$

[3] The expression $x^T y$ stands for the scalar product of two vectors x, y.

can be equivalently written as

$$\min_{x\in\mathbb{R}^p} \sum_{j=1}^{n} \max_{i=1}^{m}(a_{ij}^T x + b_{ij}). \tag{2}$$

Optimizing over the individual variables by coordinate-wise minimization in (1) *does not yield the same interior local optima as in* (2). *For instance, assume that* $m = 2$, $n = 3$, $p = 1$, *and the problem* (2) *is given as*

$$\min\ (\max\{x,0\} + \max\{-x,-1\} + \max\{-x,-2\}), \tag{3}$$

where $x \in \mathbb{R}$. *Then, when optimizing directly in form* (3), *one can see that all the interior local optima are global optimizers.*

However, when one introduces the variables $z \in \mathbb{R}^3$ *and applies coordinate-wise minimization on the corresponding problem* (1), *then there are interior local optima that are not global optimizers, for example* $x = z_1 = z_2 = z_3 = 0$, *which is an interior local optimum, but not a global optimum.*

On the other hand, optimizing over blocks of variables $\{z_1, \dots, z_n, x_i\}$ *for each* $i \in [p]$ *in case* (1) *is equivalent to optimization over individual* x_i *in formulation* (2). ■

We remark that the underlying principle which explains how seemingly 'equivalent' reformulations of a problem influence the applicability of coordinate-wise optimization was in detail explained in [17] by relating block-coordinate minimization in linear programs to constraint propagation in linear (in)equalities.

3 Main Result

We consider the pair of mutually dual linear programs

$$\begin{array}{lll|l} \min\ b^T\varphi + 1^T\alpha & \max\ \overline{\varphi}^T z + \underline{\varphi}^T y + c^T x & & \text{(4a)} \\ \alpha_j + A_{:j}^T\varphi \ge c_j & x_j \ge 0 & \forall j \in [n] & \text{(4b)} \\ \alpha_j \ge 0 & x_j \le 1 & \forall j \in [n] & \text{(4c)} \\ \varphi_i \ge \underline{\varphi}_i & y_i \ge 0 & \forall i \in [m] & \text{(4d)} \\ \varphi_i \le \overline{\varphi}_i & z_i \le 0 & \forall i \in [m] & \text{(4e)} \\ \varphi_i \in \mathbb{R} & A_{i:}^T x + y_i + z_i = b_i & \forall i \in [m] & \text{(4f)} \end{array}$$

where the primal is on the left and the dual on the right, and $A \in \mathbb{R}^{m\times n}$, $b \in \mathbb{R}^m$, $c \in \mathbb{R}^n$, $\underline{\varphi} \in (\mathbb{R} \cup \{-\infty\})^m$, and $\overline{\varphi} \in (\mathbb{R} \cup \{\infty\})^m$ (assuming $\underline{\varphi} < \overline{\varphi}$) are given constants. We optimize over variables (φ, α) in the primal and (x, y, z) in the dual. To fix notation, $A_{:j}$ and $A_{i:}$ denotes the j-th column and i-th row of A, respectively. A_{ij} stands for the entry of A in row i and column j.

Clearly, at primal optimum we have

$$\alpha_j = \max\{c_j - A_{:j}^T\varphi, 0\} \quad \forall j \in [n] \tag{5}$$

which allows us to simplify the primal as a box-constrained minimization of a convex piecewise-affine function, i.e.,

$$\min\ b^T\varphi + \sum_{j\in[n]} \max\{c_j - A_{:j}^T\varphi, 0\} \tag{6a}$$

$$\underline{\varphi}_i \le \varphi_i \le \overline{\varphi}_i \quad \forall i \in [m]. \tag{6b}$$

Analogously to Example 3, optimizing (6) along single coordinates is in one-to-one correspondence with optimizing the primal (4) along m blocks of variables where each block contains all α variables and a single variable φ_i.

Definition 1. *Let $A \in \mathbb{R}^{m\times n}$ define a bipartite graph*[4] $([m],[n],E)$ *with $m+n$ vertices where $\{i,j\} \in E$ if $a_{ij} \neq 0$. We say that A is* bipartite-acyclic *if $([m],[n],E)$ is acyclic.*

Recall that a square matrix $A \in \{0,1\}^{n\times n}$ is a *permutation matrix* if each row and each column of A contains exactly one 1 and all other entries are zero [8, §1.1]. A matrix $A \in \mathbb{R}^{m\times n}$ is a *(possibly rectangular) diagonal matrix* if $\forall i \in [m], j \in [n] : i \neq j \implies A_{ij} = 0$, i.e., only elements A_{ii}, $i \in [\min\{m,n\}]$ can be non-zero.

Definition 2. *Matrix $A \in \mathbb{R}^{m\times n}$ is called* 2-in-row *if there exist*

- *a matrix $B \in \mathbb{R}^{m\times n'}$, $0 \le n' \le n$ with at most 2 non-zero elements per row,*
- *a (possibly rectangular) diagonal matrix $D \in \mathbb{R}^{m\times(n-n')}$,*
- *and permutation matrices $P \in \mathbb{R}^{m\times m}$ and $P' \in \mathbb{R}^{n\times n}$*

such that $A = P\,[B|D]\,P'$ where $[B|D] \in \mathbb{R}^{m\times n}$ is the block matrix created by placing B and D next to each other.

Informally, matrix A is 2-in-row if it can be constructed by permuting the rows and columns of a matrix $[B|D]$ where B contains at most 2 non-zero elements per row and D is a (possibly rectangular) diagonal matrix. From this point of view, one can characterize 2-in-row matrices as matrices whose each row either contains at most 2 non-zero elements or, if some row contains 3 non-zero elements, then one of the elements needs to be the only non-zero element in its column.

Notice that the definition of a 2-in-row matrix does not forbid the case when $n' = n$ or $n' = 0$. In such case, we are left with only one of the matrices B or D and the condition on matrix A thus reads $A = PBP'$ or $A = PDP'$, respectively.

Theorem 1. *Let $A \in \{-1,0,1\}^{m\times n}$, $b \in \mathbb{Z}^m$, and $c \in \mathbb{R}^n$. If A is 2-in-row or bipartite-acyclic, then any ILM of* (6) *w.r.t. individual coordinates is a global optimum.*

Theorem 1 for bipartite-acyclic matrices can be seen as a generalization of the result for acyclic instances of Max-SAT in [15]. We are going to prove Theorem 1 using the result in [17] which says that every ILM of a polyhedral problem is a global optimum if certain constraint propagation method is able to detect feasibility of a system defined by complementary slackness conditions.

Even though the proof for 2-in-row matrices was already given in [16], we present its simplified version here, which does not require automated checking.

3.1 Coordinate-Wise Optimization of (6)

In every update of coordinate-wise minimization, we need to minimize problem (6) over a single variable φ_i, $i \in [m]$ and update it to a value from the relative interior of the set of optimizers. Objective function (6a) restricted to φ_i reads (up to a constant)

$$b_i\varphi_i + \sum_{\substack{j\in[n]\\ A_{ij}\neq 0}} \max\{k_j - A_{ij}\varphi_i, 0\} \tag{7}$$

where

$$k_j = c_j - \sum_{\substack{i'\in[m]\\ i'\neq i}} A_{i'j}\varphi_{i'} \tag{8}$$

are constants. This is the minimization of a univariate convex piecewise-affine function whose breakpoints[5] are k_j/A_{ij} for each $j \in [n]$, $A_{ij} \neq 0$. To find its minimum subject to $\underline{\varphi}_i \le \varphi_i \le \overline{\varphi}_i$, it is sufficient to consider the cases listed below.

[4] In [8, §1.1], this is called the *bipartite graph associated with A*.

[5] By a breakpoint, we mean a point of non-differentiability.

If function (7) is strictly decreasing (resp. increasing) and $\overline{\varphi}_i$ (resp. $\underline{\varphi}_i$) is finite, then update $\varphi_i := \overline{\varphi}_i$ (resp. $\varphi_i := \underline{\varphi}_i$) which is the unique minimum.

If function (7) has an (possibly unbounded) interval $[l, u]$ as its set of minimizers, then the set of minimizers subject to $\underline{\varphi}_i \leq \varphi_i \leq \overline{\varphi}_i$ is the projection of $[l, u]$ onto $[\underline{\varphi}_i, \overline{\varphi}_i]$, i.e., an interval $[l', u']$ where $l' = h_{[\underline{\varphi}_i, \overline{\varphi}_i]}(l)$, $u' = h_{[\underline{\varphi}_i, \overline{\varphi}_i]}(u)$ and $h_{[p,q]}$ denotes projection onto the set $[p, q]$, i.e., $h_{[p,q]}(x) = \max\{\min\{x, q\}, p\}$. If the projection is a bounded interval $[l', u']$, the variable is assigned the middle point from this interval, i.e., $\varphi_i := (l' + u')/2$. If the projection is unbounded, i.e., $[l', \infty)$, then we set $\varphi_i := l' + \Delta$, where $\Delta > 0$ is a fixed constant. In case of $(-\infty, u']$, the variable is updated as $\varphi_i := u' - \Delta$.

To identify which case occurred, one should analyse the slopes of the function between its breakpoints and the region of optima corresponds to the interval where the function (7) is constant. If there is no such interval, then its (unrestricted) minimum is at a breakpoint where the function changes from decreasing to increasing. In other cases, (7) is unbounded and therefore also the original problem (6) is unbounded.

3.2 Proof of Theorem 1

To prove Theorem 1, we will proceed by presenting individual propositions which are then going to be utilized for its proof. In detail, the two parts of the theorem, depending on whether the matrix is 2-in-row or bipartite-acyclic, will be proven separately.

Proposition 1. *Feasible solution $\varphi \in \mathbb{R}^m$ is optimal for (6) if there exist $x \in \mathbb{R}^n$, $y, z \in \mathbb{R}^m$ such that*

$$\begin{aligned}
x_j &\in [0,1] && \forall j \in X^U(\varphi) && \text{(9a)}\\
x_j &= 0 && \forall j \in X^0(\varphi) && \text{(9b)}\\
x_j &= 1 && \forall j \in X^1(\varphi) && \text{(9c)}\\
y_i &= 0 && \forall i \in Y^0(\varphi) && \text{(9d)}\\
y_i &\geq 0 && \forall i \in Y^+(\varphi) && \text{(9e)}\\
z_i &= 0 && \forall i \in Z^0(\varphi) && \text{(9f)}\\
z_i &\leq 0 && \forall i \in Z^-(\varphi) && \text{(9g)}\\
A_{i:}^T x + y_i + z_i &= b_i && \forall i \in [m] && \text{(9h)}
\end{aligned}$$

where $X^U(\varphi), X^0(\varphi), X^1(\varphi)$ (resp. $Y^0(\varphi), Y^+(\varphi)$ and $Z^0(\varphi), Z^-(\varphi)$) is a partition of $[n]$ (resp. $[m]$ and $[m]$) given as

$$\begin{aligned}
X^U(\varphi) &= \{j \in [n] \mid A_{:j}^T \varphi = c_j\} & Y^0(\varphi) &= \{i \in [m] \mid \varphi_i > \underline{\varphi}_i\} && \text{(10a)}\\
X^0(\varphi) &= \{j \in [n] \mid A_{:j}^T \varphi > c_j\} & Y^+(\varphi) &= \{i \in [m] \mid \varphi_i = \underline{\varphi}_i\} && \text{(10b)}\\
X^1(\varphi) &= \{j \in [n] \mid A_{:j}^T \varphi < c_j\} & Z^0(\varphi) &= \{i \in [m] \mid \varphi_i < \overline{\varphi}_i\} && \text{(10c)}\\
& & Z^-(\varphi) &= \{i \in [m] \mid \varphi_i = \overline{\varphi}_i\} && \text{(10d)}
\end{aligned}$$

Proof. Follows from the complementary slackness theorem applied to the primal-dual pair (4) and noting the substitution (5). □

Before we proceed further, we need to recall the precise connection between constraint propagation and block-coordinate minimization which we discovered in [17]. In [17], this was shown for LPs in standard form, but generalization for LPs with inequalities was also discussed. We will now apply the results of [17] to the considered primal-dual pair (4) when updating along m blocks of variables, each containing all α variables and a single $\varphi_i, i \in [m]$ variable.

In general, the proposed propagation rule sets variables in the complementary slackness system (9) to their bounds (or makes an inequality constraint hold with equality) if such equality constraint is implied by a subset of (in)equalities which correspond to the blocks of variables over which we update the primal. We will now explain to which subsets of the dual constraints the corresponding primal variables in the updated blocks correspond in this case. Since we always update over all α variables in the primal which correspond to the dual constraint $x_j \leq 1$ (or $x_j = 1$ if it is active), these constraints will be a part of the subset. Next, primal variable φ_i corresponds to the dual constraint $A_{i:}^T x + y_i + z_i = b_i$, thus this constraint will also belong to the subset. Finally, all the dual variables (along with the information whether they are non-negative, non-positive or already set to zero) belong to the subset, too.

As there are m blocks of variables, each can be naturally identified with a single $i \in [m]$. Let us now choose a block corresponding to a fixed $i \in [m]$ and let φ be a feasible solution of (6). The subset of (in)equalities consists of constraints (9a)-(9g) and the constraint $A_{i:}^T x + y_i + z_i = b_i$. The propagation algorithm queries whether this subset of (in)equalities implies[6] $y_i = 0$ (resp. $z_i = 0$, $x_j = 0$ or $x_j = 1$ for some $j \in X^U(\varphi)$).

By [17, Theorem 2], if φ is an interior local minimum, the previously discussed subset of (in)equalities is feasible for each $i \in [m]$ and does not imply that any inequality in (9) should hold strictly. We state this result in the following proposition.

Proposition 2. *Let φ be an ILM of* (6) *and $i \in [m]$ be arbitrary. The system of inequalities and equalities given by* (9a)-(9g) *and $A_{i:}^T x + y_i + z_i = b_i$ is feasible and does not imply $x_j = 0$ (resp. $x_j = 1$) for any $j \in X^U(\varphi)$.*

Proof. Since (9) is the system given by complementary slackness and updating (6) along individual φ_i variables is in correspondence to updating (4) by blocks consisting of φ_i and all α variables, it follows from [17, Theorem 2] that no further propagation can be performed and no contradiction is detected since φ is an interior local minimum. □

Recall [42, §2.3] that a point $v \in P$ is a *vertex of a polyhedron P* if for any $v', v'' \in P$, $0 < \lambda < 1$, $v = \lambda v' + (1-\lambda)v''$ implies $v = v' = v''$. In other words, a vertex can not be a strict convex combination of two different points of the polyhedron. Furthermore, a polyhedron is said to be *integral* if each of its vertices has integer coordinates.

Proposition 3. *Let $c \in \{-1, 0, 1\}^n$ and $d \in \mathbb{Z}$. The polyhedron*[7]

$$\{\, x \in [0,1]^n \mid c^T x \leq d \,\} \tag{11}$$

is integral and its projection onto each $x_j, j \in [n]$ is either $\{0\}, \{1\}, [0,1]$, or $\emptyset$.

Proof. As the hypercube $[0,1]^n$ is integral, non-integral vertices can only appear by adding the constraint $c^T x \leq d$, i.e., any non-integral vertex v would satisfy $c^T v = d$. Thus, there would exist $i \in [n]$ such that $v_i \notin \mathbb{Z}$ and $c_i \neq 0$ (otherwise we could both increase or decrease coordinate v_i and the point would still belong to (11), which would mean that v is not a vertex). Since $c^T v = d$, $c \in \{-1, 0, 1\}^n$, and $d \in \mathbb{Z}$, there must exist $i' \in [n], i' \neq i$ such that $v_{i'} \notin \mathbb{Z}$ and $c_{i'} \neq 0$. Assume that $c_i = 1, c_{i'} = -1$ (the other cases are analogous). Point $w \in \mathbb{R}^n$ defined as

$$w_k = \begin{cases} v_k & \text{if } k \neq i, i' \\ v_k + \epsilon & \text{if } k \in \{i, i'\} \end{cases} \tag{12}$$

also belongs to (11) for some suitable $\epsilon \neq 0$ which is sufficiently small in absolute value and may be both positive and negative to satisfy $x \in [0,1]^n$ due to $0 < v_i, v_{i'} < 1$. Point v is therefore not a vertex.

The property of projection is directly implied by integrality.[8] □

Proposition 4. *Let $n = 2$, $c \in \{-1, 0, 1\}^n$ and $d \in \mathbb{Z}$. If the projection of polyhedron* (11) *onto x_1 is the interval $[0,1]$ and the projection onto x_2 is also $[0,1]$, then $x = (\frac{1}{2}, \frac{1}{2})$ is feasible for* (11).

[6] In general, we say that a set of linear inequalities and equalities (in variables $x \in \mathbb{R}^n$) $Ax \leq b, A'x = b'$ implies $c^T x = d$ if $c^T x = d$ holds for all x satisfying $Ax \leq b, A'x = b'$. Symbols A, b, A', b', x are different than above.

[7] The proposition holds even if the constraint $c^T x \leq d$ is replaced by $c^T x = d$ or $c^T x \geq d$.

[8] Proof of integrality could also be done analogously to Theorem 45 in [30].

Proof. Since polyhedron (11) is bounded, it equals to the convex hull of its vertices [42, §2.3]. By Proposition 3, (11) is integral and due to $n = 2$, its vertices are always a subset of $\{(0,0),(1,0),(0,1),(1,1)\}$. This leaves 16 options for the choice of vertices based on exhaustive enumeration, from which only 7 options satisfy the assumptions on projections. For each of these 7 options, $x = (\frac{1}{2}, \frac{1}{2})$ is feasible. □

Proposition 5. *Let $b'_i = b_i - \sum_{j \in X^1(\varphi)} A_{ij}$ for $i \in [m]$ and let $A' \in \mathbb{R}^{m \times X^U(\varphi)}$ be a matrix created from A by removing columns $j \in X^0(\varphi) \cup X^1(\varphi)$. Projection of the polyhedron defined by* (9) *onto variables $x_j, j \in X^U(\varphi)$ (i.e., eliminating all y and z variables and all decided x variables) is*

$$x_j \in [0,1] \qquad \forall j \in X^U(\varphi) \tag{13a}$$
$$A'^T_{i:} x = b'_i \qquad \forall i \in Y^0(\varphi) \cap Z^0(\varphi) \tag{13b}$$
$$A'^T_{i:} x \le b'_i \qquad \forall i \in Y^+(\varphi) \cap Z^0(\varphi) \tag{13c}$$
$$A'^T_{i:} x \ge b'_i \qquad \forall i \in Y^0(\varphi) \cap Z^-(\varphi). \tag{13d}$$

Proof. By directly substituting known values of x_j by (9b) and (9c) and similarly eliminating variables y, z, we obtain (13). The case with $i \in Y^+(\varphi) \cap Z^-(\varphi)$ can be omitted because it corresponds to $A'^T_{i:} x \in \mathbb{R}$, which is always satisfied. □

Proposition 6. *Let A be a 2-in-row matrix and A' be the matrix considered in Proposition 5. Let us denote the set of columns of A' that correspond to the columns of the diagonal matrix D in Definition 2 by C, $C \subseteq X^U(\varphi)$. The projection of* (13) *onto all x_k, $k \in X^U(\varphi) - C$ (i.e., eliminating all $x_k, k \in C$) reads*

$$x_j \in [0,1] \qquad \forall j \in X^U(\varphi) - C \tag{14a}$$
$$A''^T_{i:} x \ge b''_i - 1 \qquad \forall i \in Y^0(\varphi) \cap Z^0(\varphi) \tag{14b}$$
$$A''^T_{i:} x \le b''_i \qquad \forall i \in Y^0(\varphi) \cap Z^0(\varphi) \tag{14c}$$
$$A''^T_{i:} x \le b''_i \qquad \forall i \in Y^+(\varphi) \cap Z^0(\varphi) \tag{14d}$$
$$A''^T_{i:} x \ge b''_i - 1 \qquad \forall i \in Y^0(\varphi) \cap Z^-(\varphi) \tag{14e}$$

where A'' is matrix A' from (13) *without columns in C, $b'' \in \mathbb{Z}^m$ is a vector, and each constraint* (14b)-(14e) *contains at most 2 variables.*

Proof. We can apply Fourier-Motzkin elimination [41] to eliminate all variables $x_l, l \in C$. By Definition 2, diagonality of D assures that x_l occurs in 3 constraints, namely: $x_l \ge 0, x_l \le 1$, and one of constraints

$$A''^T_{i:} x + s \cdot x_l = b'_i \tag{15a}$$
$$A''^T_{i:} x + s \cdot x_l \le b'_i \tag{15b}$$
$$A''^T_{i:} x + s \cdot x_l \ge b'_i \tag{15c}$$

where $s = A_{il} \in \{-1, 1\}$ and $x \in [0,1]^{X^U(\varphi) - C}$.

Suppose that the constraint is (15a) and $s = 1$. Then the elimination step creates inequalities (14b) and (14c) with $b''_i = b'_i$. If $s = -1$, then it creates the same inequalities except that $b''_i = b'_i + 1$.

If the constraint is (15b) (resp. (15c)), the elimination creates inequality (14d) (resp. (14e)) with $b''_i = b'_i + [\![s = -1]\!]$.[9]

The fact that each of the constraints (14) contains at most 2 variables follows from Definition 2. □

[9] The symbol $[\![\cdot]\!]$ denotes the Iverson bracket.

*Proof (**Theorem 1 for 2-in-row property**).* Let φ be an ILM of (6). If A is 2-in-row then it follows from Proposition 2, Proposition 3, and Proposition 5 that the projection of polyhedron defined by constraints (13a) and a single constraint from (13b)-(13d) onto any $x_k, k \in X^U(\varphi)$ is $[0,1]$. The same holds for the projection of polyhedron defined by constraints (14a) and a single constraint from (14b)-(14e) onto any $x_k, k \in X^U(\varphi) - C$ by Proposition 6. By Proposition 4 applied on system (14), setting $x_j = \frac{1}{2}$ for all $j \in X^U(\varphi) - C$ satisfies all constraints in (14). Therefore, (14) is feasible, hence (13) and (9) are also feasible by Proposition 6 and Proposition 5. Since (9) is feasible, φ is optimal by Proposition 1. □

For the second part of Theorem 1, we need to recall the notion of a *constraint satisfaction problem* (CSP) with constraints of higher arity and also the fact that if the *factor graph* (a.k.a. dual graph) of a CSP is a tree and the CSP is *generalized arc consistent* (GAC, a.k.a. hyper-arc consistent) without any domain-wipeout, then this CSP has a solution. The simpler case of CSPs with binary constraints is treated in [22]. For a formal introduction to CSPs with higher order and GAC, we refer the interested reader to, e.g. [12]. We are going to review the necessary concepts only informally for our specific case here.

Seeing (13b)-(13d) as constraints over discrete variables $x_j \in \{0,1\}, j \in X^U(\varphi)$ corresponds to a Boolean CSP. We say that this CSP is GAC if for any variable $x_j, j \in X^U(\varphi)$, any value $v \in \{0,1\}$, and any single constraint from (13b)-(13d) where x_j occurs, there exists an assignment for all variables involved in this constraint such that the chosen constraint is satisfied by this assignment and $x_j = v$.

Proposition 7. *Let φ be an ILM of* (6) *where $A \in \{-1,0,1\}^{m\times n}$ and $b \in \mathbb{Z}^m$. Let us consider a CSP with discrete variables $x_j \in \{0,1\}$, $j \in X^U(\varphi)$ and constraints* (13b)-(13d)*. This CSP is GAC.*

Proof. Proof by contradiction. Let us assume that there exists $i \in Y^+(\varphi) \cap Z^0(\varphi)$ and $k \in X^U(\varphi)$ such that for $x_k = 0$, there do not exist values for the remaining variables that would satisfy the constraint $A'^T_{i:} x \le b'_i$.[10] In other words,

$$A'^T_{i:} x \le b'_i \tag{16a}$$
$$x_j \in \{0,1\} \qquad \forall j \in X^U(\varphi) \tag{16b}$$

implies $x_k \neq 0$. Since $A'_{ij} \in \{-1,0,1\}$ and $b'_i \in \mathbb{Z}$, by Proposition 3,

$$A'^T_{i:} x \le b'_i \tag{17a}$$
$$x_j \in [0,1] \qquad \forall j \in X^U(\varphi) \tag{17b}$$

is not satisfiable with $x_j = 0$ (otherwise, it would have an integral solution). By Proposition 3, (17) is satisfiable only with $x_k = 1$ or not satisfiable at all.

Taking Proposition 5 into account, if the only possible value for x_j is 1, then (17) implies $x_j = 1$, which is contradictory with Proposition 2. If (17) is not satisfiable, then this is contradictory again with Proposition 2. □

Let us informally recall that a *factor graph of a CSP* is a bipartite graph with nodes corresponding to variables and constraints. The graph contains an edge between a variable node and a constraint node if the constraint involves the variable (in other words, if the variable is in the scope of the constraint).

We will now focus on the special structure of constraints (13b)-(13d) in case when A is bipartite-acyclic. Since A' is a submatrix of A, it follows that A' is bipartite-acyclic because A' defines a bipartite node-induced subgraph of the bipartite graph defined by A in the sense of Definition 1. In terms of constraint programming, this subgraph corresponds to the factor graph of the CSP considered in Proposition 7. Therefore, the factor graph is a forest, i.e., a disjoint collection of trees.

To explain the underlying idea behind the fact that GAC is a sufficient condition for satisfiability of a CSP if its factor graph is a tree[11], we can arbitrarily choose a single variable x_j,

[10] The reasoning for the other cases, i.e., $i \in Y^0(\varphi) \cap Z^0(\varphi)$ or $i \in Y^0(\varphi) \cap Z^-(\varphi)$ or $x_k = 1$ is analogous.

[11] If the factor graph is a forest, then each tree subproblem in the forest can be solved independently.

declare it to be the root of the tree, and set it an arbitrary allowed value, e.g., $x_j = 0$. By the property of GAC and acyclicity of the factor graph, there exists a joint assignment for all variables which occur in any constraint together with x_j. We can fix these variables to such assignment and remove the variable x_j along with all constraints where it is involved from the CSP. This decomposes the factor graph into a forest where each of the trees has only a single variable with assigned value due to acyclicity. Seeing that this is the same situation as before, i.e., a tree with a single decided root variable, we can repeat the previously outlined iteration recursively on each newly created tree separately. Eventually, each variable is assigned some value so that all the constraints in the CSP are satisfied.

*Proof (**Theorem 1 for bipartite-acyclic property**).* Let φ be an ILM of (6). If A is bipartite-acyclic, by Proposition 7, the CSP composed of variables $x_j \in \{0,1\}, j \in X^U(\varphi)$ and constraints (13b)-(13d) is GAC. By the assumption on A to be bipartite-acyclic, the factor graph of this CSP is acyclic, hence generalized arc consistency is a sufficient condition for existence of a solution [12]. This implies feasibility of (13) and by Proposition 5 implies feasibility of (9), thus φ is optimal by Proposition 1. □

Let us remark that the theorem in [16] allowed elements of vector b to be slightly more general in case of 2-in-row matrices, allowing also certain real values which are large in absolute value, e.g. $b_1 = \frac{9}{2}$. Such value causes the objective (6a) restricted to φ_1 to be strictly increasing, as discussed in the proof in [16], hence $\varphi_1 = \underline{\varphi}_1$ holds[12] in any ILM (and also in any global minimum) and its value does not change by coordinate-wise updates. For this particular case, since $\varphi_1 = \underline{\varphi}_1$ in any ILM, the corresponding dual variable y_1 is always allowed to be non-negative in (9). Thus, if we consider an inequality (4f) in the form, e.g.,

$$x_1 - x_2 + y_1 + z_1 = \frac{9}{2}, \tag{18}$$

then it can be satisfied by setting $y_1 = \frac{9}{2} - x_1 + x_2 - z_1$, which is always non-negative due to $x_1 \le 1$, $x_2 \ge 0$, and $z_1 \le 0$. Thus, such dual constraint is always satisfiable. Performing a simple analysis of cases on the number and sign of x variables in a single dual constraint (4f) would allow to additionally extend the range of possible values for b to certain real numbers, as it was done in [16].

4 Linear Programs with 2-in-row Constraint Matrix

Here we show that several LP relaxations of combinatorial problems correspond to the primal or dual (4) and discuss which additional constraints correspond to the assumptions of Theorem 1.

4.1 Weighted Partial Max-SAT

In weighted partial Max-SAT [40], one is given two sets of clauses, soft and hard. Each soft clause is assigned a positive weight. The task is to find values of binary variables $x_i \in \{0,1\}$, $i \in [p]$ such that all the hard clauses are satisfied and the sum of weights of the satisfied soft clauses is maximized.

We organize the m soft clauses into a matrix $S \in \{-1,0,1\}^{m \times p}$ defined as

$$S_{ci} = \begin{cases} 1 & \text{if literal } x_i \text{ is present in soft clause } c \\ -1 & \text{if literal } \neg x_i \text{ is present in soft clause } c \\ 0 & \text{otherwise} \end{cases}.$$

In addition, we denote $n^S_c = \sum_i [\![S_{ci} < 0]\!]$ to be the number of negated variables in clause c. These numbers are stacked in a vector $n^S \in \mathbb{Z}^m$. The h hard clauses are organized in a matrix $H \in \{-1,0,1\}^{h \times p}$ and a vector $n^H \in \mathbb{Z}^h$ in the same manner.

[12] If $\underline{\varphi}_1 = -\infty$ and $b_1 = \frac{9}{2}$, then the primal (4) is unbounded, dual (4) is infeasible, and (6) has no interior local minima.

The LP relaxation of this problem [40] reads

$$\max w^T s \tag{19a}$$
$$s_c \le S_{c:}^T x + n_c^S \qquad \forall c \in [m] \tag{19b}$$
$$H_{c:}^T x + n_c^H \ge 1 \qquad \forall c \in [h] \tag{19c}$$
$$x_i \in [0,1] \qquad \forall i \in [p] \tag{19d}$$
$$s_c \in [0,1] \qquad \forall c \in [m] \tag{19e}$$

where $w \in \mathbb{R}_+^m$ are the weights of the soft clauses. This is a sub-class of the dual (4) because the variables are box-constrained between 0 and 1 and inequality constraints $\le$ can be created by setting $\underline{\varphi}_i = 0$ and $\overline{\varphi}_i = \infty$, i.e., $y \ge 0$ become slack variables for the dual constraint (4f) that correspond to (19b) and z_i is forced to 0. The situation for inequality constraints $\ge$ in (19c) is analogous.

Formulation (19) satisfies the conditions of Theorem 1 for 2-in-row matrix if each of the clauses has length at most 2. In other words, optimality is guaranteed for weighted partial Max-2SAT. Also notice that if we omitted the soft clauses (19b) and instead set $v = -1$, we would obtain an instance of Min-Ones SAT, which could be generalized to weighted Min-Ones SAT. This relaxation still satisfies the requirements of Theorem 1 if all the present clauses have length at most 2. The part of Theorem 1 concerning bipartite-acyclic matrices applies to instances where the clause-variable incidence graph is acyclic and which are therefore tractable.

Let us remark that due to the result in [17], the fixed points of the constraint-propagation based algorithm in [15, §4] have the same quality as fixed points of coordinate-wise optimization with relative interior rule on formulation (6). Therefore, all interior local minima for weighted Max-SAT are global minima for instances with tractable language types and acyclic structure [15, §4.4]. Additionally, the LP relaxation is tight in these cases, hence the objective in these minima even coincides with the optimal value of the original non-relaxed problem.

4.1.1 Results

We tested the method on 800 smallest[13] instances that appeared in Max-SAT Evaluations [2] in years 2017 [1] and 2018 [3]. The results on the instances are divided into groups in Table 1 based on the minimal and maximal length of present clauses. We also evaluated this approach on 60 instances of weighted Max-2SAT from Ke Xu [56]. The highest number of logical variables in an instance was 19034 and the highest overall number of clauses in an instance was 31450. It was important to separate the instances without unit clauses (i.e. clauses of length 1) because in such cases the LP relaxation (19) has a trivial optimal solution given by $x_i = \frac{1}{2}$ for all $i \in V$.

Coordinate-wise minimization was stopped when the objective did not improve by at least $\epsilon = 10^{-7}$ after a whole cycle of updates for all variables. We report the quality of the solution as the median and mean relative difference between the optimal value and the objective reached by coordinate-wise minimization before termination.

Table 1 reports not only instances of weighted partial Max-2SAT but also instances with longer clauses, where optimality is no longer guaranteed. Nevertheless, the relative differences on instances with longer clauses still seem not too large and could be usable as bounds in a branch-and-bound scheme.

We remark that the dual formulation (6) for the specific case of weighted partial Max-SAT together with the form of the coordinate-wise updates was given in [14].

[13] Smallest in the sense of the file size. All instances could not have been evaluated due to their size and lengthy evaluation.

Instance Group Specification			Results	
Min CL	Max CL	#inst.	Mean RD	Median RD
≥ 2	any	91	0	0
1	2	123	$1.44 \cdot 10^{-9}$	$1.09 \cdot 10^{-11}$
1	3	99	$6.98 \cdot 10^{-3}$	$1.90 \cdot 10^{-7}$
1	≥ 4	487	$1.26 \cdot 10^{-2}$	$2.97 \cdot 10^{-3}$
1	2	60	$1.59 \cdot 10^{-9}$	$5.34 \cdot 10^{-10}$

Table 1: Experimental comparison of coordinate-wise minimization and exact solutions for LP relaxation on instances from [2] (first 4 rows) and [56] (last row).

4.2 Weighted Vertex Cover

Dual (4) also subsumes[14] the LP relaxation of weighted vertex cover [51, §14.3], which reads

$$\min v^T x \tag{20a}$$
$$x_i + x_j \geq 1 \qquad \forall \{i,j\} \in E \tag{20b}$$
$$x_i \in [0,1] \qquad \forall i \in V \tag{20c}$$

where V is the set of nodes and E is the set of edges of an undirected graph and $v_i \in \mathbb{R}_+$, $i \in V$ are vertex weights. This problem also satisfies the conditions of Theorem 1 for 2-in-row matrices and therefore the corresponding primal (6) will have no non-optimal interior local minima.

On the other hand, notice that formulation (20), which corresponds to dual (4) can have non-optimal interior local minima even with respect to all subsets of variables of size $|V| - 1$.

Example 4. *Consider the graph $K_{1,n}$, i.e., a complete bipartite graph with one node x_1 on one side with weight equal to $n - \frac{1}{2}$ and n nodes $x_2, ..., x_{n+1}$ on the other side, which have weights equal to 1. Then, $x = (0, 1, 1, 1, ..., 1)$ is an interior local minimum of (20) with respect to all blocks of variables of size n. Let us now consider such block $B \subset V$.*

- *If $1 \notin B$ (i.e., the variable x_1 is not in block), no update is possible due to constraints (20b).*
- *If $1 \in B$, then we could update all x_i for $i \in B, i \neq 1$ to $x_i - \epsilon$ and x_1 to $x_1 + \epsilon$ for $\epsilon \in [0,1]$ and the values of variables will remain feasible[15]. This update would change the objective by $\epsilon(n - \frac{1}{2}) - \epsilon(n-1) = \frac{1}{2}\epsilon \geq 0$ and since we are minimizing, the optimal update is for $\epsilon = 0$, i.e., keep the current x unchanged.*

Thus, x is an interior local minimum w.r.t. all blocks of variables of size $n = |V| - 1$, but has worse objective than the global optimizer $x^* = (1, 0, 0, ..., 0)$. ■

Formulation (6) for this case was given already in [53] where optimality was not proven yet, but was only experimentally observed. It reads

$$\max \sum_{\{i,j\} \in E} \varphi_{ij} + \sum_{i \in V} \min \Big\{ v_i - \sum_{j \in N_i} \varphi_{ij}, 0 \Big\} \tag{21a}$$
$$\varphi_{ij} \geq 0 \quad \forall \{i,j\} \in E \tag{21b}$$

where N_i denotes the set of neighbors of vertex i in graph (V, E). It is easy to see that Theorem 1 guarantees optimality of all interior local maxima[16] of (21). A closed-form solution of coordinate-wise updates for (21) satisfying relative interior rule was given in [53] as

$$\varphi_{ij} := \tfrac{1}{2} \max \Big\{ v_i - \textstyle\sum_{k \in N_i - \{j\}} \varphi_{ik}, 0 \Big\} + \tfrac{1}{2} \max \Big\{ v_j - \textstyle\sum_{k \in N_j - \{i\}} \varphi_{jk}, 0 \Big\}. \tag{22}$$

[14] It is only necessary to transform minimization to maximization by inverting sign in the objective (20a).

[15] We could also increase x_1 to $x_1 + \epsilon + \delta$ for $\delta \in [0, \epsilon - 1]$, but such update would never be optimal for any $\delta > 0$.

[16] The relation between interior local maxima and minima can be trivially obtained by changing maximization to minimization while inverting the sign of the objective.

Instance Group or Instance		Results	
Name	#inst.	Mean RD	Median RD
BVZ-tsukuba [7]	16	$6.03 \cdot 10^{-10}$	$1.17 \cdot 10^{-11}$
BVZ-sawtooth [45] [7]	20	$9.83 \cdot 10^{-11}$	$6.11 \cdot 10^{-12}$
BVZ-venus [45] [7]	22	$3.40 \cdot 10^{-11}$	$2.11 \cdot 10^{-12}$
KZ2-tsukuba [35]	16	$2.69 \cdot 10^{-10}$	$1.77 \cdot 10^{-10}$
KZ2-sawtooth [45] [35]	20	$4.08 \cdot 10^{-9}$	$1.56 \cdot 10^{-10}$
KZ2-venus [45] [35]	22	$5.21 \cdot 10^{-9}$	$1.74 \cdot 10^{-10}$
BL06-camel-sml [39]	1	$1.21 \cdot 10^{-11}$	
BL06-gargoyle-sml [6]	1	$6.29 \cdot 10^{-12}$	
LB07-bunny-sml [38]	1	$1.33 \cdot 10^{-10}$	

Table 2: Experimental comparison of coordinate-wise minimization on max-flow instances, the references are the original sources of the data and/or to the authors that reformulated these problems as maximum flow. The first 6 rows correspond to stereo problems, the 2 following rows are multiview reconstruction instances, the last row is a shape fitting problem.

4.3 Minimum st-Cut, Maximum Flow

Recall from [23] the usual formulation of max-flow problem between nodes $s \in V$ and $t \in V$ on a directed graph with vertex set V, edge set E and positive edge weights $c_{ij} \in \mathbb{R}_+$ for each $(i,j) \in E$, which reads

$$\max \sum_{(s,i)\in E} f_{si} \tag{23a}$$

$$0 \le f_{ij} \le c_{ij} \qquad \forall (i,j) \in E \tag{23b}$$

$$\sum_{(u,i)\in E} f_{ui} = \sum_{(j,u)\in E} f_{ju} \qquad \forall u \in V - \{s,t\}. \tag{23c}$$

Assume that there is no edge (s,t), there are no ingoing edges to s and no outgoing edges from t, then any value of f feasible for (23) is an interior local optimum w.r.t. individual coordinates by the same reasoning as in Example 2 due to the flow conservation constraint (23c), which limits each individual variable to a single value. In [16], we proposed a formulation which has no non-globally optimal interior local optima, namely

$$\min_{f\ge 0} \sum_{(i,j)\in E} \max\{c_{ij} - f_{ij}, 0\} + \sum_{\substack{(i,j)\in E\\ i\neq s}} f_{ij} + \sum_{i\in V-\{s,t\}} \max\Big\{\sum_{(j,i)\in E} f_{ji} - \sum_{(i,j)\in E} f_{ij}, 0\Big\}. \tag{24}$$

The formulation (24) however required recalculation of the optimum in the sense that if the optimal value of (24) equals O, then the optimal value of the original maximum flow problem equals $\sum_{(i,j)\in E} c_{ij} - O$.

We tested applying coordinate-wise optimization on formulation (24) on max-flow instances[17] from computer vision. The instances correspond to stereo problems, multiview reconstruction instances and shape fitting problems. We report the same statistics as with Max-SAT in Table 2.

For multiview reconstruction and shape fitting, we were able to run our algorithm only on small instances, which have approximately between $8 \cdot 10^5$ and $1.2 \cdot 10^6$ nodes and between $5 \cdot 10^6$ and $6 \cdot 10^6$ edges. On these instances, the algorithm terminated with the reported precision in 13 to 34 minutes on a laptop.

4.3.1 Simplified Formulation

As an improvement to formulation (24), we propose a simpler formulation and present a closed-form solution for coordinate-wise updates satisfying the relative interior rule which has a natural

[17] Available at https://vision.cs.uwaterloo.ca/data/maxflow.

interpretation. In particular, the maximum flow problem can be formulated as

$$\max \sum_{(s,j)\in E} f_{sj} + \sum_{i\in V-\{s,t\}} \min\{R_i, 0\} \tag{25a}$$

$$0 \le f_{ij} \le c_{ij} \quad \forall (i,j) \in E \tag{25b}$$

where $R_i = \sum_{(i,j)\in E} f_{ij} - \sum_{(j,i)\in E} f_{ji}$ for $i \in V - \{s,t\}$. This formulation does not switch maximization to minimization so that no re-calculation of the optimal value is necessary. Optimization problem (25) was obtained by adding bounds to the node variables in the corresponding dual which reads

$$\min \sum_{(i,j)\in E} c_{ij} y_{ij} \tag{26a}$$

$$y_{ij} \ge x_i - x_j \qquad \forall (i,j) \in E, i \ne s, j \ne t \tag{26b}$$

$$y_{sj} \ge 1 - x_j \qquad \forall (s,j) \in E \tag{26c}$$

$$y_{it} \ge x_i \qquad \forall (i,t) \in E \tag{26d}$$

$$y_{ij} \ge 0 \qquad \forall (i,j) \in E, \tag{26e}$$

$$x_i \in [0,1] \qquad \forall i \in V - \{s,t\}, \tag{26f}$$

and corresponds to minimum st-cut problem. In detail, if $y_{ij} = 1$, then edge (i,j) is in the cut and if $y_{ij} = 0$, then edge (i,j) is not in the cut. The cut should separate s and t, so the set of nodes connected to s after the cut will be denoted by S, and $T = V - S$ is the set of nodes connected to t. Using this notation, $x_i = [\![i \in S]\!]$. Formulation (26) is different from the classical formulation by bounding the x variables (which are usually unbounded, i.e., $x_i \in \mathbb{R}$).

Clearly, each constraint (26) contains at most two x variables and each y_{ij} variable occurs in only one constraint. Thus, Theorem 1 applies to (25) since the constraint matrix is 2-in-row.

Let us now focus on the details of coordinate-wise optimization of (25). The form of the update for a particular variable depends on whether the corresponding edge is incident[18] to s, t, or none of these special nodes. The update for variable f_{ij}, $(i,j) \in E$ reads

$$f_{sj} := \tfrac{1}{2}\left(c_{sj} + h_{[0,c_{sj}]}(R_j + f_{sj})\right) \qquad \text{if } i = s \tag{27a}$$

$$f_{it} := \tfrac{1}{2}\left(c_{it} + h_{[0,c_{it}]}(f_{it} - R_i)\right) \qquad \text{if } j = t \tag{27b}$$

$$f_{ij} := \tfrac{1}{2}\left(h_{[0,c_{ij}]}(f_{ij} - R_i) + h_{[0,c_{ij}]}(R_j + f_{ij})\right) \qquad \text{if } i \ne s, j \ne t \tag{27c}$$

where $h_{[0,c_{ij}]}$ denotes projection onto $[0, c_{ij}]$, as defined in §3.1. The derivation of the updates is based on applying the procedure described in §3.1 on problem (25).

There exists an informal natural interpretation for these updates. If the constraints $R_i = 0$ for all $i \in V - \{s,t\}$ are added to (25), we will obtain the classical flow conservation constraints and the usual LP formulation (23) where coordinate-wise updates are impossible.

For now, consider an edge $(i,j) \in E, i \ne s, j \ne t$. If $f_{ij} = R_j + f_{ij}$, then $R_j = 0$ and node j satisfies flow conservation. The same holds for node i if $f_{ij} = f_{ij} - R_i$. Now, given some value f_{ij}, we may want to change it so that $|R_j|$ is minimal (i.e., the flow is conserved in j as much as possible) while satisfying $0 \le f_{ij} \le c_{ij}$. This is achieved by setting $f_{ij} := h_{[0,c_{ij}]}(R_j + f_{ij})$. On the other hand, if we wanted to minimize $|R_i|$ by changing f_{ij} while satisfying the lower and upper bounds on flow (25b), we would set $f_{ij} := h_{[0,c_{ij}]}(f_{ij} - R_i)$. Observe that update (27c) takes the average between the mentioned values, hence it can be informally said that f_{ij} is updated so that it 'balances' the flow in both nodes i and j. For instance, if there is 'overpressure' in node i ($R_i < 0$) and 'underpressure' in node j ($R_j > 0$), the value of f_{ij} increases (if allowed by constraint $f_{ij} \le c_{ij}$) in order to increase R_i and decrease R_j. The case of $R_i > 0$ and $R_j < 0$ is symmetric and results in decreasing the value of f_{ij} if possible.

Updates (27a) also have an informal interpretation in this model, consider an edge that leads out of source, i.e., $(s,j) \in E$. Similarly as before, if we want to minimize $|R_j|$, we should

[18] Without loss of generality, we assume that $(s,t) \notin E$ and that there are no outgoing edges from t and no ingoing edges to s.

update $f_{sj} := h_{[0,c_{sj}]}(R_j + f_{sj})$, but we also want to maximize the flow coming out of s, i.e., set f_{sj} as large as possible, ideally $f_{sj} := c_{sj}$. Again, update (27a) takes the average between these values. Updates for edges that are connected to the terminal node t are clearly analogous since the terminal node would also like to obtain as much flow as possible.

All in all, performing updates (27a) for edges coming from source s can be interpreted as trying to push as much flow into the network as possible while the updates (27c) for the intermediate nodes try to propagate the flow throughout the network so that the flow could be consumed by edges connected to the terminal node t by updates (27b).

4.4 MAP Inference with Potts Potentials

Coordinate-wise minimization for the dual LP relaxation of MAP inference was intensively studied, see e.g. the review [52]. One of the formulations is

$$\min \sum_{i\in V} \max_{k\in K} \theta_i^\delta(k) + \sum_{\{i,j\}\in E} \max_{k,l\in K} \theta_{ij}^\delta(k,l) \tag{28a}$$

$$\delta_{ij}(k) \in \mathbb{R} \quad \forall \{i,j\} \in E, k \in K, \tag{28b}$$

where K is the set of labels, V is the set of nodes and E is the set of unoriented edges and

$$\theta_i^\delta(k) = \theta_i(k) - \sum_{j\in N_i} \delta_{ij}(k) \tag{29a}$$

$$\theta_{ij}^\delta(k,l) = \theta_{ij}(k,l) + \delta_{ij}(k) + \delta_{ji}(l) \tag{29b}$$

are equivalent transformations of the potentials θ. Notice that there are $2 \cdot |E| \cdot |K|$ variables, i.e., two for each direction of an edge. In [43], it is mentioned that in case of Potts interactions, which are given as $\theta_{ij}(k,l) = -[\![k \neq l]\!]$, one can add constraints

$$\delta_{ij}(k) + \delta_{ji}(k) = 0 \qquad \forall \{i,j\} \in E, k \in K \tag{30a}$$

$$-\tfrac{1}{2} \leq \delta_{ij}(k) \leq \tfrac{1}{2} \qquad \forall \{i,j\} \in E, k \in K \tag{30b}$$

to (28) without changing the optimal objective. One can therefore use constraint (30a) to reduce the overall number of variables by defining

$$\lambda_{ij}(k) = -\delta_{ij}(k) = \delta_{ji}(k) \tag{31}$$

subject to $-\frac{1}{2} \leq \lambda_{ij}(k) \leq \frac{1}{2}$. The decision of whether $\delta_{ij}(k)$ or $\delta_{ji}(k)$ should have inverted sign depends on the chosen orientation E' of the originally undirected edges E and is arbitrary. Also, given values δ satisfying (30), it holds for any edge $\{i,j\} \in E$ and pair of labels $k,l \in K$ that $\max_{k,l\in K} \theta_{ij}^\delta(k,l) = 0$, which can be seen from the properties of the Potts interactions.

Therefore, one can reformulate (28) into

$$\min \sum_{i\in V} \max_{k\in K} \theta_i^\lambda(k) \tag{32a}$$

$$-\tfrac{1}{2} \leq \lambda_{ij}(k) \leq \tfrac{1}{2} \quad \forall (i,j) \in E', k \in K, \tag{32b}$$

where the equivalent transformation in variables λ is given by

$$\theta_i^\lambda(k) = \theta_i(k) + \sum_{(i,j)\in E'} \lambda_{ij}(k) - \sum_{(j,i)\in E'} \lambda_{ji}(k) \tag{33}$$

and we optimize over $|E'| \cdot |K|$ variables λ, the graph (V, E') is the same as (V, E) except that each edge becomes oriented (in arbitrary direction). The way of obtaining an optimal solution to (28) from an optimal solution of (32) is given by (31) and depends on the chosen orientation of the edges in E'. Also observe that $\theta_i^\delta(k) = \theta_i^\lambda(k)$ for any node $i \in V$ and label $k \in K$ and therefore the optimal values will be equal. This reformulation therefore maps global optima of (32) to global optima of (28). However, it does not map interior local minima of (32) to interior local minima of (28) when $|K| \geq 3$.

Example 5. *Consider the case with $|K| = 3$ labels and a chain graph with 4 nodes. We can see the numerical example in Figure 1 where the active (maximal) labels in nodes are shown as black, inactive as white, and their transformed values are shown under them (the first number in the formula is their original value). The values of variables λ are on the corresponding edges and the orientation of the edges is from left to right. One can clearly see that each of the λ variables is in the relative interior of optimizers.*

If the variables λ are transformed into the general form of MAP inference (28) *with δ variables using* (31)*, we obtain the result in Figure 2. Clearly, the unary potentials $\theta_i^\delta(k)$ did not change by the transformation. The binary potentials $\theta_{ij}^\delta(k,l)$ have values $-3,-2,-1,0$, depending on the corresponding $\delta_{ij}(k)$ and $\delta_{ji}(l)$. One can observe that this setting of variables is not an interior local minimum because by reasoning from [53], the arc consistency closure of the maximal nodes and maximal edges in Figure 2 is empty and therefore we can decrease the objective by max-sum diffusion (i.e. by coordinate-wise updates into the relative interior).*

At the initial stage, $\delta_{12}(1)$ is not in the relative interior of optimizers, but on its boundary, we will update it. For the same reason, we will then sequentially update $\delta_{21}(1)$, $\delta_{23}(1)$, $\delta_{32}(1)$ and $\delta_{34}(1)$. Then, $\delta_{43}(1)$ would not be the optimal choice and its change would decrease the objective. ■

In problems with two labels ($K = \{1,2\}$), problem (32) is subsumed by (6) and satisfies the conditions imposed by Theorem 1 because one can rewrite the objective by observing that

$$\max_{k\in\{1,2\}} \theta_i^\lambda(k) = \max\{\theta_i^\lambda(1) - \theta_i^\lambda(2), 0\} + \theta_i^\lambda(2) \tag{34}$$

and each $\lambda_{ij}(k)$ is present only in $\theta_i^\lambda(k)$ and $\theta_j^\lambda(k)$. Thus, $\lambda_{ij}(k)$ has non-zero coefficient in the matrix A only in columns i and j. The coefficients of the variables in the objective are only $\{-1,0,1\}$ and the other conditions are straightforward.

We reported the experiments on the Potts problem in [53] where the optimality was not proven yet. A closed-form solution for coordinate-wise update of (32) satisfying relative interior rule was given as follows

$$\begin{aligned}\lambda_{ij}(k) :=& \tfrac{1}{2}h'\big(\max_{k'\in K-\{k\}} \theta_i^\lambda(k') - \theta_i^\lambda(k) + \lambda_{ij}(k)\big) - \\ & \tfrac{1}{2}h'\big(\max_{k'\in K-\{k\}} \theta_j^\lambda(k') - \theta_j^\lambda(k) - \lambda_{ij}(k)\big)\end{aligned} \tag{35}$$

where $h'(\cdot)$ stands for $h_{\left[-\frac{1}{2},\frac{1}{2}\right]}(\cdot)$, i.e., projection onto $\left[-\frac{1}{2},\frac{1}{2}\right]$.

4.5 Binarized Monotone Linear Programs

In [27], integer linear programs with at most two variables per constraint were discussed. It was also allowed to have 3 variables in some constraints if one of the variables occurred only in this constraint and in the objective function. Although the objective function in [27] was allowed to be more general, we will restrict ourselves to linear objective. It was also shown that such problems can be transformed into binarized monotone constraints over binary variables by introducing additional variables whose number is defined by the bounds of the original variables, such optimization problem reads

$$\min w^T x + e^T z \tag{36a}$$
$$Ax - Iz \le 0 \tag{36b}$$
$$Cx \le 0 \tag{36c}$$
$$x \in \{0,1\}^{n_1} \tag{36d}$$
$$z \in \{0,1\}^{n_2} \tag{36e}$$

where A, C contain exactly one -1 per row and exactly one 1 per row and all other entries are zero, I is the identity matrix. We refer the reader to [27] for details, where it is also explained

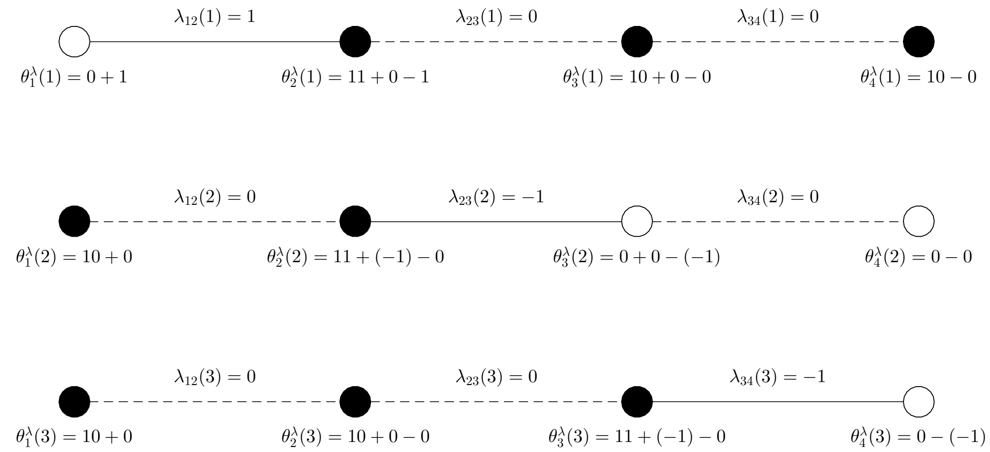

Figure 1: An example of an interior local minimum of the Potts problem, non-zero λ variables are denoted by solid lines and zero λ variables are denoted by dashed lines. All values should be halved, which was omitted for clarity and better reading.

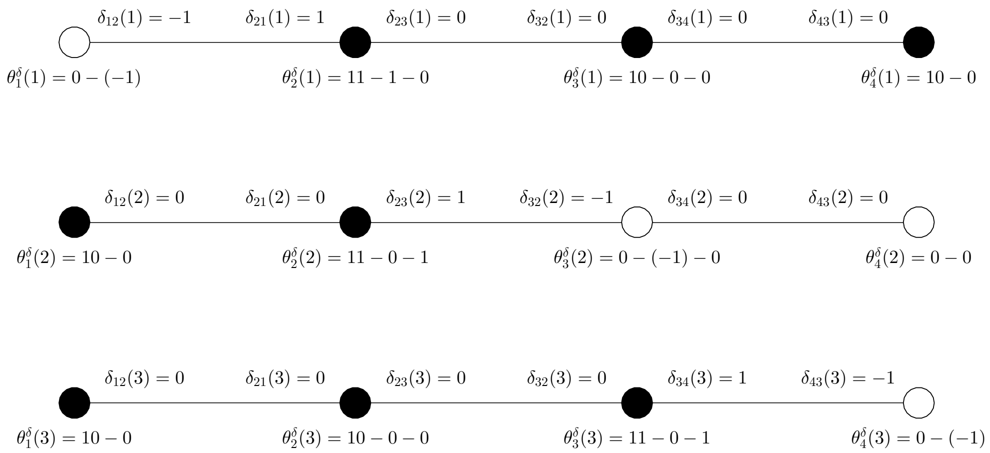

Figure 2: Corresponding problem in variables of max-sum diffusion, maximal values of $\theta_{ij}^\delta(k,l)$ are shown by solid lines in each edge, non-maximal values $\theta_{ij}^\delta(k,l)$ are not drawn. All values should be halved, which was omitted for clarity and better reading.

that the LP relaxation of (36) can be solved by min-st-cut on an associated graph. We can notice that the LP relaxation of (36) is subsumed by the dual (4) because one can change the minimization into maximization by changing the signs in w, e. Also, the relaxation satisfies the conditions given by Theorem 1.

In the paper [27], there are listed many problems which can be transformed to (36) and are also directly (without any complicated transformation) subsumed by the dual (4) and satisfy Theorem 1, for example, minimizing the sum of weighted completion times of precedence-constrained jobs (ISLO formulation in [11]), generalized independent set (forest harvesting problem in [28]), generalized vertex cover [29], clique problem [29], Min-SAT (introduced in [32], LP formulation in [27]).

For each of these problems, it is easy to verify the conditions of Theorem 1 as they contain at most two variables per constraint and if a constraint contains a third variable, then it is the only occurrence of this variable and the coefficients of the variables in the constraints are from the set $\{-1, 0, 1\}$.

The transformation presented in [27] can be applied to partial Max-SAT and vertex cover to obtain a problem in the form (36) and solve its LP relaxation. But this step is unnecessary when applying the presented coordinate-wise minimization approach.

Except for the previously mentioned problems, there also exists a suitable formulation of the roof-dual optimization problem [30, §7.2] which is subsumed by (6) and to which Theorem 1 applies. In detail, the LP relaxation of the pseudoboolean optimization problem is given as [30, equation (7.10)] whose dual is the roof-dual. Adding redundant bounds $w \leq 1$ into [30, equation (7.10)] results in a different form of the roof-dual which is amenable to coordinate-wise optimization in the sense of Theorem 1. This is easy to see since each constraint in [30, equation (7.10)] contains at most 2 variables and their coefficients are only $\{-1, 0, 1\}$. However, for practical purposes, there exist specialized algorithms for optimizing this LP relaxation as it can be seen as a bidirected flow problem [30, §7.2].

5 Linear Programs with 2-in-column Constraint Matrix

It follows from our constructive proof of Theorem 1, respectively already from the proof in [16] that if A satisfies the conditions of the theorem, then dual (4) is (half-)integral. Aside from the cases captured by Theorem 1, a question may arise whether it is possible to generalize the result for other matrices A which would yield a (half-)integral dual (4). For this purpose, we consider the notion of a 2-in-column matrix.

Definition 3. *Matrix $A \in \mathbb{R}^{m \times n}$ is called* 2-in-column *if it contains at most 2 non-zero elements in each column.*

Obviously, the transpose of a 2-in-column matrix is 2-in-row. Also, if A contains at most 1 non-zero entry in each column, then it is bipartite-acyclic by Definition 1. Thus, 2-in-column matrices are in a certain informal sense 'close' to the classes of matrices considered in Theorem 1. Despite the fact that Theorem 1 does not generalize in this manner, coordinate-wise optimization is capable of providing reasonable results for such problems.

In particular, we consider the assignment problem, shortest paths problem, and LP relaxation of maximum weight matching. We argue that even though the interior local optima of these problems may not be globally optimal in general, we experimentally show that coordinate-wise optimization frequently attains fixed points not far from global optima. For each of the considered problems, we provide a counter-example showing a non-optimal fixed point, a closed-form solution for updates satisfying the relative interior rule, and an overview of experimental results.

5.1 Assignment Problem

The assignment problem is defined by a number $n \in \mathbb{N}$ and costs $c_{ij} \in \mathbb{R}$ for each $i, j \in [n]$ and the task is to find a bijection $f\colon [n] \to [n]$ such that $\sum_{i \in [n]} c_{i,f(i)}$ is minimized. This problem can be formulated as a linear program

$$\min \sum_{i,j\in[n]} c_{ij}x_{ij} \tag{37a}$$
$$\sum_{i\in[n]} x_{ij} = 1 \qquad \forall j \in [n] \tag{37b}$$
$$\sum_{j\in[n]} x_{ij} = 1 \qquad \forall i \in [n] \tag{37c}$$
$$x_{ij} \in [0,1] \qquad \forall i,j \in [n] \tag{37d}$$

where the constraint matrix is totally unimodular [13] and thus, the optimal value of (37) equals to the optimal value of the original problem. Clearly, (37) corresponds to the dual[19] (4) with a 2-in-column matrix A as one can observe that each variable x_{ij} occurs only in a single constraint (37b) and a single constraint (37c). The dual LP to (37) reads

$$\max\ 1^T y + 1^T q + \sum_{i,j\in[n]} \min\{c_{ij} - y_j - q_i, 0\} \tag{38a}$$
$$y_j \in \mathbb{R} \qquad \forall j \in [n] \tag{38b}$$
$$q_i \in \mathbb{R} \qquad \forall i \in [n] \tag{38c}$$

and may have interior local optima which are not globally optimal.

Example 6. *Let $n = 5$ and $c_{ij} = [\![i \leq 3 \wedge j \leq 3]\!]$. Then, $y_1 = ... = y_5 = q_1 = ... = q_5 = 0$ is an interior local maximum of (38) with objective 0, but the optimal value is 1.* ■

An update for a single y_j, $j \in [n]$ satisfying the relative interior rule reads $y_j := (b_1 + b_2)/2$ where b_1 and b_2 are the two smallest[20] values among $c_{ij} - q_i$ for $i \in [n]$. The update is the same for $q_i, i \in [n]$ except that we consider the two smallest values among $c_{ij} - y_j$ for $j \in [n]$.

Even though the results are not guaranteed to be optimal, we applied coordinate-wise optimization with relative interior rule on the formulation (38). We considered the following types of randomly generated instances, based on [9, 13, 26]:

- Uniform: c_{ij} are chosen randomly uniformly from the set $[k]$ where $k \in \{10, 10^2, 10^3, 10^6\}$.
- Geometric: $c_{ij} = \lfloor \|r_i - r'_j\| \rfloor$ where $(r_i)_{i=1}^n, (r'_i)_{i=1}^n$ are randomly generated points in $[k]^2$, $k \in \{10, 10^2, 10^3, 10^6\}$ and $\|\cdot\|$ denotes the Euclidean norm.
- Two cost: $c_{ij} = 1$ with probability p, otherwise $c_{ij} = 10^6$, $p \in \{0.25, 0.5, 0.75\}$.
- Randomized Machol Wien: each c_{ij} is chosen randomly uniformly from the set $[i \cdot j]$.

For each type and each parameter setting, we compared the objective attained by coordinate-wise optimization[21] with the optimal value of the problem. We list the results in Table 3 and mention that among the 720 evaluated instances, coordinate-wise optimization was able to solve 570 of them to optimality (up to numerical precision), which includes all two cost instances, all randomized Machol Wien instances, majority of uniform instances, and less than half of geometric instances.

An unoptimized implementation of coordinate-wise optimization had on average 150 times shorter runtime when compared to a general purpose LP solver. However, there also exist specialized algorithms for solving assignment problem, such as the well-known Hungarian algorithm [37, 20].

[19] It is also necessary to switch between minimization and maximization, which can be achieved by using a cost vector with opposite sign of its components. The dual variables (y, z) in (4) are eliminated by setting all lower and upper bounds $\underline{\varphi}$, $\overline{\varphi}$ to $-\infty$ and ∞, respectively, hence the variables in (38) are unbounded.

[20] We allow $b_1 = b_2$ if there were two coinciding minima.

[21] Coordinate-wise updates were stopped when the objective did not improve at least by $\epsilon = 10^{-6}$ after a whole round of updates.

Instance Type	Results					
	$n=20$		$n=50$		$n=100$	
	Mean RD	Median RD	Mean RD	Median RD	Mean RD	Median RD
Uniform, $k=10$	$2.89\cdot10^{-7}$	$9.78\cdot10^{-8}$	$1.09\cdot10^{-8}$	$-9.31\cdot10^{-10}$	$-3.18\cdot10^{-9}$	$-1.65\cdot10^{-9}$
Uniform, $k=10^2$	$2.48\cdot10^{-9}$	$-4.96\cdot10^{-10}$	$7.08\cdot10^{-8}$	$8.08\cdot10^{-9}$	$2.63\cdot10^{-7}$	$1.35\cdot10^{-7}$
Uniform, $k=10^3$	$-1.90\cdot10^{-10}$	$-1.85\cdot10^{-10}$	$-4.68\cdot10^{-10}$	$-3.39\cdot10^{-10}$	$-7.86\cdot10^{-10}$	$-5.57\cdot10^{-10}$
Uniform, $k=10^6$	$-1.96\cdot10^{-9}$	$-1.29\cdot10^{-9}$	$-4.22\cdot10^{-10}$	$-9.22\cdot10^{-11}$	$-2.12\cdot10^{-11}$	$-2.65\cdot10^{-12}$
Geometric, $k=10$	$5.43\cdot10^{-2}$	$5.62\cdot10^{-2}$	$1.57\cdot10^{-1}$	$1.66\cdot10^{-1}$	$3.00\cdot10^{-1}$	$2.77\cdot10^{-1}$
Geometric, $k=10^2$	$3.07\cdot10^{-3}$	$4.15\cdot10^{-9}$	$1.57\cdot10^{-2}$	$6.52\cdot10^{-3}$	$3.66\cdot10^{-2}$	$3.37\cdot10^{-2}$
Geometric, $k=10^3$	$-4.13\cdot10^{-11}$	$-1.55\cdot10^{-11}$	$4.51\cdot10^{-4}$	$-8.33\cdot10^{-11}$	$4.28\cdot10^{-3}$	$3.44\cdot10^{-8}$
Geometric, $k=10^6$	$-2.30\cdot10^{-9}$	$-2.02\cdot10^{-9}$	$-1.63\cdot10^{-10}$	$-8.24\cdot10^{-11}$	$-6.97\cdot10^{-11}$	$-1.95\cdot10^{-11}$
Two cost, $p=0.25$	$-2.63\cdot10^{-4}$	$-7.29\cdot10^{-9}$	$-1.11\cdot10^{-5}$	$-3.35\cdot10^{-8}$	$-5.28\cdot10^{-6}$	$-7.64\cdot10^{-9}$
Two cost, $p=0.5$	$-3.16\cdot10^{-9}$	$-1.31\cdot10^{-9}$	$-6.27\cdot10^{-9}$	$-7.85\cdot10^{-10}$	$-4.96\cdot10^{-9}$	$-6.03\cdot10^{-9}$
Two cost, $p=0.75$	$-1.29\cdot10^{-9}$	$-1.13\cdot10^{-9}$	$-4.29\cdot10^{-9}$	$-2.99\cdot10^{-9}$	$-4.33\cdot10^{-10}$	$-4.35\cdot10^{-10}$
Rand. Machol Wien	$4.20\cdot10^{-10}$	$-8.61\cdot10^{-10}$	$-5.07\cdot10^{-10}$	$-6.90\cdot10^{-10}$	$-9.26\cdot10^{-10}$	$-6.82\cdot10^{-10}$

Table 3: Comparison of results on randomly generated instances of assignment problem. For each type and parameter setting, 20 instances were generated. The reason for negative entries is that coordinate-wise optimization found a solution with better objective than the LP solver, which is caused by numerical precision issues.

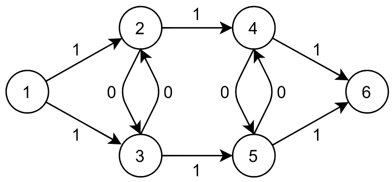

Figure 3: Directed graph with 6 nodes, 10 edges, $s = 1$, $t = 6$, and edge weights from the set $\{0, 1\}$ (indicated next to the edges). Dual feasible solution $y = (4, 3, 3, 1, 1, 0)$ is an interior local maximum of (40) with objective 2 but not a global optimum, which has value 3. The vertex indices are indicated by the numbers within the corresponding circles.

5.2 Shortest Paths

Let (V, E) be a directed graph with edge weights $c_{ij} \in \mathbb{R}_+$ for each $(i, j) \in E$, and let $s, t \in V$ be two distinguished nodes. Following [42], finding the length of the shortest path from s to t in this graph can be formulated as a linear program

$$\min \sum_{(i,j)\in E} c_{ij} f_{ij} \tag{39a}$$

$$\sum_{(i,j)\in E} f_{ij} - \sum_{(j,i)\in E} f_{ji} = \begin{cases} 1 & \text{if } i = s \\ -1 & \text{if } i = t \\ 0 & \text{otherwise} \end{cases} \qquad \forall i \in V \tag{39b}$$

$$f_{ij} \in [0, 1] \qquad \forall (i, j) \in E \tag{39c}$$

which is subsumed by the minimum-cost flow problem. Additionally, LP (39) is a subclass of the dual (4) with 2-in-column matrix A because each variable f_{ij} is present only in two constraints (39b), namely, for $i \in V$ and $j \in V$. The other properties are analogous to §5.1. The dual LP to (39) reads

$$\max\ y_s - y_t + \sum_{(i,j)\in E} \min\{c_{ij} - y_i + y_j, 0\} \tag{40a}$$

$$y_i \in \mathbb{R} \qquad \forall i \in V \tag{40b}$$

which is different from the dual in [42] as we are also upper bounding the flows by 1 in (39c).

An update satisfying relative interior rule for a single variable y_i, $i \in V$ is given by the following procedure:

1. Sort values $y_k - c_{ki}$ for $k \in N_i^-$ and $c_{ij} - y_j$ for $j \in N_i^+$ into a non-decreasing sequence $b_1, ..., b_{|N_i|}$ where N_i^- (and N_i^+) is the set of predecessors (and successors) of node i in (V, E), respectively. Thus, one can write $N_i = N_i^+ \cup N_i^-$.

2. Calculate $r = |N_i^-| + [\![i = s]\!] - [\![i = t]\!]$.

3. Update $y_i := \begin{cases} (b_r + b_{r+1})/2 & \text{if } 1 \le r \le |N_i| - 1 \\ b_1 - \Delta & \text{if } r = 0 \\ b_{|N_i|} + \Delta & \text{if } r = |N_i| \end{cases}$ where $\Delta > 0$ is a fixed constant, e.g., $\Delta = 1$.

The procedure can be derived using the general approach described in §3.1. We remark that if $N_s^+ = \emptyset$ or $N_t^- = \emptyset$, then (40) is unbounded and (39) infeasible. If $N_i = \emptyset$ for $i \in V - \{s, t\}$, then i is an isolated node and we can set y_i arbitrarily.

The formulation (40) may have non-optimal interior local maxima, example of such case is shown in Figure 3.

Nevertheless, we tested applicability of coordinate-wise optimization on (40) for randomly generated graphs. To construct the graph, we set $V = [n]$, randomly uniformly generated points $r_i, i \in V$ in hypercube $[0, 1]^q$, chose density $d \in [0, 1]$, and randomly added $\lfloor d \cdot |V| \cdot (|V| - 1) \rfloor$ edges to the graph with costs $c_{ij} = \|r_i - r_j\|$. We set $s = 1$ and $t = n$. If such graph did not contain an $s-t$ path, we added a random path from s to t with analogously created costs. We performed experiments with parameter settings $q \in \{3, 5, 10\}, |V| \in \{5, 10, 30, 100, 1000\}, d \in \{0.1, 0.5, 0.75\}$ and for each setting, 20 random graphs were generated. In all of the 900 generated instances, the relative difference was lower than $8.91 \cdot 10^{-9}$ and absolute difference lower than $5.50 \cdot 10^{-8}$. In 897 instances, the relative difference was below $3.36 \cdot 10^{-15}$ and absolute difference below $6.66 \cdot 10^{-15}$. Coordinate-wise optimization therefore solved these instances to optimality (up to numerical precision).

We also tested instances where the edge weights are generated randomly uniformly from the set $[k]$ for $k \in \{10, 10^3, 10^6\}$ and the random generation otherwise followed the process described above with 60 repetitions for each parameter setting. This resulted in 2700 instances, all of which were solved up to relative difference $4.73 \cdot 10^{-8}$ and absolute difference $4.56 \cdot 10^{-7}$. In particular, 2518 of these instances were solved up to relative difference $1.99 \cdot 10^{-16}$ and absolute difference $4.66 \cdot 10^{-10}$. Observe that the optimal value is always integral because the edge weights are integral. We remark that the rounded objective determined by coordinate-wise optimization exactly coincided with the global optimum in each case.

To obtain exact results, we used Dijkstra's algorithm whose unoptimized version was 8.3 times faster when compared with similarly unoptimized version of coordinate-wise optimization.

5.3 LP Relaxation of Maximum Weight Matching

Recall (from e.g. [18]) the maximum weight matching in an undirected graph (V, E) with edge weights $w_{ij} \in \mathbb{R}_+$, $\{i, j\} \in E$ which seeks to find a subset of edges $S \subseteq E$ such that no two adjacent edges are in S and $\sum_{\{i,j\} \in S} w_{ij}$ is maximized. There exist fast polynomial algorithms for solving this problem, such as [24]. The linear programming relaxation of this problem reads

$$\max \sum_{\{i,j\} \in E} w_{ij} x_{ij} \tag{41a}$$

$$\sum_{j \in N_i} x_{ij} \le 1 \qquad \forall i \in V \tag{41b}$$

$$x_{ij} \in [0, 1] \qquad \forall \{i, j\} \in E \tag{41c}$$

where N_i is the set of neighbors of $i \in V$ in the graph. Notice that this is just a relaxation as it does not incorporate the odd-set constraints, as opposed to the matching polytope [19] which

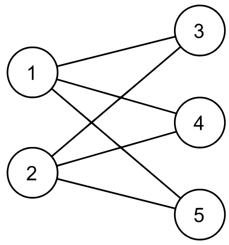

Figure 4: Complete bipartite graph $K_{2,3}$ with all edge weights equal to 1. For this instance of maximum weight matching, dual feasible point $y = (0, 0, 1, 1, 1)$ is an interior local minimum of (42) with objective 3 but the optimal value is 2.

is integral. However, in bipartite graphs, (41) is well-known to be integral and it is therefore not a relaxation [18].

Linear program (41) also corresponds to the dual (4) with 2-in-column matrix A by similar reasoning as in the previous cases discussed in §5.1 and §5.2 except that the bounds $\underline{\varphi}$ in the corresponding formulation of (4) are set to 0.

The dual of the LP relaxation (41) reads

$$\min\ 1^T y + \sum_{\{i,j\}\in E} \max\{w_{ij} - y_i - y_j, 0\} \tag{42a}$$

$$y_i \geq 0 \qquad \forall i \in V \tag{42b}$$

and may have interior local optima which are not global optima even if (V, E) is bipartite and unweighted. We provide an example in Figure 4.

An update for a single variable $y_i, i \in V$ for (42) satisfying the relative interior rule can be stated as

$$y_i := \begin{cases} (h_{[0,\infty)}(b_{|N_i|}) + h_{[0,\infty)}(b_{|N_i|-1}))/2 & \text{if } |N_i| \geq 2 \\ h_{[0,\infty)}(b_1)/2 & \text{if } |N_i| = 1 \\ 0 & \text{if } N_i = \emptyset \end{cases} \tag{43}$$

where $b_1 \leq ... \leq b_{|N_i|}$ is a non-decreasing sequence of values $w_{ij} - y_j$, $j \in N_i$ and $h_{[0,\infty)}(\cdot)$ denotes the projection onto $\mathbb{R}_+$ as defined in §3.1.

We experimentally evaluated coordinate-wise optimization on the dual (42) with the previously stated update and compared its results with the optimal values. We generated random undirected graphs with $|V| \in \{5, 10, 30, 100\}$ vertices, edge density $d \in \{0.1, 0.5, 0.75\}$, and edge weights randomly uniformly chosen from $[k]$ for $k \in \{1, 10, 10^3, 10^6\}$. For each setting of parameters, 20 random graphs were evaluated, hence there were 960 instances in total. We report the experimental results in Table 4. In most cases, coordinate-wise optimization attained the global optimum up to numerical precision, but there occur non-optimal fixed points, especially for small values of k and/or larger density d.

We believe that similar results would also be possible for closely related LP relaxation of minimum weight edge cover [21] where one minimizes in (41) and (41b) has sign $\geq$ instead of $\leq$. The dual LP relaxation is the same as (42) except for replacing max by min and vice versa.

6 Concluding Remarks

We presented two classes of linear programs that are exactly solved by coordinate-wise minimization. It was shown that dual LP relaxations of several well-known combinatorial optimization problems (partial Max-2SAT, vertex cover, minimum *st*-cut, MAP inference with Potts potentials and two labels, and other problems) belong, possibly after a reformulation, to one of these classes. We have shown experimentally (in this paper and in [53]) that the resulting methods are reasonably efficient for large-scale instances of these problems.

Parameters		$d=0.1$		$d=0.5$		$d=0.75$	
		Mean RD	Median RD	Mean RD	Median RD	Mean RD	Median RD
$\|V\|=5$	$k=1$	$-3.27\cdot10^{-10}$	$-3.27\cdot10^{-10}$	$7.35\cdot10^{-8}$	$1.01\cdot10^{-8}$	$2.00\cdot10^{-2}$	$5.59\cdot10^{-8}$
	$k=10$	$-4.13\cdot10^{-9}$	$-1.56\cdot10^{-9}$	$-5.02\cdot10^{-9}$	$-2.79\cdot10^{-9}$	$1.16\cdot10^{-10}$	$-6.72\cdot10^{-10}$
	$k=10^3$	$-1.56\cdot10^{-10}$	$-6.36\cdot10^{-12}$	$-7.41\cdot10^{-10}$	$-3.98\cdot10^{-11}$	$-6.98\cdot10^{-11}$	$-3.10\cdot10^{-11}$
	$k=10^6$	$4.05\cdot10^{-14}$	$8.56\cdot10^{-14}$	$-1.16\cdot10^{-11}$	$-3.43\cdot10^{-12}$	$-1.93\cdot10^{-11}$	$-8.19\cdot10^{-12}$
$\|V\|=10$	$k=1$	$2.48\cdot10^{-8}$	$-3.41\cdot10^{-9}$	$3.11\cdot10^{-2}$	$2.72\cdot10^{-8}$	$2.48\cdot10^{-2}$	$2.11\cdot10^{-8}$
	$k=10$	$-3.88\cdot10^{-9}$	$-3.56\cdot10^{-9}$	$1.72\cdot10^{-3}$	$-1.48\cdot10^{-9}$	$-4.79\cdot10^{-9}$	$-4.18\cdot10^{-9}$
	$k=10^3$	$-2.55\cdot10^{-11}$	$-1.27\cdot10^{-11}$	$-6.99\cdot10^{-10}$	$-5.38\cdot10^{-11}$	$-3.46\cdot10^{-10}$	$-4.13\cdot10^{-11}$
	$k=10^6$	$-5.22\cdot10^{-12}$	$-2.76\cdot10^{-12}$	$-2.69\cdot10^{-11}$	$-1.70\cdot10^{-11}$	$-2.53\cdot10^{-11}$	$-1.87\cdot10^{-11}$
$\|V\|=30$	$k=1$	$1.08\cdot10^{-4}$	$3.46\cdot10^{-8}$	$4.96\cdot10^{-4}$	$5.02\cdot10^{-9}$	$1.64\cdot10^{-2}$	$4.51\cdot10^{-9}$
	$k=10$	$-4.11\cdot10^{-9}$	$-1.28\cdot10^{-9}$	$5.40\cdot10^{-4}$	$-4.86\cdot10^{-10}$	$2.01\cdot10^{-5}$	$2.61\cdot10^{-9}$
	$k=10^3$	$-2.16\cdot10^{-9}$	$-1.52\cdot10^{-9}$	$-7.23\cdot10^{-11}$	$-7.56\cdot10^{-11}$	$-6.51\cdot10^{-11}$	$-6.80\cdot10^{-11}$
	$k=10^6$	$-2.26\cdot10^{-11}$	$-1.75\cdot10^{-11}$	$-3.64\cdot10^{-11}$	$-3.41\cdot10^{-11}$	$-5.12\cdot10^{-11}$	$-3.26\cdot10^{-11}$
$\|V\|=100$	$k=1$	$2.43\cdot10^{-4}$	$1.02\cdot10^{-8}$	$1.26\cdot10^{-9}$	$1.32\cdot10^{-9}$	$4.01\cdot10^{-3}$	$2.52\cdot10^{-11}$
	$k=10$	$2.17\cdot10^{-9}$	$3.03\cdot10^{-10}$	$2.52\cdot10^{-4}$	$5.89\cdot10^{-9}$	$6.43\cdot10^{-5}$	$3.32\cdot10^{-10}$
	$k=10^3$	$-2.50\cdot10^{-9}$	$-1.73\cdot10^{-9}$	$-5.25\cdot10^{-9}$	$-5.04\cdot10^{-9}$	$-5.46\cdot10^{-9}$	$-4.88\cdot10^{-9}$
	$k=10^6$	$-1.83\cdot10^{-9}$	$-8.30\cdot10^{-11}$	$-7.29\cdot10^{-9}$	$-6.48\cdot10^{-9}$	$-6.47\cdot10^{-9}$	$-5.92\cdot10^{-9}$

Table 4: Comparison of results on randomly generated instances of maximum weight matching. The reason for negative entries is that coordinate-wise optimization found a solution with better objective than the LP solver, which is caused by numerical precision issues.

The direct practical impact of Theorem 1 is limited because the presented dual LP relaxations satisfying its assumptions can be efficiently solved also by other approaches. Thus, max-flow/min-*st*-cut can be solved (besides well-known combinatorial algorithms such as Ford-Fulkerson) by message-passing methods such as TRW-S. Similarly, the Potts problem with two labels is tractable and can be reduced to max-flow. In general, the LP relaxations can be reduced to max-flow, as noted in §4.5. Note, however, that this does not make our result trivial because (as noted in §2) equivalent reformulations of problems may not preserve interior local minima and thus message-passing methods are not equivalent in any obvious way to our method.

It is open whether there are practically interesting classes of linear programs that are solved exactly (or at least with constant approximation ratio) by (block-)coordinate minimization and are not solvable by known combinatorial algorithms. Another interesting question is which reformulations in general preserve interior local minima and which do not.

We notice that when the assumptions of Theorem 1 are relaxed (e.g., general Max-SAT instead of Max-2SAT, or the Potts problem with any number of labels, matrix with two non-zeros per column instead of one), the method experimentally still provides good local (though not global in general) optima. All of the previously mentioned classes of LPs where coordinate-wise optimization was observed to work well are sparse in a certain sense. For example in Max-3SAT, each constraint in (19) contains up to 4 variables, whereas in the problems discussed in §5, each variable occurs in at most 2 (resp. 4 if box constraints are counted) constraints. We believe that this observation may have a practical impact on applying coordinate-wise optimization on large sparse LPs to provide relatively tight estimation on optimal value. In other words, one may not need to theoretically guarantee global optimality to obtain a good bound on optimum or even attain the optimum in practice.

Our approach can pave the way to new efficient large-scale optimization methods in the future. Certain features of our results give us hope here. For instance, our approach has an important novel feature over message-passing methods: it applies to a *constrained* convex problem (the box constraints (6b)). This can open the way to a new class of applications. Furthermore, updates along large variable blocks (which we have not explored) can speed algorithms considerably, e.g., TRW-S uses updates along subtrees of a graphical model, while max-sum diffusion uses updates along single variables.

References

[1] Ansotegui, C., Bacchus, F., Järvisalo, M., Martins, R., et al.: MaxSAT Evaluation 2017 (2017). URL https://helda.helsinki.fi/bitstream/handle/10138/233112/mse17proc.pdf

[2] Bacchus, F., Järvisalo, M., Martins, R.: MaxSAT Evaluation 2018: New developments and detailed results. Journal on Satisfiability, Boolean Modeling and Computation **11**(1), 99–131 (2019). Instances available at https://maxsat-evaluations.github.io/.

[3] Bacchus, F., Järvisalo, M.J., Martins, R., et al.: MaxSAT Evaluation 2018 (2018). URL https://helda.helsinki.fi/bitstream/handle/10138/237139/mse18_proceedings.pdf

[4] Bertsekas, D.P.: Nonlinear Programming, 2nd edn. Athena Scientific, Belmont, MA (1999)

[5] Boyd, S., Parikh, N., Chu, E., Peleato, B., Eckstein, J.: Distributed optimization and statistical learning via the alternating direction method of multipliers. Foundations and Trends in Machine learning **3**(1), 1–122 (2011)

[6] Boykov, Y., Lempitsky, V.S.: From photohulls to photoflux optimization. In: Proceedings of the British Machine Conf., vol. 3, p. 27. Citeseer (2006)

[7] Boykov, Y., Veksler, O., Zabih, R.: Markov random fields with efficient approximations. In: Proceedings. 1998 IEEE Computer Society Conf. on Computer Vision and Pattern Recognition (Cat. No. 98CB36231), pp. 648–655. IEEE (1998)

[8] Brualdi, R.A.: Combinatorial matrix classes, vol. 13. Cambridge University Press (2006)

[9] Carpaneto, G., Toth, P.: Primal-dual algrorithms for the assignment problem. Discrete Applied Mathematics **18**(2), 137–153 (1987)

[10] Chambolle, A., Pock, T.: A first-order primal-dual algorithm for convex problems with applications to imaging. Journal of Mathematical Imaging and Vision **40**(1), 120–145 (2011)

[11] Chudak, F.A., Hochbaum, D.S.: A half-integral linear programming relaxation for scheduling precedence-constrained jobs on a single machine. Operations Research Letters **25**(5), 199–204 (1999)

[12] Dechter, R.: Tractable structures for constraint satisfaction problems. In: Handbook of Constraint Programming, chap. 7. Elsevier (2006)

[13] Dell'Amico, M., Toth, P.: Algorithms and codes for dense assignment problems: the state of the art. Discrete Applied Mathematics **100**(1-2), 17–48 (2000)

[14] Dlask, T.: Unit propagation by means of coordinate-wise minimization. In: International Conference on Machine Learning, Optimization, and Data Science. Springer (2020)

[15] Dlask, T., Werner, T.: Bounding linear programs by constraint propagation: Application to Max-SAT. In: International Conference on Principles and Practice of Constraint Programming, pp. 177–193. Springer (2020)

[16] Dlask, T., Werner, T.: A class of linear programs solvable by coordinate-wise minimization. In: I.S. Kotsireas, P.M. Pardalos (eds.) Learning and Intelligent Optimization, pp. 52–67. Springer (2020)

[17] Dlask, T., Werner, T.: On relation between constraint propagation and block-coordinate descent in linear programs. In: International Conference on Principles and Practice of Constraint Programming, pp. 194–210. Springer (2020)

[18] Duan, R., Pettie, S.: Linear-time approximation for maximum weight matching. Journal of the ACM (JACM) **61**(1), 1–23 (2014)

[19] Edmonds, J.: Maximum matching and a polyhedron with 0, 1-vertices. Journal of research of the National Bureau of Standards B **69**(125-130), 55–56 (1965)

[20] Edmonds, J., Karp, R.M.: Theoretical improvements in algorithmic efficiency for network flow problems. Journal of the ACM (JACM) **19**(2), 248–264 (1972)

[21] Ferdous, S., Pothen, A., Khan, A.: New approximation algorithms for minimum weighted edge cover. In: 2018 Proceedings of the Seventh SIAM Workshop on Combinatorial Scientific Computing, pp. 97–108. SIAM (2018)

[22] Freuder, E.C.: A sufficient condition for backtrack-free search. Journal of the ACM (JACM) **29**(1), 24–32 (1982)

[23] Fulkerson, D., Ford, L.: Flows in networks. Princeton University Press (1962)

[24] Gabow, H.N.: Data structures for weighted matching and nearest common ancestors with linking. In: Proceedings of the first annual ACM-SIAM symposium on Discrete algorithms, pp. 434–443 (1990)

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

[26] Goldberg, A.V., Kennedy, R.: An efficient cost scaling algorithm for the assignment problem. Mathematical Programming **71**(2), 153–177 (1995)

[27] Hochbaum, D.S.: Solving integer programs over monotone inequalities in three variables: A framework for half integrality and good approximations. European Journal of Operational Research **140**(2), 291–321 (2002)

[28] Hochbaum, D.S., Pathria, A.: Forest harvesting and minimum cuts: a new approach to handling spatial constraints. Forest Science **43**(4), 544–554 (1997)

[29] Hochbaum, D.S., Pathria, A.: Approximating a generalization of MAX 2SAT and MIN 2SAT. Discrete Applied Mathematics **107**(1-3), 41–59 (2000)

[30] Hooker, J.: Logic-based methods for optimization: combining optimization and constraint satisfaction. Wiley series in discrete mathematics and optimization. Wiley (2000)

[31] Kappes, J.H., Andres, B., Hamprecht, F.A., Schnörr, C., Nowozin, S., Batra, D., Kim, S., Kausler, B.X., Kröger, T., Lellmann, J., Komodakis, N., Savchynskyy, B., Rother, C.: A comparative study of modern inference techniques for structured discrete energy minimization problems. Intl. Journal of Computer Vision **115**(2), 155–184 (2015)

[32] Kohli, R., Krishnamurti, R., Mirchandani, P.: The minimum satisfiability problem. SIAM Journal on Discrete Mathematics **7**(2), 275–283 (1994)

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

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

[35] Kolmogorov, V., Zabih, R.: Computing visual correspondence with occlusions via graph cuts. Tech. rep., Cornell University (2001)

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

[37] Kuhn, H.W.: The Hungarian method for the assignment problem. Naval research logistics quarterly **2**(1-2), 83–97 (1955)

[38] Lempitsky, V., Boykov, Y.: Global optimization for shape fitting. In: 2007 IEEE Conf. on Computer Vision and Pattern Recognition, pp. 1–8. IEEE (2007)

[39] Lempitsky, V., Boykov, Y., Ivanov, D.: Oriented visibility for multiview reconstruction. In: European Conf. on Computer Vision, pp. 226–238. Springer (2006)

[40] Li, C.M., Manya, F.: Maxsat, hard and soft constraints. Handbook of satisfiability **185**, 613–631 (2009)

[41] Matoušek, J., Gärtner, B.: Understanding and using linear programming (universitext). Springer-Verlag (2006)

[42] Papadimitriou, C.H., Steiglitz, K.: Combinatorial optimization: algorithms and complexity. Courier Corporation (1998)

[43] Průša, D., Werner, T.: LP relaxation of the Potts labeling problem is as hard as any linear program. IEEE Trans. on Pattern Analysis and Machine Intelligence **39**(7), 1469–1475 (2017)

[44] Průša, D., Werner, T.: LP relaxations of some NP-hard problems are as hard as any LP. In: ACM-SIAM Symp. on Discrete Algorithm (SODA), pp. 1372–1382. SIAM (2017)

[45] Scharstein, D., Szeliski, R.: A taxonomy and evaluation of dense two-frame stereo correspondence algorithms. Intl. Journal of Computer Vision **47**(1-3), 7–42 (2002)

[46] Schlesinger, M.I., Antoniuk, K.: Diffusion algorithms and structural recognition optimization problems. Cybernetics and Systems Analysis **47**, 175–192 (2011)

[47] Szeliski, R., Zabih, R., Scharstein, D., Veksler, O., Kolmogorov, V., Agarwala, A., Tappen, M., Rother, C.: A comparative study of energy minimization methods for markov random fields with smoothness-based priors. IEEE Trans. on Pattern Analysis and Machine Intelligence **30**(6), 1068–1080 (2008)

[48] Tourani, S., Shekhovtsov, A., Rother, C., Savchynskyy, B.: MPLP++: Fast, parallel dual block-coordinate ascent for dense graphical models. In: Proceedings of the European Conference on Computer Vision, pp. 251–267 (2018)

[49] Tourani, S., Shekhovtsov, A., Rother, C., Savchynskyy, B.: Taxonomy of dual block-coordinate ascent methods for discrete energy minimization. arXiv.org (2020)

[50] Tseng, P.: Convergence of a block coordinate descent method for nondifferentiable minimization. Journal of Optimization Theory and Applications **109**(3), 475–494 (2001)

[51] Vazirani, V.V.: Approximation Algorithms. Springer-Verlag New York (2001)

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

[53] Werner, T., Průša, D., Dlask, T.: Relative interior rule in block-coordinate descent. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 7559–7567 (2020)

[54] Werner, T., Průša, D.: Relative interior rule in block-coordinate minimization. arXiv.org (2019)

[55] Wright, S.J.: Coordinate descent algorithms. Mathematical Programming **151**(1), 3–34 (2015)

[56] Xu, K., Li, W.: Many hard examples in exact phase transitions with application to generating hard satisfiable instances. arXiv.org (2003). Instances available at http://sites.nlsde.buaa.edu.cn/~kexu/benchmarks/max-sat-benchmarks.htm.