The version of record of this article is published in Constraints and is available online at
https://doi.org/10.1007/s10601-023-09349-0.

Activity Propagation in Systems of Linear Inequalities and Its Relation to Block-Coordinate Descent in Linear Programs

Tomáš Dlask*, Tomáš Werner

Machine Learning Group, Department of Cybernetics, Faculty of Electrical Engineering, Czech Technical University in Prague

dlaskto2@fel.cvut.cz, werner@fel.cvut.cz

Abstract

We study a constraint propagation algorithm to detect infeasibility of a system of linear inequalities over continuous variables, which we call activity propagation. Each iteration of this algorithm chooses a subset of the inequalities and if it infers that some of them are always active (i.e., always hold with equality), it turns them into equalities. We show that this algorithm can be described as chaotic iterations and its fixed points can be characterized by a local consistency, in a similar way to traditional local consistency methods in CSP such as arc consistency. Via complementary slackness, activity propagation can be employed to iteratively improve a dual-feasible solution of large-scale linear programs in a primal-dual loop – a special case of this method is the Virtual Arc Consistency algorithm by Cooper et al. As our second contribution, we show that this method has the same set of fixed points as block-coordinate descent (BCD) applied to the dual linear program. While BCD is popular in large-scale optimization, its fixed points need not be global optima even for convex problems and a succinct characterization of convex problems optimally solvable by BCD remains elusive. Our result may open the way for such a characterization since it allows us to characterize BCD fixed points in terms of local consistencies.

Keywords: Block-Coordinate Descent, Constraint Propagation, Primal-Dual Method, Linear Programming, Virtual Arc Consistency

1 Introduction

Solving large-scale convex optimization problems (such as linear programming relaxations of large combinatorial problems encountered in artificial intelligence, computer vision or machine learning) can be challenging and one often has to resort to approximate methods. One such popular method is *block-coordinate descent* (BCD), which in every iteration optimizes the problem over a subset (block) of variables, keeping the remaining variables constant. If each block contains only a single variable, we speak about *coordinate descent*. Unfortunately, BCD fixed points can be arbitrarily far from global optima even for convex problems. The class of convex optimization problems for which BCD is known to converge to global optima is currently quite narrow, revolving around unconstrained minimization of convex functions whose non-differentiable part is separable [41].

For general (non-differentiable and/or constrained) convex problems, the set of block-optimal solutions in a BCD iteration can contain more than one element. It has been recently argued [48, 47] that in that case, one should choose an optimal solution from the relative interior of this set. BCD updates satisfying this *relative-interior rule* are not worse than any other rule to choose block-optimal solutions. Of course, this rule does not guarantee convergence to global optima.

BCD methods known as *convex message passing* are state of the art for approximately solving the dual linear programming (LP) relaxation of the MAP inference problem in graphical models [43, 26, 36] in computer vision and machine learning, which is equivalent to the weighted (or valued) CSP [42]. Examples are max-sum diffusion [29, 44], TRW-S [27], MPLP [20], or MPLP++ [40]. These methods comply to the relative-interior rule (or at least have the same fixed points as BCD with this rule) [47] and their fixed points are characterized by local consistencies – precisely, arc consistency of the active tuples (i.e., the tuples for which the corresponding inequalities in the dual LP relaxation are active [45]).

*Corresponding author.

Another approach to tackle the dual LP relaxation of weighted CSP is the *Virtual Arc Consistency (VAC) algorithm* [9] and the similar Augmenting DAG algorithm [28, 44]. Though these are not BCD methods, their fixed points are also characterized by arc consistency of active tuples. We showed in [15] that this approach is related to the primal-dual method [34, Section 5] in linear programming and generalized it to any linear program by replacing the arc-consistency algorithm with general constraint propagation in a system of linear inequalities (assuming that an initial feasible solution is provided).

It has been observed [13] that when BCD with the relative-interior rule is applied to the dual LP relaxation of SAT, it corresponds precisely to unit propagation. Moreover, there exists a connection between a form of the dominating unit-clause rule and BCD with the relative-interior rule applied to the dual LP relaxation of weighted Max-SAT [13].

The above results suggest there is a close relation between BCD applied to a linear program and constraint propagation in a system of linear inequalities. In this paper, we describe this relation precisely. While constraint propagation in a linear inequality system can be done in various ways, we study a particular propagation algorithm which we call *activity propagation*. In every iteration, this algorithm infers from a subset of inequalities that some of them are always active (i.e., always hold with equalities) and turns these inequalities into equalities. When an infeasible system is produced in this way, clearly the initial system was also infeasible. We show that activity propagation can be explained using the framework of chaotic iterations and closure operators, in a similar way to many of the popular constraint propagation methods for CSPs [2, 3]. Moreover, we show that the primal-dual approach [15] with activity propagation and BCD with[1] the relative-interior rule have the same fixed points. Thus, the question of whether a given linear program is exactly solvable by BCD can be translated to the question of whether feasibility of a certain class of linear inequality systems is decidable by activity propagation.

The structure of this paper is as follows. In Section 2, we overview the background on order theory and linear programming that we will need throughout the paper. In Section 3, we formally define both compared methods, BCD and the primal-dual approach. For the constraint-propagation-based primal-dual approach, we also define and analyze the precise form of constraint propagation that we propose. In Section 4, we show the connection between the aforementioned methods, characterize different types of fixed points of BCD by local consistency conditions, and provide a characterization of linear programs optimally solvable by BCD in terms of refutation-completeness of the associated propagator. We discuss in Section 5 how our results generalize to linear programs in any form and also explain why adding redundant constraints to a linear program may improve solvability of its dual by BCD. Appendix A shows how to compute an improving direction whenever activity propagation detects that a current solution is not optimal, which is important for a practical implementation of the primal-dual approach. Finally, Appendix B presents a connection between sets satisfying the previously mentioned local consistency condition and faces of the underlying polyhedron.

The current paper extends its earlier conference version [17] by relating our results to known facts from order theory and linear programming, which improves readability and draws a clear analogy between the proposed propagation rule and classical propagators considered in CSPs. Furthermore, we refine our results so that additional connections emerge, such as the connection between individual BCD updates and actions of a propagator, and improve the overall structure of our proofs. Finally, we provide a link between sets satisfying an associated local consistency condition and faces of the underlying polyhedron. Some parts of this paper appear in the first author's dissertation [14].

2 Preliminaries

In this section, we provide the necessary background on order theory, chaotic iterations, linear programming, and systems of linear inequalities.

2.1 Lattices, Closure Operators, and Chaotic Iterations

Let us begin by reviewing the basic concepts of order theory. We start by recalling the notions of lattice, semilattice, and complete lattice. Next, we focus on the connection between complete lattices and closure operators. We conclude by analyzing iterative applications of isotone and intensive mappings. This part is based mainly on the books [6, 11]. Its purpose is to provide background for constraint propagation since

[1] Our results also apply to general BCD methods (possibly not adhering to the relative-interior rule) if their stopping points are 'best possible' in the sense that no sequence of arbitrary BCD updates does not improve them – this will be made more precise later (Definition 4). However, such stopping points coincide with those that are attained by BCD with relative-interior rule.

locally consistent problems can be defined as common fixed points of propagators on partially ordered sets [3].

Let S be a set and $\preceq$ be a *partial order* on S, i.e., $\preceq$ is a binary relation on S that is reflexive, anti-symmetric, and transitive. Let us first recall the *duality principle* in partially ordered sets [6, Section 1][11, Section 1.20]: for any true statement about a partially ordered set $(S, \preceq)$, there is a corresponding true statement about its dual ordered set $(S, \succeq)$ where $\succeq$ is the inverse order (a.k.a. dual order), i.e., $s_1 \succeq s_2 \iff s_2 \preceq s_1$ for all $s_1, s_2 \in S$. The corresponding dual statement is obtained by replacing all (both explicit and implicit) occurrences of $\preceq$ in the statement by $\succeq$.

Inspired by the notation in [6], for $Q \subseteq S$, we define the set of all *upper bounds* and *lower bounds* on Q in S, respectively, to be

$$Q_S^{\uparrow} = \{s \in S \mid \forall q \in Q : q \preceq s\} \tag{1a}$$

$$Q_S^{\downarrow} = \{s \in S \mid \forall q \in Q : s \preceq q\}. \tag{1b}$$

Furthermore, if $q^* \in Q_S^{\uparrow}$ satisfies $q^* \preceq q$ for all $q \in Q_S^{\uparrow}$, then it is called the *least upper bound* on Q in S and we denote it by $q^* = \bigvee_S Q$. Analogously, if $q^* \in Q_S^{\downarrow}$ satisfies $q^* \succeq q$ for all $q \in Q_S^{\downarrow}$, then it is called the *greatest lower bound* on Q in S and is denoted by $q^* = \bigwedge_S Q$. The operations $\bigvee_S$ and $\bigwedge_S$ are called the *join* and the *meet*, respectively. For two-element sets $Q = \{q_1, q_2\}$, we write $q_1 \vee_S q_2$ instead of $\bigvee_S \{q_1, q_2\}$ and $q_1 \wedge_S q_2$ instead of $\bigwedge_S \{q_1, q_2\}$. In general, a partially ordered set need not have the least upper bound or greatest lower bound for each of its subsets.

In particular, $S_S^{\uparrow}$ is either empty or a singleton because if $s_1, s_2 \in S_S^{\uparrow}$, then $s_1 \preceq s_2$ and $s_2 \preceq s_1$ by definition (1a), hence $s_1 = s_2$ by anti-symmetry. If $S_S^{\uparrow}$ is non-empty, then $S_S^{\uparrow} = \{\top\}$ where $\top = \bigvee_S S \in S$ is the *top element* of S. By the duality principle, $S_S^{\downarrow}$ is also either empty or a singleton. If $S_S^{\downarrow} = \{\bot\}$, then $\bot = \bigwedge_S S \in S$ is the *bottom element* of S.[2] Again, a partially ordered set need not contain the top or the bottom element in general.

If the set S w.r.t. which the upper bounds, lower bounds, joins, and meets are taken is clear from context, then we omit the subscript S for brevity, i.e., we simplify $Q_S^{\uparrow}$, $Q_S^{\downarrow}$, $\bigvee_S$, and $\bigwedge_S$ to $Q^{\uparrow}$, $Q^{\downarrow}$, $\bigvee$, and $\bigwedge$.

2.1.1 Lattices

Definition 1. *A partially ordered set $(S, \preceq)$ is:*

- *a* meet-semilattice *if for any $s_1, s_2 \in S$, $s_1 \wedge s_2$ exists in S,*
- *a* join-semilattice *if for any $s_1, s_2 \in S$, $s_1 \vee s_2$ exists in S,*
- *a* lattice *if it is a meet-semilattice and a join-semilattice,*
- *a* complete lattice *if for any $Q \subseteq S$, both $\bigwedge Q$ and $\bigvee Q$ exist in S.*

It is easily shown by induction [11, Section 2.11] that for a meet-semilattice $(S, \preceq)$ and any non-empty finite $Q \subseteq S$, $\bigwedge Q$ exists in S. Consequently, any non-empty finite meet-semilattice $(S, \preceq)$ has a bottom element, namely $\bigwedge S$. By the duality principle, for any join-semilattice $(S, \preceq)$ and any non-empty finite $Q \subseteq S$, $\bigvee Q$ exists in S. In particular, any non-empty finite join-semilattice $(S, \preceq)$ has a top element, $\bigvee S$. A complete lattice always has both the top and the bottom element [6, Theorem 2.10].

The following well-known theorem connects these concepts for a finite partially ordered set.

Theorem 1 ([33, 11, 6])**.** *Let $(S, \preceq)$ be a non-empty finite partially ordered set. The following are equivalent:*

(a) $(S, \preceq)$ is a meet-semilattice with top element $\top$,

(b) $(S, \preceq)$ is a join-semilattice with bottom element $\bot$,

(c) $(S, \preceq)$ is a complete lattice,

(d) $(S, \preceq)$ is a lattice.

By Theorem 1, it is possible to augment any finite meet-semilattice by introducing a new artificial top element to obtain a complete lattice. Dually, one can include a bottom element in a finite join-semilattice which is known as the *lifting* operation [11, Section 1.22].

[2] For the top and the bottom element, we adopt notation from [11, Section 1.21] to avoid using 0 and 1 which we reserve for their numerical meaning.

2.1.2 (Dual) Closure Operators

Let us now discuss the connection between closure operators and complete lattices. In detail, the image of any closure operator defined on a complete lattice is a complete sublattice of this lattice (see Theorem 4) and, *vice versa*, under additional assumptions any complete sublattice of a complete lattice defines a closure operator (Theorem 3).

First, we recall some properties of mappings on partially ordered sets:

Definition 2 ([6, Section 1.4])**.** *Let $(S, \preceq)$ be a partially ordered set. Mapping $f\colon S \to S$ is*

- extensive *if* $\forall s \in S\colon s \preceq f(s)$,
- intensive *if* $\forall s \in S\colon f(s) \preceq s$,
- idempotent *if* $\forall s \in S\colon f(s) = f(f(s))$,
- isotone *if* $\forall s_1, s_2 \in S\colon s_1 \preceq s_2 \implies f(s_1) \preceq f(s_2)$,
- *a* closure operator *if it is extensive, idempotent, and isotone,*
- *a* dual closure operator *if it is intensive, idempotent, and isotone.*

We denote the *image* of a mapping $f\colon S \to S$ by $\operatorname{im} f = \{f(s) \mid s \in S\}$.

Proposition 2 (cf. [6, Section 1.4])**.** *Let $(S, \preceq)$ be a partially ordered set and $f\colon S \to S$ be an idempotent mapping. The set of fixed points of f coincides with its image, i.e., $\{s \in S \mid f(s) = s\} = \operatorname{im} f$.*

Proof. Inclusion in the $\supseteq$ direction follows from idempotence and the other direction is clear from the definition of a fixed point. $\square$

Lemma 1 ([11, Lemma 2.22(v)])**.** *Let $(S, \preceq)$ be a complete lattice and $S_1 \subseteq S_2 \subseteq S$. Then $\bigwedge S_1 \succeq \bigwedge S_2$ and $\bigvee S_2 \succeq \bigvee S_1$.*

Proof. The first claim follows from the fact that $\bigwedge S_2 = \bigwedge S_1 \wedge \bigwedge (S_2 - S_1)$ is a lower bound on $\bigwedge S_1$. The second claim follows from the duality principle. $\square$

Theorem 3 (cf. [11, Theorem 7.3])**.** *Let $(S, \preceq)$ and $(Q, \preceq)$ be complete lattices such that $Q \subseteq S$.*

(a) If the meet operations in $(S, \preceq)$ and $(Q, \preceq)$ coincide[3]*, then the mapping $f\colon S \to Q$ defined by $f(s) = \bigwedge \{s\}^{\uparrow}_{Q}$ is a closure operator.*

(b) If the join operations in $(S, \preceq)$ and $(Q, \preceq)$ coincide, then the mapping $g\colon S \to Q$ defined by $g(s) = \bigvee \{s\}^{\downarrow}_{Q}$ is a dual closure operator.

Proof. We show only (a) since (b) then follows from the duality principle. Notice that we do not need to distinguish $\bigwedge_S$ and $\bigwedge_Q$ in the definition of f because $\{s\}^{\uparrow}_{Q}$ is a subset of Q and the meet operations coincide.

We begin by extensivity: for $s \in S$, $s \in \{s\}^{\uparrow}_{S}$ holds by reflexivity of $\preceq$ and also $\{s\}^{\uparrow}_{Q} \subseteq \{s\}^{\uparrow}_{S}$ due to $Q \subseteq S$. Lemma 1 yields $f(s) = \bigwedge \{s\}^{\uparrow}_{Q} \succeq \bigwedge \{s\}^{\uparrow}_{S} = s$.

For isotony, let $s_1, s_2 \in S$ be such that $s_1 \preceq s_2$. Then $\{s_1\}^{\uparrow}_{Q} \supseteq \{s_2\}^{\uparrow}_{Q}$ by transitivity of $\preceq$ and we obtain $f(s_1) = \bigwedge \{s_1\}^{\uparrow}_{Q} \preceq \bigwedge \{s_2\}^{\uparrow}_{Q} = f(s_2)$ by Lemma 1.

For idempotency, we have $f(s) \in \{f(s)\}^{\uparrow}_{Q}$, so $f(f(s)) = \bigwedge \{f(s)\}^{\uparrow}_{Q} = f(s)$. $\square$

Focusing on the definition of f and g in Theorem 3, for any $s \in S$, we have that $f(s)$ is the least upper bound on s in Q and $g(s)$ is the greatest lower bound on s in Q. Theorem 3 has the following practical corollary.

Corollary 1. *Let $(S, \preceq)$ and $(Q, \preceq)$ be complete lattices with $Q \subseteq S$. Define the mappings $f, g\colon S \to Q$ by $f(s) = \bigwedge_Q \{s\}^{\uparrow}_{Q}$ and $g(s) = \bigvee_Q \{s\}^{\downarrow}_{Q}$. It holds that*

$$Q = \{s \in S \mid f(s) = s\} = \{s \in S \mid g(s) = s\}. \tag{2}$$

Proof. If $s \in Q$, $s \in \{s\}^{\uparrow}_{Q}$ and $s \preceq q$ holds for any $q \in \{s\}^{\uparrow}_{Q}$ by definition, so $s = \bigwedge_Q \{s\}^{\uparrow}_{Q} = f(s)$. If $s \notin Q$, $f(s) \neq s$ due to $\operatorname{im} f = Q$. The part with g is obtained dually. $\square$

[3] That is, for any $Q' \subseteq Q$ we have $\bigwedge_S Q' = \bigwedge_Q Q'$. Similarly for the join.

Algorithm 1 Iterations of a given set of mappings applied to an initial element $s \in S$ of a partially ordered set $(S, \preceq)$.

input: partially ordered set $(S, \preceq)$, initial element $s \in S$, isotone and intensive mappings $f_1, \ldots, f_n \colon S \to S$.

1: $s' \leftarrow s$
2: **while** $\exists i \in [n] : f_i(s') \neq s'$ **do**
3: Find such i.
4: Update $s' \leftarrow f_i(s')$.
5: **return** s'

The converse connection between (dual) closure operators and complete lattices also holds and we review it in the next theorem.

Theorem 4 ([6, Theorem 2.14][11, Section 7])**.** *Let $(S, \preceq)$ be a complete lattice. If $f \colon S \to S$ is a closure operator (or a dual closure operator), then $(\operatorname{im} f, \preceq)$ is a complete lattice.*

Proof. We prove the case when f is a closure operator. Let $\top = \bigvee_S S \in S$ be the top element in S (which exists because it is a complete lattice). By extensivity of f, we have that $\top \preceq f(\top) \in S$, hence $f(\top) = \top$ and $\top \in \operatorname{im} f$, so $(\operatorname{im} f, \preceq)$ has the top element $\top$.

Next, let $Q \subseteq \operatorname{im} f$ and $s = \bigwedge_S Q \in S$ be the greatest lower bound on Q in S. We will show that $s \in \operatorname{im} f$. Clearly, for all $q \in Q$, we have that $s \preceq q$ by definition of s. Consequently, $f(s) \preceq f(q) = q$ by isotony of f and by q being a fixed point of f (due to $q \in \operatorname{im} f$). Therefore, $f(s) \preceq \bigwedge_S Q = s$. By extensivity of f and anti-symmetry of $\preceq$, $f(s) = s$, so $s \in \operatorname{im} f$. Theorem 1 yields that $(\operatorname{im} f, \preceq)$ is a complete lattice.

The case when f is a dual closure operator is analogous. $\square$

2.1.3 Chaotic Iterations

(Dual[4]) closure operators frequently appear in the field of constraint propagation. An example is the arc consistency closure [5, Section 3.3.1]. For more details, we refer to the partial order over domain-based tightenings and Φ-closure in [5, Section 3.2] and domain reduction in [2, Section 3.1]. The closure is typically obtained by iteratively applying *propagators*, which are usually intensive (or extensive, cf. Footnote 4) and isotone mappings, but may also be idempotent, commutative, or semi-commutative [5, 2, 3]. This general framework was originally studied in [2, 3] under the name *chaotic iteration* and many constraint propagation algorithms, such as the well-known AC-3 algorithm, can be seen as instantiations of this framework.

We outline a particular version of chaotic iterations in Algorithm 1. We are given a finite set of isotone and intensive[5] mappings on a finite partially ordered set $(S, \preceq)$. For an input element $s \in S$, the algorithm finds the greatest common fixed point s' of the mappings such that $s' \preceq s$ (as stated by Theorem 5).

Theorem 5 (cf. [3, Theorem 1])**.** *Let $(S, \preceq)$ be a finite partially ordered set, $s \in S$, and $f_1, \ldots, f_n \colon S \to S$ be isotone and intensive mappings. Algorithm 1 terminates in a finite number of iterations and returns the greatest common fixed point s' of mappings $f_1, \ldots, f_n$ such that $s' \preceq s$.*

Proof. In each iteration of the algorithm, the current s' strictly decreases w.r.t. $\preceq$ by intensivity of f_i. This implies that $s' \preceq s$ and, by finiteness of S, that the algorithm terminates after a finite number of iterations. The fact that s' is a common fixed point of all the mappings f_i follows directly from the condition on line 2 of the algorithm.

To prove that s' is the greatest common fixed point among those satisfying $s' \preceq s$, we use induction. In detail, we show that $s^* \preceq s'$ holds during the whole run of the algorithm for any common fixed point $s^* \in S$ such that $s^* \preceq s$. The base case is clear: $s^* \preceq s = s'$. For the inductive step, by isotony of f_i we have $s^* = f_i(s^*) \preceq f_i(s')$ where the equality is given by s^* being a common fixed point. $\square$

[4] Although the operators are actually dual closure operators according to the usual formalism, it is common to call them just 'closure operators' in constraint programming literature. This distinction is only technical as it can be easily corrected by considering the dual setting where the order is formally reversed.

[5] We state chaotic iterations and related results for intensive mappings and dual closures, although they can be stated (by the duality principle) also for extensive mappings and closures.

Note that, by Theorem 5, the value returned by Algorithm 1 is independent of the way of choosing i on line 3. An additional property of Algorithm 1 is given in Theorem 6.

Theorem 6. *Let $(S, \preceq)$ be a finite partially ordered set and $f_1, \dots, f_n \colon S \to S$ be isotone and intensive mappings. Let $g \colon S \to S$ be the mapping such that $g(s)$ is the greatest common fixed point of $f_1, \dots, f_n$ such that $g(s) \preceq s$ (i.e., $g(s)$ is the output of Algorithm 1 for input s). Then g is a dual closure operator.*

Proof. Intensivity and idempotence is trivial. To show isotony, let $s_1, s_2 \in S$ satisfy $s_1 \preceq s_2$. Since both $g(s_1)$ and $g(s_2)$ are common fixed points of the mappings and $g(s_1) \preceq s_1 \preceq s_2$ by intensivity, we necessarily have $g(s_1) \preceq g(s_2)$ because otherwise $g(s_2)$ would not be the greatest common fixed point such that $g(s_2) \preceq s_2$. $\square$

Remark 1. *Let $(S, \preceq)$ be a finite partially ordered set and $f_1, \dots, f_n \colon S \to S$ be dual closure operators. Let $g \colon S \to S$ be the dual closure operator defined by Algorithm 1 (as in Theorem 6). By Theorem 4, $(\operatorname{im} f_1, \preceq), \dots, (\operatorname{im} f_n, \preceq)$, and $(\operatorname{im} g, \preceq)$ are complete lattices. These lattices satisfy*

$$\operatorname{im} g = \{s \in S \mid g(s) = s\} = \{s \in S \mid \forall i \in [n] : f_i(s) = s\} = \bigcap_{i=1}^{n} \operatorname{im} f_i \tag{3}$$

which follows from Proposition 2 and the fact that fixed points of g are precisely the common fixed points of $f_1, \dots, f_n$.

2.2 Linear Programming and Always-Active Inequalities

The linear programming (LP) problem seeks to minimize or maximize a linear function of a finite number of variables over the set of solutions of a finite number of linear inequalities and equalities (i.e., a convex polyhedron). It is well-known [34, Section 2.1] that any linear program can be transformed in linear time to one of the restricted LP forms, e.g., containing only equalities and non-negative variables, or containing only inequalities and unconstrained variables.

To every linear program, one can construct its dual linear program. This construction is symmetric, so one can speak about mutually dual linear programs. We will work with the particular form of a primal-dual pair

$$
\begin{aligned}
&\max c^T x && \min b^T y && \text{(4a)}\\
&Ax = b && y \in \mathbb{R}^m && \text{(4b)}\\
&x \geq 0 && A^T y \geq c && \text{(4c)}
\end{aligned}
$$

where $A \in \mathbb{R}^{m\times n}$, $b \in \mathbb{R}^m$, $c \in \mathbb{R}^n$ are constants and $x \in \mathbb{R}^n, y \in \mathbb{R}^m$ are variables. We denote by x_j the j-th component of vector x (similarly for y, b, c) and by A^i and A_j the i-th row and j-th column of A where $i \in [m] = \{1, \dots, m\}$ and $j \in [n] = \{1, \dots, n\}$, respectively. The transpose of A is denoted by A^T. We will refer to the left-hand problem of (4) as the primal and to the right-hand problem as the dual. Note that we always write a constraint and the corresponding dual variable on the same line.

The primal and dual linear programs are related by duality theorems:

Theorem 7 (Strong duality [37, 31])**.** *For any primal-dual pair, exactly one of the following cases happens:*

(a) The primal and the dual are both feasible and their optimal values coincide.

(b) The primal and the dual are both infeasible.

(c) The primal is unbounded and the dual is infeasible or vice versa.

Theorem 8 (Complementary slackness [31, 34])**.** *Let $x \in \mathbb{R}^n$ and $y \in \mathbb{R}^m$ be feasible for the primal and the dual* (4)*, respectively. The following are equivalent:*

(a) x and y are optimal for the primal and the dual, respectively,

(b) for each $j \in [n]$ it holds that $x_j(A_j^T y - c_j) = 0$, i.e., $x_j = 0$ or $A_j^T y = c_j$.

For brevity, we define the mappings $\sigma \colon \mathbb{R}^n \to 2^{[n]}$ and $\tau \colon \mathbb{R}^m \to 2^{[n]}$ by

$$\sigma(x) = \{j \in [n] \mid x_j = 0\} \tag{5a}$$

$$\tau(y) = \{j \in [n] \mid A_j^T y = c_j\}, \tag{5b}$$

so that $\sigma(x)$ is the index set of the primal constraints (4c) that are *active* (i.e., satisfied with equality[6]) at x. Similarly, $\tau(y)$ is the index set of the dual constraints (4c) that are active at y. Using this notation, statement (b) in Theorem 8 can be expressed as $\tau(y) \cup \sigma(x) = [n]$.

In the sequel, we will need the notion of *relative interior* of a set $S \subseteq \mathbb{R}^n$, denoted by $\operatorname{ri} S$ [30, Section A2.1]. It is the topological interior of S relative to the affine hull of S, i.e., $x \in \operatorname{ri} S$ if and only if there exists a ball $B_r(x)$ centered at x with a positive radius r such that $B_r(x) \cap \operatorname{aff} S \subseteq S$. For every convex set S we have $\operatorname{ri} S \subseteq S$ and, in particular, $\operatorname{ri} S \neq \emptyset$ if and only if $S \neq \emptyset$. For a singleton set S we have $\operatorname{ri} S = S$. The relative interior of a line segment, $[x, x'] = \{(1-\alpha)x + \alpha x' \mid 0 \leq \alpha \leq 1\}$ where $x, x' \in \mathbb{R}^n$ are such that $x \neq x'$, is this line segment without its endpoints, $\operatorname{ri}[x, x'] = [x, x'] - \{x, x'\}$. Note, in contrast, that the topological interior of any line segment in $\mathbb{R}^n$ is empty unless $n = 1$. In convex optimization, the intuitive notion of 'interior' is often better captured by relative interior than topological interior: e.g., the set of optimal solutions of a linear program is a convex polyhedron, which however need not have full dimension, hence its topological interior can be empty but its relative interior is non-empty.

Now we are able to state the next theorem:

Theorem 9 (Strict complementary slackness [21, 25, 22, 49])**.** *Let $x \in \mathbb{R}^n$ and $y \in \mathbb{R}^m$ be feasible for the primal and the dual* (4)*, respectively. The following are equivalent:*

(a) x and y are in the relative interior of the set of optimal solutions of the primal and the dual, respectively,

(b) $\{\sigma(x), \tau(y)\}$ is a partition of $[n]$, i.e., for each $j \in [n]$, either $x_j = 0$ or $A_j^T y = c_j$.

As a corollary, for any x and y in the relative interior of the set of optimal solutions of the primal and the dual, respectively, the partition $\{\sigma(x), \tau(y)\}$ is the same. This unique partition is sometimes called the optimal partition of $[n]$ [25, 1, 22, 32].

Definition 3. *Let $A \in \mathbb{R}^{m\times n}$, $b \in \mathbb{R}^m$, and $j \in [n]$. We say that the inequality $x_j \geq 0$ is* always active[7] *in the system $Ax = b$, $x \geq 0$ if for every $x \in \mathbb{R}^n$ it holds that $Ax = b$, $x \geq 0$ implies $x_j = 0$.*[8]

All always-active inequalities in a system of linear inequalities and equalities can be characterized using relative interior:

Theorem 10 (cf. [23, Theorem 8])**.** *Let $A \in \mathbb{R}^{m\times n}$, $b \in \mathbb{R}^m$, $j \in [n]$, and*

$$x^* \in \operatorname{ri}\{x \in \mathbb{R}^n \mid Ax = b,\ x \geq 0\}. \tag{6}$$

System $Ax = b$, $x \geq 0$ implies $x_j = 0$ if and only if $j \in \sigma(x^*)$, i.e., $x^*_j = 0$.

Proof. Consider the primal-dual pair (4) with $c = 0$, i.e., zero objective function in the primal. Trivially, feasibility is equivalent to optimality for the primal, so the primal (and hence also the dual) is feasible and bounded.

Let y^* be in the relative interior of the set of optimal solutions of the dual (4). Any x feasible for the primal necessarily satisfies complementary slackness with dual-optimal y^*, i.e., $\tau(y^*) \cup \sigma(x) = [n]$. Therefore, the system implies $x_j = 0$ for all $j \in [n] - \tau(y^*) = \sigma(x^*)$ where the set equality $[n] - \tau(y^*) = \sigma(x^*)$ is given by strict complementary slackness. Since x^* satisfies $Ax^* = b$, $x^* \geq 0$ and $x^*_j > 0$ for all $j \in [n] - \sigma(x^*)$, the inequality $x_j \geq 0$ for $j \in [n] - \sigma(x^*)$ is not always active. □

3 Compared Methods

In this section, we formalize both considered approaches in detail, applied to the problem (4). We start by BCD in Section 3.1 and then recall the primal-dual approach in Section 3.2 where we focus on a specific constraint propagation method.

We assume that both linear programs (4) are feasible and bounded and that a dual-feasible solution y is provided. Furthermore, we expect that a finite collection $\mathcal{B} \subseteq 2^{[m]}$ of subsets ('blocks') of dual variables is given. Note that $\mathcal{B}$ can be also seen as a collection of subsets of primal constraints (4b).

For both methods, the goal is to improve this dual-feasible solution, ideally to make it dual-optimal. For brevity of notation, we will assume that the set of blocks $\mathcal{B}$, matrix A, and vectors b and c are fixed in Sections 3 and 4, and Appendices A and B.

[6] Recall that an inequality $c^T x \geq d$ is *active* at a point x if $c^T x = d$ [38, 19, 7].

[7] This term is used, e.g., in [19]. The equivalent term *implied equality* is used in [39, 23].

[8] We abbreviate the phrase 'for every $x \in \mathbb{R}^n$ it holds that $Ax = b$, $x \geq 0$ implies $x_j = 0$' by just '$Ax = b$, $x \geq 0$ implies $x_j = 0$', understanding that the quantifier $\forall x$ is implicitly present.

3.1 Block-Coordinate Descent and Relative-Interior Rule

Suppose we apply BCD to the dual (4). For brevity of notation, we formulate this optimization problem as the unconstrained minimization of the extended-valued function $f\colon \mathbb{R}^m \to \mathbb{R} \cup \{\infty\}$ defined by

$$f(y) = \begin{cases} b^T y & \text{if } A^T y \geq c \\ \infty & \text{otherwise} \end{cases}. \tag{7}$$

To formulate BCD, we introduce a new notation: for $B = \{i_1, \dots, i_{|B|}\} \subseteq [m]$ and $y \in \mathbb{R}^m$, by y_B we denote the restriction of y onto the set B, i.e., $y_B = (y_{i_1}, \dots, y_{i_{|B|}})$ where the order of the components of the vector y_B is defined by the natural total order on $[m]$. As noted above, we assume that a set $\mathcal{B} \subseteq 2^{[m]}$ of blocks of variables is given and an initial feasible solution $y = y^1$ is available (i.e., $f(y^1) < \infty$). BCD in each iteration chooses a single block $B \in \mathcal{B}$ and minimizes the function f over variables y_B while keeping the remaining variables $y_{[m]-B}$ fixed, i.e., updates y^k to some y^{k+1} satisfying[9]

$$y_B^{k+1} \in \operatorname*{argmin}_{y' \in \mathbb{R}^B} f(y', y^k_{[m]-B}) \tag{8a}$$

$$y_i^{k+1} = y_i^k \qquad \forall i \in [m] - B. \tag{8b}$$

By repeating updates (8) with different blocks $B \in \mathcal{B}$, the points y^k remain feasible for the dual and the sequence of objective values $b^T y^k$ is non-increasing.

The set of block-optimal solutions, $\operatorname{argmin}_{y'_B \in \mathbb{R}^B} f(y'_B, y_{[m]-B}) \subseteq \mathbb{R}^B$, is a non-empty convex polyhedron. If this polyhedron contains more than one point, we need to choose a single point of this polyhedron. It was recently argued [48, 47] that this point should be chosen from the relative interior of the polyhedron, which was called the *relative-interior rule*. Thus, the update condition (8) has to be modified as

$$y_B^{k+1} \in \operatorname{ri} \operatorname*{argmin}_{y' \in \mathbb{R}^B} f(y', y^k_{[m]-B}) \tag{9a}$$

$$y_i^{k+1} = y_i^k \qquad \forall i \in [m] - B. \tag{9b}$$

As the relative interior of a non-empty convex set is always non-empty (see Section 2.2), an update satisfying (9) is always possible.

Definition 4 ([48, 47])**.** *A point y feasible to the dual* (4) *is*

- *a* local minimum[10] *(LM) of the dual w.r.t. $\mathcal{B}$ if*

$$y_B \in \operatorname*{argmin}_{y' \in \mathbb{R}^B} f(y', y_{[m]-B}) \tag{10}$$

  (i.e., (8) *for $y^{k+1} = y^k = y$) holds for all $B \in \mathcal{B}$,*

- *an* interior local minimum *(ILM) of the dual w.r.t. $\mathcal{B}$ if*

$$y_B \in \operatorname{ri} \operatorname*{argmin}_{y' \in \mathbb{R}^B} f(y', y_{[m]-B}) \tag{11}$$

  (i.e., (9) *for $y^{k+1} = y^k = y$) holds for all $B \in \mathcal{B}$,*

- *a* pre-interior local minimum *(pre-ILM) of the dual w.r.t. $\mathcal{B}$ if there is an ILM y' such that y is in a face of the polyhedron $\{\, y \in \mathbb{R}^m \mid A^T y \geq c \}$ containing y' in its relative interior. Since the number of faces is finite in our case, this means that the set of pre-ILMs is the closure of the set of ILMs.*

Clearly, every ILM is a pre-ILM and every pre-ILM is an LM [48, 47]. The fixed points of BCD algorithm following the updates (8) are local minima and fixed points of BCD with the relative-interior rule (9) are interior local minima. We will not use the definition of pre-ILM explicitly but instead rely on its properties, given by the following theorem.

[9] Note the notation abuse in (8a): $(y', y^k_{[m]-B}) \in \mathbb{R}^{[m]}$ denotes the concatenation of the components of the vectors $y' \in \mathbb{R}^B$ and $y^k_{[m]-B} \in \mathbb{R}^{[m]-B}$ in the right order. E.g., for $m = 5$ and $B = \{2, 3\}$, we have $y' = (y'_2, y'_3)$, $y^k_{[m]-B} = (y^k_1, y^k_4, y^k_5)$, and $(y', y^k_{[m]-B}) = (y^k_1, y'_2, y'_3, y^k_4, y^k_5)$.

[10] We emphasise that this is different from the usual notion of a local minimum in optimization: here (by Definition 4), the objective in a local minimum cannot be improved by any single update (8) instead of an arbitrary update within some neighborhood.

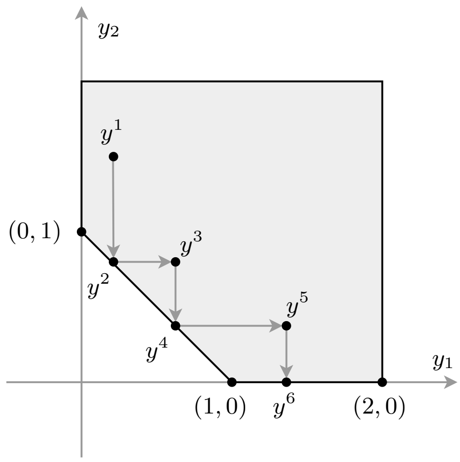

Figure 1: Five iterations of coordinate descent with the relative-interior rule starting from the initial point y^1. We minimize the objective function $b^T y = y_2$ over the shaded feasible set.

Theorem 11 ([48, 47])**.** *Let $(B_i)_{i=1}^\infty$ be a sequence of blocks $B_i \in \mathcal{B}$ that contains each element of $\mathcal{B}$ an infinite number of times. Let $(y^i)_{i=1}^\infty$ be a sequence produced by the BCD method, where the blocks are visited in the order given by $(B_i)_{i=1}^\infty$.*

(a) If $(y^i)_{i=1}^\infty$ satisfies (9) *and y^1 is an ILM, then y^i is an ILM for all i.*

(b) If $(y^i)_{i=1}^\infty$ satisfies (9) *and y^1 is a pre-ILM, then y^i is an ILM for some i.*

(c) If $(y^i)_{i=1}^\infty$ satisfies (8) *and y^1 is a pre-ILM, then $b^T y^i = b^T y^1$ for all i.*

(d) If $(y^i)_{i=1}^\infty$ satisfies (9) *and y^1 is not a pre-ILM, then $b^T y^i < b^T y^1$ for some i.*

Therefore, if y^1 is a pre-ILM, the objective cannot be improved by any further BCD iterations (8), even with the relative-interior rule (9). On the other hand, if y^1 is not a pre-ILM, BCD with the relative-interior rule (9) inevitably improves the objective after a finite number of iterations. In this sense, pre-ILMs are the 'best possible' kind of fixed points of BCD (we mentioned this already in Footnote 1). It also follows that the relative-interior rule is not worse (in this sense) than any other rule to choose non-unique block-optimal solutions in BCD.

We illustrate Definition 4 and Theorem 11 on the following example, adapted from [48, 47, 14].

Example 1. *Let the dual* (4) *minimize the objective $b^T y = y_2$ over variables $y = (y_1, y_2) \in \mathbb{R}^2$ subject to $0 \le y_1 \le 2$, $0 \le y_2 \le 2$, and $y_1 + y_2 \ge 1$, see Figure 1. Let the set of blocks be $\mathcal{B} = \{\{1\}, \{2\}\}$, which corresponds to coordinate descent since both $B \in \mathcal{B}$ are singleton sets. We will iterate update* (9)*, alternating between $B = \{1\}$ and $B = \{2\}$. The initial point $y^1 \in \mathbb{R}^2$ is shown in the figure.*

First we take $B = \{2\}$, i.e., we minimize y_2 over the line segment that is the intersection of the feasible set and the vertical straight line passing through y^1. The argmin-set in (9) *is a singleton, hence taking its relative interior does not change it (thus, we would get the same result using update* (8)*). This results in point y^2 and the objective decreases. Point y^2 is an LM (w.r.t. $\mathcal{B}$) because both of its components are block-wise optimal.*

Next we take $B = \{1\}$, i.e., we minimize y_2 over the horizontal line segment passing through y^2. Since the objective is constant horizontally, it cannot be decreased and the argmin-set is the whole line segment. Note that rule (8) *would allow us to choose any point of this line segment as the next point y^3, e.g., we could set $y^3 = y^2$ and stop the algorithm. However, the relative-interior rule* (9) *forces us to choose y^3 in the interior of the line segment, as shown in the figure. This does not decrease the objective but obtains the 'room' to decrease the objective in the next iteration.*

Indeed, taking now $B = \{2\}$ allows us to move to point y^4 and decrease the objective. The next two updates move us to point y^6, which is an ILM (hence also a pre-ILM) and, in this particular example, also a global minimum. All future updates will stay in the relative interior of the line segment $[(1,0),(2,0)]$.

The set of LMs (w.r.t. $\mathcal{B}$) is $[(0,1),(1,0)] \cup [(1,0),(2,0)]$, the set of pre-ILMs is $[(1,0),(2,0)]$ and the set of ILMs is $\operatorname{ri}[(1,0),(2,0)]$. The global minima coincide with the pre-ILMs in this example (however, this is not true in general).

Remark 2. *One can ask how to choose the block set $\mathcal{B}$. Though the theory presented here applies to any set $\mathcal{B}$, for a concrete problem at hand it is natural to choose $\mathcal{B}$ so that the updates* (8) *or* (9) *can be computed in closed-form, resulting in efficient algorithms [20, 40, 47, 18]. If this is impossible (e.g., the blocks are too large), one can use general LP solvers. In particular, interior-point methods can be used to compute the updates* (9) *since they return a point from the relative interior of the set of optimal solutions.*

Remark 3. *In* local search methods*, one seeks to improve a current feasible solution by choosing a next solution with a better objective from a certain neighborhood. Thus, BCD can be seen as a local search where, for a block $B \subseteq [m]$, the neighborhood of y is the set $\{y' \in \mathbb{R}^m \mid A^T y' \geq c, \forall i \in [m] - B : y_i = y'_i\}$. Since this set is infinite in general, BCD can be seen as very large-scale neighborhood search [35] and choosing different blocks B is analogous to variable neighborhood descent [24]. This view is arguable, however, because one usually speaks about local search only in discrete optimization, while BCD is usually applied to continuous problems.*

3.2 Primal-Dual Approach

Let us now focus on the latter of the two approaches to approximately solve linear programs, primal-dual approach with constraint propagation, which we proposed in [15]. Given a dual-feasible solution y, the complementary slackness conditions for primal-dual pair (4) can be written in terms of the primal variables x as

$$Ax = b \tag{12a}$$
$$x_j = 0 \qquad \forall j \in [n] - \tau(y) \tag{12b}$$
$$x_j \geq 0 \qquad \forall j \in \tau(y) \tag{12c}$$

where $\tau(y)$ is defined by (5).

By Theorem 8, system (12) is feasible if and only if y is dual optimal. Furthermore, by Farkas' lemma [31, Section 6.4], system (12) is feasible if and only if Farkas' alternative system

$$b^T \bar{y} < 0 \tag{13a}$$
$$A_j^T \bar{y} \geq 0 \qquad \forall j \in \tau(y) \tag{13b}$$

is infeasible. Moreover, any $\bar{y}$ satisfying (13) is a dual-improving direction from y, so one can update $y \leftarrow y + \theta\bar{y}$ for a suitable step size $\theta > 0$ and improve the dual objective.

If the set of dual-feasible solutions is unbounded in direction $\bar{y}$, choosing an arbitrary $\theta > 0$ keeps the point $y + \theta\bar{y}$ dual-feasible and pushes the dual objective arbitrarily low. This happens if and only if $A_j^T \bar{y} \geq 0$ for all $j \in [n]$. In such a case, the dual is unbounded and the primal is infeasible (which contradicts our assumptions from the beginning of Section 3).

Assuming that the dual is bounded in direction $\bar{y}$ (i.e., $A_j^T \bar{y} < 0$ for at least one $j \in [n]$), Proposition 12 shows how to compute θ such that the updated point y remains feasible and the dual objective improves.

Proposition 12 (cf. [34, Theorem 5.2])**.** *Let y be feasible for the dual* (4)*, let $\bar{y}$ satisfy* (13) *and $\exists j \in [n]\colon A_j^T \bar{y} < 0$, and let*

$$\theta = \min_{\substack{j \in [n] \\ A_j^T \bar{y} < 0}} \frac{c_j - A_j^T y}{A_j^T \bar{y}} > 0. \tag{14}$$

Then $y' = y + \theta\bar{y}$ is feasible for the dual (4) *and $b^T y > b^T y'$.*

Proof. First, note that $\theta > 0$ which follows from the fact that if $A_j^T \bar{y} < 0$, then $j \notin \tau(y)$ by (13b), hence $c_j - A_j^T y < 0$ by definition of $\tau(y)$ in (5b). This together with $A_j^T \bar{y} < 0$ in (14) implies that each term in the minimum (14) is positive.

To prove feasibility of y', i.e., $A_j^T y' \geq c_j$ for all $j \in [n]$, we consider two cases. If $A_j^T \bar{y} \geq 0$, then $A_j^T y' = A_j^T y + \theta A_j^T \bar{y} \geq A_j^T y \geq c_j$ where we used $\theta > 0$ and the assumption that y is feasible. If $A_j^T \bar{y} < 0$, then $\theta \leq (c_j - A_j^T y)/A_j^T \bar{y}$ by definition of θ, which is equivalent to $A_j^T y' = A_j^T y + \theta A_j^T \bar{y} \geq c_j$. Also, $b^T y' = b^T y + \theta b^T \bar{y} < b^T y$ due to (13a) and $\theta > 0$. $\square$

It is easy to see that the step size (14) is optimal in the sense that it provides the greatest possible improvement of the dual objective along the direction $\bar{y}$ from the current point y.

Algorithm 2 Primal-dual approach for approximate optimization of the dual (4).

input: dual-feasible solution y, propagation rules for (12).

while constraint propagation detects infeasibility of system (12) **do**
    Find an improving direction $\bar{y}$ satisfying (13).
    Update $y \leftarrow y + \theta\bar{y}$ where θ is defined by (14).
return y (At this point, we are unable to prove that (12) is infeasible.)

Following [15], these properties can be used to iteratively optimize a linear program if an initial dual-feasible solution y is provided. We construct system (12) and if it is infeasible, we find an improving direction $\bar{y}$ satisfying (13) and update $y \leftarrow y + \theta\bar{y}$ using (14). By repeating this iteration, we obtain a better and better upper bound on the common optimal value of (4). Eventually, if (12) becomes feasible, the current y is optimal for the dual. This optimization scheme is related to the primal-dual method [34, Section 5], where complementary slackness is not enforced strictly but its violation is minimized instead.

Since checking feasibility of a system of linear equalities and inequalities is in general as hard as solving a linear program, we proposed in [15] to detect infeasibility of (12) using constraint propagation. However, some forms of constraint propagation may not detect infeasibility each time when the system is infeasible (i.e., they are refutation-incomplete), so the approach may generally terminate in a non-optimal dual solution [15]. Moreover, the step size may not be calculated exactly but only approximately. A general scheme of the primal-dual approach with exact line search is outlined in Algorithm 2.

Remark 4. *Unlike the primal-dual method, Algorithm 2 need not terminate after a finite number of iterations in general. Indeed, it is known that the VAC / Augmenting DAG algorithm need not terminate in finite time – to fix this, a version of capacity scaling has been proposed [9, 28, 44]. We propose a similar modification and state sufficient conditions for finiteness of Algorithm 2 in [14, Section 2.2.1]. Similarly, the BCD methods* (8) *and* (9) *also need not converge after a finite number of updates. However, one easily obtains fixed points of BCD and of Algorithm 2 (with the aforementioned modification) in practice up to machine precision. We do not discuss convergence in more detail because it is not needed for this paper.*

3.2.1 Activity Propagation in a System of Linear Inequalities

Constraint propagation can be seen as a special kind of inference: in each iteration, new valid constraints are inferred from the current set of constraints using a *propagation rule*, and added to the current constraint set (which is usually done implicitly by replacing some of the initial constraints). Whenever the current constraint set becomes infeasible, the initial constraint set is clearly also infeasible. For a system of linear inequalities and equalities over continuous variables, the inferred constraints are again linear inequalities or equalities. While in [15] we discussed this process in full generality, here we focus on the following natural propagation rule:

> *Choose a subset of the inequalities and equalities, infer which inequalities are always active (see Definition 3) in this subset, and turn them into equalities.*[11]

We call constraint propagation with this rule *activity propagation*. It is our key observation in this paper that the primal-dual approach with this particular form of constraint propagation has the same fixed points as BCD, when both are applied to a linear program.

Activity propagation, as defined above, is applicable to a system of linear constraints in any form, including both inequality and equality constraints over non-negative or real-valued variables. The connection with BCD holds even in these more general cases but further in this section we focus on the particular case of system (12). We will comment on the other forms later in Section 5.

Remark 5. *Activity propagation has been used, under various names, in several existing methods. One example is the VAC algorithm [9] and the Augmenting DAG algorithm [28, 44], where the primal problem* (4) *is the basic LP relaxation of the weighted CSP and activity propagation is equivalent to the arc-consistency algorithm [15]. The approach proposed in [15] to upper-bound the LP relaxation of weighted Max-SAT is (up to technical details) another example. If the minimization of a convex piecewise-affine function is expressed as a linear program, then activity propagation strictly subsumes the sign relaxation technique introduced in [46] and further developed in [12].*

[11] E.g., turning the inequality $x_1 \leq 1$ into equality means changing it to $x_1 = 1$. Clearly, this can be seen as adding the inequality $x_1 \geq 1$ to the system, obtaining thus the system $x_1 \leq 1$, $x_1 \geq 1$.

Activity propagation, when applied to system (12), works as follows. We first initialize $J = \tau(y)$. Then, in every iteration, we choose a subset $B \in \mathcal{B}$ of equalities (12a) (where $\mathcal{B} \subseteq 2^{[m]}$ was introduced at the beginning of Section 3), decide if the system

$$
\begin{aligned}
A^i x &= b_i && \forall i \in B && \text{(15a)}\\
x_j &= 0 && \forall j \in [n] - J && \text{(15b)}\\
x_j &\geq 0 && \forall j \in J && \text{(15c)}
\end{aligned}
$$

implies[12] $x_j = 0$ for some $j \in J$ and if so, we remove all such indices j from J. Eventually, if the set J shrinks so much that system (15) becomes infeasible, then system (12) is also infeasible, so y is not optimal and can be improved.

In the rest of this section, we analyze activity propagation in detail. We show that the activity propagator associated with each $B \in \mathcal{B}$ is a (dual) closure operator and activity propagation can be seen as chaotic applications of these operators.

Definition 5. *Let $B \subseteq [m]$. A set $J \subseteq [n]$ is* B-consistent *if system* (15) *is feasible and does not imply $x_j = 0$ for any $j \in J$ (i.e., does not contain any always-active inequality), i.e., if the system*

$$
\begin{aligned}
A^i x &= b_i && \forall i \in B && \text{(16a)}\\
x_j &= 0 && \forall j \in [n] - J && \text{(16b)}\\
x_j &> 0 && \forall j \in J && \text{(16c)}
\end{aligned}
$$

is feasible. For $\mathcal{B} \subseteq 2^{[m]}$, J is $\mathcal{B}$-consistent if it is B-consistent for each $B \in \mathcal{B}$.

Proposition 13. *Let $B \subseteq [m]$. If $J, J' \subseteq [n]$ are B-consistent, so is $J \cup J'$.*

Proof. If (16) is satisfied by x and x' for J and J', respectively, then it is satisfied by $(x+x')/2$ for $J \cup J'$ due to $(x_j + x'_j)/2 > 0 \iff (x_j > 0 \vee x'_j > 0)$ which follows from non-negativity of the components of x and x'. ☐

Proposition 13 says that for a fixed $B \subseteq [m]$, the collection of all B-consistent subsets of $[n]$ is closed under union. This is a natural property of many local consistencies, called 'stability' under union in [5, Definition 3.17]. In other words, this collection is a join-semilattice with respect to the ordering by the set inclusion and its join is the set union. However, it is not a (complete) lattice as it need not have a bottom element.

In order to overcome this, we add the bottom element $\perp$ to this collection by the lifting operation (recall Section 2.1.1). We then equip the set $\mathcal{J} = 2^{[n]} \cup \{\perp\}$ with the partial order $\sqsubseteq$ defined by

$$
J \sqsubseteq J' \iff (J = \perp \vee (J, J' \subseteq [n] \wedge J \subseteq J')) \tag{17}
$$

where $\vee$ and $\wedge$ denotes here the logical disjunction and conjunction, respectively, and $J, J' \in \mathcal{J}$. Consequently, for any $B \subseteq [m]$, the set

$$
\mathcal{J}_B = \{J \subseteq [n] \mid J \text{ is } B\text{-consistent}\} \cup \{\perp\} \subseteq \mathcal{J}, \tag{18}
$$

partially ordered by $\sqsubseteq$, is a complete lattice by Theorem 1. The join operation of this lattice is the binary operation $\sqcup$ on $\mathcal{J}$ defined by[13]

$$
J \sqcup J' = \begin{cases} J & \text{if } J' \sqsubseteq J \\ J' & \text{if } J \sqsubseteq J' \\ J \cup J' & \text{otherwise} \end{cases}. \tag{19}
$$

According to Theorem 3b, the complete lattice $(\mathcal{J}_B, \sqsubseteq)$ gives rise to the dual closure operator $p_B \colon \mathcal{J} \to \mathcal{J}_B$ defined by

$$
p_B(J) = \bigsqcup \{J' \in \mathcal{J}_B \mid J' \sqsubseteq J\}. \tag{20}
$$

The set $p_B(J)$ is the *B-consistency closure* of $J \in \mathcal{J}$. Note that the assumptions of Theorem 3b are satisfied because $\mathcal{J}_B \subseteq \mathcal{J}$ and the complete lattices $(\mathcal{J}_B, \sqsubseteq)$ and $(\mathcal{J}, \sqsubseteq)$ have the same join operation, namely $\sqcup$.

[12] Recalling Definition 3, system (15) implies $x_j = 0$ if $x_j = 0$ holds for all x satisfying (15), i.e., $x_j = 0$ is an always-active inequality in (15). This can be decided, e.g., by Theorem 10 or by projecting polyhedron (15) onto the j-th coordinate (the projection is the singleton set $\{0\}$ if and only if (15) implies $x_j = 0$). Alternatively, one can maximize x_j subject to (15) and the maximum equals 0 if and only if (15) implies $x_j = 0$.

[13] Recall that $\perp \sqsubseteq J$ for any $J \in \mathcal{J}$, so if the elements J, J' are not comparable by $\sqsubseteq$, they are subsets of $[n]$, hence set union in the last case in (19) is well-defined.

Proposition 14. *A set $J \subseteq [n]$ is B-consistent if and only if $p_B(J) = J$.*

Proof. Apply Corollary 1 to the complete lattices $(\mathcal{J}, \sqsubseteq)$ and $(\mathcal{J}_B, \sqsubseteq)$ and the associated dual closure operator p_B. Note that $\perp$ is not a subset of $[n]$, so it is not B-consistent despite that $p_B(\perp) = \perp$. □

Note, combining Proposition 14 with Proposition 2 yields a different characterization of the set $\mathcal{J}_B$, namely

$$\mathcal{J}_B = \{J \in \mathcal{J} \mid p_B(J) = J\} = \operatorname{im} p_B. \tag{21}$$

Next, for any $J \subseteq [n]$ and $B \subseteq [m]$, we define

$$X_B(J) = \{x \in \mathbb{R}^n \mid x \text{ satisfies (15)}\}. \tag{22}$$

It is easy to see from (15) that if $J \subseteq J'$, then $X_B(J) \subseteq X_B(J')$. We show in Appendix B that the restriction of X_B to B-consistent sets is a bijection between B-consistent sets and non-empty faces of the polyhedron $X_B([n])$ and that the face lattice of $X_B([n])$ is order-isomorphic to the lattice $(\mathcal{J}_B, \sqsubseteq)$.

Theorem 15. *Let $J \subseteq [n]$ and $B \subseteq [m]$. The following are equivalent:*

(a) system (15) *is feasible,*

(b) $p_B(J) \neq \perp$,

(c) $p_B(J)$ is the union of all B-consistent subsets of J, i.e., $p_B(J)$ is the greatest B-consistent subset of J,

(d) $p_B(J) = J - \{\, j \in J \mid$ (15) *implies $x_j = 0 \,\}$, i.e., system* (15) *implies $x_j = 0$ if and only if $j \in [n] - p_B(J)$.*

If these statements hold, then $p_B(J) = [n] - \sigma(x^*)$ for every $x^* \in \operatorname{ri} X_B(J)$, where σ was defined in (5a)*.*

Proof. (a) $\Longrightarrow$ (b): If (15) is feasible, then

$$J' = J - \{\, j \in J \mid (15) \text{ implies } x_j = 0 \,\} \tag{23}$$

is B-consistent by definition. Also, $J' \sqsubseteq J$, hence $p_B(J) \neq \perp$ due to $\perp \sqcup J' = J'$.

(b) $\Longrightarrow$ (c): If $p_B(J) \neq \perp$, $p_B(J) = \bigcup\{J' \in \mathcal{J}_B \mid J' \subseteq J\}$ because $\perp$ is the identity element of $\sqcup$ and operation $\sqcup$ coincides with $\cup$ when applied to subsets of $[n]$.

(c) $\Longrightarrow$ (a): Since $p_B(J) \neq \perp$, intensivity of p_B implies $p_B(J) \subseteq J$, hence $X_B(p_B(J)) \subseteq X_B(J)$. By definition of a B-consistent set, system (15) is feasible for $p_B(J)$, i.e., $X_B(p_B(J)) \neq \emptyset$. Consequently, $X_B(J) \neq \emptyset$, i.e., (15) is feasible.

(c) $\Longrightarrow$ (d) by contradiction: Let J' be defined as in (23) and $p_B(J) \neq J'$. As discussed above, J' is B-consistent and $J' \subseteq J$, so $p_B(J) \supsetneq J'$. Consequently, there exists $j \in p_B(J) - J'$. By definition of J', (15) implies $x_j = 0$, i.e., $x_j = 0$ holds for all $x \in X_B(J) = X_B(J')$. Moreover, $p_B(J) \subseteq J$ implies $X_B(p_B(J)) \subseteq X_B(J)$, thus $x_j = 0$ holds for all $x \in X_B(p_B(J))$. Having $j \in p_B(J)$ is contradictory with $p_B(J)$ being B-consistent (recall Proposition 14).

(d) $\Longrightarrow$ (b): Trivial because (23) is not equal to $\perp$.

The last claim follows from (d) by applying Theorem 10 to system (15). □

Remark 6. *Even though this is not important for our theory, Theorem 15 shows how to compute p_B in practice. In detail, it suffices to check feasibility of* (15) *and, if* (15) *is feasible, compute a relative-interior point of its solution set (or identify all always-active inequalities, recall Footnote 12). In general, both steps can be reduced to a linear program. For the latter step, see, e.g., [19, 32]. However, if the blocks are small and the problem has a suitable structure, it is possible to solve these problems in closed form without using a general-purpose LP solver (see, e.g., [15] or [12]).*

By statement (d) of Theorem 15, the map p_B can be interpreted as the *propagator* for activity propagation. Note that p_B satisfies the typical properties of constraint propagators, i.e., idempotence, intensivity, and isotony [3]. Moreover, p_B holds on to the intuitive idea of a propagator: it makes an inference based on local information (namely, a subset B of equalities (12a)). Using this propagator, we formulate the *propagation algorithm* in Algorithm 3, which enforces $\mathcal{B}$-consistency of an input set $J \in \mathcal{J}$.

Since the set $\mathcal{J}$ is finite and propagator p_B for each $B \in \mathcal{B}$ is intensive and isotone, Algorithm 3 is an instance of Algorithm 1, describing chaotic iterations. By Theorem 5, the value returned by Algorithm 3 is independent of the order in which the mappings p_B are applied. Thus, we can denote the value returned by Algorithm 3 as $p_{\mathcal{B}}(J)$. For any $J \in \mathcal{J}$, $p_{\mathcal{B}}(J)$ is the greatest common fixed point of the mappings $p_B, B \in \mathcal{B}$ such that $p_{\mathcal{B}}(J) \sqsubseteq J$. By Theorem 6, the mapping $p_{\mathcal{B}} \colon \mathcal{J} \to \mathcal{J}$ is a dual closure operator and $p_{\mathcal{B}}(J)$ is the *$\mathcal{B}$-consistency closure* of $J \in \mathcal{J}$.

Algorithm 3 Propagation algorithm $p_{\mathcal{B}}$ applied to input $J \in \mathcal{J}$.

input: $J \in \mathcal{J}$

$J' \leftarrow J$

while $\exists B \in \mathcal{B} : p_B(J') \neq J'$ **do**

  Find such a set B.

  $J' \leftarrow p_B(J')$

return J'

Proposition 16. *A set $J \subseteq [n]$ is $\mathcal{B}$-consistent if and only if $p_{\mathcal{B}}(J) = J$.*

Proof. Consider the following chain of equivalences:

$$\begin{aligned} J \text{ is } \mathcal{B}\text{-consistent} &\iff \forall B \in \mathcal{B} \colon J \text{ is } B\text{-consistent} && \text{(24a)} \\ &\iff \forall B \in \mathcal{B} \colon p_B(J) = J && \text{(24b)} \\ &\iff p_{\mathcal{B}}(J) = J && \text{(24c)} \end{aligned}$$

where (24a) is given by Definition 5 and (24b) follows from Proposition 14. Equivalence (24c) follows from the definition of $p_{\mathcal{B}}$ in Algorithm 3. ☐

In analogy to (18), we can define the set

$$\mathcal{J}_{\mathcal{B}} = \{J \subseteq [n] \mid J \text{ is } \mathcal{B}\text{-consistent}\} \cup \{\bot\} \tag{25}$$

which satisfies

$$\mathcal{J}_{\mathcal{B}} = \{J \in \mathcal{J} \mid p_{\mathcal{B}}(J) = J\} = \operatorname{im} p_{\mathcal{B}} = \bigcap_{B \in \mathcal{B}} \operatorname{im} p_B = \bigcap_{B \in \mathcal{B}} \mathcal{J}_B \tag{26}$$

where the set equalities follow from Proposition 16, Proposition 2, Remark 1, and (21), respectively. Additionally, Theorem 4 yields that $(\mathcal{J}_{\mathcal{B}}, \sqsubseteq)$ is the complete lattice induced by $p_{\mathcal{B}}$. The join operation of this complete lattice is again $\sqcup$.

Remark 7. *The properties of p_B and $p_{\mathcal{B}}$ are analogous to the properties of, e.g., the arc-consistency propagator and arc-consistency closure, respectively. In more general view, the propagator resembles domain-based constraint propagation [5] and $p_{\mathcal{B}}$ corresponds to Φ-closure in [5].*

Example 2. *Let $m = 3$, $n = 4$, $J = \{2, 3, 4\}$, $\mathcal{B} = \{\{1, 2\}, \{3\}\}$, and the system $Ax = b$ from* (12a) *be*

$$\begin{aligned} 3x_1 + x_2 &= 1 && \text{(27a)} \\ x_2 + 2x_3 &= 1 && \text{(27b)} \\ -2x_2 + 5x_3 - x_4 &= 0 && \text{(27c)} \end{aligned}$$

where the equalities are numbered 1–3 from top to bottom. Recall that $x_2, x_3, x_4 \geq 0$ due to (12b) *and $x_1 = 0$ due to* (12c) *(note that $1 \notin J$). We now demonstrate the run of Algorithm 3.*

First, see that $p_{\{3\}}(J) = J$ because equality (27c) *can be easily satisfied by, e.g., $x_2 = x_3 = 1$ and $x_4 = 3$. So, this equality does not (together with non-negativity of x_2, x_3, x_4) imply that any of these variables are zero and $p_{\{3\}}$ is therefore not applied.*

Second, we apply propagator $p_{\{1,2\}}$. From (27a)*, it follows that $x_2 = 1$ due to $x_1 = 0$. Combining this with* (27b) *yields that $x_3 = 0$. Hence, equalities* (27a)*-*(27b) *together with $x_1 = 0$ imply $x_3 = 0$, i.e., $p_{\{1,2\}}(J) = \{2, 4\}$ and x_3 is set to zero (via* (12c) *and $3 \notin p_{\{1,2\}}(J)$).*[14]

Third, we return to propagator $p_{\{3\}}$. Due to $x_3 = 0$, (27c) *can be satisfied only with $x_2 = x_4 = 0$. In other words,* (27c) *together with $x_3 = 0$ and non-negativity of x_2 and x_4 implies $x_2 = x_4 = 0$, i.e., all variables are now zero and $p_{\{3\}}(\{2, 4\}) = \emptyset$.*

Finally, we again apply propagator $p_{\{1,2\}}$ and find that equalities (27a)*-*(27b) *cannot be satisfied if all variables are zero. Hence, the propagation algorithm detected a contradiction, i.e., $p_{\{1,2\}}(\emptyset) = \bot$ and $p_{\mathcal{B}}(J) = \bot$.*[15]

[14]Note, we cannot set $x_2 = 1$ because there is no inequality $x_2 \leq 1$ (or $x_2 \geq 1$) in system (12) that could be made active. We say more on this in Section 5.3.

[15]In general, it does not necessarily hold that $p_B(\emptyset) = \bot$. E.g., if $b = 0$ (i.e., system (16a) is homogeneous), then (16) is feasible even with $J = \emptyset$ and $p_B(\emptyset) = \emptyset \neq \bot$.

Algorithm 4 Algorithmic scheme for approximate optimization of the dual (4) using activity propagation.

input: dual-feasible solution y

1: **while** $p_{\mathcal{B}}(\tau(y)) = \perp$ **do**

2: Find an improving direction $\bar{y}$ satisfying (13).

3: Update $y \leftarrow y + \theta\bar{y}$ where θ is (14).

4: **return** y

Let us recall system (12). If $p_{\mathcal{B}}(\tau(y)) = \perp$, then (12) is infeasible. This follows from the fact that if a subsystem (15) (with $J = \tau(y)$) implies $x_j = 0$ for some $j \in J$, then also the whole system (12) implies $x_j = 0$. Furthermore, if a subsystem (15) is infeasible, then so is (12). On the other hand, $p_{\mathcal{B}}(\tau(y)) \neq \perp$ in general does not imply that (12) is feasible.

As discussed earlier, if (12) is infeasible, there exists an improving direction $\bar{y}$ satisfying (13). We note that such an improving direction can be constructed from the history of the propagation, but we postpone this technical procedure to Appendix A as it is not essential to characterize the fixed points of this method.

In Algorithm 4 (which is a special case of Algorithm 2), we show how activity propagation can be applied to iteratively improve a feasible solution of the dual (4). Note, Algorithm 4 is precisely Algorithm 2 where the constraint propagation method is activity propagation.

4 Relation Between the Approaches

In this section, we will show a close connection between Algorithm 4 (based on constraint propagation) and BCD. In detail, we will prove that the stopping points of Algorithm 4 are the pre-ILMs of the dual (4) w.r.t. $\mathcal{B}$ and that the ILMs of the dual (4) w.r.t. $\mathcal{B}$ are the points y for which the set $\tau(y)$ is $\mathcal{B}$-consistent. Then it follows from Theorem 11 that BCD with the relative-interior rule (and, consequently, any BCD method) cannot improve the objective when initialized in any stopping point of Algorithm 4. Vice versa, Algorithm 4 cannot improve the objective when initialized in any pre-ILM.

We will proceed as follows. We begin in Section 4.1 by identifying a connection between the actions of propagators and BCD updates. Using these results, we characterize the kinds of local minima in BCD using local consistency conditions in Section 4.2. In turn, this allows us to state the relation between optimality of BCD and strength of the propagator in Section 4.3.

4.1 Connection Between Propagators and BCD Updates

For a fixed set $B \subseteq [m]$, the dual (4) restricted to block of variables y_B, together with the corresponding primal on the left, reads[16]

$$\begin{aligned}
&\max d^T x && \min \sum_{i\in B} b_i y_i && && \text{(28a)}\\
&A^i x = b_i && y_i \in \mathbb{R} && \forall i \in B && \text{(28b)}\\
&x_j \geq 0 && \sum_{i\in B} A_{ij} y_i \geq d_j && \forall j \in [n] && \text{(28c)}
\end{aligned}$$

where $d_j = c_j - \sum_{i\in[m]-B} A_{ij} y_i$ are constants and A_{ij} is the element of A in row i and column j. Clearly $\sum_{i\in B} A_{ij} y_i = d_j$ if and only if $j \in \tau(y)$, hence the complementary slackness conditions for the pair (28) (i.e., the conditions for block-optimality of y_B) are (15) where $J = \tau(y)$. Similarly, the strict complementary slackness conditions for (28) are (16) where $J = \tau(y)$. A consequence of this observation is stated in Lemma 2.

Lemma 2. *Let y be feasible for the dual* (4) *and let $B \subseteq [m]$. Then*

(a) block of variables y_B satisfies (10) *if and only if $p_B(\tau(y)) \neq \perp$,*

(b) block of variables y_B satisfies (11) *if and only if $p_B(\tau(y)) = \tau(y)$, i.e., $\tau(y)$ is B-consistent.*

Proof. (a): Clearly, y_B is optimal for the dual (28) if and only if there exists x feasible for the primal (28) satisfying complementary slackness conditions. The complementary slackness conditions are equivalent

[16] The primal in the pair (28) is of course different from the primal in the pair (4).

to (15) for $J = \tau(y)$. By statements (a) and (b) in Theorem 15, (15) for $J = \tau(y)$ is feasible if and only if $p_B(\tau(y)) \neq \bot$.

(b): Block y_B satisfies (11) if and only if there exists an optimal solution x for the primal problem (28) satisfying strict complementary slackness conditions with y_B. This is equivalent to feasibility of (16) for $J = \tau(y)$, i.e., to B-consistency of $\tau(y)$, hence $p_B(\tau(y)) = \tau(y)$ by Proposition 14. $\square$

Let us now focus on the relation between the propagators p_B and block-coordinate updates (9).

Lemma 3. *Let y be feasible for the dual (4) and let $B \subseteq [m]$. Let y^* be the result of applying BCD iteration (9) to y w.r.t. block B, i.e., $y^* = (y^*_B, y_{[m]-B})$ where $y^*_B \in \operatorname{ri} \operatorname{argmin}_{y' \in \mathbb{R}^B} f(y', y_{[m]-B})$. Then*

(a) $p_B(\tau(y)) = \bot$ if and only if $b^T y > b^T y^*$,

(b) if $p_B(\tau(y)) \neq \bot$, then $b^T y = b^T y^*$ and $p_B(\tau(y)) = \tau(y^*)$.

Proof. It follows from Lemma 2a that $p_B(\tau(y)) = \bot$ holds if and only if condition (10) was not satisfied, i.e., y_B was not block-optimal and updating it results in improved objective, i.e., $b^T y > b^T y^*$. On the other hand, if (10) was satisfied, y_B was block-optimal and objective does not improve, i.e., $p_B(\tau(y)) \neq \bot$ implies $b^T y = b^T y^*$.

For the remaining statement, suppose that y_B was block-optimal, i.e., $p_B(\tau(y)) \neq \bot$ by Lemma 2a. Since any optimal solution to the primal (28) needs to satisfy complementary slackness with any dual-optimal y_B, the set of primal-optimal solutions coincides with x feasible for (15) where $J = \tau(y)$. Let x^* be from the relative interior of (15) for $J = \tau(y)$, i.e., from the relative interior of the set of optimal solutions of the primal (28). By the last statement in Theorem 15 together with feasibility of (15) for $J = \tau(y)$, we have that $p_B(\tau(y)) = [n] - \sigma(x^*) = \tau(y^*)$ where the second equality holds by strict complementary slackness (Theorem 9). $\square$

Remark 8. *Lemmas 2 and 3 provide a new insight into BCD updates with the relative-interior rule. Let y be dual-feasible. In a single update of y over a block $B \in \mathcal{B}$ to obtain y^*, exactly one of the following cases happens:*

(a) If y_B already satisfies condition (11), then $\tau(y) = \tau(y^*)$ and $b^T y = b^T y^*$.

(b) If y_B satisfies (10) but not (11), then $\tau(y) \supsetneq \tau(y^*)$ and $b^T y = b^T y^*$.

(c) If y_B does not satisfy (10), then $b^T y > b^T y^*$.

Suppose we try to improve y by BCD updates (9) where the sequence $(B_k)_{k=1}^{\infty}$ is such that each $B \in \mathcal{B}$ occurs in it an infinite number of times. Since the set $\tau(y)$ can shrink only a finite number of times, case (b) can happen only a finite number of times in a row. Hence, when applying relative-interior updates (9) for consecutive $B \in \mathcal{B}$ from the sequence, either the objective $b^T y$ improves after a finite number of iterations or an ILM w.r.t. $\mathcal{B}$ is attained (cf. Theorem 11).

Case (b) corresponds to the situation when y_B is on the relative boundary of the set of block-optimal solutions (i.e., it is block-optimal but not in the relative interior of the set of block-optima). By an update satisfying the relative-interior rule, such y_B is moved to the relative interior and the set $\tau(y)$ shrinks. Geometrically, even if the current y is block-optimal, choosing a block-optimal solution from the relative interior shifts the current solution to a face of higher dimension, thus providing 'more room' for improvement in subsequent iterations [47] (recall Example 1). Given our point of view with propagators, it corresponds to identifying all always-active inequalities in the system (15). On the other hand, if a block-optimal y_B is kept on the relative boundary, this corresponds to a propagator that cannot identify all always-active inequalities, but possibly only some of them.

Remark 9. *Lemma 3 implies that, to compute the value of the propagator $p_B(J)$ where $J = \tau(y)$, one can also (besides the options mentioned in Remark 6) compute a relative-interior solution of the dual (28). Assuming that (15) is feasible, it suffices to know the indices of dual constraints active at the aforementioned relative-interior solution to compute $p_B(\tau(y))$. Although this seems to suggest that computing $p_B(\tau(y))$ is simpler than computing a relative-interior update (9), we explain in Appendix A that one still needs a certain dual relative-interior solution to compute the improving direction for the primal-dual approach with activity propagation. In summary, the complexity of computing a BCD update from y satisfying the relative-interior rule and computing $p_B(\tau(y))$ is similar in practice, which is not surprising.*

The connection between the propagators and BCD for multiple consecutive iterations is given by the following theorem.

Theorem 17. *Let y^1 be a feasible point for the dual* (4)*. Let $(B_k)_{k=1}^{\infty}$ be a sequence of blocks from $\mathcal{B}$. Let $(y^k)_{k=1}^{\infty}$ be a sequence satisfying* (9) *w.r.t. blocks $(B_k)_{k=1}^{\infty}$. Let $(J_k)_{k=1}^{\infty}$ be the sequence defined by $J_1 = \tau(y^1)$ and $J_{k+1} = p_{B_k}(J_k)$ for all $k \geq 1$. Then, for every k it holds that:*

(a) if $J_k = \perp$, then $b^T y^k < b^T y^1$,

(b) if $J_k \neq \perp$, then $b^T y^k = b^T y^1$ and $\tau(y^k) = J_k$.

Proof. We proceed by induction. The base case with $k = 1$ holds by definition due to $b^T y^1 = b^T y^1$ and $J_1 = \tau(y^1) \neq \perp$. For the inductive step with $k \geq 1$, y^{k+1} originated from y^k by updating block B_k and $J_{k+1} = p_{B_k}(J_k)$. We consider the following cases:

- If $p_{B_k}(\tau(y^k)) \neq \perp$ and $J_k \neq \perp$, we have that $b^T y^k = b^T y^{k+1}$ and $p_{B_k}(\tau(y^k)) = \tau(y^{k+1})$ by Lemma 3b. By induction hypothesis, due to $J_k \neq \perp$, we have $b^T y^1 = b^T y^k$ and $\tau(y^k) = J_k$. Therefore, $\tau(y^{k+1}) = p_{B_k}(\tau(y^k)) = p_{B_k}(J_k) = J_{k+1} \neq \perp$ and $b^T y^1 = b^T y^k = b^T y^{k+1}$.
- If $p_{B_k}(\tau(y^k)) = \perp$ and $J_k \neq \perp$, then by Lemma 3a, $b^T y^k > b^T y^{k+1}$. Similarly to the previous case, by induction hypothesis: since $J_k \neq \perp$, it holds that $b^T y^1 = b^T y^k$ and $\tau(y^k) = J_k$. Thus, $p_{B_k}(\tau(y^k)) = p_{B_k}(J_k) = J_{k+1} = \perp$ and $b^T y^1 = b^T y^k > b^T y^{k+1}$.
- If $J_k = \perp$, by induction hypothesis $b^T y^k < b^T y^1$ and by isotony of propagator, $J_{k+1} = p_{B_k}(J_k) = p_{B_k}(\perp) = \perp$. Since updates (9) never worsen the objective, $b^T y^1 > b^T y^k \geq b^T y^{k+1}$. ☐

Example 3. *Let the matrix A, vector b and set $\mathcal{B}$ be as in Example 2. In addition, let $c = (3, 6, 6, 0)$, so that the dual* (4) *reads*

$$
\begin{aligned}
\min \quad & y_1 + y_2 && && (29a)\\
& 3y_1 && \geq 3 && (29b)\\
& y_1 + y_2 - 2y_3 && \geq 6 && (29c)\\
& 2y_2 + 5y_3 && \geq 6 && (29d)\\
& -y_3 && \geq 0 && (29e)
\end{aligned}
$$

where the constraints (29b)-(29e) *correspond to the primal variables x_1-x_4.*

For $y^1 = (3, 3, 0)$, the set of active dual constraints is $J = \tau(y^1) = \{2, 3, 4\}$ (which is the same initial set J as in Example 2). We now initialize BCD at y^1. As given by the theorems above, the set of active dual constraints after each update of block B will be the same as if the propagator p_B was applied (to be precise, this holds until the propagator returns $\perp$). Moreover, the propagator returns $\perp$ if and only if the corresponding BCD update improves the objective.

For $B = \{3\}$, y^1 is block-optimal and also in the relative interior of the set of block-optimal solutions which is in correspondence to $p_{\{3\}}(\tau(y^1)) = \tau(y^1)$ (cf. Lemma 2b for y^1 and $B = \{3\}$).

Next, y^1 is block-optimal for $B = \{1, 2\}$ but not in the relative interior of the set of block-optimal solutions. Updating the values of $y^1_{\{1,2\}}$ to the relative interior of the set of block-optimal solutions results in, e.g., $y^2 = (2, 4, 0)$. Although the relative interior contains multiple elements, we will have $p_{\{1,2\}}(\tau(y^1)) = \tau(y^2) = \{2, 4\}$ for any of them (cf. Lemma 3b for y^1 and $B = \{1, 2\}$).

Now, y^2 is block-optimal for $B = \{3\}$ but not in the relative interior of the set of block-optimal solutions. A relative-interior update for this coordinate results in, e.g., $y^3 = (2, 4, -\frac{1}{5})$. So, we have $p_{\{3\}}(\tau(y^2)) = \tau(y^3) = \emptyset$ (cf. Lemma 3b for y^2 and $B = \{3\}$).

Finally, y^3 is not block-optimal for $B = \{1, 2\}$. Updating this block with the relative-interior rule results in, e.g., $y^4 = (2 - \frac{1}{5}, 4 - \frac{1}{5}, -\frac{1}{5})$ and improves the objective from $b^T y^3 = 6$ to $b^T y^4 = 6 - \frac{2}{5}$. Note that $p_{\{1,2\}}(\tau(y^3)) = \perp$ (cf. Lemma 3a for y^3 and $B = \{1, 2\}$).

4.2 Final Characterization

The results from the previous section allow us to characterize all types of (local) minima occurring in BCD in terms of the propagation algorithm.

Theorem 18. *Let y be feasible for the dual* (4)*. Then*

(a) y is an LM of the dual (4) *w.r.t. $\mathcal{B}$ if and only if $p_B(\tau(y)) \neq \perp$ for all $B \in \mathcal{B}$,*

(b) y is an ILM of the dual (4) *w.r.t. $\mathcal{B}$ if and only if $p_{\mathcal{B}}(\tau(y)) = \tau(y)$, i.e., $\tau(y)$ is $\mathcal{B}$-consistent,*

(c) y is an optimal solution of the dual (4) *if and only if $p_{[m]}(\tau(y)) \neq \perp$,*

(d) y is in the relative interior of the set of optimal solutions of the dual (4) *if and only if $\tau(y)$ is $[m]$-consistent,*

(e) y is a pre-ILM of dual (4) *w.r.t. $\mathcal{B}$ if and only if $p_{\mathcal{B}}(\tau(y)) \neq \bot$.*

Proof. (a): By definition, y is an LM of dual (4) w.r.t. $\mathcal{B}$ if (10) holds for all $B \in \mathcal{B}$. Applying Lemma 2a, this is equivalent to $p_B(\tau(y)) \neq \bot$ for all $B \in \mathcal{B}$.

(b): Analogous, except that we use Lemma 2b and recall Proposition 16.

(c): Dual optimality of y is equivalent to (10) with $B = [m]$. The claim now follows from Lemma 2a.

(d): Similar, by Lemma 2b.

(e): Let $(B_k)_{k=1}^{l}$ be a finite sequence of blocks $B_k \in \mathcal{B}$ such that

$$p_{B_l}(p_{B_{l-1}}(\cdots p_{B_2}(p_{B_1}(\tau(y)))\cdots)) = p_{\mathcal{B}}(\tau(y)). \tag{30}$$

Note, $(B_k)_{k=1}^{l}$ can be obtained, e.g., by storing the individual blocks as they were applied in Algorithm 3. Let us extend this finite sequence into an infinite sequence $(B_k)_{k=1}^{\infty}$ so that $(B_k)_{k=1}^{\infty}$ contains each element of $\mathcal{B}$ an infinite number of times. Next, define sequences $(J_k)_{k=1}^{\infty}$ and $(y^k)_{k=1}^{\infty}$ based on the sequence $(B_k)_{k=1}^{\infty}$ and $y^1 = y$ as in Theorem 17.

If $J_{l+1} = p_{\mathcal{B}}(\tau(y)) = \bot$, Theorem 17a yields $b^T y^1 > b^T y^{l+1}$. This together with Theorem 11c implies that $y = y^1$ is not a pre-ILM because updates (9) cannot improve the objective from a pre-ILM.

On the other hand, if $J_{l+1} = p_{\mathcal{B}}(\tau(y)) \neq \bot$, then $p_B(J_{l+1}) = J_{l+1} \neq \bot$ for all $B \in \mathcal{B}$, hence $J_k \neq \bot$ for all k, so $b^T y^1 = b^T y^k$ by Theorem 17b. Combining this with Theorem 11d yields that $y = y^1$ is a pre-ILM. $\square$

Corollary 2 summarizes and connects the results given by Theorem 18.

Corollary 2. *Let y be a feasible point for the dual* (4)*. The following implications and equivalences hold (for better readability, equivalent statements are boxed in gray):*

$$\begin{array}{ccc}
y \text{ is an LM of dual (4) w.r.t. } \mathcal{B} & \Leftarrow & y \text{ is an ILM of dual (4) w.r.t. } \mathcal{B} \\
\Updownarrow & & \Updownarrow \\
p_B(\tau(y)) \neq \bot \text{ for all } B \in \mathcal{B} & & p_{\mathcal{B}}(\tau(y)) = \tau(y) \\
\Uparrow & & \Updownarrow \\
y \text{ is a pre-ILM of dual (4) w.r.t. } \mathcal{B} & \Leftarrow & \tau(y) \text{ is } \mathcal{B}\text{-consistent} \\
\Updownarrow & & \Uparrow \\
p_{\mathcal{B}}(\tau(y)) \neq \bot & \Leftarrow & y \text{ is in the relative interior of the set of optimal solutions of dual (4)} \\
\Uparrow & & \Updownarrow \\
y \text{ is an optimal solution of dual (4)} & & p_{[m]}(\tau(y)) = \tau(y) \\
\Updownarrow & & \Updownarrow \\
p_{[m]}(\tau(y)) \neq \bot & \Leftarrow & \tau(y) \text{ is } [m]\text{-consistent}
\end{array}$$

Proof. The equivalences follow from Theorem 18. To prove the implications, it suffices to show for any $J \subseteq [n]$ that

$$\begin{array}{ccccc}
p_{[m]}(J) = J & \overset{\text{(a)}}{\Longrightarrow} & p_{\mathcal{B}}(J) = J & & \\
\text{(b)}\Downarrow & & \text{(c)}\Downarrow & & \\
p_{[m]}(J) \neq \bot & \overset{\text{(d)}}{\Longrightarrow} & p_{\mathcal{B}}(J) \neq \bot & \overset{\text{(e)}}{\Longrightarrow} & \forall B \in \mathcal{B}\colon p_B(J) \neq \bot.
\end{array} \tag{31}$$

(a): $p_{[m]}(J) = J$ is equivalent to J being $[m]$-consistent by Proposition 14, i.e., (16) is feasible for $B = [m]$ by definition. This clearly implies feasibility of (16) for any $B \subseteq [m]$ as then (16) contains only a subset of all the equalities, so J is $\mathcal{B}$-consistent which is equivalent to $p_{\mathcal{B}}(J) = J$ by Proposition 16.

(b) and (c): Trivial due to $J \neq \bot$.

(d): Denote $p_{[m]}(J) = J' \neq \bot$, so (16) is feasible for J' and $B = [m]$ because J' is $[m]$-consistent by Proposition 14. As in the previous case, this implies $p_{\mathcal{B}}(J') = J'$. By (24), J' is a common fixed point of propagators p_B, $B \in \mathcal{B}$ and a subset of J. As discussed in Section 3.2.1, $p_{\mathcal{B}}(J)$ is the greatest common fixed point of these propagators such that $p_{\mathcal{B}}(J) \sqsubseteq J$, so $\bot \neq J' \sqsubseteq p_{\mathcal{B}}(J)$. Due to $\bot \sqsubseteq J'$, we have $p_{\mathcal{B}}(J) \neq \bot$.

(e): By contrapositive: if $p_B(J) = \bot$ for some $B \in \mathcal{B}$, then the propagation algorithm (Algorithm 3) terminates with $p_{\mathcal{B}}(J) = \bot$. $\square$

4.3 BCD Optimality in Terms of Propagation Strength

As a consequence of the obtained results, we are able to characterize linear programs exactly solvable by BCD. Precisely, we characterize linear programs where (pre-)ILMs are optima of the dual (4) in terms of refutation-completeness (i.e., the ability to always detect infeasibility) of activity propagation.

Corollary 3. *Let $\mathcal{B} \subseteq 2^{[m]}$. The following are equivalent:*

(a) every ILM y of the dual (4) *w.r.t. $\mathcal{B}$ is a global minimum of the dual* (4)*,*

(b) every pre-ILM y of the dual (4) *w.r.t. $\mathcal{B}$ is a global minimum of the dual* (4)*,*

(c) for all y feasible for the dual (4) *we have $p_{[m]}(\tau(y)) = \bot \implies p_{\mathcal{B}}(\tau(y)) = \bot$, i.e., if* (12) *is infeasible, then the propagation algorithm detects it ($p_{\mathcal{B}}(\tau(y)) = \bot$),*

(d) for all y feasible for the dual (4)*, if $\tau(y)$ is $\mathcal{B}$-consistent, then* (12) *is feasible.*

Moreover, if these statements hold, then the set of global minima coincides with the set of pre-ILMs of the dual (4) *w.r.t. $\mathcal{B}$.*

Proof. Since every ILM is also a pre-ILM, we have (b) $\implies$ (a). To show (a) $\implies$ (b), let y^1 be a pre-ILM. By Theorem 11b and 11c, after performing a finite number of relative-interior updates (9) from y, we attain an ILM with the same objective.

Theorem 18b yields (a) $\iff$ (d). The contrapositive of statement (c) is $p_{\mathcal{B}}(\tau(y)) \neq \bot \implies p_{[m]}(\tau(y)) \neq \bot$. The equivalence (b) $\iff$ (c) is now immediate from the implications and equivalences summarized in Corollary 2.

One inclusion in the last statement follows already from (b). We prove the remaining part by contradiction: if a global minimum is not a pre-ILM, then, by Theorem 11d, the objective must improve after a finite number of updates, which is impossible. □

See that statements (a) and (b) in Corollary 3 state that the BCD fixed points are global optima whereas statement (c) means that the propagation algorithm is able to detect infeasibility of any infeasible system.

As this result was already stated in the conference version of this paper [17], it led to a simplification of the proof in [16] where a class of linear programs solvable by BCD was identified. This simplified proof was, along with a newly identified class of such linear programs, given in [18]. Recently, we used Corollary 3 to prove optimality of BCD on yet another class of linear programs in [14, Section 5.2.1].

5 Other Forms of Linear Programs

Linear programs can come in various forms [34, Section 2.1] which can be easily transformed to each other, preserving global optima. One can ask if the connection between constraint propagation and BCD holds also for different forms than (4). This question is non-trivial because transformations that preserve global optima do not necessarily preserve (pre-)ILMs [18]. We show that if we use activity propagation as defined at the beginning of Section 3.2.1, the two approaches remain equivalent independently of the formulation.

5.1 Primal with Inequalities and Non-negative Variables

For example, consider the primal-dual pair

$$
\begin{aligned}
\max c^T x &\qquad& \min b^T y &\qquad (32a)\\
Ax \le b &\qquad& y \ge 0 &\qquad (32b)\\
x \ge 0 &\qquad& A^T y \ge c. &\qquad (32c)
\end{aligned}
$$

The primal problem (32) (on the left-hand side) can be equivalently reformulated [34, Section 2.1][31, Section 4.1] by introducing non-negative slack variables $s_i \ge 0$, $i \in [m]$ (where m is the number of rows of A) which yields the primal-dual pair

$$
\begin{aligned}
\max c^T x &\qquad& \min b^T y &\qquad (33a)\\
Ax + s = b &\qquad& y \in \mathbb{R}^m &\qquad (33b)\\
x \ge 0 &\qquad& A^T y \ge c &\qquad (33c)\\
s \ge 0 &\qquad& y \ge 0 &\qquad (33d)
\end{aligned}
$$

which is in the form (4). See that the duals (32) and (33) are identical, hence also BCD applied to them is identical.

Activity propagation for the case of (33) corresponds to deciding which s_i and x_j are implied to be zero. Clearly, setting $s_i = 0$ corresponds to setting $A^i x = b_i$ and enforcing $s_i > 0$ is equivalent to $A^i x < b_i$. Thus, instead of rewriting (32) into (33), we can apply propagation directly to the primal (32) except that when considering system (15) for some $B \in \mathcal{B}$, we will instead of a single set J use two sets, $J_1 \subseteq [m]$ and $J_2 \subseteq [n]$, that indicate which of the original inequalities need to hold with equality, i.e., we will use

$$
\begin{aligned}
A^i x &\le b_i && \forall i \in B \cap J_1 && \text{(34a)}\\
A^i x &= b_i && \forall i \in B \cap ([m] - J_1) && \text{(34b)}\\
x_j &\ge 0 && \forall j \in J_2 && \text{(34c)}\\
x_j &= 0 && \forall j \in [n] - J_2 && \text{(34d)}
\end{aligned}
$$

instead of (15). Deciding which inequalities among (34a) in (34) are always active[17] by considering a set $J_1 \subseteq [m]$ is in one-to-one correspondence with deciding which inequalities $s_i \ge 0$ in

$$
\begin{aligned}
A^i x + s_i &= b_i && \forall i \in B && \text{(35a)}\\
s_i &\ge 0 && \forall i \in J_1 && \text{(35b)}\\
s_i &= 0 && \forall i \in [m] - J_1 && \text{(35c)}\\
x_j &\ge 0 && \forall j \in J_2 && \text{(35d)}\\
x_j &= 0 && \forall j \in [n] - J_2 && \text{(35e)}
\end{aligned}
$$

are always active. In particular, (34) contains an always-active inequality if and only if (35) contains an always-active inequality.

5.2 Primal with Inequalities and Unconstrained Variables

The second common form of a primal-dual pair is

$$
\begin{aligned}
\max c^T x &\qquad& \min b^T y && \text{(36a)}\\
Ax \le b &\qquad& y \ge 0 && \text{(36b)}\\
x \in \mathbb{R}^n &\qquad& A^T y = c. && \text{(36c)}
\end{aligned}
$$

By complementary slackness, y is optimal for the dual if and only if there exists $x \in \mathbb{R}^n$ such that

$$
\begin{aligned}
A^i x &\le b_i && \forall i \in \sigma'(y) && \text{(37a)}\\
A^i x &= b_i && \forall i \in [m] - \sigma'(y) && \text{(37b)}
\end{aligned}
$$

where $\sigma'(y) = \{i \in [m] \mid y_i = 0\}$ (in analogy to (5a)).

From this point, we could completely repeat the reasoning from Section 3.2.1 and prove the same theorems as in Section 4. We would infer from

$$
\begin{aligned}
A^i x &\le b_i && \forall i \in B \cap \sigma'(y) && \text{(38a)}\\
A^i x &= b_i && \forall i \in B \cap ([m] - \sigma'(y)) && \text{(38b)}
\end{aligned}
$$

whether some of the inequalities (38a) are always active. As an example, for any y feasible for the dual (36): y is an ILM of the dual (36) w.r.t. $\mathcal{B}$ if and only if for each $B \in \mathcal{B}$, (38) is feasible and no inequality from (38a) is always active in the system (38).

5.3 Redundant Constraints

It was observed in [18] that adding redundant constraints into a linear program has significant influence on its solvability by BCD. Using our results, we are able to explain this quite naturally.

[17] In general and analogously to Definition 3, the inequality $C^i x \le d_i$ is always active in the system $Ax = b, Cx \le d$ if $Ax = b, Cx \le d$ implies $C^i x = d_i$.

As an example, consider the following LP relaxation of the weighted vertex cover problem on a graph (V, E) with vertex weights $w : V \to \mathbb{R}^+$ together with its dual

$$\min w^T x \qquad \max \textstyle\sum_{\{i,j\}\in E} y_{ij} \tag{39a}$$
$$x_i + x_j \geq 1 \qquad y_{ij} \geq 0 \qquad \forall \{i,j\} \in E \tag{39b}$$
$$x_i \geq 0 \qquad \textstyle\sum_{j\in N_i} y_{ij} \leq w_i \qquad \forall i \in V \tag{39c}$$

where N_i is the set of neighbors of vertex i in the graph. If we optimize the primal by coordinate descent or the dual by coordinate ascent, there are interior local minima and maxima, respectively, that are not global optima [18, 14]. However, if we add redundant constraints $x \leq 1$ to the primal, we obtain

$$\min w^T x \qquad \max \textstyle\sum_{\{i,j\}\in E} y_{ij} + \sum_{i\in V} z_i \tag{40a}$$
$$x_i + x_j \geq 1 \qquad y_{ij} \geq 0 \qquad \forall \{i,j\} \in E \tag{40b}$$
$$x_i \geq 0 \qquad z_i + \textstyle\sum_{j\in N_i} y_{ij} \leq w_i \qquad \forall i \in V \tag{40c}$$
$$x_i \leq 1 \qquad z_i \leq 0 \qquad \forall i \in V. \tag{40d}$$

Note that, at optimum of the dual, we have $z_i = \min\{w_i - \sum_{j\in N_i} y_{ij}, 0\}$ for each $i \in V$, so variables z can be eliminated and the dual simplified to

$$\max_{y\geq 0} \Big(\sum_{\{i,j\}\in E} y_{ij} + \sum_{i\in V} \min\{w_i - \sum_{j\in N_i} y_{ij}, 0\} \Big). \tag{41}$$

We showed [18] that coordinate ascent in (41) is equivalent to block-coordinate ascent of the dual (40) w.r.t. $|E|$ blocks, each consisting of three variables y_{ij}, z_i, z_j where $\{i,j\} \in E$, and that its interior local maxima are global maxima.

The explanation for the difference between (non-)optimality for the two formulations lies in the fact that in case of (39), we can only propagate equality in primal constraints (39b) and $x_i = 0$. However, in (40), we are also able to propagate $x_i = 1$ due to the added constraint $x_i \leq 1$. This results in a stronger propagation algorithm which is even refutation-complete in this case.

In [18, Section 4.3] we observed a similar phenomenon for the LP formulation of min-*st*-cut and its dual, maximum flow. Adding redundant bounds $0 \leq x_i \leq 1$ on variables in min-*st*-cut results in global optimality of coordinate ascent for the dual (rewritten to a form analogous to (41)). In contrast, the dual of the usual LP formulation of min-*st*-cut (i.e., without these bounds) is not amenable to coordinate ascent as it is not even possible to change any single dual variable while staying within the feasible set. This difference is now explained by the possibility of the underlying propagation algorithm to set the primal variables to their bounds, i.e., set $x_i = 0$ or $x_i = 1$ which is not possible if variables x are unbounded.

Our results therefore shed light on which constraints are useful in terms of propagation or BCD even though they are redundant from the point of global optimality.

6 Conclusion

Even though propagation in a system of linear inequalities can be performed in many ways, we have proposed a propagation algorithm which not only has natural and useful properties but also allows a full characterization of types of local minima in BCD. Additionally, there is a tight connection between the fixed points of BCD with relative-interior rule (or any BCD method whose fixed points are the 'best possible', i.e., pre-ILMs) and the fixed points of the primal-dual approach based on this propagation algorithm. Despite the fact that both algorithms may not reach a global optimum, none of the algorithms can improve the fixed points of the other.

We argued that the propagation algorithm can be generalized to linear programs in any form. In detail, BCD in the dual for a given set of blocks $\mathcal{B}$ corresponds to propagating which primal constraints given by complementary slackness should be active and which inactive while inferring only from subsets of the constraints given by sets in $\mathcal{B}$.

We believe that our findings are interesting for the theory of BCD as they explain what kind of local consistency is reached by any BCD algorithm (both with or without relative-interior rule) on any linear program. E.g., as shown in [47], since both TRW-S [27] and max-sum diffusion [29, 44] satisfy the relative-interior rule, their fixed point conditions are equivalent to the proposed local consistency condition if applied to the specific LP formulations which these algorithms optimize.

The tight connection between the decidability of feasibility of a system of linear inequalities by (generally refutation-incomplete) propagation and BCD (Corollary 3) provides theoretical ground for analysis of BCD in terms of constraint propagation. Identifying in which special cases a certain kind of propagation is refutation-complete (i.e., it is always able to decide feasibility) is of interest in the constraint programming community [8, 10]. Such analysis of activity propagation may lead to a different characterization of linear programs solvable by BCD. Moreover, it may result in better design for choices of blocks of variables so that the propagation is more effective and BCD may reach better stopping points. Since we already stated this result in the previous version of this paper [17], it led to the discovery of new classes of linear programs solvable by BCD [18, 14]. Even though these classes are relatively narrow, it is open whether they are the only ones. This connection also precisely explains the differences in applicability of BCD caused by minor changes in the formulation of the optimized linear program, as discussed in Section 5.3.

The practical impact of these results is mainly focused on approximately optimizing challenging large-scale linear programs which are not solvable by off-the-shelf LP solvers due to their super-linear space complexity. Propagation algorithms subsumed (up to technical details) by the proposed one were previously derived ad-hoc for specific linear programs [9, 15, 28, 46, 12] where they provided useful solutions which were often close to global optima. Presenting all of these algorithms in a single framework may simplify design of similar algorithms in the future.

Acknowledgments: 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 (grants SGS19/170/OHK3/3T/13 and SGS22/061/OHK3/1T/13).

A Computing an Improving Feasible Direction

As discussed at the beginning of Section 3.2, if (12) is infeasible, there exists an improving feasible direction $\bar{y}$ satisfying (13). We describe one way of obtaining such a direction based on the propagation algorithm (Algorithm 3). We remark that conditions (13) define a whole convex cone of improving directions and our algorithm finds one of them based on its precise implementation.

Let y be a dual-feasible point such that $p_{\mathcal{B}}(\tau(y)) = \bot$. This implies infeasibility of (12), i.e., non-optimality of y. Consider sequences $(B_l)_{l=1}^L$ and $(J_l)_{l=1}^L$ where $J_1 \supsetneq J_2 \supsetneq \cdots \supsetneq J_L$, $J_1 = \tau(y)$, $J_{l+1} = p_{B_l}(J_l)$ for every $l \in [L-1]$, and $p_{B_L}(J_L) = \bot$. To construct $\bar{y}$, we use the primal-dual pair

$$
\begin{array}{llll}
\max 0 & \min b^T \hat{y}^l & & \text{(42a)} \\
A^i x = b_i & \hat{y}^l_i \in \mathbb{R} & \forall i \in B_l & \text{(42b)} \\
x_j = 0 & - & \forall j \in [n] - J_l & \text{(42c)} \\
x_j \geq 0 & A_j^T \hat{y}^l \geq 0 & \forall j \in J_l & \text{(42d)} \\
- & \hat{y}^l_i = 0 & \forall i \in [m] - B_l. & \text{(42e)}
\end{array}
$$

and proceed as outlined in Algorithm 5. Note that the primal (42) is a feasibility problem identical to (15) if $J_l = J$ and $B_l = B$. Even though Algorithm 5 may seem complicated, it is easy to see that in cases when the blocks B are small, problem (42) is also small (and thus could even be solvable in closed-form). Correctness of Algorithm 5 is given by the following theorem.

Theorem 19. *Let $J_1 = \tau(y)$. If $p_{\mathcal{B}}(\tau(y)) = \bot$, Algorithm 5 returns a vector $\bar{y}^1$ satisfying* (13).

Proof. We will proceed by induction, i.e., we claim that for each $l \in [L]$, $\bar{y}^l$ satisfies $A_j^T \bar{y}^l \geq 0$ for all $j \in J_l$ and $b^T \bar{y}^l = b^T \bar{y}^L < 0$ is maintained during the whole algorithm. Thus, eventually $A_j^T \bar{y}^1 \geq 0$ holds for all $j \in J_1 = \tau(y)$.

For the base case with $l = L$, primal (42) is infeasible due to $p_{B_L}(J_L) = \bot$ (see Theorem 15) and dual (42) is therefore unbounded since it is always feasible. Thus, there exists $\hat{y}^L$ feasible to the dual (42) that satisfies $b^T \hat{y}^L < 0$. By feasibility, $A_j^T \hat{y}_i^L \geq 0$ for all $j \in J_L$.

For the inductive step, let $l \leq L - 1$. If condition on line 3 is not satisfied, $A_j^T \bar{y}^l \geq 0$ holds for all $j \in J_l$ trivially by setting $\bar{y}^l$ equal to $\bar{y}^{l+1}$ on line 7 due to our inductive hypothesis.

If condition on line 3 is satisfied, let us focus on (42). Since $p_{B_l}(J_l) = J_{l+1} \neq \bot$, primal (42) is feasible with optimal value 0, which is attained by all feasible solutions. Let x^l and $\hat{y}^l$ be in the relative interior of the set of optimal solutions of the primal and dual (42), respectively. Since $x^l, \hat{y}^l$ are from the relative

Algorithm 5 Construction of improving direction

input: sequences of sets $(B_l)_{l=1}^L$ and $(J_l)_{l=1}^L$ satisfying $J_{l+1} = p_{B_l}(J_l)$ for every $l \in [L-1]$, and $p_{B_L}(J_L) = \bot$

1: Set $\bar{y}^L \leftarrow \hat{y}^L$ where $\hat{y}^L$ is feasible for the dual (42) with $l = L$ and $b^T\hat{y}^L < 0$.
2: **for** $l \in \{L-1, L-2, \dots, 2, 1\}$ in descending order **do**
3:     **if** $\exists j \in J_l - J_{l+1}$: $A_j^T\bar{y}^{l+1} < 0$ **then**
4:         Find $\hat{y}^l$ from the relative interior of the optimal solution set of dual (42) for l.
5:         Set $\bar{y}^l \leftarrow \bar{y}^{l+1} + \delta_l\hat{y}^l$ where $\delta_l = \max\limits_{\substack{j \in J_l - J_{l+1} \\ A_j^T\bar{y}^{l+1} < 0}} -A_j^T\bar{y}^{l+1}/A_j^T\hat{y}^l$.
6:     **else**
7:         Set $\bar{y}^l \leftarrow \bar{y}^{l+1}$.
8: **return** $\bar{y}^1$

interior, they satisfy strict complementary slackness (see Theorem 9), i.e., $x_j = 0 \iff A_j^T\hat{y}^l > 0$ for all $i \in J_l$. By the last statement in Theorem 15, $x_j = 0 \wedge A_j^T\hat{y}^l > 0$ holds for all $j \in J_l - J_{l+1}$ because $p_{B_l}(J_l) = J_{l+1}$. For completeness, $x_j > 0 \wedge A_j^T\hat{y}^l = 0$ holds for all $j \in J_{l+1}$.

Notice that δ_l is well-defined because condition on line 3 was satisfied. Moreover, $\delta_l > 0$ due to both $-A_j^T\bar{y}^{l+1}$ and $A_j^T\hat{y}^l$ being positive by definition of δ_l.

We consider the following cases to prove that $A_j^T\bar{y}^l \geq 0$ for all $j \in J_l$:

- If $j \in J_{l+1}$, then $A_j^T\bar{y}^{l+1} \geq 0$ by inductive hypothesis and $A_j^T\hat{y}^l = 0$ by strict complementary slackness, hence $A_j^T\bar{y}^l = A_j^T\bar{y}^{l+1} \geq 0$.
- If $j \in J_l - J_{l+1}$, then $A_j^T\hat{y}^l > 0$. If $A_j^T\bar{y}^{l+1} \geq 0$, then $A_j^T\bar{y}^l = A_j^T\bar{y}^{l+1} + \delta_l A_j^T\hat{y}^l \geq 0$. On the other hand, if $A_j^T\bar{y}^{l+1} < 0$, it holds by definition of δ_l that $\delta_l \geq -A_j^T\bar{y}^{l+1}/A_j^T\hat{y}^l$, which is after a simple reformulation equivalent to $A_j^T\bar{y}^l = A_j^T\bar{y}^{l+1} + \delta_l A_j^T\hat{y}^l \geq 0$.

Finally, it holds by strong duality that $b^T\bar{y}^l = 0$, which yields $b^T\bar{y}^l = b^T\bar{y}^{l+1} + \delta b^T\hat{y}^l = b^T\bar{y}^{l+1} < 0$. ☐

B Faces and B-consistent Sets

In this section, we explain the geometric meaning of B-consistent sets, as defined in Section 3.2.1. In detail, we will show that the set of B-consistent sets is order-isomorphic to the set of non-empty faces of the polyhedron $X_B([n])$ (defined in (22)). Consequently, the lattice $(\mathcal{J}_B, \sqsubseteq)$ (see (18)) is isomorphic to the face lattice of $X_B([n])$.

The faces of a convex polyhedron are usually defined using valid inequalities (or supporting hyperplanes) [50, Section 2.1]. However, faces can be also equivalently obtained by forcing subsets of inequalities to be active [38, Section 5.6]. We use this latter definition. Moreover, we define faces only for the polyhedron $X_B([n])$ (see (22)) where $B \subseteq [m]$ is fixed.

Definition 6. *Let $B \subseteq [m]$. A set $F \subseteq \mathbb{R}^n$ is a face of the polyhedron $X_B([n])$ if $F = \emptyset$ or $F = X_B(J)$ for some $J \subseteq [n]$. The set of all faces of $X_B([n])$ is denoted by*

$$\mathcal{F}_B = \{F \mid F \text{ is a face of } X_B([n])\}. \tag{43}$$

It is immediate that the set of all faces of $X_B([n])$ is finite. Moreover, the set of all faces is closed under intersections, as shown by the following corollary.

Corollary 4 ([50, Proposition 2.3])**.** *Let $B \subseteq [m]$. If $F, F' \in \mathcal{F}_B$, then $F \cap F' \in \mathcal{F}_B$.*

Proof. This is clear if $F = \emptyset$ or $F' = \emptyset$. Otherwise, there are $J, J' \subseteq [n]$ such that $F = X_B(J)$ and $F' = X_B(J')$. Hence, $X_B(J) \cap X_B(J') = X_B(J \cap J') \in \mathcal{F}_B$. ☐

Thus, the face set of the polyhedron $X_B([n])$ forms a finite meet-semilattice w.r.t. the partial order given by set inclusion where the meet operation is set intersection. Moreover, $X_B([n])$ is the top element of this meet-semilattice, so $(\mathcal{F}_B, \subseteq)$ is a complete lattice by Theorem 1. This is known as the *face lattice* [50, 4, 37].

We will now describe the connection between B-consistent sets and the faces of polyhedron $X_B([n])$. Firstly, let us point our attention to the fact that we could require the set J to be B-consistent in Definition 6. We formulate a stronger statement in the following theorem.

Theorem 20. *Let $B \subseteq [m]$. For any non-empty $F \in \mathcal{F}_B$, there exists a unique B-consistent set $J \subseteq [n]$ such that $F = X_B(J)$. Conversely, for any B-consistent set $J \subseteq [n]$, the face $X_B(J)$ is non-empty.*

Proof. For the first part, let us show that at least one such set exists. By definition, since $F \in \mathcal{F}_B$ is non-empty, there exists $J' \subseteq [n]$ such that $F = X_B(J')$. Clearly, we have that $X_B(J') = X_B(p_B(J'))$ and $p_B(J')$ is B-consistent by Theorem 15. To show that this set is unique, let us proceed by contradiction. Let $J_1, J_2 \subseteq [n]$ be B-consistent sets such that $X_B(J_1) = X_B(J_2) = F$ and $J_1 \neq J_2$. Without loss of generality, assume $J_2 - J_1 \neq \emptyset$ and let $j^* \in J_2 - J_1$ be arbitrary. We have that $x_{j^*} = 0$ for all $x \in X_B(J_1)$ due to $j^* \notin J_1$, so J_2 is not B-consistent as (15) for J_2 implies $x_{j^*} = 0$ and $j^* \in J_2$.

For the other part, it is clear that any B-consistent set defines a face of the polyhedron by Definition 6. Moreover, this face is non-empty due to (15) being feasible for any B-consistent set J. □

Following Theorem 20, X_B can be interpreted as a bijection between B-consistent sets and the set of non-empty faces of $X_B([n])$.[18]

As already noted earlier in Section 3.2.1, X_B is an isotone mapping, i.e., if $J \subseteq J' \subseteq [n]$, then $X_B(J) \subseteq X_B(J')$. The converse relation also holds if we restrict ourselves to B-consistent sets:

Proposition 21. *Let $B \subseteq [m]$ and $J, J' \subseteq [n]$ be B-consistent. If $X_B(J) \subseteq X_B(J')$, then $J \subseteq J'$.*

Proof. By contradiction: let $X_B(J) \subseteq X_B(J')$ and $J \nsubseteq J'$. The latter implies that there is $j \in [n]$ such that $j \in J$ and $j \notin J'$. By Definition 5, there exists $x^* \in X_B(J)$ satisfying (16) for J with $x^*_j > 0$. However, by definition of $X_B(J')$ (see (22)), we have $x_j = 0$ for all $x \in X_B(J')$ (due to $j \notin J'$), so $x^* \notin X_B(J')$, which contradicts $X_B(J) \subseteq X_B(J')$. □

It seems natural to extend the mapping X_B to obtain the isotone bijection $X'_B \colon \mathcal{J}_B \to \mathcal{F}_B$ defined by

$$X'_B(J) = \begin{cases} X_B(J) & \text{if } J \subseteq [n] \\ \emptyset & \text{if } J = \bot \end{cases}. \tag{44}$$

Clearly, we have that $J \sqsubseteq J' \iff X'_B(J) \subseteq X'_B(J')$ for any $J, J' \in \mathcal{J}_B$. The lattices $(\mathcal{J}_B, \sqsubseteq)$ and $(\mathcal{F}_B, \subseteq)$ are therefore *order-isomorphic* and X'_B is a *lattice isomorphism* [11].[19]

Remark 10. *In some formalisms, $\emptyset$ does not belong to the face lattice of some polyhedra. To be precise, if $(S, \subseteq)$ is the face lattice of some polyhedron, then $(S - \{\emptyset\}, \subseteq)$ is a lattice if and only if $\bigcap\{F \mid F \in S - \{\emptyset\}\} \neq \emptyset$, i.e., if there is a minimal non-empty face [4, Section 8]. As an example, for the non-negative orthant $X_\emptyset([n]) = \{x \in \mathbb{R}^n \mid x \geq 0\}$, we have $\bigcap\{F \mid F \in \mathcal{F}_\emptyset - \{\emptyset\}\} = \{0\}$ and $(\mathcal{F}_\emptyset - \{\emptyset\}, \subseteq)$ is a lattice where the bottom element is $\{0\}$ (i.e., the singleton set containing the origin).*

Consequently, if $(\mathcal{F}_B - \{\emptyset\}, \subseteq)$ is a lattice, then $(\mathcal{J}_B - \{\bot\}, \sqsubseteq)$ is a lattice too. These lattices are again order-isomorphic and X_B is the lattice isomorphism.

References

[1] I. Adler and R. D. Monteiro. A geometric view of parametric linear programming. *Algorithmica*, 8(1):161–176, 1992.

[2] K. R. Apt. From chaotic iteration to constraint propagation. In *International Colloquium on Automata, Languages, and Programming*, pages 36–55. Springer, 1997.

[3] K. R. Apt. The rough guide to constraint propagation. In *Conf. on Principles and Practice of Constraint Programming*, pages 1–23. Springer, 1999.

[4] A. Bachem and M. Grötschel. *New aspects of polyhedral theory.* Inst. für Ökonometrie und Operations Research, 1980.

[5] C. Bessiere. Constraint propagation. In *Handbook of Constraint Programming*, chapter 3. Elsevier, 2006.

[6] T. Blyth. *Lattices and Ordered Algebraic Structures.* Universitext. Springer London, 2005.

[18] In analogy to [4, Section 4], for a B-consistent set J and $F = X_B(J)$, $[m] - J$ is the *equality set* of F. Additionally, the lattice $(\mathcal{J}_B, \sqsubseteq)$ is similar to the *equality set lattice* of $X_B([n])$ [4, Section 8].

[19] Following on Footnote 18, this result is analogous to the fact that, for a polyhedron M, the face lattice of M is anti-isomorphic to the equality set lattice of M [4, Diagram 8.1].

[7] S. Boyd and L. Vandenberghe. *Convex optimization*. Cambridge university press, 2004.

[8] D. A. Cohen and P. G. Jeavons. The power of propagation: when gac is enough. *Constraints*, 22(1):3–23, 2017.

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

[10] M. C. Cooper and S. Živný. The power of arc consistency for csps defined by partially-ordered forbidden patterns. In *Proceedings of the 31st Annual ACM/IEEE Symposium on Logic in Computer Science*, pages 652–661, 2016.

[11] B. A. Davey and H. A. Priestley. *Introduction to lattices and order*. Cambridge university press, 2002.

[12] T. Dlask. *Minimizing Convex Piecewise-Affine Functions by Local Consistency Techniques*. Master's thesis, Czech Technical University in Prague, Faculty of Electrical Engineering, 2018.

[13] T. Dlask. Unit propagation by means of coordinate-wise minimization. In *International Conference on Machine Learning, Optimization, and Data Science*, pages 688–699. Springer, 2020.

[14] T. Dlask. *Block-Coordinate Descent and Local Consistencies in Linear Programming*. Ph.D. thesis, available online `https://dspace.cvut.cz/handle/10467/102874?locale-attribute=en`, Czech Technical University in Prague, Faculty of Electrical Engineering, 2022.

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

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

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

[18] T. Dlask and T. Werner. Classes of linear programs solvable by coordinate-wise minimization. *Annals of Mathematics and Artificial Intelligence*, 90(7):777–807, 2022.

[19] R. M. Freund, R. Roundy, and M. J. Todd. Identifying the set of always-active constraints in a system of linear inequalities by a single linear program. Technical Report 1674-85, Massachusetts Institute of Technology, Alfred P. Sloan School of Management, 1985.

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

[21] A. J. Goldman and A. W. Tucker. *Theory of Linear Programming*, pages 53–97. Princeton University Press, 1956.

[22] H. J. Greenberg. The use of the optimal partition in a linear programming solution for postoptimal analysis. *Operations Research Letters*, 15(4):179–185, 1994.

[23] H. J. Greenberg. Consistency, redundancy, and implied equalities in linear systems. *Annals of Mathematics and Artificial Intelligence*, 17(1):37–83, 1996.

[24] P. Hansen, N. Mladenović, J. Brimberg, and J. A. M. Pérez. Variable neighborhood search. In *Handbook of metaheuristics*, chapter 3. Springer, 2010.

[25] B. Jansen, C. Roos, T. Terlaky, and J.-P. Vial. Interior-point methodology for linear programming: duality, sensitivity analysis and computational aspects. In *Optimization in Planning and Operation of Electric Power Systems*, pages 57–123. Springer, 1993.

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

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

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

[29] V. Kovalevsky and V. Koval. A diffusion algorithm for decreasing energy of max-sum labeling problem. *Glushkov Institute of Cybernetics, Kiev, USSR*, 1975. Unpublished.

[30] C. Lemaréchal and J.-B. Hiriart-Urruty. *Fundamentals of Convex Analysis.* Springer Grundlehren Text Editions, Springer Verlag, New York, 2004.

[31] J. Matoušek and B. Gärtner. *Understanding and using linear programming.* Springer-Verlag, 2006.

[32] S. Mehrotra and Y. Ye. Finding an interior point in the optimal face of linear programs. *Mathematical Programming*, 62(1):497–515, 1993.

[33] J. B. Nation. Notes on lattice theory. `https://math.hawaii.edu/~jb/math618/Nation-LatticeTheory.pdf`, 1998.

[34] C. H. Papadimitriou and K. Steiglitz. *Combinatorial optimization: algorithms and complexity.* Courier Corporation, 1998.

[35] D. Pisinger and S. Ropke. Large neighborhood search. In *Handbook of metaheuristics*, chapter 13. Springer, 2010.

[36] B. Savchynskyy. Discrete graphical models – an optimization perspective. *Foundations and Trends in Computer Graphics and Vision*, 11(3-4):160–429, 2019.

[37] A. Schrijver. *Theory of linear and integer programming.* John Wiley & Sons, 1998.

[38] A. Schrijver. *Combinatorial optimization: polyhedra and efficiency.* Springer Science & Business Media, 2004.

[39] J. Telgen. Identifying redundant constraints and implicit equalities in systems of linear constraints. *Management Science*, 29(10):1209–1222, 1983.

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

[41] P. Tseng. Convergence of a block coordinate descent method for nondifferentiable minimization. *J. Optim. Theory Appl.*, 109(3):475–494, June 2001.

[42] S. Živný. *The Complexity of Valued Constraint Satisfaction Problems.* Cognitive Technologies. Springer, 2012.

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

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

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

[46] T. Werner. On coordinate minimization of piecewise-affine functions. Technical Report CTU-CMP-2017-05, Department of Cybernetics, Faculty of Electrical Engineering, Czech Technical University in Prague, September 2017.

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

[48] T. Werner and D. Průša. Relative interior rule in block-coordinate minimization. *ArXiv*, 2019.

[49] S. Zhang. On the strictly complementary slackness relation in linear programming. In *Advances in Optimization and Approximation*, pages 347–361. Springer, 1994.

[50] G. M. Ziegler. *Lectures on Polytopes.* Springer-Verlag, New York, 1994.