Polyhedral Diameters and Applications to Optimization

by

Sean Kafer

A thesis
presented to the University of Waterloo
in fulfillment of the
thesis requirement for the degree of
Doctor of Philosophy
in
Combinatorics & Optimization

Waterloo, Ontario, Canada, 2022

Examining Committee Membership

The following served on the Examining Committee for this thesis. The decision of the Examining Committee is by majority vote.

External Examiner: Steffen Borgwardt
Assoc. Professor, Dept. of Mathematical and Statistical Sciences,
University of Colorado Denver

Supervisor: Laura Sanità
Assoc. Professor, Dept. of Mathematics and Computer Science,
Eindhoven University of Technology

Internal Members: Levent Tunçel
Professor, Dept. of Combinatorics & Optimization,
University of Waterloo

Ricardo Fukasawa
Professor, Dept. of Combinatorics & Optimization,
University of Waterloo

Internal-External Member: Rafael Oliveira
Asst. Professor, Cheriton School of Computer Science,
University of Waterloo

Author's Declaration

This thesis consists of material all of which I authored or co-authored: See Statement of Contributions included in the thesis. This is a true copy of the thesis, including any required final revisions, as accepted by my examiners.

I understand that my thesis may be made electronically available to the public.

Statement of Contributions

This thesis contains joint work with Alex Black, Jesús De Loera, and my supervisor Laura Sanità. Details can be found in Section 1.1.

Abstract

The Simplex method is the most popular algorithm for solving linear programs (LPs). Geometrically, it moves from an initial vertex solution to an improving neighboring one, selected according to a *pivot* rule. Despite decades of study, it is still not known whether there exists a pivot rule that makes the Simplex method run in polynomial time.

Circuit-augmentation algorithms are generalizations of the Simplex method, where in each step one is allowed to move along a fixed set of directions, called the circuits, that is a superset of the edges of a polytope. The number of circuit augmentations has been of interest as a proxy for the number of steps in the Simplex method, and the circuit-diameter of polyhedra has been studied as a lower bound to the combinatorial diameter of polyhedra. We show that in the circuit-augmentation framework the Greatest Improvement and Dantzig pivot rules are NP-hard, even for 0/1 LPs. On the other hand, the Steepest Descent pivot rule can be carried out in polynomial time in the 0/1 setting, and the number of circuit augmentations required to reach an optimal solution according to this rule is strongly-polynomial for 0/1 LPs. We introduce a new rule in the circuit-augmentation framework which we call Asymmetric Steepest Descent. We show both that it can be computed in polynomial time and that it reaches an optimal solution of an LP in a polynomial number of augmentations. It was not previously known that such a rule was possible. We further show a weakly-polynomial bound on the circuit diameter of rational polyhedra.

We next show that the circuit-augmentation framework can be exploited to make novel conclusions about the classical Simplex method itself: In particular, as a byproduct of our circuit results, we prove that (i) computing the shortest (monotone) path to an optimal solution on the 1-skeleton of a polytope is NP-hard, and hard to approximate within a factor better than 2, and (ii) for 0/1-LPs (i.e., those whose vertex solutions are in $\{0,1\}^n$), a monotone path of strongly-polynomial length can be constructed using steepest improving edges.

Inspired by this, we further examine the lengths of other monotone paths generated by some local decision rules – which we call *edge rules* – including two modifications of the classical Shadow Vertex pivot rule. We leverage the techniques we use for analyzing edge rules to devise pivot rules for the Simplex method on 0/1-LPs that generate the same monotone paths as their edge rule counterparts. In particular, this shows that there exist pivot rules for the Simplex method on 0/1 LPs that reach an optimal solution by performing only a polynomial number of non-degenerate pivots, answering an open question in the literature.

Acknowledgements

First and most of all, thank you to my supervisor, Laura Sanità. In our seven years working together so far, she has taught me so much and taken great care to ensure my success. Thank you also to Jesús De Loera. I am lucky and grateful to have been able to work with him so consistently throughout my time as a PhD student, and I have learned much from him by doing so. I am grateful to my committee for reading this thesis, for their suggestions, and for a great, insightful discussion.

Thank you to Chaitanya Swamy for helping keep my graduate studies smooth and seamless by being my co-supervisor. I thank Alex Black for his contributions to our work together. Finally, thank you to Evelyne, to my parents, and to my friends for making everything better and everything easier.

Table of Contents

List of Figures

Chapter 1

Introduction

Linear Programs (LPs) are one of the most powerful mathematical tools for tackling optimization problems. While various algorithms have been proposed for solving LPs in the past decades, probably the most popular method remains the *Simplex method*, introduced by G. B. Dantzig in the 1940's. In this thesis, we assume the input LP is given in the general maximization format

$$\max \{ \, \boldsymbol{c}^\intercal \boldsymbol{x} \; : \; A\boldsymbol{x} = \boldsymbol{b}, \, D\boldsymbol{x} \leq \boldsymbol{d}, \, \boldsymbol{x} \in \mathbb{R}^n \, \} \, . \tag{1.1}$$

for integer matrices A and D of sizes $m_A \times n$ and $m_D \times n$ respectively, and integer vectors $\boldsymbol{b}$, $\boldsymbol{d}$, and $\boldsymbol{c}$. Here the objective function is maximized over the polyhedron $P = \{\boldsymbol{x} \in \mathbb{R}^n : A\boldsymbol{x} = \boldsymbol{b}, \, D\boldsymbol{x} \leq \boldsymbol{d}\}$ of feasible solutions.

The Simplex method starts with an initial feasible *basis* of the LP, where informally a basis is a minimal set of tight constraints that uniquely define a vertex solution. In each step, it moves in an improving direction to an *adjacent basis* until an optimal solution is found or unboundedness is detected. Two bases are adjacent if they can be obtained from each other via a single exchange two tight constraints. The choice of which improving direction to select is done according to some *pivot rule*. Geometrically, the Simplex method with a given pivot rule generates a path from an initial vertex solution of an LP to an optimal one by moving along the *edges* – i.e., the 1-dimensional faces of the underlying polyhedron. We define the *1-skeleton* of a polyhedron P to be the graph obtained by taking the vertices and edges of P to be the vertices and edges of the graph. Then the Simplex method generates a path on the 1-skeleton. Furthermore, this path is *monotone* – i.e., the objective function of each vertex on this path is greater than that of the previous vertex.

Despite having been used and studied for more than 70 years, a longstanding open question in the theory of optimization [92] states: *is there a version of the Simplex method (i.e., a choice of pivot rule) that can solve LPs in polynomial time*? The question of finding an efficient pivot rule is a fundamental difficulty in studying the Simplex method.

Due to the fact that it has proven to be exceedingly difficult to study the number of pivots performed by the Simplex method, it is often studied by proxy. Given a polyhedron P, the *combinatorial diameter* (or simply *diameter*) of P is the maximum length of a shortest path between any two vertices on the 1-skeleton of P. Since the Simplex method follows a path on the 1-skeleton of the feasible region P of an LP, the combinatorial diameter is a lower bound on the number of pivots that the Simplex method must perform. In order for there to exist a pivot rule that can solve LPs in polynomial time, it is necessary that there at least exists a polynomial bound on the combinatorial diameter of polyhedra.

The most famous conjecture in this area is the *Hirsch Conjecture* which stated that the diameter of a d-dimensional polyhedron with f facets is at most $f-d$. While this conjecture has been disproved (first for unbounded polyhedra by Klee and Walkup in 1967 [68], and then for bounded ones by Francisco Santos much later in 2012 [83]), it is still an open question whether there exists some polynomial function that bounds the diameter. This is referred to as the *polynomial Hirsch Conjecture* (see e.g. [83]).

Although the study of the diameter is motivated in large part due to its ability to act as a proxy for the performance of the Simplex method, there are significant gaps between these two topics. First, the Simplex method does not follow an arbitrary path on the 1-skeleton, but rather follows a monotone path. Given a polyhedron P and an objective function $\boldsymbol{c}$ to be maximized, the *monotone diameter* of P is the maximum possible length of a shortest monotone path from a vertex to a $\boldsymbol{c}$-maximizer, where the maximum is taken over all possible objective functions $\boldsymbol{c}$ and all possible initial vertices. The monotone diameter is also a lower bound on the number of pivots required by the Simplex method, and the monotone diameter can be substantially larger than the combinatorial diameter. For example, the monotone diameter of the Birkhoff polytope is $n/2$, while it's combinatorial diameter is only 2. However much the diameter is a relevant proxy for the behavior of the Simplex method, the monotone diameter is a strictly more relevant one.

Second, the existence of a short (monotone) path between two vertices does not easily translate to an instance of a polynomial pivot rule. While the monotone diameter tells us the maximum length of a *shortest* monotone path, we show that *computing* a shortest monotone path is NP-hard:

Theorem 1. *Given an LP and an initial feasible solution, finding the shortest (monotone) path to an optimal solution is NP-hard. Furthermore, unless P=NP, it is hard to approxi-*

mate within a factor strictly better than two. This holds even when the feasible region is a 0/1 polytope.

This implies that for *any* efficiently-computable pivot rule, the Simplex method cannot be guaranteed to reach an optimal solution via a minimum number of edge augmentations (i.e., non-degenerate pivots), unless P=NP. Then even if there *is* a polynomial bound on the monotone diameter, in order for the Simplex method to have any hope of having a polynomial pivot rule, it must further be the case that polynomial-length monotone paths can be generated in polynomial time.

Then to bridge this conceptual gap between the monotone diameter and the Simplex method, we formalize the notion of so-called *edge rules*: An edge rule takes as input a vertex solution of an LP and prescribes a way to choose an adjacent improving vertex solution. This is in contrast to a pivot rule for the Simplex method, which takes as input a basis of an LP and prescribes a way to choose an adjacent improving basis. Though related, these concepts differ in the presence of degeneracy, wherein there may be exponentially many adjacent vertices, and where the computational problem of selecting an improving adjacent vertex is not (in general) any easier than solving the original LP itself. Although both edge rules and pivot rules ultimately generate monotone paths on the 1-skeleton of the underlying polyhedron, it is not clear how to construct a pivot rule which generates a *particular* path if it needs to perform degenerate basis exchanges to do so. This suggests the following question:

Question 1. *Given an edge rule, can one define a pivot rule for the Simplex method that always generates the same path as that edge rule?*

In much of this thesis, we will investigate edge rules, pivot rules, and Question 1 in the setting of 0/1-LPs. These are problems of the form $\max\{\boldsymbol{c}^\intercal \boldsymbol{x} : \boldsymbol{x} \in P\}$, where P is a 0/1 polytope, i.e., its vertices have coordinates in $\{0, 1\}$. A polynomial pivot rule is not even known for 0/1-LPs. This state of affairs remains despite the fact that the diameter of a 0/1 polytope is bounded by the dimension of the polytope [74] (an obstacle for general polytopes) and that 0/1-LPs arise frequently from combinatorial optimization problems.

It is known (see [86]) that paths of (strongly) polynomial length in a 0/1 polytope can be constructed using *any* augmentation oracle that yields an improving adjacent vertex in (strongly) polynomial time. However, results of this type do not give insights on the performance of the Simplex method run over 0/1-LPs because (a) using an oracle in this way may require modifying the original objective function, resulting in paths that are not even monotone with respect to the original objective function (for this reason, they do not even yield edge rules), and (b) as discussed above, in the presence of degeneracy a path of polynomial length does not immediately translate into a sequence of basis exchanges.

Degeneracy is indeed a very crucial challenge in the analysis of the Simplex method. Given a general LP, one can often assume that the LP is non-degenerate by, for example, applying perturbation techniques. However, 0/1 polytopes are often highly degenerate and when working with 0/1-LPs, perturbation changes P into a polytope that is no longer 0/1. Thus, perturbation cannot be performed without loss of generality to understand the behavior of the Simplex for the original 0/1-LP. Degeneracy might make the Simplex method *cycle* and hence not terminate [93]. There are several pivot rules that can avoid cycling [93, 92, 72, 91]. However, none of these pivot rules guarantee a polynomial bound on the number of degenerate basis exchanges, and they might even require an exponential number of pivots before moving to a different vertex (this phenomenon is called *stalling*). While cycling may be avoided easily, resolving stalling in polynomial time is as hard as solving the general problem of finding a polynomial pivot rule for the Simplex method (see e.g., [73]). This observation holds even for 0/1-LPs. A first stepping stone in resolving this problem is finding a pivot rule which at least guarantees a polynomial number of non-degenerate pivots. Hence, the following natural question arises as stated for instance in [69]:

Question 2. *Is there a pivot rule for the Simplex method that guarantees a (strongly) polynomial number of non-degenerate pivots on* 0/1*-LPs?*

One of the results of this thesis is to give a complete affirmative answer to this question by simultaneously investigating Question 1. We present three efficiently computable edge rules which generate paths of polynomial length in 0/1-LPs. We call these rules the Steepest Edge, Slim Shadow, and Ordered Shadow rules (We note here that, for the sake of brevity, we may at times refer to an edge rule as simply a *rule* if it is clear from context that we are referring to an edge rule). Then we show that there exist pivot rules for 0/1-LPs which generate the same paths:

Theorem 2. *On any* 0/1*-LP of the form (1.1), the Simplex method with a True Steepest Edge pivot rule reaches an optimal solution by performing a strongly polynomial number of non-degenerate pivots. Furthermore, it generates the same monotone path as the Steepest Edge rule.*

Theorem 3. *On any* 0/1*-LP of the form (1.1), the Simplex method with the Slim Shadow pivot rule reaches an optimal solution by performing no more than n non-degenerate pivots. Furthermore, it generates the same monotone path as the Slim Shadow rule.*

Theorem 4. *On any 0/1-LP of the form (1.1) whose feasible region has dimension d, the Ordered Shadow pivot rule reaches an optimal solution by performing no more than d non-degenerate pivots. Furthermore, it generates the same monotone path as the Ordered Shadow rule.*

The methods we used to arrive at these results are of independent interest: they came as a consequence of thinking about a much more general family of algorithms that includes the Simplex method. *Circuit-augmentation* algorithms are extensions of the Simplex method where we have many more choices of improving directions available at each step – more than just the edges of the polyhedron. Many of our results, like e.g. Theorem 1, are valid in this more general family of algorithms which we now introduce to the reader.

Given a polyhedron, its *circuits* are all potential edges that can arise by translating some of the inequalities in its description (i.e., by varying the right hand side). Circuits are important not just in the development of linear optimization [13, 80], but also they appear very naturally in other areas of application where polyhedra need to be decomposed [70]. Formally:

Definition 1 (See [58])**.** *Given a polyhedron of the form $P = \{\boldsymbol{x} \in \mathbb{R}^n : A\boldsymbol{x} = \boldsymbol{b},\ D\boldsymbol{x} \leq \boldsymbol{d}\}$, a non-zero vector $\boldsymbol{g} \in \mathbb{R}^n$ is a circuit if*

(i) $\boldsymbol{g} \in \ker(A)$, *and*

(ii) $\operatorname{supp}(D\boldsymbol{g})$ *is an inclusion-wise minimal set in the collection*

$$\{\operatorname{supp}(D\boldsymbol{y}) : \boldsymbol{y} \in \ker(A), \boldsymbol{y} \neq \boldsymbol{0}\}.$$

Here $\ker(Z)$ denotes the kernel of the matrix Z and $\operatorname{supp}(\boldsymbol{z})$ denotes the support of the vector $\boldsymbol{z}$. As we will see later, from a geometric perspective the circuits can capture the extreme rays of a cone containing any improving direction, and allow one to represent any augmentation as a sum of at most n circuit-augmentations. To represent the circuits with a finite set, we can normalize them in various ways. Following [19, 25, 37, 49], we denote by $\mathcal{C}(A, D)$ the (finite) set of circuits with co-prime integer components.

Given an initial feasible point of an LP, a circuit-augmentation algorithm at each iteration moves as far as possible along an improving circuit-direction while maintaining feasibility. This is done until an optimal solution is found (or unboundedness is detected). Circuits and circuit-augmentation algorithms have appeared in several papers and books on linear and integer optimization (see [14, 15, 17, 19, 25, 23, 36, 37, 54, 61, 63, 75, 80, 88, 18] and the many references therein). In particular, the authors of [37] considered linear programs in equality form and analyzed in detail three circuit-pivot rules that guarantee notable bounds on the number of steps performed by a circuit-augmentation algorithm to reach an optimal solution.

Given a feasible point $\boldsymbol{x} \in P$, the proposed circuit-pivot rules are as follows:

(i) *Greatest Improvement circuit-pivot rule:* select a circuit $\boldsymbol{g} \in \mathcal{C}(A, D)$ that maximizes the objective improvement $\boldsymbol{c}^\intercal(\alpha\boldsymbol{g})$, among all circuits $\boldsymbol{g}$ and $\alpha \in \mathbb{R}_{>0}$ such that $\boldsymbol{x} + \alpha\boldsymbol{g} \in P$.

(ii) *Dantzig circuit-pivot rule:* select a circuit $\boldsymbol{g} \in \mathcal{C}(A, D)$ that maximizes $\boldsymbol{c}^\intercal\boldsymbol{g}$, among all circuits $\boldsymbol{g}$ such that $\boldsymbol{x} + \varepsilon\boldsymbol{g} \in P$ for some $\varepsilon > 0$.

(iii) *Steepest Descent circuit-pivot rule:* select a circuit $\boldsymbol{g} \in \mathcal{C}(A, D)$ that maximizes $\frac{\boldsymbol{c}^\intercal\boldsymbol{g}}{||\boldsymbol{g}||_1}$, among all circuits $\boldsymbol{g}$ such that $\boldsymbol{x} + \varepsilon\boldsymbol{g} \in P$ for some $\varepsilon > 0$.

Note that these circuit-pivot rules are direct extensions of three famous pivot rules proposed for the Simplex method. Unfortunately, the Simplex method with these pivot rules can require an exponential number of edge steps before reaching an optimal solution [57, 62, 67]. When all circuits are considered as possible directions to move, much better bounds can be given. Most notably, the Greatest Improvement circuit-pivot rule guarantees a *polynomial* bound on the number of steps performed by a circuit-augmentation algorithm on LPs in equality form (see [37, 61, 75] and references therein), a result which we extend to general LPs of the form (1.1). However, the set of circuits in general can have an exponential cardinality, and therefore selecting the best circuit according to the previously mentioned rules is not an easy optimization problem. The authors of [37] leave as an open question the complexity of computing each of the above rules. We investigate this question, as well as the question of how *approximate* solutions to these circuit-pivot rules can be used to design or analyze circuit-augmentation algorithms.

First we settle the computational complexity of the circuit-pivot rules (i) and (ii).

Theorem 5. *The Greatest Improvement and Dantzig circuit-pivot rules are NP-hard.*

We prove this theorem by showing that computing a circuit, according to both the Greatest Improvement and the Dantzig circuit-pivot rule, is already hard to do when P is a 0/1 polytope. In particular, we focus on the case when P is the matching polytope of a bipartite graph. We characterize the circuits of the more general *fractional* matching polytope, i.e., the polytope given by the standard LP-relaxation for the matching problem on general graphs. This builds on the known graphical characterization of adjacency given in [82, 7].

Then, we construct a reduction from the NP-hard Hamiltonian path problem. The heart of the reduction yields the following interesting corollary.

Corollary 1. *Given a feasible vertex solution of the bipartite matching polytope and an objective function, it is NP-hard to decide whether there is a neighbor vertex that is optimal.*

With the above corollary, the hardness result stated in Theorem 1 can be easily derived. Even more, combining this corollary with the characterization of circuits for the fractional matching polytope, we can show that the hardness result of Theorem 1 holds more generally for *circuit*-paths, i.e., paths constructed by a circuit-augmentation algorithm (see Section 3.1 for related results).

We next make a very useful observation: Any polynomial time γ-approximation algorithm for the Greatest Improvement circuit-pivot rule optimization problems yields an increase of at most a multiplicative γ-factor on the running time of the corresponding circuit-augmentation algorithm – this follows from an extension of the analysis given by [37]. This simple observation turns out to be quite useful, and it plays a key role in our subsequent results. We therefore formally state its main implication in the next lemma.

Lemma 1. *Consider an LP in the general form (1.1). Denote by δ the maximum absolute value of the determinant of any $n \times n$ submatrix of $\binom{A}{D}$. Let $\boldsymbol{x}_0$ be an initial feasible solution, and let $\gamma \geq 1$. Using a γ-approximate Greatest Improvement circuit-pivot rule, we can reach an optimal solution $\boldsymbol{x}^*$ of (1.1) with $\mathcal{O}\big(n\gamma \log\big(\delta\, \boldsymbol{c}^\intercal(\boldsymbol{x}^* - \boldsymbol{x}_0)\big)\big)$ augmentations. Furthermore, if all vertices of the feasible region have integer coordinates, we can reach an optimal solution with $\mathcal{O}\big(n\gamma \log\big(\boldsymbol{c}^\intercal(\boldsymbol{x}^* - \boldsymbol{x}_0)\big)\big)$ augmentations.*

We are able to leverage Lemma 1 to give a new bound on the number of Steepest Descent augmentations required to solve an LP whose feasible region is bounded. In particular, we show that in this setting, a Steepest Descent augmentation is an approximate Greatest Improvement augmentation

Theorem 6. *Consider the LP $\max\{\, \boldsymbol{c}^\intercal \boldsymbol{x} : \boldsymbol{x} \in P \,\}$ of the form (1.1) where P is bounded. Let ω_1 denote the minimum 1-norm distance from any vertex $\boldsymbol{v} \in P$ to any facet F of P such that $\boldsymbol{v} \notin F$. Let M_1 be the maximum 1-norm distance between any pair of vertices of P. Using a Steepest Descent circuit-pivot rule, a circuit-augmentation algorithm reaches an optimal solution $\boldsymbol{x}^*$ from any initial feasible solution $\boldsymbol{x}_0$, performing at most*

$$\mathcal{O}\left(n^2 \frac{M_1}{\omega_1} \log\left(\delta\, \boldsymbol{c}^\intercal(\boldsymbol{x}^* - \boldsymbol{x}_0)\right)\right)$$

augmentations.

The authors of [37] gave bounds on the number of steps taken by such a circuit-augmentation algorithm for LPs in equality form. Later, Borgwardt and Viss [25] extended these results to LPs in general form, however we note that they extend the *definition* of the Steepest Descent circuit-pivot rule to general form LPs in a way that differs from the definition used here. In particular, they define the rule to choose the circuit that maximizes $\boldsymbol{c}^\intercal \boldsymbol{g}/\|D\boldsymbol{g}\|_1$ rather than $\boldsymbol{c}^\intercal \boldsymbol{g}/\|\boldsymbol{g}\|_1$, as we do here. Note that these definitions coincide for LPs in standard equality form. The bounds given by [37] and [25] both depend on the size of the set of circuits (when they are normalized so as to make this set finite) and the number of different values the objective function takes on that set. However, such bounds are a bit opaque, and often difficult to analyze.

Instead, in Theorem 6 we get another type of bound of independent application. Our bound is more comparable to that obtained with the Greatest Improvement circuit-pivot rule, since it depends more explicitly on the input description of the LP. On the other hand, the version of Steepest Descent considered in [25] can be computed in polynomial time (by solving an auxiliary LP), whereas for the definition considered here, we are only able to do so in the special case of 0/1-LPs, as we will discuss later. We also note that, except in special cases, none of the bounds known for the number of Steepest Descent augmentations necessary to solve an LP are polynomial.

Considered together, Theorems 5 and 6 and the results of [37] and [25] suggest the following natural question:

Question 3. *Does there exist a circuit-pivot rule that is both computable in polynomial time and can solve an LP with only a polynomial number of augmentations?*

We provide an affirmative answer by proposing a new rule inspired by the work of Schultz and Weismantel [85] which we call the Asymmetric Steepest Descent (ASD) circuit-pivot rule. In Theorems 11 and 12, we show that this circuit-pivot rule satisfies both criteria. To do this, we show that an ASD circuit augmentation is a $\frac{1}{m_B}$-approximate Greatest Improvement augmentation, and then we leverage the technique of [25] to show that it can be computed in polynomial time.

It must be noted that both the result of Borgwardt and Viss [25] regarding Steepest Descent and our results regarding ASD compute circuits by solving an auxiliary LP, and therefore the polynomiality of their run-time assumes that one already has access to an efficient LP solver (e.g. the ellipsoid method). To the extent that these results are therefore not practical for developing LP solvers, they should still be interpreted as valuable computational complexity results establishing that it is not always NP-Hard to optimize over the (exponentially large) set of circuits, nor is the possibility of an efficient LP solver based on circuit-augmentation categorically hopeless. That said, we note that in follow-up work to [25], the authors explore the possibility that the auxiliary LP they use is substantially easier to solve in practice than the original LP, and achieve some promising computational results [23]. It is an open question whether something similar can be achieved without assuming the use of an efficient LP solver. We note that very recently, the authors of [35] explored a similar concept to analyze circuit-augmentation algorithms and the circuit diameter for LPs and polyhedra in equality form.

Finally, we consider the consequences of Lemma 1 on the circuit diameter. First, as an easy consequence, we get a polynomial bound on the circuit diameter of any rational polyhedron, which to the best of our knowledge was not observed before. We recall that, as usual, the encoding length of a rational number $\frac{p}{q}$ is defined as $\lceil \log(p+1) \rceil + \lceil \log(q+1) \rceil + 1$.

Theorem 7. *There exists a polynomial function $f(m, \alpha)$ that bounds above the circuit diameter of any rational polyhedron $P = \{\boldsymbol{x} \in \mathbb{R}^n : A\boldsymbol{x} = \boldsymbol{b},\, D\boldsymbol{x} \leq \boldsymbol{d}\}$ with $m = m_A + m_D$ row constraints and maximum encoding length among the coefficients in its description equal to α.*

Note that this was not quite yet implied by the work of [37] as their results were only stated and proved for polyhedra expressed in equality form. We note that their results require an explicit extension to general form LPs, because in the setting of circuits, changing *descriptions* of a polyhedron or LP can dramatically change the set of circuits. In particular, it is known that putting an LP into standard equality form can introduce *exponentially many* new circuits (see [25, 49]).

Second, we consider the special case of *k-lattice polytopes*, i.e., those whose vertices are contained in the set $\{0, \ldots, k\}^n$. We prove the following:

Theorem 8. *The circuit diameter of a k-lattice polytope is at most on the order of* $\mathrm{poly}(n)\log(kn)$.

Asymptotically, this is a better function of the parameter k than the best known bounds on the combinatorial diameter of k-lattice polytopes which are linear in both n and k [40, 41].

1.1 Organization of This Thesis

We start in Chapter 3 by investigating questions related to circuits. As discussed previously, this is in part because the results we obtain in the circuit setting form the foundation for much of the work we do later on.

In Section 3.1, we establish the hardness of the Greatest Improvement and Dantzig circuit pivot rules. In Section 3.2 we further explore circuit-augmentation algorithms by extending the results of [37] in addition to establishing new results. We then use our results to analyze the Steepest Descent circuit-pivot rule. In Section 3.3, we introduce and analyze the ASD circuit-pivot rule and show that it can be computed in polynomial time. In Section 3.4 we consider the implications of our results on the circuit diameter.

In Chapter 4, we introduce four edge rules for 0/1-LPs and analyze the lengths of the paths they generate. In Section 4.1, we show that at a vertex solution of a 0/1-LP, a Steepest Descent circuit is in fact an edge. We therefore define the Steepest Edge rule for 0/1-LPs, and – by leveraging our results about the Steepest Descent circuit-pivot rule – we show that it generates a path of polynomial length and that it can be computed in polynomial time. In Section 4.2 we likewise show that at a vertex solution of a 0/1-LP in SEF, an ASD circuit is an edge. We therefore define the ASD rule for 0/1-LPs in SEF, and show that it generates a path of polynomial length. In Section 4.3, we introduce two edge rules for 0/1-LPs, the Slim Shadow and Ordered Shadow rules, which are modifications of the classical Shadow Vertex pivot rule. We show that they generate paths of length n and d, respectively, where d is the dimension of the underlying polyhedron.

Finally, in Chapter 5, we extend the edge rules given in Chapter 4 to pivot rules which each generate the same path as their respective edge rule counterparts. In Section 5.1 we do so for the Steepest Edge rule, in Section 5.2 we do so for the ASD rule, and in Section 5.3 we do so for the Slim Shadow and Ordered Shadow rules. The results of Sections 5.1 and 5.3 answer Question 2. In Section 5.4, we explore connections between our proposed pivot rules and classical algorithms in combinatorial optimization.

All work in this thesis is joint work with Laura Sanità. The results of Sections 3.1, 3.2, 3.4, and 4.1 are also joint work with Jesús De Loera, and those of Sections 3.1, 3.2, and 4.1 appear in [38]. The results of Sections 4.3, 5.1, 5.3, and 5.4 are also joint work with Alex Black and Jesús De Loera, and appear in [12].

Chapter 2

Preliminaries

Given a vector $\boldsymbol{z} \in \mathbb{R}^n$, we will use the notation $\boldsymbol{z}(i)$ to denote the i^{th} component of $\boldsymbol{z}$, and given a subset X of $[n]$ we will use the notation $\boldsymbol{z}(X)$ to denote the restriction of $\boldsymbol{z}$ to the components indexed by X. We further let $\operatorname{supp}(\boldsymbol{z}) = \{\, i \in [n] : \boldsymbol{z}(i) \neq 0 \,\}$ denote the support of $\boldsymbol{z}$. Given a matrix $Z \in \mathbb{R}^{m \times n}$ we will use the notation Z_i to denote the i^{th} row of Z, and given a subset X of $[m]$ we will use the notation Z_X to denote the submatrix of Z obtained by taking only the rows indexed by X. Given a matrix Z, we let $\ker(Z)$ denote its kernel.

As stated in Chapter 1, we consider LPs in the general form (1.1)

$$\max \{\; \boldsymbol{c}^\intercal \boldsymbol{x} \;:\; A\boldsymbol{x} = \boldsymbol{b},\, D\boldsymbol{x} \leq \boldsymbol{d},\, \boldsymbol{x} \in \mathbb{R}^n \;\}$$

for integer matrices A and D of sizes $m_A \times n$ and $m_D \times n$ respectively, and integer vectors $\boldsymbol{b}$, $\boldsymbol{d}$, and $\boldsymbol{c}$. In all parts of this thesis, we make the assumptions that (i) the matrix A has full row-rank, which can be made without loss of generality by removing redundant equalities, and that (ii) the matrix $\binom{A}{D}$ has rank n, to ensure that the feasible region is pointed. In particular chapters, we may make further assumptions, and we will state these later in this section as well as at the beginning of the appropriate chapter. We allow for the possibility that m_A is equal to 0, corresponding to the case where the LP has no equality constraints (If m_B is equal to 0, then the assumption that $\binom{A}{D}$ has rank n would imply that the feasible region has just one solution, so we do not consider that case). Here the objective function is maximized over the polyhedron $P = \{\boldsymbol{x} \in \mathbb{R}^n : A\boldsymbol{x} = \boldsymbol{b},\, D\boldsymbol{x} \leq \boldsymbol{d}\}$ of feasible solutions. Given such an LP or polyhedron, we denote by δ the maximum absolute value of the determinant of any $n \times n$ submatrix of $\binom{A}{D}$ (the notation δ will only be used in circumstances where the matrix it describes is clear from context).

Given an extreme point $\boldsymbol{x}'$ of P, we define the *feasible cone* at $\boldsymbol{x}'$ to be the set of all directions $\boldsymbol{z}$ such that $\boldsymbol{x}' + \varepsilon\boldsymbol{z} \in P$ for some $\varepsilon > 0$. That is, it is the set $\mathcal{C}(\boldsymbol{x}') = \{\, \boldsymbol{z} \in \mathbb{R}^n : A\boldsymbol{z} = \boldsymbol{0},\, \overline{D}\boldsymbol{z} \leq \boldsymbol{0} \,\}$ where $\overline{D}$ denotes row submatrix of D given by the indices of the inequalities of $D\boldsymbol{x} \leq \boldsymbol{d}$ that are tight at $\boldsymbol{x}'$. The extreme rays of the feasible cone at $\boldsymbol{x}'$ correspond to the *edge-directions* at $\boldsymbol{x}'$.

Let $\boldsymbol{z}^1, \dots, \boldsymbol{z}^t$ be generators of the extreme rays of the feasible cone at $\boldsymbol{x}'$. Then for each $i \in [t]$, there exists a unique extreme point $\boldsymbol{x}^i$ of P which is adjacent to $\boldsymbol{x}'$ in P such that $\boldsymbol{z}^i = \alpha(\boldsymbol{x}^i - \boldsymbol{x}')$ for some positive scalar α. In this circumstance, we say that $(\boldsymbol{x}^i - \boldsymbol{x})$ is an *edge-direction* at $\boldsymbol{x}'$. We therefore say that any generator $\boldsymbol{z}^i$ of an extreme ray of $\mathcal{C}(\boldsymbol{x}')$ *corresponds to an edge-direction at* $\boldsymbol{x}'$ *in* P.

We let $N_{\boldsymbol{c}}(\boldsymbol{x}')$ denote the set of $\boldsymbol{c}$-improving neighbors (i.e., adjacent vertices) of $\boldsymbol{x}'$ for a vertex $\boldsymbol{x}'$ of P.

An edge-direction $\boldsymbol{g}$ at $\boldsymbol{x}'$ in P is called a steepest edge-direction if it maximizes $\frac{\boldsymbol{c}^\intercal \boldsymbol{g}}{||\boldsymbol{g}||_1}$ among all edge-directions at $\boldsymbol{x}'$ in P. We make special note of the fact that "steepness" here is measured with respect to the 1-norm instead of the usual 2-norm used in [50] and elsewhere. This will hold true throughout this entire thesis in *all* instances where steepness is measured.

The extreme points of P and the one-dimensional faces form the *graph of the polytope* P, also called the 1-*skeleton* of P.

An *edge rule* is a process which, given an LP of the form (1.1) with feasible region P and a vertex $\boldsymbol{x}$ of P, prescribes a way for choosing a $\boldsymbol{c}$-improving edge-direction at $\boldsymbol{x}$. Given an initial vertex $\boldsymbol{x}^0$, an edge rule generates a $\boldsymbol{c}$-monotone path from $\boldsymbol{x}^0$ to an optimal solution, where the path is obtained by taking the edges chosen by the edge rule. We say that an edge rule is computable in polynomial time on an LP (P) if, for and all possible starting vertices $\boldsymbol{x}^0$, at each vertex on the path generated by the rule, the problem of computing the edge chosen by the rule can be solved in polynomial time. We say that an edge rule is computable in polynomial time on a family of LPs if it is computable in polynomial time on all members of that family.

In this thesis, we will consider edge rules which are computable in polynomial time for the family of 0/1-LPs. Recall that for the sake of brevity, we may at times refer to an edge rule as simply a *rule* if it is clear from context that we are referring to an edge rule.

For some of our results, we will rely next on the following result of Frank and Tardos [51]:

Lemma 2 ([51])**.** *Let* $\boldsymbol{c} \in \mathbb{R}^n$ *be a rational vector, and* α *be a positive integer. Define* $N := (n+1)!2^{n\alpha} + 1$*. Then one can compute an integral vector* $\boldsymbol{c}' \in \mathbb{Z}^n$ *satisfying:*

(a) $\|\boldsymbol{c}'\|_\infty \leq 2^{4n^3} N^{n(n+2)}$;

(b) *Consider any rational LP of the form* $\max\{\, \boldsymbol{c}^\intercal \boldsymbol{x} \;:\; A'\boldsymbol{x} \leq \boldsymbol{b}',\, \boldsymbol{x} \in \mathbb{R}^n \,\}$*, where the encoding length of any entry of* A' *is at most* α*. Then,* $\boldsymbol{x} \in \mathbb{R}^n$ *is an optimal solution to that LP if and only if it is an optimal solution to* $\max\{\, \boldsymbol{c}'^\intercal \boldsymbol{x} \;:\; A'\boldsymbol{x} \leq \boldsymbol{b}',\, \boldsymbol{x} \in \mathbb{R}^n \,\}$.

2.1 The Simplex Method

We start by noting that in all cases where we consider applying the Simplex method to an LP (i.e., the contents of Chapter 5), it will be under the assumption that the feasible region of this LP is a 0/1 polytope. Then in Chapter 5, we will assume without loss of generality that the set of inequalities contains $\boldsymbol{x} \geq \boldsymbol{0}$. We will now justify a geometric framework through which we may define a pivot rule. For the purpose of explaining this geometric framework, we will give a brief review of the Simplex method here in the typical language of tableaux manipulation, which is how the it is implemented in practice. For general background on the Simplex method and its implementation, see [8, 84, 93].

Suppose that we are given a 0/1-LP of the form (1.1). We have to start by putting our LP into standard equality form. Since we will only consider applications to 0/1-LPs, it suffices to add slack variables, as all original variables already satisfy non-negativity by assumption. Then after we add slack variables we get an LP of the form

$$\max\{\, \boldsymbol{c}'^\intercal \boldsymbol{x} :\; A'\boldsymbol{x} = \boldsymbol{b}',\, \boldsymbol{x} \geq \boldsymbol{0},\, \boldsymbol{x} \in \mathbb{R}^{n'} \,\},$$

where $A' \in \mathbb{Z}^{m' \times n'}$, $\boldsymbol{c}' \in \mathbb{Z}^{n'}, \boldsymbol{b}' \in \mathbb{Z}^{m'}$. We remark here that we do not require that the added slack variables always take on 0/1 values at vertices. Though we put the LP into equality form for the purposes of executing the Simplex method, the results in this thesis will only rely on the fact that the *original LP* is 0/1.

Since we assumed earlier that the equality matrix A of our original LP has full row-rank, we have that A' has full row-rank m'. A *basis* B is a subset of $[n']$ (the column indices of the matrix A') of size m' such that the columns of A' indexed by B are linearly independent. For the purposes of this description, we will use the notation $A'_{[B]}$ to denote the submatrix of A' obtained by taking only the columns indexed by B, and likewise we will use the notation $A'_{[j]}$ to denote the j^{th} column of A'. Though this notation is non-standard, outside of this explanation and Subsection 5.1.1, we will not need to reuse this notation. Given a basis B, assume that it is an ordered set and let $B(j)$ be the j-th element of the set. The variables with indices in B are also referred to as *basic variables*, and the variables with indices in $N := \{1, \ldots, n'\} \setminus B$ are referred to as the *nonbasic variables.* A basis B is uniquely associated with a *basic solution* $\boldsymbol{x}'$ defined as follows: $\boldsymbol{x}'(B) = A'^{-1}_{[B]}\boldsymbol{b}'$ and $\boldsymbol{x}'(N) = \mathbf{0}$. If $\boldsymbol{x}' \geq \mathbf{0}$, then it is called a *basic feasible solution,* and the corresponding basis is called a *feasible basis.*

The Simplex method can be described compactly as follows; see, e.g., [8, 84, 93] for a more detailed treatment.

- Start with any feasible basis B be and repeat the following steps:

  1. Compute the reduced costs for the nonbasic variables

  $$\bar{\boldsymbol{c}}'(N)^\intercal = \boldsymbol{c}'(N)^\intercal - \boldsymbol{c}'(B)^\intercal A'^{-1}_{[B]} A'_{[N]}.$$

  If $\bar{\boldsymbol{c}}'(N) \leq \mathbf{0}$ the basis is optimal: the algorithm terminates. Otherwise, choose $j : \bar{\boldsymbol{c}}'(j) > 0$. The particular choice of j depends on the pivot rule used. The variable associated with column j is the *entering* variable.

  2. Compute $\boldsymbol{u} = A'^{-1}_{[B]} A'_{[j]}$. If $\boldsymbol{u} \leq \mathbf{0}$, the optimal cost is unbounded: the algorithm terminates.

  3. Else, some component of $\boldsymbol{u}$ is positive. Compute the *ratio test*:

  $$r^* := \min_{i \in [m'] \,:\, \boldsymbol{u}(i) > 0} \left(\frac{\boldsymbol{x}'(B(i))}{\boldsymbol{u}(i)} \right). \tag{2.1}$$

  4. Let ℓ be such that $r^* = \frac{\boldsymbol{x}'(B(\ell))}{\boldsymbol{u}(\ell)}$. If there is a tie, the particular choice of ℓ depends on the pivot rule used. The variable associated with ℓ is the *leaving* variable. Form a new basis replacing $B(\ell)$ with j. This step is called a *pivot.* Go back to 1.

At a high level, the Simplex method moves between feasible bases, and requires a pivot rule, which given a basis, decides which *adjacent basis* to move to (in particular, how to choose the entering variable and the leaving variable at each step).

Each of the Simplex method's basis exchanges corresponds geometrically to a *pivot direction* in which to move from the current basic feasible solution $\boldsymbol{x}$. Formally, note that at Step 1 of the algorithm, each index $j \in N$ yields a pivot direction $\bar{\boldsymbol{z}}^j$ defined as $\bar{\boldsymbol{z}}^j(B) = A'^{-1}_{[B]} A'_{[j]}$, $\bar{\boldsymbol{z}}^j(j) = 1$, and $\bar{\boldsymbol{z}}^j(i) = 0$ for all $i \in N \setminus \{\, j \,\}$. The Simplex method selects a pivot direction which is improving with respect to the objective function.

If moving (by a non-zero amount) in a pivot direction $\boldsymbol{g}$ from $\boldsymbol{x}'$ *maintains* feasibility, we call this a *non-degenerate pivot*. That is, a non-degenerate pivot has the property that $\boldsymbol{x}' + \varepsilon \boldsymbol{g}$ is feasible for some $\varepsilon > 0$. In this case, our new basis yields a new basic feasible solution $\boldsymbol{y}'$, where $\boldsymbol{x}'$ and $\boldsymbol{y}'$ are *adjacent* in the feasible region – that is, contained in a common edge. If instead, moving in this direction from $\boldsymbol{x}'$ would *violate* feasibility, we call this a degenerate pivot. A degenerate pivot has the property that $\boldsymbol{x}' + \varepsilon \boldsymbol{g}$ is not feasible for any $\varepsilon > 0$. In this case, though the basis changes, the new basic feasible solution still corresponds to the extreme point $\boldsymbol{x}'$.

We can now describe our geometric framework. Given a feasible basis B of our LP in standard equality form, let $\boldsymbol{x}'$ be the basic feasible solution associated to B, and $\boldsymbol{x}''$ be the corresponding vertex of P (i.e., the vector obtained from $\boldsymbol{x}'$ by removing the slack variables). With a slight abuse of terminology, we say that $\boldsymbol{x}''$ is the vertex of P *associated to* B.

The basic feasible solution $\boldsymbol{x}'$ is identified by $m' + |N|$ linearly independent constraints *of the equality form LP* that are tight at $\boldsymbol{x}'$: namely, the constraints $A'\boldsymbol{x} = \boldsymbol{b}'$ plus the non-negativity constraints for the nonbasic variables. Let $\bar{N} \subseteq N$ be the set of indices for the nonbasic variables that are original variables for P (i.e., non-slack variables). We can naturally associate n tight constraints *of our original LP* that identify $\boldsymbol{x}''$ as follows: we consider (i) the equalities $A\boldsymbol{x} = \boldsymbol{b}$, plus (ii) the non-negativity constraints $\boldsymbol{x}(j) \geq 0$ for $j \in \bar{N}$, plus (iii) the subset of inequalities associated to each nonbasic slack variable. It is easy to see that these constraints are linearly independent. Let D^B be the submatrix of D induced by the rows of the tight inequality constraints (ii) and (iii). We define the *basic cone* associated to B as

$$\mathcal{C}(B) = \{\, \boldsymbol{z} \in \mathbb{R}^n : \, A\boldsymbol{z} = \boldsymbol{0}, \, D^B \boldsymbol{z} \leq \boldsymbol{0} \,\} .$$

One observes that given a feasible basis B and its corresponding basic feasible solution $\boldsymbol{x}'$, the available pivot directions at $\boldsymbol{x}'$ project to the extreme rays of $\mathcal{C}(B)$.

Clearly, the basic cone associated to B contains the feasible cone at $\boldsymbol{x}''$, i.e., $\mathcal{C}(B) \supseteq \mathcal{C}(\boldsymbol{x}'')$, as the system defining $\mathcal{C}(B)$ is a relaxation of the system defining $\mathcal{C}(\boldsymbol{x}'')$. A generator $\bar{\boldsymbol{z}}$ of an extreme ray of $\mathcal{C}(B)$ is the projection of the pivot direction given by a non-degenerate basis exchange if and only if $\bar{\boldsymbol{z}}$ also generates an extreme ray of $\mathcal{C}(\boldsymbol{x}'')$. That is, $\bar{\boldsymbol{z}}$ is the projection of a non-degenerate pivot direction at $\boldsymbol{x}'$ given by the basis B if and only if $\bar{\boldsymbol{z}}$ also corresponds to an edge-direction at $\boldsymbol{x}''$ in P. Given this, we can partially define a pivot rule in terms of the *original* LP (i.e., the LP before it was put into standard equality form) by explaining which extreme ray of $\mathcal{C}(B)$ to choose as a pivot direction. In particular, this corresponds to the choice of the variable *entering* the basis. When it is clear from context that a vertex $\boldsymbol{x}''$ is a vertex solution of P corresponding to some feasible basis B, we may informally refer to $\mathcal{C}(B)$ as just the basic cone at $\boldsymbol{x}''$.

2.2 Circuits of Polyhedra

Here we will provide some background and fundamental results on the circuits of polyhedra. Recall that formally, the circuits are defined as follows:

Definition 2 (See [58]). *Given a polyhedron of the form* $P = \{\boldsymbol{x} \in \mathbb{R}^n : A\boldsymbol{x} = \boldsymbol{b},\ D\boldsymbol{x} \leq \boldsymbol{d}\}$, *a non-zero vector* $\boldsymbol{g} \in \mathbb{R}^n$ *is a circuit if*

(i) $\boldsymbol{g} \in \ker(A)$, *and*

(ii) $\operatorname{supp}(D\boldsymbol{g})$ *is an inclusion-wise minimal set in the collection*

$$\{\operatorname{supp}(D\boldsymbol{y}) : \boldsymbol{y} \in \ker(A), \boldsymbol{y} \neq \boldsymbol{0}\}.$$

Another way of understanding condition (ii) is that a circuit is parallel to an inclusion-wise maximal set of the hyperplanes defined by the inequalities $D\boldsymbol{x} \leq \boldsymbol{d}$ (which, in a minimal description, define the *facets* of P). This interpretation makes it clear geometrically why all edge-directions are also circuits. By this definition, if $\boldsymbol{g}$ is a circuit, then so is $\alpha\boldsymbol{g}$ for all $\alpha \neq 0$, and therefore there are infinitely many circuits. To represent the circuits with a finite set, we can normalize them in various ways. Following [19, 25, 37, 49, 63, 38], we denote by $\mathcal{C}(A, D)$ the (finite) set of circuits with co-prime integer components.

The following proposition gives an equivalent condition under which a vector $\boldsymbol{g}$ is a circuit:

Proposition 1 (See [63]). *Given a polytope* $P = \{\boldsymbol{x} \in \mathbb{R}^n : A\boldsymbol{x} = \boldsymbol{b}, D\boldsymbol{x} \leq \boldsymbol{d}\}$, *a non-zero vector* $\boldsymbol{g} \in \mathbb{R}^n$ *is a circuit iff* $\boldsymbol{g}$ *is a unique (up to scaling) nonzero solution of* $\{\, A\boldsymbol{z} = \boldsymbol{0},\ D'\boldsymbol{z} = \boldsymbol{0} \,\}$ *where* D' *is a row submatrix of* D.

The following proposition implies that the circuits can be collected into equivalence classes based on the support of their product with D:

Proposition 2 (See [58]). *Let* $\boldsymbol{g} \in \mathcal{C}(A, D)$, $\boldsymbol{v} \in \ker(A)$, *and suppose* $\operatorname{supp}(B\boldsymbol{g}) = \operatorname{supp}(B\boldsymbol{v})$. *Then* $\boldsymbol{g} = \alpha\boldsymbol{v}$ *for some* $\alpha \in \mathbb{R}$.

Then for each such equivalence class, $\mathcal{C}(A, D)$ contains two representatives, $\boldsymbol{g}$ and $-\boldsymbol{g}$.

This implies a fundamental and extremely useful property of the circuits, called the *sign-compatible representation property* of circuits. We say two vectors $\boldsymbol{v}$ and $\boldsymbol{w}$ are *sign-compatible with respect to* D if the i-th components of the vectors $(D\boldsymbol{v})$ and $(D\boldsymbol{w})$ satisfy $(D\boldsymbol{v})(i) \cdot (D\boldsymbol{w})(i) \geq 0$ for all $1 \leq i \leq m_D$. The representation property is as follows:

Proposition 3 (See [58]). *Let* $\boldsymbol{v} \in \ker(A) \setminus \{ \mathbf{0} \}$. *Then we can express* $\boldsymbol{v}$ *as* $\boldsymbol{v} = \sum_{i=1}^{k} \alpha_i \boldsymbol{g}^i$ *such that for all* $1 \leq i \leq k$

- $\boldsymbol{g}^i \in \mathcal{C}(A, D)$,
- $\boldsymbol{g}^i$ *and* $\boldsymbol{v}$ *are sign-compatible with respect to* D *and* $\operatorname{supp}(D\boldsymbol{g}^i) \subseteq \operatorname{supp}(D\boldsymbol{v})$,
- $\alpha_i \in \mathbb{R}_{\geq 0}$,
- *and* $k \leq n$.

That is, any feasible direction can be decomposed into a sum of at most n circuits which are all sign compatible with the original direction (and thus, each other). This property allows for an alternative and equivalent definition of circuits

Proposition 4 (See [49]). *The set* $\mathcal{C}(A, D)$ *is the unique (up to re-scaling) inclusion-wise minimal set with the sign-compatible representation property.*

Essentially, it is the sign compatible representation property that allows circuits to be used as a set of augmentation directions for solving LPs:

Proposition 5 (See [58]). *Consider an LP of the form (1.1) with feasible region* P*, and let* $\boldsymbol{x}'$ *be a non-optimal solution. Then there exists* $\boldsymbol{g} \in \mathcal{C}(A, D)$ *such that* $\boldsymbol{x}' + \varepsilon \boldsymbol{g} \in P$ *for some* $\varepsilon > 0$ *and* $\mathbf{c}^\intercal \boldsymbol{g} > 0$.

A *circuit-path* is a finite sequence of feasible solutions $\boldsymbol{x}_0, \boldsymbol{x}_1, \ldots, \boldsymbol{x}_q$ satisfying $\boldsymbol{x}_{i+1} = \boldsymbol{x}_i + \alpha_i \boldsymbol{g}_i$, where $\boldsymbol{g}_i \in \mathcal{C}(A, D)$ and $\alpha_i > 0$ is such that $\boldsymbol{x}_i + \alpha_i \boldsymbol{g}_i \in P$ but $\boldsymbol{x}_i + (\alpha_i + \varepsilon)\boldsymbol{g}_i \notin P$ for all $\varepsilon > 0$ (i.e., the augmentation is maximal). Note that $\boldsymbol{x}_i$ is not necessarily a vertex of P. We say that such a circuit-path has length q. A circuit-path is called monotone if each $\boldsymbol{g}_i$ satisfies $\mathbf{c}^\intercal \boldsymbol{g}_i > 0$ (i.e., it is an improving circuit).

Given two points $\boldsymbol{x}$ and $\boldsymbol{y}$ in P, the circuit distance from $\boldsymbol{x}$ to $\boldsymbol{y}$ is the length of a shortest circuit-path from $\boldsymbol{x}$ to $\boldsymbol{y}$. Due to the requirement that circuit augmentations are maximal, circuit-paths are not necessarily reversible, and so the circuit distance from $\boldsymbol{x}$ to $\boldsymbol{y}$ is not necessarily the same as the circuit distance from $\boldsymbol{y}$ to $\boldsymbol{x}$.

Given a polyhedron P, the circuit diameter of P is the maximum circuit distance between any pair of vertices of P. Here, we must address an important technicality. The conditions in Definition 1 are perfectly well-defined even if we are given a description of P with redundant inequalities. Indeed, in an optimization setting, we are likely to be given P by some linear description, and there is usually no guarantee that this description won't contain redundant information. However – as is evident from condition (ii) especially – adding redundant inequalities to a linear system can grow the set of circuits. In fact, as mentioned in Chapter 1, is known in general that equivalent polyhedra do not necessarily have equivalent sets of circuits. However, we would like to be able speak of the circuit diameter as being purely a property of the geometry of P. This would be a desirable property, and would give the circuit diameter more parity with the combinatorial diameter.

Therefore, when considering the circuit diameter (i.e., in Section 3.4), we will assume that the given description of P is *minimal*, by which we mean the following:

Definition 3 (See e.g. [32]). *Given a polyhedron P (considered as a geometric object), we say that $\{\boldsymbol{x} \in \mathbb{R}^n : A\boldsymbol{x} = \boldsymbol{b},\ D\boldsymbol{x} \leq \boldsymbol{d}\}$ is a minimal description of P if it satisfies the following:*

(i) $P = \{\boldsymbol{x} \in \mathbb{R}^n : A\boldsymbol{x} = \boldsymbol{b},\ D\boldsymbol{x} \leq \boldsymbol{d}\}$

(ii) No inequality of $\{\boldsymbol{x} \in \mathbb{R}^n : A\boldsymbol{x} = \boldsymbol{b},\ D\boldsymbol{x} \leq \boldsymbol{d}\}$ can be made an equation without changing the solution set.

(iii) No inequality or equation of $\{\boldsymbol{x} \in \mathbb{R}^n : A\boldsymbol{x} = \boldsymbol{b},\ D\boldsymbol{x} \leq \boldsymbol{d}\}$ can be omitted without changing the solution set.

In Section 3.4, we prove the following lemma, which resolves the dilemma described above:

Lemma 3. *Let P be any polyhedron where $\{\boldsymbol{x} \in \mathbb{R}^n : A\boldsymbol{x} = \boldsymbol{b},\ D\boldsymbol{x} \leq \boldsymbol{d}\}$ and $\{\boldsymbol{x} \in \mathbb{R}^n : A'\boldsymbol{x} = \boldsymbol{b}',\ D'\boldsymbol{x} \leq \boldsymbol{d}'\}$ are both minimal descriptions of P. Then $\mathcal{C}(A, D) = \mathcal{C}(A', D')$.*

This shows that if we restrict ourselves only to descriptions of P which are minimal, then in fact the circuits of P *are* independent of its description. This is known to be true for full-dimensional polyhedra as they are known to have inequality-form descriptions which are unique up to re-scaling of the inequalities. However, this is not immediately obvious for polyhedra which are not full-dimensional, as their descriptions have no such uniqueness property. To our knowledge, this was not previously observed.

A circuit-augmentation algorithm computes a monotone circuit-path starting at a given initial feasible solution, and it performs augmentations along circuits until an optimal solution is reached (or unboundedness is detected). The circuit $\boldsymbol{g}$ to use at each augmentation is usually chosen according to some *circuit-pivot rule*. As discussed before, the authors of [37] consider three such rules, each of which gives rise to a corresponding optimization problem.

The optimization problem that arises when following the Greatest Improvement circuit-pivot rule will be called $\text{Great}(P, \boldsymbol{x}, \boldsymbol{c})$, and is as follows:

$$\begin{aligned} \max \quad & \boldsymbol{c}^\intercal(\alpha \boldsymbol{g}) \\ \text{s.t.} \quad & \boldsymbol{g} \in \mathcal{C}(A, D) \\ & \alpha > 0 \\ & \boldsymbol{x} + \alpha \boldsymbol{g} \in P. \end{aligned}$$

The optimization problem that arises when following the Dantzig circuit-pivot rule will be called $\text{Dan}(P, \boldsymbol{x}, \boldsymbol{c})$, and is as follows:

$$\begin{aligned} \max \quad & -\boldsymbol{c}^\intercal \boldsymbol{g} \\ \text{s.t.} \quad & \boldsymbol{g} \in \mathcal{C}(A, D) \\ & \boldsymbol{g} \in \mathcal{C}(\boldsymbol{x}). \end{aligned}$$

The optimization problem that arises when following the Steepest Descent circuit-pivot rule will be called $\text{Steep}(P, \boldsymbol{x}, \boldsymbol{c})$, and is defined as follows:

$$\begin{aligned} \max \quad & -\frac{\boldsymbol{c}^\intercal \boldsymbol{g}}{||\boldsymbol{g}||_1} \\ \text{s.t.} \quad & \boldsymbol{g} \in \mathcal{C}(A, D) \\ & \boldsymbol{g} \in \mathcal{C}(\boldsymbol{x}). \end{aligned}$$

A maximal augmentation given by an optimal solution to $\text{Great}(P, \boldsymbol{x}, \boldsymbol{c})$ is called a Greatest Improvement augmentation. A Dantzig augmentation and a Steepest Descent augmentation are defined similarly. In this thesis, we will only use maximal augmentations, and therefore will omit the word "maximal" henceforth.

Let $\boldsymbol{x}_{\max}$ and $\boldsymbol{x}_{\min}$ be a maximizer and minimizer, respectively, of the objective function in an LP problem of the form (1.1). We will use the following lemma from [37] based on well-known estimates of [2]:

Lemma 4 (See Lemma 1 in [37])**.** *Let $\varepsilon > 0$ be given. Let $\boldsymbol{c}$ be an integer vector. Define $f^{\min} := \boldsymbol{c}^\intercal \boldsymbol{x}_{\min}$, $f^{\max} := \boldsymbol{c}^\intercal \boldsymbol{x}_{\max}$. Suppose that $f^k = \boldsymbol{c}^\intercal \boldsymbol{x}_k$ is the objective function value of the solution $\boldsymbol{x}_k$ at the k-th iteration of an augmentation algorithm. Furthermore, suppose that the algorithm guarantees that for every augmentation k,*

$$(f^{k+1} - f^k) \geq \beta(f^{\max} - f^k).$$

Then the algorithm reaches a solution with objective value less than $f^{\max} - \varepsilon$ in no more than $2\log((f^{\max} - f^{\min})/\varepsilon)/\beta$ augmentations.

We now state the following easy lemma – a version of which appears in [37] – that we reprove for completeness and to establish that it holds for general form LPs:

Lemma 5. *Let $\bar{\boldsymbol{x}}$ be any feasible solution of the LP problem (1.1). Then with a sequence of at most n maximal augmentations, we can reach a vertex solution $\hat{\boldsymbol{x}}$ of (1.1) such that $\boldsymbol{c}^\intercal \hat{\boldsymbol{x}} \geq \boldsymbol{c}^\intercal \bar{\boldsymbol{x}}$.*

Proof. Let $T = \{\, i : D_i \bar{\boldsymbol{x}} = \boldsymbol{d}(i) \,\}$. If $\bar{\boldsymbol{x}}$ is not a vertex, then we can select any direction $\boldsymbol{g} \in \ker \binom{A}{D_T}$ such that $\boldsymbol{c}^\intercal \boldsymbol{g} \geq 0$, and such that for some $\varepsilon > 0$, $\tilde{\boldsymbol{x}} := \bar{\boldsymbol{x}} + \varepsilon \boldsymbol{g}$ satisfies $D_i \tilde{\boldsymbol{x}} \leq \boldsymbol{d}(i)$ for all $i \notin T$. We then use $\boldsymbol{g}$ to perform a maximal step $\alpha \boldsymbol{g}$ at $\bar{\boldsymbol{x}}$. Since the step is maximal, there exists an index $i \notin T$ such that $D_i(\bar{\boldsymbol{x}} + \alpha \boldsymbol{g}) = \boldsymbol{d}(i)$. This enables us to grow the set T at the new feasible solution. Furthermore, $\boldsymbol{c}^\intercal(\bar{\boldsymbol{x}} + \alpha \boldsymbol{g}) \geq \boldsymbol{c}^\intercal \bar{\boldsymbol{x}}$. We can iterate this process, and note that the number of linearly independent rows of $\binom{A}{D_T}$ increases by one at each step. Therefore, after at most $n - \operatorname{rank}(A)$ iterations we arrive at a vertex $\hat{\boldsymbol{x}}$.

Note that the above argument does not require the use of circuits, but it requires only that the selected directions are improving with respect to $\boldsymbol{c}$. By the sign-compatible representation property of circuits though, at any non-optimal point $\bar{\boldsymbol{x}}$, there always exists an improving direction that is a circuit. □

2.3 Other Prior Work

Simplex There are many different pivot rules for the Simplex method, and the list of pivot rules that have already been studied is too large to cover here in detail; we refer the reader to [92] for a general taxonomy.

In 1972, Klee & Minty [67] showed for the first time that pivot rules may exhibit exponential behavior – even in the case where short monotone paths exist. They constructed an explicit set of examples such that Dantzig's original pivot rule requires exponentially many steps. The algorithm could be tricked into visiting all 2^d vertices of a deformed cube to find a path between two nodes which are only one step apart in the 1-skeleton of the cube. Later, Zadeh found that bad exponential behavior may appear even in nice families such as network flow problems [96]. Today, all popular pivot rules for the Simplex method are known to require an exponential number of steps to solve some concrete "twisted" linear programs (see [3, 4, 43, 52, 57, 56, 60, 71, 92, 96, 99] and references therein). We also know that pivot rules as decision problems are hard in the sense of complexity theory [1, 44, 48].

Today, only a few highly structured families of LPs have reasonable bounds of efficiency for the Simplex method. A notable family of LPs is the family of network flow linear programs which have been shown to be solvable in polynomial time by Orlin [76]. Dadush & Hähnle [33] – inspired by prior work of Brunsch & Röglin [27] and Eisenbrand & Vempala [47] – studied the Simplex method with the Shadow Vertex pivot rule of K. H. Borgwardt [16]. In the remainder of this thesis, we drop the word "Vertex" from the name for sake of brevity. They studied this rule over *low-curvature* polyhedra. Intuitively, a polyhedron is low-curvature when the hyperplanes at the boundary meet vertices at sharp angles, i.e., their tangent cones are slim. Dadush and Hänhle obtained a diameter bound of $O(\frac{d^2}{\delta}\ln\frac{d}{\delta})$ for d-dimensional polyhedra with curvature parameter $\delta \in (0, 1]$. They showed that, starting from some initial vertex, an optimal vertex can be found using an expected $O(\frac{d^3}{\delta}\ln\frac{d}{\delta})$ Simplex pivots, each requiring $O(md)$ time to compute, where m is the number of constraints. An initial feasible solution can be found using $O(\frac{md^3}{\delta}\ln\frac{d}{\delta})$ pivot steps. Their analysis of the Shadow pivot rule differs in that they study the behavior of the pivot rule in relation to the geometry of the normal cones. Borgwardt referred to this perspective as the dual Shadow pivot rule in and used the dual perspective to prove his bounds as well. We note that 0/1 polytopes do not have bounded curvature or small subdeterminants because the coefficients of defining inequalities can be extremely large [98].

Inspired by the work of Y. Ye [95], Kitahara & Mizuno [65, 64], showed that in specific context of LPs in the *standard equality form* (SEF)

$$\max\{\, \boldsymbol{c}^\intercal \boldsymbol{x} \;:\; A\boldsymbol{x} = \boldsymbol{b},\, \boldsymbol{x} \geq \boldsymbol{0} \,\}, \tag{2.2}$$

the number of different basic feasible solutions (BFSs) generated by the Simplex method using Dantzig's pivot rule is bounded by $m_A\lceil n\frac{\gamma}{\tau}\log(n\frac{\gamma}{\tau})\rceil$, where τ and γ are the minimum and the maximum values of all the positive elements of primal BFSs. Their results showed that the number of non-degenerate pivots performed by the Simplex method with Dantzig's rule is strongly polynomial for a *subclass* of 0/1-LPs: namely, those that can be expressed in standard equality form with all extreme points having variables in $\{0, 1\}$ (e.g., the Birkhoff polytope). We stress that while every 0/1 polyhedral region can be written in standard form by adding slack variables, at an extreme point such variables could potentially take values not inside $\{0, 1\}$. Thus, they may not satisfy the hypothesis of [65, 64]. As such, the strongly polynomial bound of [65, 64] does not extend to all 0/1-LPs, and so these results do not give a complete answer to Question 2.

It is important we stress that when we speak of the Simlpex method here, we mean the traditional version of the primal Simplex method, based only on local basic feasible solution information (as taught in most courses and used in most software). It is not easy to compare results in this setting with generalizations or variations of the Simplex method. Work outside of the traditional version of the Simplex method has achieved results which are very interesting in their own right, but do not deal with the same algorithm. In particular, different assumptions may be made about the input, the algorithms may be stated in different oracle models – such as augmentation and verification oracles – or they may move in directions other than edges at a vertex solution (see [28, 29, 30, 37, 40, 38] and the many references therein). For example, there are "variations" of the Simplex method for which the number of non-degenerate pivots (or simply augmentations, when bases are not even considered) is strongly polynomially bounded, but such results require modifying the objective function to the point that the path of edges followed may or may not be monotonically improving in the original objective function [40, 30].

Diameter Besides studies of upperbounds on the combinatorial diameter for general polytopes, there is a long history of studies of such upperbounds for some special classes of polytopes. In particular, many researchers have considered the combinatorial diameter of polytopes corresponding to classical combinatorial optimization problems. Prominent examples of polytopes for which the combinatorial diameter has been widely studied are Transportation and Network Flow polytopes [5, 6, 20, 21, 26], Matching polytopes [6, 31], Traveling Salesman (TSP) polytopes [79, 59], and many others. Currently the best known general upperbound on the diameter is exponential in n [89, 90].

Since for every polytope the set of circuit directions contains all edge directions, the combinatorial diameter is always an upper bound on the *circuit diameter*, it's circuit analogue. Thus even if the Hirsch Conjecture does not hold for the combinatorial diameter, it may be true for the circuit diameter. In particular, Borgwardt et al. [19] conjectured that the circuit diameter is at most $f - d$ for every d-dimensional polytope with f facets. We refer the reader to [22] for recent progress on this conjecture. The circuit diameter has also been studied in the context of well-known problems in combinatorial optimization [20, 19, 63, 24].

Chapter 3

Circuits

3.1 Hardness of Some Circuit-Pivot Rules

3.1.1 The Circuits of the Fractional Matching Polytope

Let G be a simple connected graph with nodes $V(G)$ and edges $E(G)$. We assume $|V(G)| \geq 3$. Given $v \in V(G)$, we let $\delta_G(v)$ denote the edges of $E(G)$ incident with v. We call a node $v \in V(G)$ a *leaf* if $|\delta_G(v)| = 1$, and let $L(G)$ denote the set of leaf nodes of G. Furthermore, for $X \subseteq E$ and $\boldsymbol{x} \in \mathbb{R}^{E(G)}$, we let $\boldsymbol{x}(X)$ denote $\sum_{e \in X} \boldsymbol{x}(e)$.

Let $P_{\text{FMAT}}(G)$ denote the fractional matching polytope of G, which is defined by the following (minimal) linear system:

$$\boldsymbol{x}\left(\delta_G(v)\right) \leq 1, \qquad \text{for all } v \in V(G) \setminus L(G). \tag{3.1}$$

$$\boldsymbol{x} \geq \mathbf{0}. \tag{3.2}$$

In this section, we fully characterize the circuits of $P_{\text{FMAT}}(G)$. We will prove that, if $\boldsymbol{x}$ is a circuit of $P_{\text{FMAT}}(G)$, then $\text{supp}(\boldsymbol{x})$ induces a connected subgraph of G that has a very special structure: namely, it belongs to one of the five classes of graphs $(\mathcal{E}_1, \mathcal{E}_2, \mathcal{E}_3, \mathcal{E}_4, \mathcal{E}_5,)$ listed below.

(i) Let $\mathcal{E}_1$ denote the set of all subgraphs $F \subseteq G$ such that F is an even cycle.

(ii) Let $\mathcal{E}_2$ denote the set of all subgraphs $F \subseteq G$ such that F is an odd cycle.

(iii) Let $\mathcal{E}_3$ denote the set of all subgraphs $F \subseteq G$ such that F is a simple path.

(iv) Let $\mathcal{E}_4$ denote the set of all subgraphs $F \subseteq G$ such that F is a connected graph satisfying $F = C \cup P$, where C and P are an odd cycle and a non-empty simple path, respectively, that intersect only at an endpoint of P (see Figure 3.1).

(v) Let $\mathcal{E}_5$ denote the set of all subgraphs $F \subseteq G$ such that F is a connected graph with $F = C_1 \cup P \cup C_2$, where C_1 and C_2 are odd cycles, and P is a (possibly empty) simple path satisfying the following: if P is non-empty, then C_1 and C_2 are node-disjoint and P intersects each C_i exactly at its endpoints (see Figure 3.2); if P is empty then C_1 and C_2 intersect only at one node v (see Figure 3.3).

We will associate a set of circuits to the subgraphs in the above families by defining the following five sets of vectors. It is worth noticing that similar elementary moves appeared in [39] in applications of Gröbner bases in combinatorial optimization.

$$\mathcal{C}_1 = \bigcup_{F \in \mathcal{E}_1} \left\{ \boldsymbol{g} \in \{-1,0,1\}^{E(G)} : \begin{array}{ll} \boldsymbol{g}(e) \neq 0 & \text{iff } e \in E(F) \\ \boldsymbol{g}(\delta_F(v)) = 0 & \forall v \in V(F) \end{array} \right\},$$

$$\mathcal{C}_2 = \bigcup_{F \in \mathcal{E}_2} \left\{ \boldsymbol{g} \in \{-1,0,1\}^{E(G)} : \begin{array}{ll} \boldsymbol{g}(e) \neq 0 & \text{iff } e \in E(F) \\ \boldsymbol{g}(\delta_F(w)) \neq 0 & \text{for one } w \in V(F) \\ \boldsymbol{g}(\delta_F(v)) = 0 & \forall v \in V(F) \setminus \{w\} \end{array} \right\},$$

$$\mathcal{C}_3 = \bigcup_{F \in \mathcal{E}_3} \left\{ \boldsymbol{g} \in \{-1,0,1\}^{E(G)} : \begin{array}{ll} \boldsymbol{g}(e) \neq 0 & \text{iff } e \in E(F) \\ \boldsymbol{g}(\delta_F(v)) = 0 & \forall v : |\delta_F(v)| = 2 \end{array} \right\},$$

$$\mathcal{C}_4 = \bigcup_{F = (P \cup C) \in \mathcal{E}_4} \left\{ \boldsymbol{g} \in \mathbb{Z}^{E(G)} : \begin{array}{ll} \boldsymbol{g}(e) \neq 0 & \text{iff } e \in E(F) \\ \boldsymbol{g}(\delta_F(v)) = 0 & \forall v : |\delta_F(v)| \geq 2 \\ \boldsymbol{g}(e) \in \{-1,1\} & \forall e \in E(C) \\ \boldsymbol{g}(e) \in \{-2,2\} & \forall e \in E(P) \end{array} \right\},$$

$$\mathcal{C}_5 = \bigcup_{F = (C_1 \cup P \cup C_2) \in \mathcal{E}_5} \left\{ \boldsymbol{g} \in \mathbb{Z}^{E(G)} : \begin{array}{ll} \boldsymbol{g}(e) \neq 0 & \text{iff } e \in E(F) \\ \boldsymbol{g}(\delta_F(v)) = 0 & \forall v \in V(F) \\ \boldsymbol{g}(e) \in \{-1,1\} & \forall e \in E(C_1 \cup C_2) \\ \boldsymbol{g}(e) \in \{-2,2\} & \forall e \in E(P) \end{array} \right\}.$$

See Figure 3.4 for an example of a vector $\boldsymbol{g} \in \mathcal{C}_5$.

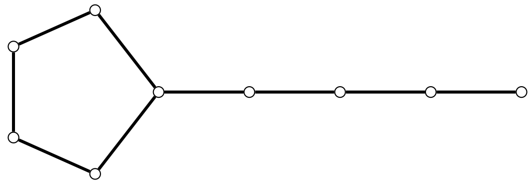

Figure 3.1: An Example of a subgraph belonging to $\mathcal{E}_4$

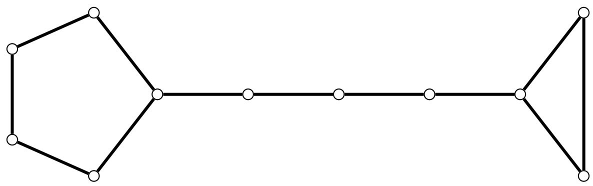

Figure 3.2: An Example of a subgraph belonging to $\mathcal{E}_5$ where P is non-empty.

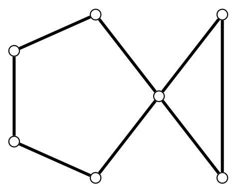

Figure 3.3: An Example of a subgraph belonging to $\mathcal{E}_5$ where P is empty.

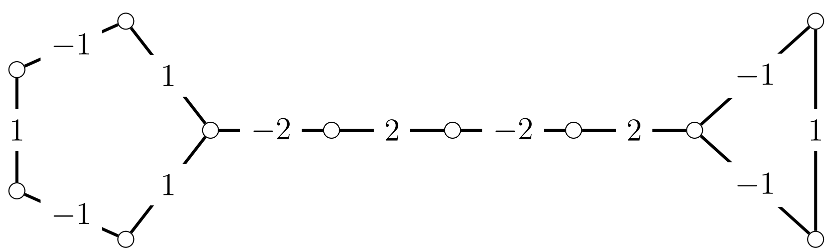

Figure 3.4: Example of a vector $\boldsymbol{g} \in \mathcal{C}_5$. Each edge e is labeled with $\boldsymbol{g}(e)$.

It is known that the vectors of $\mathcal{C}_1 \cup \cdots \cup \mathcal{C}_5$ correspond to edge-directions of $P_{\text{FMAT}}(G)$ (see e.g. [7, 82]). Let us denote by $\mathcal{C}(P_{\text{FMAT}}(G))$ the set of circuits of $P_{\text{FMAT}}(G)$ with co-prime integer components.

Lemma 6. *$\mathcal{C}(P_{\text{FMAT}}(G)) = \mathcal{C}_1 \cup \mathcal{C}_2 \cup \mathcal{C}_3 \cup \mathcal{C}_4 \cup \mathcal{C}_5$.*

Proof. Since the vectors of $\mathcal{C}_1 \cup \cdots \cup \mathcal{C}_5$ correspond to edge-directions of $P_{\text{FMAT}}(G)$, it remains to be shown that all circuits belong to one of these sets.

Let B denote the constraint matrix corresponding to the inequality constraints (3.1). In what follows, the rows of B will be indexed by $V(G) \setminus L(G)$, and the columns of B will be indexed by $E(G)$. With this notation, we can treat $\text{supp}(\mathsf{B}\boldsymbol{x})$ and $\text{supp}(\boldsymbol{x})$ as a subset of $V(G)$ or $E(G)$, respectively. Let $\boldsymbol{g} \in \mathcal{C}(P_{\text{FMAT}}(G))$, and let $G(\boldsymbol{g})$ be the subgraph of G induced by the edges in $\text{supp}(\boldsymbol{g})$. Note that, by definition, the vector $\binom{\mathsf{B}\boldsymbol{g}}{\boldsymbol{g}}$ is support-minimal.

First we note that $G(\boldsymbol{g})$ is connected. Otherwise, restricting $\boldsymbol{g}$ to the edges of any component of $G(\boldsymbol{g})$ gives a vector $\boldsymbol{f}$ with $\text{supp}(\mathsf{B}\boldsymbol{f}) \subseteq \text{supp}(\mathsf{B}\boldsymbol{g})$ and $\text{supp}(\boldsymbol{f}) \subsetneq \text{supp}(\boldsymbol{g})$, contradicting that $\boldsymbol{g}$ is a circuit.

Now, suppose that $G(\boldsymbol{g})$ contains no cycles. Let P be any edge-maximal path in $G(\boldsymbol{g})$, with endpoints u and w. Note that $\text{supp}(\mathsf{B}\boldsymbol{g}) \supseteq \{u, w\} \setminus L(G)$. Let $\boldsymbol{f} \in \{-1, 1\}^{E(G)}$ be a vector that satisfies (i) $\boldsymbol{f}(e) \neq 0$ if and only if $e \in E(P)$, and (ii) $\boldsymbol{f}(\delta_P(v)) = 0\ \forall v \neq u, w$. Note that $\boldsymbol{f} \in \mathcal{C}_3$. Then, $\text{supp}(\mathsf{B}\boldsymbol{f}) = \{u, w\} \setminus L(G) \subseteq \text{supp}(\mathsf{B}\boldsymbol{g})$, and $\text{supp}(\boldsymbol{f}) \subseteq \text{supp}(\boldsymbol{g})$. Therefore, it must be that the edges of $G(\boldsymbol{g})$ are exactly $E(P)$, and $\boldsymbol{g}(\delta_P(v)) = 0$ for all $v \in V(G) \setminus \{u, w\}$. Thus, $\boldsymbol{g} = \boldsymbol{f}$ or $\boldsymbol{g} = -\boldsymbol{f}$. In any case, $\boldsymbol{g} \in \mathcal{C}_3$.

Now, suppose that $G(\boldsymbol{g})$ contains an even cycle C. Let $\boldsymbol{f} \in \{-1, 1\}^{E(G)}$ be a vector that satisfies (i) $\boldsymbol{f}(e) \neq 0$ if and only if $e \in E(C)$, and (ii) $\boldsymbol{f}(\delta_C(v)) = 0\ \forall v \in V(C)$. Note that $\boldsymbol{f} \in \mathcal{C}_1$. Then, $\text{supp}(\mathsf{B}\boldsymbol{f}) = \emptyset \subseteq \text{supp}(\mathsf{B}\boldsymbol{g})$, and $\text{supp}(\boldsymbol{f}) \subseteq \text{supp}(\boldsymbol{g})$. Therefore, it must be that the edges of $G(\boldsymbol{g})$ are exactly $E(C)$, and $\boldsymbol{g}(\delta_C(v)) = 0$ for all $v \in V(G)$. Thus, $\boldsymbol{g} = \boldsymbol{f}$ or $\boldsymbol{g} = -\boldsymbol{f}$. In any case, $\boldsymbol{g} \in \mathcal{C}_1$.

We are left with the case where $G(\boldsymbol{g})$ contains at least one cycle, but it does not contain any even cycle. In this case, first we state an easy claim that gives some more structure for the graph $G(\boldsymbol{g})$.

Claim 1. *Under the assumption that $G(\boldsymbol{g})$ contains at least one cycle, but it does not contain an even cycle, any two odd cycles in $G(\boldsymbol{g})$ must share at most one node.*

Proof. Let $C, D \subseteq G(\boldsymbol{g})$ be two odd cycles, and suppose, for the sake of contradiction, that $|V(C) \cap V(D)| \geq 2$. Then C can be written as the union of two edge-disjoint paths $C_1 \cup C_2$ where C_1 is some sub-path of C such that $V(C_1) \cap V(D) = \{u, v\}$ where u and v are the endpoints of C_1, and $E(C_1) \cap E(D) = \emptyset$. Since D is a cycle, we can decompose D into two sub-paths D_1 and D_2 each with endpoints u and v. Since $|E(D)|$ is odd, for *exactly* one $i \in \{1, 2\}$, $|E(D_i)|$ is even. Note that since $V(C_1) \cap V(D_i) = \{u, v\}$, $C_1 \cup D_i$ is a cycle for all $i \in \{1, 2\}$, and therefore there exists $i \in \{1, 2\}$ such that $C_1 \cup D_i$ is an even cycle, a contradiction with the assumption. □

Suppose that $G(\boldsymbol{g})$ contains at least two distinct odd cycles C_1 and C_2. Since $G(\boldsymbol{g})$ is connected, then either these two cycles share a node or there exists a simple path P in $G(\boldsymbol{g})$ connecting them. In particular, we can choose P so that $E(P) \cap E(C_i) = \emptyset$ for $i \in \{1, 2\}$. Let $F = C_1 \cup P \cup C_2$ (where $E(P) = \emptyset$ if C_1 and C_2 share a node). Let $\boldsymbol{f} \in \mathbb{Z}^{E(G)}$ be a vector that satisfies (i) $\boldsymbol{f}(e) \neq 0$ if and only if $e \in E(F)$, (ii) $\boldsymbol{f}(\delta_F(v)) = 0 \; \forall v \in V(F)$, (iii) $\boldsymbol{f}(e) \in \{-1, 1\}$ for all $e \in E(C_1 \cup C_2)$, and (iv) $\boldsymbol{f}(e) \in \{-2, 2\}$ for all $e \in E(P)$. Note that $\boldsymbol{f} \in \mathcal{C}_5$. Then $\mathrm{supp}(\mathsf{B}\boldsymbol{f}) = \emptyset \subseteq \mathrm{supp}(\mathsf{B}\boldsymbol{g})$, and $\mathrm{supp}(\boldsymbol{f}) \subseteq \mathrm{supp}(\boldsymbol{g})$. Therefore, it must be that the edges of $G(\boldsymbol{g})$ are exactly $E(F)$, and $\boldsymbol{g}(\delta_G(v)) = 0$ for all $v \in V(G)$. Thus, $\boldsymbol{g} \in \mathcal{C}_5$.

Finally, suppose that $G(\boldsymbol{g})$ contains exactly one odd cycle C. If there exists a node $w \in V(C)$ such that $\boldsymbol{g}(\delta_G(w)) \neq 0$, then let $\boldsymbol{f} \in \{-1, 1\}^{E(G)}$ be a vector that satisfies (i) $\boldsymbol{f}(e) \neq 0$ if and only if $e \in E(C)$, and (ii) $\boldsymbol{f}(\delta_C(v)) = 0 \; \forall v \in V(C) \setminus \{w\}$. Note that $\boldsymbol{f} \in \mathcal{C}_2$. Then, $\mathrm{supp}(\mathsf{B}\boldsymbol{f}) = \{w\} \subseteq \mathrm{supp}(\mathsf{B}\boldsymbol{g})$, and $\mathrm{supp}(\boldsymbol{f}) \subseteq \mathrm{supp}(\boldsymbol{g})$. Therefore, it must be that the edges of $G(\boldsymbol{g})$ are exactly $E(C)$, and $\boldsymbol{g}(\delta_C(v)) = 0$ for all $v \in V(G) \setminus \{w\}$. Thus, it must be that $\boldsymbol{g} \in \mathcal{C}_2$.

We are left with the case where $\boldsymbol{g}(\delta_G(v)) = 0$ for all $v \in V(C)$. Note that this is not possible if $\mathrm{supp}(\boldsymbol{g}) = E(C)$, because C is an odd cycle. Then let P be any simple path in $G(\boldsymbol{g})$ which is inclusion-wise maximal subject to the condition that $E(P) \cap E(C) = \emptyset$ and $|V(P) \cap V(C)| = \{u\}$, where u is an endpoint of P. Let $F = C \cup P$, and let $w \in V(G)$ be the unique node such that $|\delta_F(w)| = 1$. Let $\boldsymbol{f} \in \mathbb{Z}^{E(G)}$ be a vector that satisfies (i) $\boldsymbol{f}(e) \neq 0$ if and only if $e \in E(F)$, (ii) $\boldsymbol{f}(\delta_F(v)) = 0 \; \forall v \in V(F) \setminus \{w\}$, (iii) $\boldsymbol{f}(e) \in \{-1, 1\}$ for all $e \in E(C)$, and (iv) $\boldsymbol{f}(e) \in \{-2, 2\}$ for all $e \in E(P)$. Note that $\boldsymbol{f} \in \mathcal{C}_4$. Then $\mathrm{supp}(\mathsf{B}\boldsymbol{f}) = \{w\} \setminus L(G) \subseteq \mathrm{supp}(\mathsf{B}\boldsymbol{g})$, and $\mathrm{supp}(\boldsymbol{f}) \subseteq \mathrm{supp}(\boldsymbol{g})$. Therefore, it must be that the edges of $G(\boldsymbol{g})$ are exactly $E(F)$, and $\boldsymbol{g}(\delta_F(v)) = 0$ for all $v \in V(G) \setminus \{w\}$. Thus, it must be that $\boldsymbol{g} \in \mathcal{C}_4$.

In all the above cases, $\boldsymbol{g} \in \mathcal{C}_1 \cup \cdots \cup \mathcal{C}_5$, as desired. □

Finally, we remark that if G is bipartite, then both the edge directions and the circuits of $P_{\mathrm{FMAT}}(G)$ are given by $\mathcal{C}_1 \cup \mathcal{C}_3$. Furthermore, when G is bipartite, we have that $P_{\mathrm{FMAT}}(G)$ is a 0/1 polytope. These facts are relevant to the hardness reduction in the next section.

3.1.2 Hardness Reduction

The purpose of this section is to show that the Dantzig and greatest-improvement circuit-pivot rules are NP-hard to compute. We will prove this via reduction from the directed Hamiltonian path problem. In particular, let $D = (N, F)$ be a directed graph with $n = |N|$, and let $s, t \in N$ be two given nodes. We will construct a suitable auxiliary undirected graph H, cost function $\boldsymbol{c}$, and a matching M in H with characteristic vector χ^M, such that the following holds: D contains a directed Hamiltonian s,t-path if and only if an optimal solutions to $\text{Dan}(P_{\text{FMAT}}(H), \chi^M, \boldsymbol{c})$ and $\text{Great}(P_{\text{FMAT}}(H), \chi^M, \boldsymbol{c})$ attain a certain objective function value.

We start by constructing $H = (V, E)$. For each node $v \in N \setminus \{t\}$ we create two copies v_a and v_b in V. For all $v \in N \setminus \{t\}$, we let $v_a v_b \in E$. For all arcs $uv \in F$, with $u, v \neq t$, we add an edge $u_b v_a \in E$. That is, every in-arc at a node v corresponds to an edge incident with v_a, and every out-arc at v corresponds to an edge incident with v_b. We add t in V, and for all arcs $ut \in F$, we have that $u_b t \in E$. Finally, we add nodes s' and t', where $s' s_a \in E$ and $tt' \in E$ (see Figure 3.5).

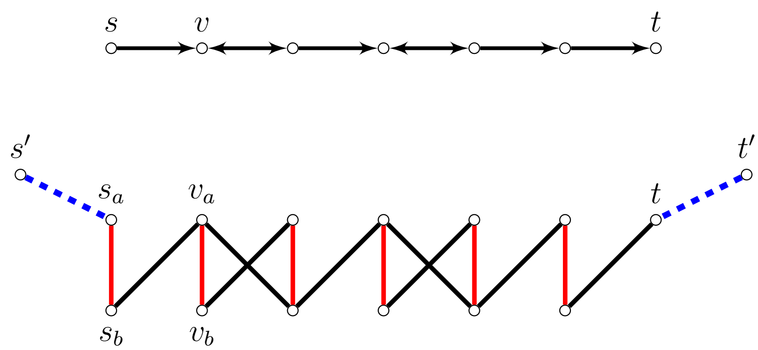

Figure 3.5: An example of a digraph D above with the corresponding auxiliary graph H below.

Now we define the cost function $\boldsymbol{c}$. We set $\boldsymbol{c}(v_a v_b) = 0$ for all $v \in N \setminus \{t\}$, $\boldsymbol{c}(s' s_a) = -W = -\boldsymbol{c}(tt')$ such that $W \in \mathbb{Z}$, $W \gg |E|$, and let all other edges have cost -1. Finally, we let

$$M = \left\{ v_a v_b : v \in N \setminus \{t\} \right\} \cup \{tt'\}$$

be a matching in H. Recall that since H is bipartite, $P_{\text{FMAT}}(H)$ is a 0/1 polytope, and its circuits are given by $\mathcal{C}_3 \cup \mathcal{C}_5$. Note also that the minimum possible objective function value of any matching is $-W - n + 1$.

Now, consider the optimization problem

$$\begin{aligned} \max \quad & -\boldsymbol{c}^\intercal \boldsymbol{g} \\ \text{s.t.} \quad & \boldsymbol{g} \in \mathcal{C}(P_{\text{FMAT}}(H)) \\ & \boldsymbol{\chi}^M + \boldsymbol{g} \in P_{\text{FMAT}}(H). \end{aligned} \tag{3.3}$$

Note that since $\boldsymbol{c}^\intercal \chi^M = W$, the optimal value of (3.3) is at most 2W + n - 1.

Theorem 9. *There exists a directed Hamiltonian s,t-path in D if and only if the optimal value of (3.3) is $2W + n - 1$.*

Proof. ($\Rightarrow$) Suppose that there exists a directed Hamiltonian s,t-path

$$P = (sv^1, v^1v^2, \cdots, v^{k-1}v^k, v^kt)$$

in D. Then P can be naturally associated to an M-alternating path P' in H with endpoints s' and t', as follows:

$$P' = (s's_a, s_as_b, s_bv_a^1, v_a^1v_b^1, v_b^1v_a^2, v_a^2v_b^2, \cdots, v_a^{k-1}v_b^{k-1}, v_b^{k-1}v_a^k, v_a^kv_b^k, v_b^kt, tt').$$

Let $\boldsymbol{g}$ be defined as

$$\boldsymbol{g}(e) := \begin{cases} 1 & \text{if } e \in E(P') \setminus M, \\ -1 & \text{if } e \in M, \\ 0 & \text{otherwise.} \end{cases}$$

Then $\boldsymbol{g} \in \mathcal{C}_3$, and is therefore a circuit of $P_{\text{FMAT}}(H)$. Note that $\chi^M + \boldsymbol{g} \in P_{\text{FMAT}}(H)$, and $-\boldsymbol{c}^\intercal \boldsymbol{g} = 2W + n - 1$. Thus, $\boldsymbol{g}$ is a feasible solution to (3.3) with the claimed objective function value.

($\Leftarrow$) Now suppose that there is a solution $\boldsymbol{g}$ to (3.3), with objective function value $2W + n - 1$. First, we argue that the support of $\boldsymbol{g}$ is indeed an M-alternating path with endpoints s' and t'.

Recall that $\boldsymbol{g} \in \mathcal{C}_1 \cup \mathcal{C}_3$. In either case, $\boldsymbol{g} \in \{1, 0, -1\}^E$. By our choice of W, since $-\boldsymbol{c}^\intercal \boldsymbol{g} = 2W + n - 1$, we have that $\boldsymbol{g}(s's_a) = 1$ and $\boldsymbol{g}(tt') = -1$. Then, since s' and t' are not in any cycles of H, $\boldsymbol{g} \in \mathcal{C}_3$ and its support is an s', t'-path. It follows that $\boldsymbol{g}$ has at most $|V| - 1$ non-zero entries. Two of the non-zero entries are $\boldsymbol{g}(s's_a)$ and $\boldsymbol{g}(tt')$, and of those that remain, exactly half have value 1. Thus,

$$-c^\intercal \boldsymbol{g} \leq 2W + \frac{1}{2}(|V| - 3) = 2W + \frac{1}{2}((2n + 1) - 3) = 2W + n - 1.$$

It is clear that the above inequality holds tight only if $\boldsymbol{g}(e) = 1$ for $\frac{1}{2}(|V| - 3)$ edges of $E \setminus \{s's_a, tt'\}$, all of which have $\boldsymbol{c}(e) = -1$, and $\boldsymbol{c}(f) = 0$ for all edges f such that $\boldsymbol{g}(f) = -1$. Since the number of edges e with $\boldsymbol{g}(e) = 1$ equals the number of edges f with $\boldsymbol{g}(f) = -1$, we have that $|\operatorname{supp}(\boldsymbol{g})| = |V| - 1$, and therefore $\operatorname{supp}(\boldsymbol{g})$ is a path P' spanning H. Furthermore, all edges of M are in $E(P')$. By removing the first and the last edge of P', and by contracting all edges of M that are the form $(v_a v_b)$ (for $v \in N$), we obtain a path that naturally corresponds to a directed Hamiltonian s', t'-path in D. □

Theorem 5. *The Greatest Improvement and Dantzig circuit-pivot rules are NP-hard.*

Proof. We will prove this by showing that the optimization problems $\operatorname{Dan}(P_{\mathrm{FMAT}}(G), \boldsymbol{x}, \boldsymbol{c})$ and $\operatorname{Great}(P_{\mathrm{FMAT}}(G), \boldsymbol{x}, \boldsymbol{c})$ are NP-hard to solve. In particular, we will show that they are hard when G is taken to be the graph H in the above hardness reduction.

For any circuit $\boldsymbol{y} \in \mathcal{C}(P_{\mathrm{FMAT}}(H))$, we have $\chi^M + 1\boldsymbol{y} \in P$, and $\chi^M + \alpha\boldsymbol{y} \notin P$ for any $\alpha > 1$. Therefore, for all $\boldsymbol{y} \in \mathcal{C}(P_{\mathrm{FMAT}}(H))$ such that $-\boldsymbol{c}^\intercal \boldsymbol{y} > 0$, we have

$$\max\{-\boldsymbol{c}^\intercal(\alpha\boldsymbol{y}) : \chi^M + \alpha\boldsymbol{y} \in P_{\mathrm{FMAT}}(H),\ \alpha > 0\} = -\boldsymbol{c}^\intercal\boldsymbol{y}.$$

Therefore, the optimization problem (3.3) is, in fact, equivalent to both $\operatorname{Dan}(P_{\mathrm{FMAT}}(H), \boldsymbol{x}, \boldsymbol{c})$ and $\operatorname{Great}(P_{\mathrm{FMAT}}(H), \boldsymbol{x}, \boldsymbol{c})$. Thus, Theorem 9 implies that there exists a directed Hamiltonian s, t-path in D if and only if the optimal values of $\operatorname{Dan}(P_{\mathrm{FMAT}}(H), \boldsymbol{x}, \boldsymbol{c})$ and $\operatorname{Great}(P_{\mathrm{FMAT}}(H), \boldsymbol{x}, \boldsymbol{c})$ are both equal to $2W + n - 1$. □

We highlight that since H is bipartite, the polytope $P_{\mathrm{FMAT}}(H)$ is 0/1. That is, this hardness result holds even for LPs defined over 0/1 polytopes.

3.1.3 Hardness implications

Here we prove that the reductions in the previous section have interesting hardness implications for the Simplex method.

Corollary 1. *Given a vertex of the bipartite matching polytope and an objective function, it is NP-hard to decide whether there is a neighboring vertex that is optimal.*

Proof. Consider again the hardness reduction in Subsection 3.1.2, and note that the optimal solution of $\operatorname{Great}(P_{\mathrm{FMAT}}(H), \chi^M, \boldsymbol{c})$ is a circuit $\boldsymbol{g}$ that corresponds to an edge-direction at χ^M. Consider the LP obtained by minimizing $\boldsymbol{c}^\intercal\boldsymbol{x}$ over $P_{\mathrm{FMAT}}(H)$, and take χ^M as an initial vertex solution. By Theorem 9, there is a neighboring optimal solution with objective function value $-W-n+1$ (the minimum possible value) if and only if the initial directed graph has a Hamiltonian path. The result follows. □

We can now prove Theorem 1, which we restate for convenience.

Theorem 1. *Given an LP and an initial feasible solution, finding the shortest (monotone) path to an optimal solution is NP-hard. Furthermore, unless P=NP, it is hard to approximate within a factor strictly better than two.*

Proof. Once again, consider the hardness reduction in Subsection 3.1.2, and the LP obtained by minimizing $\boldsymbol{c}^\intercal \boldsymbol{x}$ over $P_{\text{FMAT}}(H)$. By Theorem 9, in order for a Hamiltonian path to exist on D, the optimal solution of this LP must have objective function value $-W-n+1$, so without loss of generality, we can assume that this is the case. Take χ^M as the initial vertex solution. Under the latter assumption, as noted in the proof of the previous corollary, there is a neighboring optimal solution to χ^M if and only if D has a Hamiltonian path. This implies the following: (i) if D has a Hamiltonian path, then there is a shortest (monotone) path to an optimal solution on the 1-skeleton of $P_{\text{FMAT}}(H)$ that consists of one edge; (ii) if D does not have a Hamiltonian path, then any shortest (monotone) path to an optimal solution has at least two edges.

Then suppose, for the sake of a contradiction, that we can approximate the shortest (monotone) path to an optimal solution within a factor $\gamma < 2$. Then given any directed Hamiltonian path instance, we can reduce the problem as in Subsection 3.1.2 and use this approximation algorithm to get a path to the optimal solution of the fractional matching instance whose length is γL where L is the length of a shortest path. Since $1 < \gamma < 2$, $\gamma L < 2$ iff $L = 1$, which itself holds iff D has a Hamiltonian path. The result follows. □

As mentioned in the introduction, our result implies that for any efficiently-computable pivot rule, the Simplex method cannot be guaranteed to reach an optimal solution via a minimum number of non-degenerate pivots, unless P=NP. In a way, this result is similar in spirit to some hardness results proven about the vertices that the Simplex method can visit during its execution [48, 44, 1].

The following hardness result also holds for circuit-paths, via the exact same argument.

Theorem 10. *Given an LP and an initial feasible solution, finding the shortest (monotone) circuit-path to an optimal solution is NP-hard. Furthermore, unless P=NP, it is hard to approximate within a factor strictly better than two.*

3.2 Circuit Augmentation Algorithms

We start with the following formal definitions of approximate Greatest Improvement augmentations and of approximate Greatest Improvement circuit-pivot rules.

Definition 4. *Let $\gamma \geq 1$, $\boldsymbol{x} \in P$, and $\alpha^* \boldsymbol{g}^*$ be a Greatest Improvement augmentation at $\boldsymbol{x}$. We say that an augmentation $\alpha \boldsymbol{g}$ is a γ-approximate Greatest Improvement augmentation at $\boldsymbol{x}$, if*

$$\alpha \boldsymbol{c}^\intercal \boldsymbol{g} \geq \frac{1}{\gamma}(\alpha^* \boldsymbol{c}^\intercal \boldsymbol{g}^*).$$

That is, $\alpha \boldsymbol{g}$ is a γ-approximate Greatest Improvement augmentation if the objective function improvement gained from the augmentation $\alpha \boldsymbol{g}$ is at least a γ fraction of the objective function improvement gained from the augmentation $\alpha^* \boldsymbol{g}^*$.

Definition 5. *A circuit-pivot rule is a γ-approximate Greatest Improvement circuit-pivot rule if at any point $\boldsymbol{x} \in P$, the augmentation produced by the rule at $\boldsymbol{x}$ is a γ-approximate Greatest Improvement augmentation at $\boldsymbol{x}$.*

As mentioned in the introduction, we define

$$\delta := \max \left\{ \left| \det \begin{pmatrix} A \\ D \end{pmatrix} \right| \right\},$$

where the max is taken over all $n \times n$ submatrices $\binom{\bar{A}}{\bar{D}}$ of $\binom{A}{D}$ such that $\binom{\bar{A}}{\bar{D}}$ has rank n. Furthermore, we let ω_1 be the minimum 1-norm distance from any vertex to any facet not containing it. Formally, let $vert(P)$ be the set of vertices of P. For a given $\boldsymbol{v} \in vert(P)$, let $\mathcal{F}(\boldsymbol{v})$ be the set of feasible points of P that lie on any facet F of P with $\boldsymbol{v} \notin F$.

$$\omega_1 := \min_{\boldsymbol{v} \in vert(P),\ \boldsymbol{f} \in \mathcal{F}(\boldsymbol{v})} \|\boldsymbol{v} - \boldsymbol{f}\|_1.$$

Finally, we let M_1 be the maximum 1-norm distance between any pair of vertices; i.e,

$$M_1 := \max_{\boldsymbol{v}_1, \boldsymbol{v}_2 \in vert(P)} \|\boldsymbol{v}_1 - \boldsymbol{v}_2\|_1.$$

3.2.1 Approximate Greatest Improvement augmentations

Let us recall the statement of Lemma 1:

Lemma 1. *Consider an LP in the general form (1.1). Denote by δ the maximum absolute value of the determinant of any $n \times n$ submatrix of $\binom{A}{D}$. Let $\boldsymbol{x}_0$ be an initial feasible solution, and let $\gamma \geq 1$. Using a γ-approximate Greatest Improvement circuit-pivot rule, we can reach an optimal solution $\boldsymbol{x}^*$ of (1.1) with $\mathcal{O}\big(n\gamma \log\big(\delta\, \boldsymbol{c}^\intercal(\boldsymbol{x}^* - \boldsymbol{x}_0)\big)\big)$ augmentations. Furthermore, if all extreme points of the feasible region have integer coordinates, we can reach an optimal solution with $\mathcal{O}\big(n\gamma \log\big(\boldsymbol{c}^\intercal(\boldsymbol{x}^* - \boldsymbol{x}_0)\big)\big)$ augmentations.*

Proof. Let $\boldsymbol{x}_k$ be the solution at the k-th iteration of an augmentation algorithm. By the sign-compatible representation property of the circuits,

$$\boldsymbol{x}^* - \boldsymbol{x}_k = \sum_{i=1}^{p} \alpha_i \boldsymbol{g}^i$$

where $\boldsymbol{g}^i \in \mathcal{C}(A, D)$ and $p \leq n$. Note that, as a consequence of Proposition 3, for any i we have that $\alpha_i \geq 0$ and $\boldsymbol{x}_k + \alpha_i \boldsymbol{g}^i$ is feasible. More precisely, $A(\boldsymbol{x}_k + \alpha_i \boldsymbol{g}^i) = A\boldsymbol{x}_k + \alpha_i A\boldsymbol{g}^i = A\boldsymbol{x}_k = \boldsymbol{b}$. Furthermore, we know $D\boldsymbol{x}_k \leq \boldsymbol{d}$ and $D\boldsymbol{x}^* = D(\boldsymbol{x}_k + \sum_{i=1}^{p} \alpha_i \boldsymbol{g}^i) \leq \boldsymbol{d}$. This and the sign-compatibility of $\boldsymbol{g}^i$ and $(\boldsymbol{x}^* - \boldsymbol{x}_k)$ implies that $D(\boldsymbol{x}_k + \alpha_i \boldsymbol{g}^i) \leq \boldsymbol{d}$ for all $i = 1 \dots p$.

We then have

$$0 < \boldsymbol{c}^\intercal(\boldsymbol{x}^* - \boldsymbol{x}_k) = \boldsymbol{c}^\intercal \sum_{i=1}^{p} \alpha_i \boldsymbol{g}^i = \sum_{i=1}^{p} \alpha_i \boldsymbol{c}^\intercal \boldsymbol{g}^i \leq n\Delta,$$

where $\Delta > 0$ is the largest value of $\alpha \boldsymbol{c}^\intercal \boldsymbol{z}$ over all $\boldsymbol{z} \in \mathcal{C}(A, D)$ and $\alpha > 0$ for which $\boldsymbol{x}_k + \alpha \boldsymbol{z}$ is feasible. Note that in particular, $\Delta \geq \alpha_i \boldsymbol{c}^\intercal \boldsymbol{g}^i$ for all $i \in [p]$. Equivalently, we get

$$\Delta \geq \frac{\boldsymbol{c}^\intercal(\boldsymbol{x}^* - \boldsymbol{x}_k)}{n}.$$

Now let $\alpha \boldsymbol{z}$ be a γ-approximate Greatest Improvement augmentation applied to $\boldsymbol{x}_k$, leading to $\boldsymbol{x}_{k+1} := \boldsymbol{x}_k + \alpha \boldsymbol{z}$. Since $\alpha \boldsymbol{c}^\intercal \boldsymbol{z} \geq \frac{1}{\gamma}\Delta$, we get

$$\boldsymbol{c}^\intercal(\boldsymbol{x}_{k+1} - \boldsymbol{x}_k) = \alpha \boldsymbol{c}^\intercal \boldsymbol{z} \geq \frac{1}{\gamma}\Delta \geq \frac{\boldsymbol{c}^\intercal(\boldsymbol{x}^* - \boldsymbol{x}_k)}{\gamma n}.$$

Thus, we have at least a factor of $\beta = \frac{1}{\gamma n}$ of objective function value increase at each augmentation. Applying Lemma 4 with $\epsilon = 1/\delta^2$ then yields a solution $\bar{\boldsymbol{x}}$ with $\boldsymbol{c}^\intercal(\boldsymbol{x}^* - \bar{\boldsymbol{x}}) < 1/\delta^2$, obtained within at most $4n\gamma\,\log(\delta\,\boldsymbol{c}^\intercal(\boldsymbol{x}^* - \boldsymbol{x}_0))$ augmentations.

By Lemma 5, a vertex solution $\boldsymbol{x}'$ with $\boldsymbol{c}^\intercal\boldsymbol{x}' \geq \boldsymbol{c}^\intercal\bar{\boldsymbol{x}}$ can be reached from $\bar{\boldsymbol{x}}$ in at most n additional augmentations. It remains to prove that $\boldsymbol{x}'$ is optimal.

Suppose $\boldsymbol{x}'$ is a non-optimal vertex. There exist subsets T_1 and T_2 of $\{1, \ldots, m_D\}$ such that $\boldsymbol{x}'$ is the unique solution to

$$\begin{pmatrix} A \\ D_{T_1} \end{pmatrix} \boldsymbol{x} = \begin{pmatrix} \boldsymbol{b} \\ \boldsymbol{d}(T_1) \end{pmatrix},$$

and $\boldsymbol{x}^*$ is the unique solution to

$$\begin{pmatrix} A \\ D_{T_2} \end{pmatrix} \boldsymbol{x} = \begin{pmatrix} \boldsymbol{b} \\ \boldsymbol{d}(T_2) \end{pmatrix}.$$

Let $\delta_1 = |\det \binom{A}{D_{T_1}}|$ and $\delta_2 = |\det \binom{A}{D_{T_2}}|$. By Cramer's rule, the entries of $\boldsymbol{x}'$ are integer multiples of $\frac{1}{\delta_1}$ and the entries of $\boldsymbol{x}^*$ are integer multiples of $\frac{1}{\delta_2}$. Then, by letting $\delta' = \operatorname{lcm}(\delta_1, \delta_2)$, we have that the entries of $(\boldsymbol{x}' - \boldsymbol{x}^*)$ are integer multiples of $\frac{1}{\delta'}$. Since $\boldsymbol{c}$ is an integer vector, we have that $\boldsymbol{c}^\intercal(\boldsymbol{x}' - \boldsymbol{x}^*) \geq \frac{1}{\delta'}$, and by the definition of δ, we have that $\frac{1}{\delta'} \geq \frac{1}{\delta^2}$. This is a contradiction to the fact that $\boldsymbol{c}^\intercal(\boldsymbol{x}^* - \boldsymbol{x}') < 1/\delta^2$.

Note that if all the vertex solutions of the given LP have integer coordinates, then we can instead set $\epsilon = 1$ instead of $\epsilon = 1/\delta^2$. □

The proof of Lemma 1 closely mimics the arguments used in [37], though we note again that their proof is only done for equality form LPs. Lemma 1 also establishes that the result obtained by [37] regarding the number Greatest Improvement augmentations needed to solve an equality form LP extends to general-form LPs (trivially, by taking $\gamma = 1$). The extension of this result to general-form LPs has valuable implications on the circuit diameter, which we explore later in Section 3.4.

3.2.2 Steepest Descent Circuit-Pivot Rule

Using the approximation result developed above, we give a new bound on the number of Steepest Descent augmentations needed to solve an LP with a bounded feasible region.

Theorem 6. *Consider the LP* $\max\{\, \boldsymbol{c}^\intercal \boldsymbol{x} : \boldsymbol{x} \in P \,\}$ *of the form (1.1) where* P *is bounded. Let* ω_1 *denote the minimum 1-norm distance from any vertex* $\boldsymbol{v} \in P$ *to any facet* F *of* P *such that* $\boldsymbol{v} \notin F$*. Let* M_1 *be the maximum 1-norm distance between any pair of vertices of* P*. Using a Steepest Descent circuit-pivot rule, a circuit-augmentation algorithm reaches an optimal solution* $\boldsymbol{x}^*$ *from any initial feasible solution* $\boldsymbol{x}_0$*, performing at most*

$$\mathcal{O}\left(n^2 \frac{M_1}{\omega_1} \log\left(\delta\, \boldsymbol{c}^\intercal (\boldsymbol{x}^* - \boldsymbol{x}_0)\right)\right)$$

augmentations.

Proof. First, we can apply Lemma 5 to move from $\boldsymbol{x}_0$ to a vertex solution $\boldsymbol{x}'$ of the LP in at most n steps.

Let $\hat{\boldsymbol{z}}$ be an optimal solution to $\mathrm{Steep}(P, \boldsymbol{x}', \boldsymbol{c})$, and let $\boldsymbol{z} := \frac{1}{\|\hat{\boldsymbol{z}}\|_1} \hat{\boldsymbol{z}}$. Note that $\boldsymbol{z}$ is a circuit of P, being a rescaling of $\hat{\boldsymbol{z}} \in \mathcal{C}(A, D)$. Let $\alpha \boldsymbol{z}$ be a Steepest Descent augmentation at $\boldsymbol{x}'$. Similarly, let $\hat{\boldsymbol{z}}^{\boldsymbol{*}}$ be an optimal solution to $\mathrm{Great}(P, \boldsymbol{x}', \boldsymbol{c})$, let $\boldsymbol{z}^* := \frac{1}{\|\hat{\boldsymbol{z}}^{\boldsymbol{*}}\|_1} \hat{\boldsymbol{z}}^{\boldsymbol{*}}$ and let $\alpha^* \boldsymbol{z}^*$ be a Greatest Improvement augmentation at $\boldsymbol{x}'$. Then we have that $(\boldsymbol{c}^\intercal \hat{\boldsymbol{z}})/\|\hat{\boldsymbol{z}}\|_1 \geq (\boldsymbol{c}^\intercal \hat{\boldsymbol{z}}^{\boldsymbol{*}})/\|\hat{\boldsymbol{z}}^{\boldsymbol{*}}\|_1$, and so $\boldsymbol{c}^\intercal \boldsymbol{z} \geq \boldsymbol{c}^\intercal \boldsymbol{z}^*$. Therefore

$$\alpha \boldsymbol{c}^\intercal \boldsymbol{z} \geq \alpha \boldsymbol{c}^\intercal \boldsymbol{z}^* = \left(\frac{\alpha}{\alpha^*}\right) (\alpha^* \boldsymbol{c}^\intercal \boldsymbol{z}^*).$$

Since the augmentation $\alpha \boldsymbol{z}$ is maximal, we have that at the point $\boldsymbol{x}' + \alpha \boldsymbol{z}$, there exists some facet of our feasible region which contains $\boldsymbol{x}' + \alpha \boldsymbol{z}$ but not $\boldsymbol{x}'$. Then $\omega_1 \leq \|(\boldsymbol{x}' + \alpha \boldsymbol{z}) - \boldsymbol{x}'\|_1 = \alpha \|\boldsymbol{z}\|_1$. Since $\|\boldsymbol{z}\|_1 = 1$, it follows that $\alpha \geq \omega_1$. Since $\boldsymbol{x}' + \alpha^* \boldsymbol{z}^*$ is feasible, we have that $\|(\boldsymbol{x}' + \alpha^* \boldsymbol{z}^*) - \boldsymbol{x}'\|_1$ is at most the maximum 1-norm distance from $\boldsymbol{x}'$ to any other feasible point. As above, it follows that α^* is at most the maximum 1-norm distance from $\boldsymbol{x}'$ to any other feasible point. Since the function $f(\boldsymbol{y}) = \|\boldsymbol{y} - \boldsymbol{x}'\|_1$ is convex, this maximum is achieved at a vertex. It follows that $\alpha^* \leq M_1$.

Given these bounds on α and α^*, it follows that

$$\alpha \boldsymbol{c}^\intercal \boldsymbol{z} \geq \left(\frac{\omega_1}{M_1}\right) (\alpha^* \boldsymbol{c}^\intercal \boldsymbol{z}^*). \tag{3.4}$$

Now let $\bar{\boldsymbol{x}} = \boldsymbol{x}' + \alpha \boldsymbol{z}$. By Lemma 5, a vertex solution $\hat{\boldsymbol{x}}$ can be found from $\bar{\boldsymbol{x}}$ in at most $n-1$ additional augmentations (e.g., using again Steepest Descent augmentations, but on a sequence of face-restricted LPs) with $\boldsymbol{c}^\intercal \hat{\boldsymbol{x}} \geq \boldsymbol{c}^\intercal \bar{\boldsymbol{x}}$. Then we have that $\hat{\boldsymbol{x}} - \bar{\boldsymbol{x}}$ is an $\left(\frac{\omega_1}{M_1}\right)$-approximate Greatest Improvement augmentation at $\boldsymbol{x}'$, and since $\hat{\boldsymbol{x}}$ is also

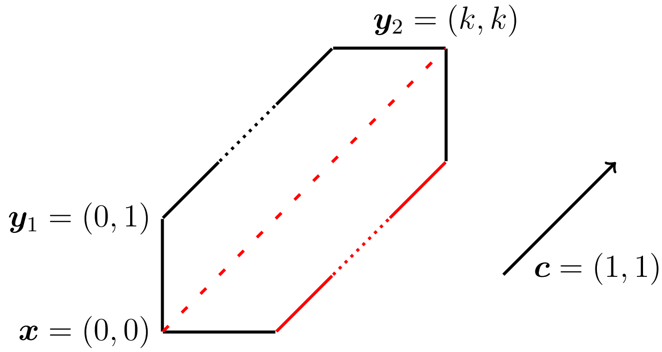

Figure 3.6: This gives a family of examples (parameterized by k) where i) moving along the edges incident at a vertex yields an arbitrarily bad approximation of moving along the Greatest Improvement circuit, and ii) a Steepest Descent augmentation at $\boldsymbol{x}$ is a (tight) $\frac{M_1}{\omega_1}$-approximate Greatest Improvement augmentation. This polygon has vertices $\boldsymbol{x} = (0,0), \boldsymbol{y}_1 = (0,1), \boldsymbol{y}_2 = (k,k), (k,k-1), (k-1,k)$ and $(1,0)$. One can check that at $\boldsymbol{x}$, $\boldsymbol{y}_1$ is both a Steepest Descent augmentation as well as a steepest edge, while $\boldsymbol{y}_2$ is a Greatest Improvement augmentation. We have that $\boldsymbol{c}^\intercal \boldsymbol{y}_1 = \frac{1}{2k}\boldsymbol{c}^\intercal \boldsymbol{y}_2 = \frac{\omega_1}{M_1}\boldsymbol{c}^\intercal \boldsymbol{y}_2$.

a vertex, we can continue to apply this procedure. Since it takes at most n Steepest Descent augmentations to find such an $\left(\frac{\omega_1}{M_1}\right)$-approximate Greatest Improvement augmentation, it follows from Lemma 1 that from an initial solution $\boldsymbol{x}_0$, we can reach $\boldsymbol{x}^*$ in $\mathcal{O}\left(n^2 \frac{M_1}{\omega_1} \log\left(\delta \boldsymbol{c}^\intercal (\boldsymbol{x}^* - \boldsymbol{x}_0)\right)\right)$ Steepest Descent augmentations. □

We note that the inequality (3.4) yields the following corollary.

Corollary 2. *Let $\boldsymbol{x}$ be a vertex solution of an LP with bounded feasible region. A Steepest Descent circuit augmentation at $\boldsymbol{x}$ is an $\left(\frac{M_1}{\omega_1}\right)$-approximate Greatest Improvement augmentation.*

The example given by Figure 3.6 shows that the approximation factor $\frac{M_1}{\omega_1}$ can be tight.

3.3 Asymmetric Steepest Descent Circuit-Pivot Rule

Given $P = \{\boldsymbol{x} \in \mathbb{R}^n : A\boldsymbol{x} = \boldsymbol{b},\, D\boldsymbol{x} \leq \boldsymbol{d}\}$, a point $\boldsymbol{x} \in P$ (considered to be fixed), and $\boldsymbol{z} \in \ker(A)$, let

$$N(\boldsymbol{z}; D, \boldsymbol{x}) \coloneqq \sum_{\substack{i\in[m_D]:\\ D_i\boldsymbol{x}<d(i)\\ D_i\boldsymbol{z}\geq 0}} \left(\frac{1}{d(i) - D_i\boldsymbol{x}}\right) D_i\boldsymbol{z}.$$

Note that if we consider $\boldsymbol{x}$ to be fixed, then the coefficient $\left(\frac{1}{d(i)-D_i\boldsymbol{x}}\right)$ is a constant independent of $\boldsymbol{z}$.

Definition 6. *Given an LP of the form (1.1) with feasible region P, an Asymmetric Steepest Descent (ASD) circuit at $\boldsymbol{x}$ is a circuit $\boldsymbol{g} \in \mathcal{C}(A, D)$ that maximizes $\frac{\boldsymbol{c}^\intercal \boldsymbol{z}}{N(\boldsymbol{z};D,\boldsymbol{x})}$ over all $\boldsymbol{z} \in \mathcal{C}(A, D)$ such that $\boldsymbol{x} + \epsilon\boldsymbol{z} \in P$ for some $\epsilon > 0$.*

We take a moment to justify the name "Asymmetric Steepest Descent." We remind the reader that the authors of [25] extend the Steepest Descent circuit-pivot rule of [37] to general form LPs by defining it to choose the circuit that maximizes $\boldsymbol{c}^\intercal\boldsymbol{g}/\|D\boldsymbol{g}\|_1$. Although it has the potential to be confusing in the context of the rest of this thesis, it is *this* definition of Steepest Descent that inspires the name "Asymmetric Steepest Descent" – by way of contrast. This is in part because the authors of [25] introduce a framework to compute Steepest Descent circuits (according to their definition), and we will use the same framework to compute ASD circuits. Further, while the expression $\|D\boldsymbol{g}\|_1$ can be thought of as treating all rows of D "symmetrically", the ASD circuit-pivot rule treats the rows of D "asymmetrically": When evaluating the expression $N(\boldsymbol{z}; D, \boldsymbol{x})$, we only consider the inequalities that $\boldsymbol{z}$ is "tightening," and furthermore each of these rows affect the expression differently depending on how close they are to tight at $\boldsymbol{x}$.

The optimization problem that arises when following the ASD circuit-pivot rule will be called ASD$(P, \boldsymbol{x}, \boldsymbol{c})$, and is defined as follows:

$$\begin{aligned} \max \quad & \frac{\boldsymbol{c}^\intercal\boldsymbol{g}}{N(\boldsymbol{g}; D, \boldsymbol{x})} \\ \text{s.t.} \quad & \boldsymbol{g} \in \mathcal{C}(A, D), \\ & \boldsymbol{g} \in \mathcal{C}(\boldsymbol{x}). \end{aligned}$$

Note that the value of $\frac{\boldsymbol{c}^\intercal\boldsymbol{z}}{N(\boldsymbol{z};D,\boldsymbol{x})}$ does not change when we scale $\boldsymbol{z}$ by some positive multiple. It is possible that $N(\boldsymbol{z}; D, \boldsymbol{x}) = 0$. In such a case, if $\boldsymbol{c}^\intercal\boldsymbol{z} = 0$, we will regard

$\frac{\boldsymbol{c}^\intercal \boldsymbol{z}}{N(\boldsymbol{z};D,\boldsymbol{x})}$ to be 0, and otherwise we will regard it to be positive or negative infinity, depending on the sign of $\boldsymbol{c}^\intercal \boldsymbol{z}$. In particular, this can happen if the given LP is unbounded. In such a case, for any point $\boldsymbol{x} \in P$, there exists a circuit $\boldsymbol{g}$ with $\boldsymbol{c}^\intercal \boldsymbol{g} > 0$ such that $\boldsymbol{x} + \alpha \boldsymbol{g} \in P$ for all $\alpha > 0$. This latter condition implies that $D_i \boldsymbol{g} \leq 0$ for all i, and thus $N(\boldsymbol{g}; D, \boldsymbol{x}) = 0$. This implies the following:

Lemma 7. *An LP of the form (1.1) is unbounded if and only if for any feasible solution $\boldsymbol{x}$, an ASD circuit $\boldsymbol{g}$ at $\boldsymbol{x}$ has $\frac{\boldsymbol{c}^\intercal \boldsymbol{g}}{N(\boldsymbol{g};D,\boldsymbol{x})} = \infty$. Furthermore, in such a case, any circuit $\boldsymbol{g}$ with $\frac{\boldsymbol{c}^\intercal \boldsymbol{g}}{N(\boldsymbol{g};D,\boldsymbol{x})} = \infty$ is an ASD circuit.*

3.3.1 Computing ASD circuits

We will rely on some definitions and results in [25], restated here. The authors define the cone $C_{A,D}$ to be

$$C_{A,D} = \{ (\boldsymbol{z}, \boldsymbol{y}^+, \boldsymbol{y}^-) \in \mathbb{R}^{n+2m_D} : A\boldsymbol{z} = \boldsymbol{0},\ D\boldsymbol{z} = \boldsymbol{y}^+ - \boldsymbol{y}^-,\ \boldsymbol{y}^+, \boldsymbol{y}^- \geq \boldsymbol{0} \}.$$

They then show that $C_{A,D}$ is generated by extreme rays of two forms. First are "trivial" extreme rays of the form $(\boldsymbol{0}, \boldsymbol{e}_j, \boldsymbol{e}_j)$ where $\boldsymbol{e}_j$ denotes an elementary coordinate vector (of the appropriate dimension). Second are extreme rays of the form $(\boldsymbol{g}, \boldsymbol{y}^+, \boldsymbol{y}^-)$ where $\boldsymbol{g} \in C(A, D)$, $\boldsymbol{y}_i^+ = \max\{0, D_i \boldsymbol{g}\}$, and $\boldsymbol{y}_i^- = \max\{0, -D_i \boldsymbol{g}\}$. They show that not all vectors of the first form generate extreme rays of $C_{A,D}$, but all vectors of the second form do (See Theorem 3 of [25]).

Given a point $\boldsymbol{x} \in P$, they further define the cone $C_{A,D,\boldsymbol{x}}$ to be the cone $C_{A,D}$ intersected with the hyperplanes $\boldsymbol{y}_i^+ = 0$ for all i such that $D_i \boldsymbol{x} = \boldsymbol{d}(i)$. They show (Theorem 5 in [25]) that the (nontrivial) extreme rays of $C_{A,D,\boldsymbol{x}}$ correspond exactly to those circuits $\boldsymbol{g}$ such that $\boldsymbol{g} \in \mathcal{C}(\boldsymbol{x})$. Furthermore, they show that for all vectors $\boldsymbol{u}$ such that $\boldsymbol{x} + \boldsymbol{u} \in P$, $(\boldsymbol{u}, \boldsymbol{y}^+, \boldsymbol{y}^-) \in C_{A,D,\boldsymbol{x}}$ where $\boldsymbol{y}_i^+ = \max\{0, D_i \boldsymbol{u}\}$, and $\boldsymbol{y}_i^- = \max\{0, -D_i \boldsymbol{u}\}$.

We will take a similar approach as [25] to prove the following theorem:

Theorem 11. *Consider an LP (P) of the form (1.1) with feasible region P and a feasible solution $\boldsymbol{x}' \in P$. In polynomial time, we can:*

- *Determine if (P) is unbounded.*
- *If (P) is not unbounded, determine if $\boldsymbol{x}'$ is optimal.*
- *If (P) is not unbounded and $\boldsymbol{x}'$ is not optimal, compute an optimal solution to* $\mathrm{ASD}(P, \boldsymbol{x}', \boldsymbol{c})$

Note that we are not claiming to be able to compute an ASD circuit in any circumstance. In particular, if the original LP is unbounded, then the method described here will not necessarily return an ASD circuit, but will instead certify unboundedness (If, however, the certificate is also a circuit, it will be an ASD circuit). Unfortunately this implies that computing an ASD circuit is at least as hard as determining whether or not an LP is unbounded. This casts doubt on the possibility that an ASD circuit can be computed without using an LP solver as a subroutine.

Similarly to [25], in order to prove Theorem 11, we will intersect $C_{A,D,\boldsymbol{x}}$ with an appropriate halfspace, creating a polyhedron $P_{A,D,\boldsymbol{x}}$ whose optimal solution with respect to $\boldsymbol{c}$ will be an ASD circuit (if $\boldsymbol{x}$ is not optimal for (P) and if (P) is not unbounded). In particular, we consider the following polyhedron:

Definition 7. *Let $P_{A,D,\boldsymbol{x}}$ be the polyhedron obtained from $C_{A,D,\boldsymbol{x}}$ by intersecting it with the halfspace defined by the inequality*

$$\boldsymbol{h}^\intercal \boldsymbol{y}^+ = \sum_{\substack{i \in [m_D] \\ D_i \boldsymbol{x} < \boldsymbol{d}(i)}} \left(\frac{1}{\boldsymbol{d}(i) - D_i \boldsymbol{x}} \right) \boldsymbol{y}_i^+ \leq 1.$$

Consider an extreme ray generator $(\boldsymbol{g}, \boldsymbol{y}^+, \boldsymbol{y}^-)$ of $C_{A,D,\boldsymbol{x}}$ with $\boldsymbol{g} \neq \boldsymbol{0}$. Then since $\boldsymbol{y}_i^+ = \max\{0, D_i \boldsymbol{g}\}$, we have that for all $i \in [m_D]$ with $D_i \boldsymbol{g} \geq 0$, $D_i \boldsymbol{g} = \boldsymbol{y}^+$. This leads to the following observation:

Observation 1. *Let $(\boldsymbol{g}, \boldsymbol{y}^+, \boldsymbol{y}^-)$ generate an extreme ray of $C_{A,D,\boldsymbol{x}}$ with $\boldsymbol{g} \neq \boldsymbol{0}$. Then*

$$N(\boldsymbol{g}; D, \boldsymbol{x}) = \sum_{\substack{i \in [m_D] \\ D_i \boldsymbol{x} < \boldsymbol{d}(i)}} \left(\frac{1}{\boldsymbol{d}(i) - D_i \boldsymbol{x}} \right) \boldsymbol{y}_i^+ = \boldsymbol{h}^\intercal \boldsymbol{y}^+.$$

Lemma 8. *Let (P) be an LP of the form (1.1) with feasible region P, and let $\boldsymbol{x}' \in P$. Then (P) is unbounded if and only if the LP problem $(P') = \max\{\boldsymbol{c}^\intercal \boldsymbol{z} : (\boldsymbol{z}, \boldsymbol{y}^+, \boldsymbol{y}^-) \in P_{A,D,\boldsymbol{x}'}\}$ is unbounded.*

Proof. Suppose that (P) is unbounded. Then there exists a vector $\boldsymbol{u} \in \mathbb{R}^n$ such that $\boldsymbol{x}' + \alpha \boldsymbol{u} \in P$ for all $\alpha > 0$ and $\boldsymbol{c}^\intercal \boldsymbol{u} > 0$. By Theorem 5 of [25], $\alpha(\boldsymbol{u}, \boldsymbol{y}^+, \boldsymbol{y}^-) \in C_{A,D,\boldsymbol{x}_0}$ for all $\alpha > 0$, where $\boldsymbol{y}_i^+ = \max\{0, D_i \boldsymbol{u}\}$ and $\boldsymbol{y}_i^- = \max\{0, -D_i \boldsymbol{u}\}$ for all i. It remains to show that $\alpha(\boldsymbol{u}, \boldsymbol{y}^+, \boldsymbol{y}^-) \in P_{A,D,\boldsymbol{x}'}$ for all $\alpha > 0$. That is, we must show that $\boldsymbol{h}^\intercal \boldsymbol{y}^+ = 0$ (Note that $\boldsymbol{h}$ and $\boldsymbol{y}^+$ are both $\geq \boldsymbol{0}$, so $\boldsymbol{h}^\intercal \boldsymbol{y}^+ \geq 0$). Since $\boldsymbol{x}' + \alpha \boldsymbol{u} \in P$ for all $\alpha > 0$, we have that $D\boldsymbol{u} \leq \mathbf{0}$, and so $\boldsymbol{y}^+ = \mathbf{0}$. Thus, $\boldsymbol{h}^\intercal \boldsymbol{y}^+ = 0$, so $\alpha(\boldsymbol{u}, \boldsymbol{y}^+, \boldsymbol{y}^-) \in P_{A,D,\boldsymbol{x}'}$ for all $\alpha > 0$, and so (P') is unbounded, as desired.

Now suppose instead that (P') is unbounded. Then there exists $(\boldsymbol{u}, \boldsymbol{y}^+, \boldsymbol{y}^-)$ such that $\alpha(\boldsymbol{u}, \boldsymbol{y}^+, \boldsymbol{y}^-) \in P_{A,D,\boldsymbol{x}'}$ for all $\alpha > 0$ and $\boldsymbol{c}^\intercal \boldsymbol{u} > 0$. Then $\boldsymbol{h}^\intercal \boldsymbol{y}^+ = 0$, and so $\boldsymbol{y}_i^+ = 0$ for all $i \in [m_D]$ such that $D_i \boldsymbol{x}' < \boldsymbol{d}(i)$. Furthermore, we have by the definition of $P_{A,D,\boldsymbol{x}'}$ that $\boldsymbol{y}_i^+ = 0$ for all i such that $D_i \boldsymbol{x}' = \boldsymbol{d}(i)$. Then since $D_i \boldsymbol{u} = \boldsymbol{y}_i^+ - \boldsymbol{y}_i^-$, we have that $D_i \boldsymbol{u} \leq 0$ for all $i \in [m_D]$. This implies that $\boldsymbol{x}' + \alpha \boldsymbol{u} \in P$ for all $\alpha > 0$. Since $\boldsymbol{c}^\intercal \boldsymbol{u} > 0$ this implies that (P) is unbounded, as desired. □

We can now prove Theorem 11.

Proof of Theorem 11. Consider the LP problem $(P') = \max\{\, \boldsymbol{c}^\intercal \boldsymbol{z} : (\boldsymbol{z}, \boldsymbol{y}^+, \boldsymbol{y}^-) \in P_{A,D,\boldsymbol{x}'} \,\}$. Consider the result of solving (P') starting from the vertex $\mathbf{0}$ using any LP solver that runs in polynomial time and returns an optimal extreme point solution if one exists. If (P) is unbounded, then by Lemma 8, solving (P') will determine that (P') is unbounded, and this will certify that (P) is unbounded. Then we may assume that (P) is bounded. We have that $\boldsymbol{x}'$ is optimal for (P) if and only if $\mathbf{0}$ is optimal for (P'), so if $\boldsymbol{x}'$ is optimal, solving (P') will determine as much. Then we may assume further that $\boldsymbol{x}'$ is not optimal for (P).

In this case, (P') is also not unbounded, and $\mathbf{0}$ is not optimal for (P'). Then solving (P') returns some vertex solution $(\boldsymbol{z}, \boldsymbol{y}^+, \boldsymbol{y}^-)$. It remains to show that $\boldsymbol{z}$ is the scaling of an optimal solution to $\mathrm{ASD}(P, \boldsymbol{x}', \boldsymbol{c})$ (Note that $\boldsymbol{z}$ may not itself be an optimal solution to $\mathrm{ASD}(P, \boldsymbol{x}', \boldsymbol{c})$ if it does not have co-prime integer components, and is therefore not in $\mathcal{C}(A, D)$).

We have that $\boldsymbol{z} \neq \mathbf{0}$ since otherwise $\mathbf{0}$ is optimal for (P'), contradicting our assumption that it is not. Then $\boldsymbol{z}$ is a scaling of a circuit $\boldsymbol{g}$ in $\mathcal{C}(A, D)$. Now, every circuit in $\mathcal{C}(A, D)$ which is strictly feasible at $\boldsymbol{x}'$ corresponds to some vector $(\boldsymbol{z}', \boldsymbol{y}'^+, \boldsymbol{y}'^-)$ which is either a vertex of $P_{A,D,\boldsymbol{x}'}$ or generates an extreme ray of $P_{A,D,\boldsymbol{x}'}$. Note that if any such vector generates an extreme ray, then $\boldsymbol{c}^\intercal \boldsymbol{z}' \leq 0$, as (P') is not unbounded. Therefore, the strictly feasible circuits at $\boldsymbol{x}'$ with positive objective function value all correspond to vertices of $P_{A,D,\boldsymbol{x}'}$.

Let $\boldsymbol{g}'$ be a circuit in $\mathcal{C}(A, D)$ which is strictly feasible at $\boldsymbol{x}'$ and with $\boldsymbol{c}^\intercal \boldsymbol{g}' > 0$. Then $\boldsymbol{g}'$ corresponds to a vertex $(\boldsymbol{z}', \boldsymbol{y}'^+, \boldsymbol{y}'^-)$ of $P_{A,D,\boldsymbol{x}'}$. Since $(\boldsymbol{z}, \boldsymbol{y}^+, \boldsymbol{y}^-)$ is optimal for (P'), we have that $\boldsymbol{c}^\intercal \boldsymbol{z} \geq \boldsymbol{c}^\intercal \boldsymbol{z}'$. Since $C_{A,D,\boldsymbol{x}'}$ has only the vertex $\mathbf{0}$, all vertices of $P_{A,D,\boldsymbol{x}'}$ other than $\mathbf{0}$ lie in the hyperplane $\boldsymbol{h}^\intercal \boldsymbol{y}^+ = 1$. This, combined with Observation 1 gives that

$$\frac{\boldsymbol{c}^\intercal \boldsymbol{z}}{N(\boldsymbol{z}; D, \boldsymbol{x}')} \geq \frac{\boldsymbol{c}^\intercal \boldsymbol{z}'}{N(\boldsymbol{z}'; D, \boldsymbol{x}')}.$$

Since $\boldsymbol{z}$ and $\boldsymbol{z}'$ are scalings of $\boldsymbol{g}$ and $\boldsymbol{g}'$, respectively, we have that

$$\frac{\boldsymbol{c}^\intercal \boldsymbol{g}}{N(\boldsymbol{g}; D, \boldsymbol{x}')} \geq \frac{\boldsymbol{c}^\intercal \boldsymbol{g}'}{N(\boldsymbol{g}'; D, \boldsymbol{x}')}.$$

Then $\boldsymbol{g}$ is an ASD circuit (and can be easily recovered from $\boldsymbol{z}$), as desired. □

3.3.2 Solving LPs with ASD Circuits

The purpose of this section is to prove the following:

Theorem 12. *Consider an LP (P) of the form (1.1) with feasible region P, and let $\boldsymbol{x}^0$ be any initial feasible solution of (P). If (P) has an optimal solution, then some optimal solution $\boldsymbol{x}^*$ to (P) can be reached starting from $\boldsymbol{x}^0$ in at most*

$$\mathcal{O}\left(m_D n \log\left(\delta \boldsymbol{c}^\intercal(\boldsymbol{x}^* - \boldsymbol{x}^0)\right)\right)$$

ASD circuit augmentations.

Proof. We will show that at any feasible point $\boldsymbol{x}'$, an ASD circuit augmentation is an m_D-approximate Greatest Improvement augmentation. Then the result follows from Lemma 1.

Let $\boldsymbol{g}'$ be an optimal solution to $\text{ASD}(P, \boldsymbol{x}', \boldsymbol{c})$ and let $\boldsymbol{g}^*$ be an optimal solution to $\text{Great}(P, \boldsymbol{x}', \boldsymbol{c})$. Then by definition, we have that

$$\frac{\boldsymbol{c}^\intercal \boldsymbol{g}'}{N(\boldsymbol{g}'; D, \boldsymbol{x}')} \geq \frac{\boldsymbol{c}^\intercal \boldsymbol{g}^*}{N(\boldsymbol{g}^*; D, \boldsymbol{x}')}.$$

Let $\boldsymbol{z}' = \alpha' \boldsymbol{g}'$ and $\boldsymbol{z}^* = \alpha^* \boldsymbol{g}^*$ be the maximal ASD and Greatest Improvement circuit augmentations at $\boldsymbol{x}'$, respectively. Then since $\frac{\boldsymbol{c}^\intercal \boldsymbol{z}}{N(\boldsymbol{z}; D, \boldsymbol{x}')}$ is invariant under scaling of $\boldsymbol{z}$, we have that

$$\frac{\boldsymbol{c}^\intercal \boldsymbol{z}'}{N(\boldsymbol{z}'; D, \boldsymbol{x}')} \geq \frac{\boldsymbol{c}^\intercal \boldsymbol{z}^*}{N(\boldsymbol{z}^*; D, \boldsymbol{x}')},$$

and therefore

$$\boldsymbol{c}^\intercal \boldsymbol{z}' \geq \frac{N(\boldsymbol{z}'; D, \boldsymbol{x}')}{N(\boldsymbol{z}^*; D, \boldsymbol{x}')} \boldsymbol{c}^\intercal \boldsymbol{z}^*.$$

Then it remains to show that $\frac{N(\boldsymbol{z}; D, \boldsymbol{x}')}{N(\boldsymbol{z}^*; D, \boldsymbol{x}')} \geq \frac{1}{m_D}$. In fact, we will show that $N(\boldsymbol{z}; D, \boldsymbol{x}) \geq 1$ and $N(\boldsymbol{z}; D, \boldsymbol{x}) \leq m_D$ for all $\boldsymbol{x}$ and *all* maximal augmentations $\boldsymbol{z}$ at $\boldsymbol{x}$, regardless of whether or not $\boldsymbol{z}$ is a circuit. Since $\boldsymbol{x}+\boldsymbol{z}$ is feasible, Each summand in

$$\sum_{\substack{i\in[m_D]:\\ D_i\boldsymbol{x}<d(i)\\ D_i\boldsymbol{z}\geq 0}} \left(\frac{1}{d(i)-D_i\boldsymbol{x}}\right) D_i\boldsymbol{z}$$

is in $[0,1]$. Furthermore, since $\boldsymbol{z}$ is a maximal augmentation at $\boldsymbol{x}$, there exists an index $l\in[m_D]$ such that $D_l\boldsymbol{z}=\boldsymbol{d}(l)-D_l\boldsymbol{x}$. That is, there exists a summand that equals 1, so $N(\boldsymbol{z};D,\boldsymbol{x})\geq 1$. Similarly, since each summand is in $[0,1]$ and since there are at most m_D of them, we have that $N(\boldsymbol{z};D,\boldsymbol{x})\leq m_D$, as desired. □

Together, Theorems 11 and 12 give the following corollary:

Corollary 3. *Consider an LP* (P) *of the form (1.1). A circuit-augmentation algorithm with an ASD circuit-pivot rule reaches an optimal solution in polynomial time.*

3.4 Circuit Diameter

Here we explore the implications that Lemma 1 has when bounding the circuit diameter of general polyhedra as well as for LPs defined over lattice polytopes. We remind the reader that in this section, we assume that polyhedra are represented with *minimal* descriptions, as defined in Definition 3. We will first prove a lemma that will not only be useful for a later proof, but is also of independent interest for the study of circuits.

Lemma 3. *Let* P *be any polyhedron where* $\{\boldsymbol{x}\in\mathbb{R}^n : A\boldsymbol{x}=\boldsymbol{b},\ D\boldsymbol{x}\leq\boldsymbol{d}\}$ *and* $\{\boldsymbol{x}\in\mathbb{R}^n : A'\boldsymbol{x}=\boldsymbol{b}',\ D'\boldsymbol{x}\leq\boldsymbol{d}'\}$ *are both minimal descriptions of* P*. Then* $\mathcal{C}(A,D)=\mathcal{C}(A',D')$*.*

Proof. We will first show that $\ker(A)=\ker(A')$. Consider any point $\boldsymbol{x}'\in P$ such that $D_i\boldsymbol{x}'<d(i)$ for all $i\in[m_D]$. Note that such a point exists because we assume that P has more than one solution (see Chapter 2). For any vector $\boldsymbol{z}\in\mathbb{R}^n$, we have that $\boldsymbol{x}'+\varepsilon\boldsymbol{z}\in P$ for some $\varepsilon>0$ iff $\boldsymbol{z}\in\ker(A)$. Likewise, $\boldsymbol{x}'+\varepsilon\boldsymbol{z}\in P$ for some $\varepsilon>0$ iff $\boldsymbol{z}\in\ker(A')$. Then $\ker(A)=\ker(A')$, as desired.

We will now show that $\mathcal{C}(A',D)=\mathcal{C}(A',D')$. As shown in Theorem 6.17 of [32], each inequality in a minimal description of P defines a facet of P. Furthermore, in any description of P, each of its facets must be described by some inequality. Then it suffices to show that given any minimal description of P, replacing any facet-defining inequality with any other inequality that defines that same facet gives a new minimal description with the same set of circuits. Given this, one can replace the inequalities of $D\boldsymbol{x} \leq \boldsymbol{d}$ with inequalities of $D'\boldsymbol{x} \leq \boldsymbol{d}'$ one at a time, giving the desired result.

Then suppose that the system $D''\boldsymbol{x} \leq \boldsymbol{d}''$ is obtained from the system $D\boldsymbol{x} \leq \boldsymbol{d}$ by replacing an inequality $D_i^\intercal \boldsymbol{x} \leq \boldsymbol{d}(i)$ with the inequality $\boldsymbol{h}^\intercal \boldsymbol{x} \leq u$ (where we will then say that $D_i'' = \boldsymbol{h}$ and $\boldsymbol{d}''(i) = u$), and let F be the facet of P that they both define. We have that $\boldsymbol{g}$ is a circuit in $\mathcal{C}(A', D'')$ if and only if it is the unique (up to scaling) solution to a system

$$\begin{bmatrix} A' \\ D_J'' \end{bmatrix} \boldsymbol{x} = \boldsymbol{0}$$

for some $J \subseteq [m_D]$ such that rank $\binom{A'}{D_J''} = n - 1$ and $\binom{A'}{D_J''}$ has full row-rank. Consider the flat H affinely spanned by the vertices of F. Then we have both that

$$H = \{\, \boldsymbol{x} \in \mathbb{R}^n : D_i''\boldsymbol{x} = \boldsymbol{d}'(i) \,\} \cap \{\, \boldsymbol{x} \in \mathbb{R}^n : A'\boldsymbol{x} = \boldsymbol{b}' \,\}$$

and that

$$H = \{\, \boldsymbol{x} \in \mathbb{R}^n : D_i\boldsymbol{x} = \boldsymbol{d}(i) \,\} \cap \{\, \boldsymbol{x} \in \mathbb{R}^n : A'\boldsymbol{x} = \boldsymbol{b}' \,\}.$$

Then as before, $\binom{A'}{D_i''^\intercal}\boldsymbol{x} = \binom{\boldsymbol{b}'}{\boldsymbol{d}''(i)}$ if and only if $\binom{A'}{D_i^\intercal}\boldsymbol{x} = \binom{\boldsymbol{b}'}{\boldsymbol{d}(i)}$, and therefore $\binom{A'}{D_i''^\intercal}$ and $\binom{A'}{D_i^\intercal}$ have the same kernel. Since all other inequalities of $D\boldsymbol{x} \leq \boldsymbol{d}$ and $D''\boldsymbol{x} \leq \boldsymbol{d}''$ are the same, this implies that

$$\begin{bmatrix} A' \\ D_J'' \end{bmatrix} \boldsymbol{g} = \boldsymbol{0}$$

if and only if

$$\begin{bmatrix} A' \\ D_J \end{bmatrix} \boldsymbol{g} = \boldsymbol{0},$$

and so $\boldsymbol{g} \in \mathcal{C}(A', D)$ if and only if $\boldsymbol{g} \in \mathcal{C}(A', D'')$, as desired. □

We recall the statement of Theorem 7:

Theorem 7. *There exists a polynomial function $f(m, \alpha)$ that bounds above the circuit diameter of any rational polyhedron $P = \{\boldsymbol{x} \in \mathbb{R}^n : A\boldsymbol{x} = \boldsymbol{b}, D\boldsymbol{x} \leq \boldsymbol{d}\}$ with m row constraints and maximum encoding length among the coefficients in its description equal to α.*

Proof. We may assume that P is pointed and has more than one vertex, as otherwise the statement is trivial. Then rank $\binom{A}{D} = n$, and so $m = m_A + m_D \geq n$. Recall that, by possibly scaling, we can assume that all coefficients in its description (i.e., all entries of $A, D, \boldsymbol{b}, \boldsymbol{d}$)

are integers. Let $\boldsymbol{x}$ and $\bar{\boldsymbol{x}}$ be two vertices of P. Let $T := \{i : D_i\bar{\boldsymbol{x}} = \boldsymbol{d}(i)\}$. Let $\boldsymbol{c}^\intercal$ be the vector obtained by adding the rows of A and D_T. By construction, using this vector as an objective function, $\bar{\boldsymbol{x}}$ is the unique optimal solution to the LP problem (1.1). Lemma 1 shows that we can reach $\bar{\boldsymbol{x}}$ from $\boldsymbol{x}$ with $\mathcal{O}(n \log\left(\delta\, \boldsymbol{c}^\intercal(\bar{\boldsymbol{x}} - \boldsymbol{x})\right))$ augmentations.

We define

$$\bar{\delta} := \max\left\{\left|\det\begin{pmatrix}\bar{A}\\ \bar{D}\end{pmatrix}\right|\right\},$$

where the max is taken over all $n \times n$ submatrices $\binom{\bar{A}}{\bar{D}}$ of $\binom{A|\boldsymbol{b}}{D|\boldsymbol{d}}$ such that $\binom{\bar{A}}{\bar{D}}$ has rank n. Note that $\boldsymbol{c}^\intercal(\bar{\boldsymbol{x}} - \boldsymbol{x}) \leq ||\boldsymbol{c}||_\infty ||\bar{\boldsymbol{x}} - \boldsymbol{x}||_1$. The result then follows by observing that $\log(||\boldsymbol{c}||_\infty) = \mathcal{O}(\alpha + \log m)$, $\log(||\bar{\boldsymbol{x}} - \boldsymbol{x}||_1) \leq \log(2n\bar{\delta})$ (using Cramer's rule), and $\log(\delta) \leq \log(\bar{\delta}) = \mathcal{O}(n(\alpha + \log n))$. □

3.4.1 Lattice Polytopes

We show that for any k-lattice polytope P, the circuit diameter is bounded by a function which is polynomial in n and logarithmic in k. This is in contrast with the combinatorial diameters of such polytopes which is $\Theta(kn)$.

Theorem 8. *The circuit diameter of a k-lattice polytope is at most $\mathcal{O}(\text{poly}(n)\log(kn))$.*

Proof. Let $\boldsymbol{x}_1$ and $\boldsymbol{x}_2$ be any two vertices of a k-lattice polytope P, and let $\{\boldsymbol{x} \in \mathbb{R}^n : A\boldsymbol{x} = \boldsymbol{b},\ D\boldsymbol{x} \leq \boldsymbol{d}\}$ be any minimal description of P with integer data. Let $\boldsymbol{c}$ be any integer objective function such that $\boldsymbol{x}_2$ is the unique maximizer of $\boldsymbol{c}^\intercal\boldsymbol{x}$ over all $\boldsymbol{x} \in P$. Ultimately, we will use Lemma 1 to bound the circuit distance from $\boldsymbol{x}_1$ to $\boldsymbol{x}_2$, but to achieve the desired result, it will be necessary to bound the term $\log(\delta(\boldsymbol{c}^\intercal(\boldsymbol{x}_1 - \boldsymbol{x}_2))$. To do this, we will use Lemma 2 to assume without loss of generality that the entries of $\boldsymbol{c}$ are not too large. This requires first showing that there exists a minimal description of P in which the encoding length of any entry of $\binom{A}{D}$ is not too large, which will conveniently bound δ as well. By Lemma 3, using a different minimal representation of P does not change the set of circuits, and therefore does not change the circuit diameter. We rely on the technique used in the proof of Corollary 26 of [98]. The technique is used to bound the entries of matrices defining full-dimensional 0/1-polytopes, but is readily generalized to k-lattice polytopes which may or may not be full-dimensional.

In the case where P is not full-dimensional, we first extend P to a full-dimensional polytope P': Since P is not full-dimensional, there exists some index $i \in [n]$ such that the elementary basis vector $\boldsymbol{e}_i$ is not in $\ker(A)$. That is, for all $\boldsymbol{x} \in P$, we have that

$\boldsymbol{x} + \alpha \boldsymbol{e}_i \notin P$ for all $\alpha \neq 0$. Let P' be the polytope obtained by taking the convex hull of P and $P + \boldsymbol{e}_i$. We can repeat this until we have a polytope P' which is full-dimensional. Note that by construction, the same elementary basis vector $\boldsymbol{e}_i$ will not be chosen twice, so the coordinates of the vertices of P' are integers between 0 and $k+1$. Note also that P appears as a face of P'. Since it is full-dimensional, the polytope P' has some minimal description $P' = \{ \boldsymbol{x} \in \mathbb{R}^n : D'\boldsymbol{x} \leq d' \}$ with integer data which is unique up to rescaling. We will now show that there exists such a description in which the entries of D' are bounded.

Let D'_j be any row of D'. Since the description of P' is minimal, each inequality of the system $D'\boldsymbol{x} \leq \boldsymbol{d}'$ defines a facet of P'. Let $\boldsymbol{v}_0, \ldots, \boldsymbol{v}_{n-1}$ be vertices of P' whose affine span is a hyperplane H containing a facet F of P' which is defined by the equality $D_j'^{\intercal}\boldsymbol{x} = \boldsymbol{d}(j)$. By translating all of the the points $\boldsymbol{v}_0, \ldots, \boldsymbol{v}_{n-1}$ by $-\boldsymbol{v}_0$ and letting $\boldsymbol{w}_i = \boldsymbol{v}_i - \boldsymbol{v}_0$ for $1 \leq i \leq n-1$, we get that the points $\boldsymbol{0}, \boldsymbol{w}_1, \ldots, \boldsymbol{w}_{n-1}$ affinely span the hyperplane H' defined by the equation $D_j'^{\intercal}\boldsymbol{x} = 0$. Let V be the matrix in $\{ -(k+1), \ldots, -1, 0, 1, \ldots, k+1 \}^{(n-1)\times n}$ whose i-th row is $\boldsymbol{w}_i$. Then we have that H' is equivalently given by an equality $\boldsymbol{h}^{\intercal}\boldsymbol{x} = 0$ with $\boldsymbol{h}(i) \in \{ \pm \det(V^i) \}$ where $V^i \in \{ -(k+1), \ldots, -1, 0, 1, \ldots, k+1 \}^{(n-1)\times(n-1)}$ is obtained from V by deleting the i-th column of V. Thus, the hyperplane H containing the facet F of P' can be equivalently given by the equality $\boldsymbol{h}^{\intercal}\boldsymbol{x} = \boldsymbol{h}^{\intercal}\boldsymbol{v}_0$, and so the inequality $D_j'^{\intercal}\boldsymbol{x} \leq \boldsymbol{d}'(j)$ in the system defining P' can be replaced by the inequality $\boldsymbol{h}^{\intercal}\boldsymbol{x} \leq \boldsymbol{h}^{\intercal}\boldsymbol{v}_0$. Note that this preserves the fact that our description has integer data.

We now bound the absolute value of the entries in $\boldsymbol{h}$ by bounding determinant of V^i. Since the entries of V^i are bounded in absolute value by $k+1$, we have by Hadamard's inequality that $|\det(V^i)| \leq (k+1)^n n^{n/2}$. Thus, we may assume without loss of generality that the system $D'\boldsymbol{x} \leq \boldsymbol{d}'$ defining P' is such that the entries of D' are bounded (in absolute value) by $(k+1)^n n^{n/2}$. Since P appears as a face of P', we can obtain a minimal description of P by setting some inequalities of $D'\boldsymbol{x} \leq \boldsymbol{d}'$ to equations and by removing some redundant inequalities. Therefore, we may assume without loss of generality that in the system $A\boldsymbol{x} = \boldsymbol{b}$, $D\boldsymbol{x} \leq \boldsymbol{d}$ defining P, the entries of $\binom{A}{D}$ are also bounded (in absolute value) by $(k+1)^n n^{n/2}$.

This implies that the encoding length of $\binom{A}{D}$ is at most $\log((k+1)^n n^{n/2})$. Then by Lemma 2, we may assume that

$$||\boldsymbol{c}||_\infty \leq 2^4 n^3 \Big((n+1)!((k+1)^n n^{n/2})^n + 1 \Big)^{n(n+2)}.$$

Note therefore that $\log(||\boldsymbol{c}||_\infty)$ is at most on the order of $\text{poly}(n)\log(kn)$. Now, by Lemma 1, we know that there exists a circuit-path (given by the Greatest Improvement circuit-pivot rule) from $\boldsymbol{x}_1$ to $\boldsymbol{x}_2$ whose length is $\mathcal{O}\big(n \log(\delta \boldsymbol{c}^{\intercal}(\boldsymbol{x}_1 - \boldsymbol{x}_2))\big)$, where again δ is the maximum absolute value of any subdeterminant of $\binom{A}{D}$. We have that $\boldsymbol{c}^{\intercal}(\boldsymbol{x}_1 - \boldsymbol{x}_2) \leq$

$n \cdot ||\boldsymbol{x}_1 - \boldsymbol{x}_2||_\infty \cdot ||\boldsymbol{c}||_\infty \leq nk \cdot ||\boldsymbol{c}||_\infty$. Furthermore, by another application of Hadamard's inequality, we get that

$$\delta \leq \left((k+1)^n n^{n/2}\right)^n n^{n/2}.$$

Thus, $\log(\delta \boldsymbol{c}^\intercal(\boldsymbol{x}_1 - \boldsymbol{x}_2))$ is at most $\mathcal{O}(\text{poly}(n)\log(kn))$, and therefore so is the length of the circuit-path from $\boldsymbol{x}_1$ to $\boldsymbol{x}_2$, as desired. □

Chapter 4

Edges Rules in 0/1-LPs

4.1 Steepest Edge Rule

In this section, we explore the implications that Theorem 6 has in the case of 0/1-LPs, eventually proving Theorem 13 and Theorem 15. We start with the following lemma, which shows that at a vertex of a 0/1 polytope a Steepest Descent circuit is the same as a steepest edge:

Lemma 9. *Consider a problem in the general form (1.1) whose feasible region P is a 0/1 polytope, and let $\boldsymbol{x}$ be a non-optimal vertex of P. Then the optimal solution to* $\mathrm{Steep}(P, \boldsymbol{x}, \boldsymbol{c})$ *corresponds to an edge-direction at $\boldsymbol{x}$, and can be computed in polynomial time.*

Proof. Consider the optimal objective function value of $\mathrm{Steep}(P, \boldsymbol{x}, \boldsymbol{c})$. It is not difficult to see that this value is bounded above by the optimal objective function value of the following optimization problem $\mathcal{Q}$:

$$
\begin{aligned}
\max \quad & \boldsymbol{c}^{\mathsf{T}} \boldsymbol{z} \\
\text{s.t.} \quad & \|\boldsymbol{z}\|_1 \leq 1 && (4.1) \\
& \boldsymbol{z} \in \mathcal{C}(\boldsymbol{x}) && (4.2)
\end{aligned}
$$

over all $\boldsymbol{z} \in \mathbb{R}^n$. This is true since if $\boldsymbol{g} \in \mathcal{C}(A, D)$ is a feasible solution to $\mathrm{Steep}(P, \boldsymbol{x}, \boldsymbol{c})$, then $\frac{\boldsymbol{g}}{\|\boldsymbol{g}\|_1}$ is a feasible solution of $\mathcal{Q}$ with the same objective function value.

Let $P_{\mathcal{Q}}$ denote the feasible region of $\mathcal{Q}$. Note that $P_{\mathcal{Q}}$ is the feasible cone at $\boldsymbol{x}$ in P – given by the constraint (4.2) – intersected with an n-dimensional cross-polytope – given by the constraint (4.1). The constraint (4.1) can be modeled using the linear constraints

$$\boldsymbol{v}^\intercal \boldsymbol{z} \leq 1 \quad \text{for all } \boldsymbol{v} \in \{\, 1, -1 \,\}^n . \tag{4.3}$$

It follows that $P_{\mathcal{Q}}$ is a polytope, and therefore $\mathcal{Q}$ is a feasible bounded LP. There exists an optimal vertex $\boldsymbol{y}$ of $P_{\mathcal{Q}}$ which is determined by n linearly independent constraints of $P_{\mathcal{Q}}$.

Since $\boldsymbol{x} \in \{\, 0, 1 \,\}^n$ and P is a 0/1 polytope, each entry of $\boldsymbol{x}$ is either equal to its upper bound or its lower bound. Thus, the feasible cone at $\boldsymbol{x}$ lies within a single orthant of $\mathbb{R}^n$. This implies that among all the linear constraints that model $\|\boldsymbol{z}\|_1 \leq 1$, only one is facet defining. In particular, it is the constraint $\boldsymbol{v}^\intercal \boldsymbol{x} \leq 1$ where $\boldsymbol{v} = \mathbf{1} - 2\boldsymbol{x}$. That is, $\boldsymbol{v}$ is defined by

$$\boldsymbol{v}(i) := \begin{cases} 1 & \text{if } \boldsymbol{x}(i) = 0, \\ -1 & \text{if } \boldsymbol{x}(i) = 1. \end{cases} \tag{4.4}$$

Therefore, $\boldsymbol{y}$ is contained in at least $n-1$ facets corresponding to inequalities that describe the feasible cone at $\boldsymbol{x}$. Since $\boldsymbol{x}$ is not optimal, $\boldsymbol{y} \neq \mathbf{0}$.

As a consequence of this, we have that the optimal solution of $\mathcal{Q}$ corresponds to an edge-direction of P incident with $\boldsymbol{x}$. It follows that the optimal solution of Steep$(P, \boldsymbol{x}, \boldsymbol{c})$ is an edge-direction of P incident with $\boldsymbol{x}$, and since it is the optimal solution of an LP of polynomial size, it can be computed in polynomial time. □

We recall that in [25], it was shown that for their different definition of the Steepest Descent circuit pivot rule, a Steepest Descent circuit augmentation can be computed in polynomial time for *all* LPs. Despite similarities in the two versions of the Steepest Descent circuit-pivot rule, the technique they employ cannot be straightforwardly applied to work for the definition used here, and our computability result in Lemma 9 is not implied by theirs.

Since the set of edges is contained in the set of circuits, Lemma 9 implies that at an extreme point solution of an LP defined over a 0/1 polytope, the Steepest Descent circuit and the steepest edge are the same. This motivates the following definition:

Definition 8. *Given a 0/1 LP (P) of the form (1.1) with feasible region P, let $\boldsymbol{x}$ be a vertex of P, and let $\boldsymbol{v} = \mathbf{1} - 2\boldsymbol{x}$. The Steepest Edge rule selects an edge direction that maximizes $\frac{\boldsymbol{c}^\intercal \boldsymbol{z}}{\boldsymbol{v}^\intercal \boldsymbol{z}}$ over all edge directions at $\boldsymbol{x}$.*

That is, the Steepest Edge rule chooses to move along the steepest edge incident to $\boldsymbol{x}$, as the name suggests. Although we could clearly define the Steepest Edge rule for arbitrary

LPs (and instead define it to choose the edge that maximizes $\frac{c^\intercal z}{\|z\|_1}$), we define it this way because this definition will be more compatible with our later aim to extend this edge rule to a Simplex pivot rule which follows the same path. Since we do not analyze the Steepest Edge rule outside the context of 0/1-LPs, the loss of generality is not impactful for this thesis.

The fact that the Steepest Edge rule and the Steepest Descent circuit-pivot rule follow the same path in the setting of 0/1-LPs means that, in this setting, results about the Steepest Descent circuit-pivot rule apply to the Steepest Edge rule as well. We now combine Corollary 2 and Lemma 9 to show the following:

Theorem 13. *Let $\boldsymbol{x}$ be a vertex solution of a 0/1-LP with n variables. An augmentation along a steepest edge is an n-approximate Greatest Improvement circuit-augmentation.*

Proof. At a vertex, a Steepest Descent circuit is also a steepest edge. Therefore, Corollary 2 also applies to a steepest edge. Since P is a 0/1 polytope, we immediately have that $M_1 \leq n$. We now show that $\omega_1 \geq 1$. Let $\boldsymbol{v}$ be any vertex of P and let F be any facet of P which does not contain $\boldsymbol{v}$. By reflecting and translating P, we may assume without loss of generality that $\boldsymbol{v} = \boldsymbol{0}$ (Note that these operations do not change the 1-norm distance between any pair of points in P). It therefore suffices to show that for any facet F not containing $\boldsymbol{0}$, $\|\boldsymbol{y}\|_1 \geq 1$ for all $\boldsymbol{y} \in F$. Since F is a 0/1 polytope not containing $\boldsymbol{0}$, this clearly holds for all vertices of F. Since all other points in F are convex combinations of the vertices of F, it also holds for all other points in F. Therefore, $\frac{\omega_1}{M_1} \geq \frac{1}{n}$. □

We are now ready to prove the following theorem:

Theorem 14. *Given a problem in the general form (1.1) whose feasible region P is a 0/1 polytope, a circuit-augmentation algorithm with a Steepest Descent circuit-pivot rule reaches an optimal solution by performing a strongly-polynomial number of augmentations. Furthermore, if the initial solution is a vertex, the algorithm follows the path on the 1-skeleton of P generated by the Steepest Edge rule.*

Proof. Let us call $(LP1)$ the given LP problem of the form (1.1) whose feasible region is P. Since P is a 0/1 polytope, for the sake of the analysis we can assume that the maximum absolute value of any element in A and B is $\leq \frac{n^{n/2}}{2^n}$ [98]. Apply Lemma 2 to $(LP1)$ to get an equivalent objective function $\boldsymbol{c}'$. Finally, let $(LP2) := \max \{ \boldsymbol{c}'^\intercal \boldsymbol{x} \ : \ A\boldsymbol{x} = \boldsymbol{b},\ B\boldsymbol{x} \leq \boldsymbol{d},\ \boldsymbol{x} \in \mathbb{R}^n \}$.

Let $\boldsymbol{x}_0$ and $\boldsymbol{x}^*$ be the initial solution and the optimal solution, respectively. Note that by the equivalence of $\boldsymbol{c}$ and $\boldsymbol{c}'$, $\boldsymbol{x}^*$ is optimal for both $(LP1)$ and $(LP2)$. By performing at most n augmentations, we can assume $\boldsymbol{x}_0$ is a vertex. First, we will show that Theorem 14 holds for $(LP2)$. Then, we will show that a circuit-augmentation algorithm traverses the same edge-walk when solving $(LP2)$ and $(LP1)$ when using the Steepest Descent circuit-pivot rule. This will prove the statement.

Recall that the Steepest Descent circuit-pivot rule selects at each step an improving circuit $\boldsymbol{g}$ that maximizes $\frac{\boldsymbol{c}^\intercal \boldsymbol{g}}{\|\boldsymbol{g}\|_1}$. Since the feasible region of $(LP2)$ is a 0/1 polytope, we can apply Lemma 9. Therefore, each augmentation corresponds to moving from a vertex to an adjacent vertex. Furthermore, the total number of augmentations can be bounded via Theorem 13 and Lemma 1 by

$$\mathcal{O}\left(n^2 \log\left(\boldsymbol{c}'^\intercal(\boldsymbol{x}^* - \boldsymbol{x}_0)\right)\right).$$

We now address the term $\log\left(\boldsymbol{c}'^\intercal(\boldsymbol{x}^* - \boldsymbol{x}_0)\right)$. Since $\boldsymbol{x}_0$ and $\boldsymbol{x}^*$ are both in $\{0,1\}^n$, we have that $\log\left(\boldsymbol{c}'^\intercal(\boldsymbol{x}^* - \boldsymbol{x}_0)\right) \leq \log(\|\boldsymbol{c}'\|_1) \leq \log(n\|\boldsymbol{c}'\|_\infty)$, which is polynomial in n due to Lemma 2(a). Therefore, the number of augmentations required to solve $(LP2)$ is strongly-polynomial in the input size.

To finish our proof, it remains to show that when the circuit-augmentation algorithm is applied to $(LP1)$, it performs the same edge-walk as it does when it is applied to $(LP2)$. To see this, we will rely on the polyhedral characterization of the problem $\mathrm{Steep}(P, \boldsymbol{x}, \boldsymbol{c})$, used in the proof of Lemma 9. As explained there, the edge-direction $\boldsymbol{g}$ selected by our algorithm applied to $(LP2)$ is an optimal solution to the LP $\max\{\boldsymbol{c}'^\intercal \boldsymbol{x} : \boldsymbol{x} \in P_{\mathcal{Q}}\}$, which describes $\mathrm{Steep}(P, \boldsymbol{x}, \boldsymbol{c}')$. Note that the maximum absolute value of a matrix-coefficient of this LP is also at most $\frac{n^{n/2}}{2^n}$. Therefore, due to Lemma 2(b), $\boldsymbol{g}$ is an optimal solution to $\max\{\boldsymbol{c}'^\intercal \boldsymbol{x} : \boldsymbol{x} \in P_{\mathcal{Q}}\}$ (i.e., $\mathrm{Steep}(P, \boldsymbol{x}, \boldsymbol{c}')$) if and only if it is an optimal solution to $\max\{\boldsymbol{c}^\intercal \boldsymbol{x} : \boldsymbol{x} \in P_{\mathcal{Q}}\}$ (i.e., $\mathrm{Steep}(P, \boldsymbol{x}, \boldsymbol{c})$). Therefore, the circuit-augmentation algorithm implemented according to the steepest-descent circuit-pivot rule, performs the exact same pivots for the objective functions $\boldsymbol{c}'$ and $\boldsymbol{c}$. □

This implies the following theorem:

Theorem 15. *Given an LP in the general form (1.1) whose feasible region P is a 0/1 polytope, then*

(i) The Steepest Edge rule can be computed in polynomial, and it generates a monotone path between any vertex and the optimum whose length is strongly-polynomial in the input size of the LP.

(ii) When the feasible region P is a non-degenerate 0/1 polytope, the Simplex method with a Steepest Edge pivot rule reaches an optimal solution in strongly-polynomial time.

Note that while the number of augmentations performed is polynomial in n and is independent from the number of rows $m_A + m_B$, actually computing the movement along a steepest edge to the next vertex solution requires a number of operations which also depends on m_B. This fact and Theorem 14 imply Part (i) of Theorem 15. In the context of the Simplex method, moving to a neighboring vertex along a steepest edge might require several *degenerate* basis exchanges. Having a pivot rule that implies a strongly-polynomial bound on the total number of basis exchanges remains an open question, but if the polytope is non-degenerate, then the two concepts coincide, hence we obtain Part (ii) of Theorem 15.

We conclude this section with a few remarks. First, we mention that while the Steepest Edge rule can be computed in polynomial time, this of course relies on the use of an existing efficient LP solver. As such, this makes Theorem 15 a result which is not of *practical* use for the direct purpose of solving LPs. However, the fact that it is computable in polynomial time is still of significant importance. As discussed in Chapter 1, in order for there to exist a pivot rule for the Simplex method which solves an LP in polynomial time, we require that it is not NP-Hard to generate monotone paths of polynomial length. We show in Theorem 15 that this holds in the setting of 0/1 LPs.

Second, we want to mention that we were interested in proving a strongly-polynomial bound on the number of augmentations, without trying to obtain the best possible such bound. For example, one could possibly get a better bound on the number of augmentations using a recent result of Eisenbrand et al. [46] instead of the result of Frank and Tardos [51].

Third, we remark that our main focus was bounding the number of augmentations performed by the Steepest Edge rule *in particular*. This is because Steepest Edge is a classical decision rule and one of our objectives here is to understand its performance. Regarding our use of Lemma 2, the authors of [51] observe that when applying Lemma 2 to LPs defined over 0/1 polytopes, it suffices to choose $N = n + 1$ rather than $N = (n + 1)!2^{n\alpha} + 1$, which is required in general. We remark that we cannot take $N = n + 1$ in this context because, while our original LP is defined over a 0/1 polytope, the *auxiliary* LP $\mathcal{Q}$ used in Lemma 9 is not. In order to argue that the algorithm performs the same edge-walk in $(LP1)$ as it does in $(LP2)$, we require that our new objective function $\boldsymbol{c}'$ is equivalent to our original objective function $\boldsymbol{c}$ not just for those two LPs, but also for $\mathcal{Q}$. By taking $N = n+1$ in Lemma 2 and by instead following the path of steepest edges with respect to $\boldsymbol{c}'$, we could get a shorter path which is monotone with respect to $\boldsymbol{c}$. However, this path would *not* necessarily be the path of steepest edges with respect to $\boldsymbol{c}$, and this is the path whose length we aim to analyze. Hence, we rely on the fully general version of Lemma 2.

Finally, as already mentioned, in the context of pivot rules for the Simplex method, the name "Steepest Descent" often refers to a normalization according to the 2-norm of a vector, rather than the 1-norm. Since for any vector $\boldsymbol{g} \in \mathbb{R}^n$, we have $\sqrt{n}||\boldsymbol{g}||_2 \geq ||\boldsymbol{g}||_1 \geq ||\boldsymbol{g}||_2$, it is not difficult to note that the bound on the length of the circuit-path obtained in Theorem 14 still holds if we normalize according to the 2-norm.

4.2 ASD Edge Rule

Mirroring Section 4.1, in this section we explore the implications that Theorem 12 has in the case of 0/1-LPs in the standard equality form (2.2). We start with the following lemma:

Lemma 10. *Consider an LP problem in equality form (2.2) whose feasible region P is a 0/1 polytope, and let $\boldsymbol{x}$ be a non-optimal vertex of P. Then the optimal solution to* $\mathrm{ASD}(P, \boldsymbol{x}, \boldsymbol{c})$ *corresponds to an edge-direction at $\boldsymbol{x}$.*

Proof. We first remark that since P is a 0/1 polytope, the LP is not unbounded. Suppose that $\boldsymbol{z}$ is any feasible direction at $\boldsymbol{x}$. In the case of an equality form LP, the inequality matrix B is the identity matrix $-I$, so we have that

$$N(\boldsymbol{z}; -I, \boldsymbol{x}) = \sum_{\substack{i \in [n]: \\ \boldsymbol{x}(i) > 0 \\ \boldsymbol{z}(i) \leq 0}} \left(\frac{1}{\boldsymbol{x}(i)} \right) - \boldsymbol{z}(i).$$

Since the feasible region P is a 0/1 polytope and since $\boldsymbol{x}$ is a vertex, if $\boldsymbol{x}(i) > 0$, then $\boldsymbol{x}(i) = 1$. Then the above sum is equal to

$$\sum_{\substack{i \in [n]: \\ \boldsymbol{x}(i) = 1 \\ \boldsymbol{z}(i) \leq 0}} -\boldsymbol{z}(i) = \sum_{\substack{i \in [n]: \\ \boldsymbol{x}(i) = 1 \\ \boldsymbol{z}(i) < 0}} -\boldsymbol{z}(i) = \sum_{\substack{i \in [n]: \\ \boldsymbol{x}(i) = 1}} -\boldsymbol{z}(i),$$

where the last equality follows from the fact that $\boldsymbol{z}$ is a feasible direction at $\boldsymbol{x}$, so $\boldsymbol{z}(i) < 0$ if and only if $\boldsymbol{x}(i) = 1$. Then if we let $\boldsymbol{v}$ be defined by

$$\boldsymbol{v}(i) := \begin{cases} -1 & \text{if } \boldsymbol{x}(i) = 1 \\ 0 & \text{if } \boldsymbol{x}(i) = 0 \end{cases}$$

(i.e., $\boldsymbol{v} = -\boldsymbol{x}$), we have that

$$N(\boldsymbol{z}; -I, \boldsymbol{x}) = \boldsymbol{v}^\intercal \boldsymbol{z}.$$

The optimal objective function value of $\mathrm{ASD}(P, \boldsymbol{x}, \boldsymbol{c})$ is bounded above by the optimal objective function value of the following optimization problem $\mathcal{Q}$:

$$\begin{aligned} \max \quad & \boldsymbol{c}^\intercal \boldsymbol{z} \\ \text{s.t.} \quad & \boldsymbol{v}^\intercal \boldsymbol{z} \leq 1 \\ & \boldsymbol{z} \in \mathcal{C}(\boldsymbol{x}) \end{aligned}$$

over all $\boldsymbol{z} \in \mathbb{R}^n$. We first observe that the feasible region $P_{\mathcal{Q}}$ of $\mathcal{Q}$ is not unbounded. Otherwise, there exists a feasible direction $\boldsymbol{z}' \neq \boldsymbol{0}$ at $\boldsymbol{x}$ such that $\boldsymbol{x} + \alpha \boldsymbol{z}'$ is feasible for all $\alpha > 0$, contradicting that P is bounded. Then there exists an optimal vertex $\boldsymbol{y}$ of $P_{\mathcal{Q}}$ which is determined by n linearly independent constraints of $\mathcal{Q}$. As in the proof of Lemma 9, $\boldsymbol{y}$ is contained in at least $n-1$ facets corresponding to inequalities that describe the feasible cone at $\boldsymbol{x}$. Since $\boldsymbol{x}$ is not optimal, $\boldsymbol{y} \neq \boldsymbol{0}$.

Then as before, we have that the optimal solution of $\mathcal{Q}$ corresponds to an edge-direction of P incident with $\boldsymbol{x}$. It follows that the optimal solution of $\mathrm{ASD}(P, \boldsymbol{x}, \boldsymbol{c})$ is an optimal solution to $\mathcal{Q}$, and is therefore an edge-direction of P incident with $\boldsymbol{x}$. □

This implies that at an extreme point solution of an SEF LP defined over a 0/1 polytope, an ASD circuit and an ASD edge are the same.

Definition 9. *Given a 0/1 LP (P) in the standard equality form (2.2) with feasible region P, let $\boldsymbol{x}$ be a vertex of P, and let $\boldsymbol{v} = -\boldsymbol{x}$. The ASD edge rule selects an edge direction that maximizes $\frac{\boldsymbol{c}^\intercal \boldsymbol{z}}{\boldsymbol{v}^\intercal \boldsymbol{z}}$ over all edge directions at $\boldsymbol{x}$.*

The proof of the following theorem closely follows that of Theorem 14

Theorem 16. *Given a problem in the standard equality form (2.2) whose feasible region P is a 0/1 polytope, a circuit-augmentation algorithm with n ASD circuit-pivot rule reaches an optimal solution by performing a strongly-polynomial number of augmentations. Furthermore, if the initial solution is a vertex, the algorithm follows the path on the 1-skeleton of P generated by the ASD rule.*

Proof. Let us call $(LP1)$ the given LP problem of the form (2.2) whose feasible region is P. Since P is a 0/1 polytope, for the sake of the analysis we can assume that the maximum absolute value of any element in A is $\leq \frac{n^{n/2}}{2^n}$ [98]. Apply Lemma 2 to $(LP1)$ to get an equivalent objective function $\boldsymbol{c}'$. Finally, let $(LP2) := \max \{ \boldsymbol{c}'^\intercal \boldsymbol{x} \ : \ A\boldsymbol{x} = \boldsymbol{b}, \ \boldsymbol{x} \geq \boldsymbol{0}, \ \boldsymbol{x} \in \mathbb{R}^n \}$.

Let $\boldsymbol{x}_0$ and $\boldsymbol{x}^*$ be the initial solution and the optimal solution, respectively . Note that by the equivalence of $\boldsymbol{c}$ and $\boldsymbol{c}'$, $\boldsymbol{x}^*$ is optimal for both $(LP1)$ and $(LP2)$. By performing at most n augmentations, we can assume $\boldsymbol{x}_0$ is a vertex. First, we will show that Theorem 16 holds for $(LP2)$. Then, we will show that a circuit-augmentation algorithm traverses the same edge-walk when solving $(LP2)$ and $(LP1)$ when using the ASD circuit-pivot rule. This will prove the statement.

Since the feasible region of $(LP2)$ is a 0/1 polytope, we can apply Lemma 10. Therefore, each augmentation corresponds to moving from a vertex to an adjacent vertex. Furthermore, by Theorem 12 the total number of augmentations is

$$\mathcal{O}\left(n^2 \log\left(\boldsymbol{c}'^{\intercal}(\boldsymbol{x}^* - \boldsymbol{x}_0)\right)\right).$$

We now address the term $\log\left(\boldsymbol{c}'^{\intercal}(\boldsymbol{x}^* - \boldsymbol{x}_0)\right)$. Since $\boldsymbol{x}_0$ and $\boldsymbol{x}^*$ are both in $\{0,1\}^n$, we have that $\log\left(\boldsymbol{c}'^{\intercal}(\boldsymbol{x}^* - \boldsymbol{x}_0)\right) \leq \log(\|\boldsymbol{c}'\|_1) \leq \log(n\|\boldsymbol{c}'\|_\infty)$, which is polynomial in n due to Lemma 2(a). Therefore, the number of augmentations required to solve $(LP2)$ is strongly-polynomial in the input size.

To finish our proof, it remains to show that when the circuit-augmentation algorithm is applied to $(LP1)$, it performs the same edge-walk as it does when it is applied to $(LP2)$. To see this, we will rely on the polyhedral characterization of the problem ASD$(P, \boldsymbol{x}, \boldsymbol{c})$, used in the proof of Lemma 10. As explained there, the edge-direction $\boldsymbol{g}$ selected by our algorithm applied to $(LP2)$ is an optimal solution to the LP $\max\{\boldsymbol{c}'^{\intercal}\boldsymbol{x} : \boldsymbol{x} \in P_{\mathcal{Q}}\}$, which describes ASD$(P, \boldsymbol{x}, \boldsymbol{c}')$. Note that the maximum absolute value of a matrix-coefficient of this LP is also at most $\frac{n^{n/2}}{2^n}$. Therefore, due to Lemma 2(b), $\boldsymbol{g}$ is an optimal solution to $\max\{\boldsymbol{c}'^{\intercal}\boldsymbol{x} : \boldsymbol{x} \in P_{\mathcal{Q}}\}$ (i.e., ASD$(P, \boldsymbol{x}, \boldsymbol{c}')$) if and only if it is an optimal solution to $\max\{\boldsymbol{c}^{\intercal}\boldsymbol{x} : \boldsymbol{x} \in P_{\mathcal{Q}}\}$ (i.e., ASD$(P, \boldsymbol{x}, \boldsymbol{c})$). Therefore, the circuit-augmentation algorithm implemented according to the ASD circuit-pivot rule, performs the exact same pivots for the objective functions $\boldsymbol{c}'$ and $\boldsymbol{c}$. □

Together, Theorem 16 and Lemma 10 imply the following theorem:

Theorem 17. *Given an LP in the standard equality form (2.2) whose feasible region P is a 0/1 polytope, the ASD rule can be computed in polynomial time, and it generates a monotone path between any vertex and the optimum whose length is strongly-polynomial in the input size of the LP.*

4.3 Shadow Vertex Edge Rules

The Shadow pivot rule is a fundamental tool in the study of the average case run-time of the Simplex method initiated by Borgwardt in [16]. This pivot rule gave rise to several algorithmic developments, including the arrival of *smoothed complexity* [34, 87, 94]. Intuitively, given a LP $\max\{\,\boldsymbol{c}^\intercal\boldsymbol{x} : \boldsymbol{x} \in P\,\}$ starting at a vertex $\boldsymbol{x}_0 \in P$, they find an auxiliary vector $\boldsymbol{v}$ such that $\boldsymbol{v}^\intercal\boldsymbol{x}$ is minimized at $\boldsymbol{x}_0$ which allows them to project P to a polygon. The two paths of this polygon guide the possible improving paths for P.

We present a novel modification of the Shadow pivot rule that instead comes from the theory of *monotone path polytopes* first introduced by Billera and Sturmfels in [10] (see Chapter 9 of [97] for an introduction and [9] for more details on their structure). The vertices of the monotone path polytope are in natural correspondence with paths that the modified Shadow pivot rule may choose. Billera and Sturmfels named these paths *coherent monotone paths*. We will continue to use that name throughout because we will allow for situations that have not been considered in earlier treatments of Shadow pivot rule. Namely, in contrast to the original setup of [16, 34, 87, 94] we do not require that non-degeneracy conditions hold on the LPs we consider.

4.3.1 The Shadow Rule for General Polyhedra

For the original version of the Shadow pivot rule, refer to Chapter 1 of the book of Borgwardt [16]. In this section, we initially consider a general LP $\max\{\,\boldsymbol{c}^\intercal\boldsymbol{x} : \boldsymbol{x} \in Q\,\}$ where Q is any polyhedron in the general form. For now, Q is not necessarily 0/1.

We will prove results for general LPs in Lemma 11 and Corollary 4. Later we will restrict those results to 0/1-LPs to prove the bounds on the lengths of monotone paths generated by two new edge rules for 0/1-LPs: the *Slim Shadow rule* and the *Ordered Shadow rule*. Like in the original Shadow pivot rule, we follow the edges guided by a shadow. Later, in Section 5.3, we transform these two rules into actual pivot rules.

In the context of Shadow rules, we have a special type of monotone paths called *coherent monotone paths*, which are constructed with both $\boldsymbol{c}$ and an additional vector $\boldsymbol{v}$. Taking $\boldsymbol{c}$ and $\boldsymbol{v}$ together, we obtain a projection $\pi : \mathbb{R}^n \to \mathbb{R}^2$ given by $\pi(\boldsymbol{x}) = (\boldsymbol{v}^\intercal\boldsymbol{x}, \boldsymbol{c}^\intercal\boldsymbol{x})$. Applying this projection to Q yields a polygon $\pi(Q)$. This polygon is often called a *shadow* of Q.

To define coherence, we require the notion of an *upper path* of $\pi(Q)$, depicted in Figure 4.1. Let F_0 and F_1 denote the $\boldsymbol{e}_1$-minimal and $\boldsymbol{e}_1$-maximal faces of $\pi(Q)$ respectively. Then let $\boldsymbol{u}^0$ and $\boldsymbol{u}^1$ be the $\boldsymbol{e}_2$-maximal vertices of F_0 and F_1, respectively. By convexity, the line segment between $\boldsymbol{u}^0$ and $\boldsymbol{u}^1$ is contained in $\pi(Q)$. Every point on the polygon lies either above or below this line segment, since the segment travels from an $\boldsymbol{e}_1$-minimum to an $\boldsymbol{e}_1$-maximum. The upper vertices are precisely the set of vertices that lie above or on this segment. Formally, let $L : \mathbb{R} \to \mathbb{R}$ be the equation of the line affinely spanned by $\boldsymbol{u}^0$

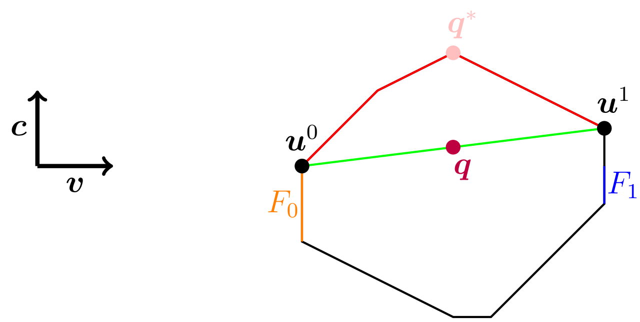

Figure 4.1: The red path is the upper path of the polygon. The edges F_0 and F_1, are the $\boldsymbol{e}_1$-minimal and $\boldsymbol{e}_1$-maximal faces of the polygon respectively. The green line segment is exactly the line segment from $\boldsymbol{u}^0$, the $\boldsymbol{e}_2$-maximum of F_0, to $\boldsymbol{u}^1$, the $\boldsymbol{e}_2$-maximum of F_1. The choice of $\boldsymbol{q}^*$ and $\boldsymbol{q}$ from the proof of Lemma 11 are shown as well.

and $\boldsymbol{u}^1$. Let $\boldsymbol{u}$ be a vertex of $\pi(Q)$. Then $\boldsymbol{u}$ is a vertex of the upper path precisely when $L(\boldsymbol{e}_1^\intercal \boldsymbol{u}) \leq \boldsymbol{e}_2^\intercal \boldsymbol{u}$. The upper vertices form a path from $\boldsymbol{u}^0$ to $\boldsymbol{u}^1$ called the upper path.

A $\boldsymbol{v}$-monotone path Γ in Q is called $\boldsymbol{c}$-*coherent* if π maps Γ to the upper path of $\pi(Q)$. Namely, all vertices of Γ must be sent to either vertices or interior points of edges in the upper path, and every vertex of the upper path must have a vertex sent to it from Γ. Note that this latter condition implies that the first vertex, $\boldsymbol{x}_0$, is the $\boldsymbol{c}$-maximum of the $\boldsymbol{v}$-minimal face of Q. A $\boldsymbol{v}$-monotone path is called *coherent* if there exists some $\boldsymbol{c} \in \mathbb{R}^n$ such that it is $\boldsymbol{c}$-coherent. In [10], they observed that not all monotone paths on a polytope are coherent, and in [11], the authors showed that even on the octahedron, some monotone paths are not coherent. We reproduce that example in Figure 4.2.

To find a $\boldsymbol{c}$-coherent $\boldsymbol{v}$-monotone path in Q, at a vertex $\boldsymbol{x}_i$ we maximize the slope in the polygon $\pi(Q)$ among all $\boldsymbol{v}$-improving edges starting at $\boldsymbol{x}_i$. Concretely, we have

$$\boldsymbol{x}_{i+1} = \operatorname*{argmax}_{\boldsymbol{u} \in N_{\boldsymbol{v}}(\boldsymbol{x}_i)} \left(\frac{\boldsymbol{c}^\intercal (\boldsymbol{u} - \boldsymbol{x}_i)}{\boldsymbol{v}^\intercal (\boldsymbol{u} - \boldsymbol{x}_i)} \right).$$

Here we make no assumption that the LP is non-degenerate. Furthermore, the choice of $\boldsymbol{x}_{i+1}$ here may not be unique when $\boldsymbol{c}$ is not generic. When it is not unique, one chooses any maximizer. From this observation, we obtain a general notion of a Shadow rule.

Definition 10. *Let* (Q) *be an LP with feasible region* Q *and objective function* $\boldsymbol{c}^\intercal \boldsymbol{x}$*. Then a Shadow rule constructs a path on the 1-skeleton of* Q *from a starting vertex* $\boldsymbol{x}_0$ *to an*

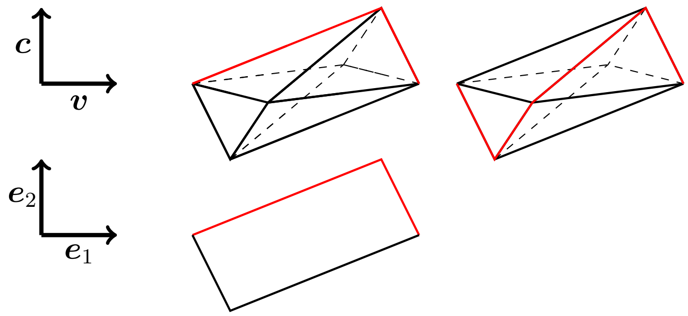

Figure 4.2: In the top of the center of the picture, a $\boldsymbol{c}$-coherent $\boldsymbol{v}$-monotone path is drawn in red on the octahedron $\diamond^3$. The corresponding shadow $\pi(\diamond^3)$ is on the bottom of the center part of the picture where the upper path corresponding to the coherent monotone path is highlighted in red. Under the projection π, $\boldsymbol{v}$ and $\boldsymbol{c}$ induce the x and y coordinates, respectively as indicated by the arrows. On the right side of the picture is an example of an incoherent monotone path on the octahedron from [11].

optimal LP solution by choosing the next vertex in the path via

$$\boldsymbol{x}_{i+1} = \operatorname*{argmax}_{\boldsymbol{u} \in N_{\boldsymbol{v}}(\boldsymbol{x}_i)} \left(\frac{\boldsymbol{c}^\intercal (\boldsymbol{u} - \boldsymbol{x}_i)}{\boldsymbol{v}^\intercal (\boldsymbol{u} - \boldsymbol{x}_i)} \right),$$

where $\boldsymbol{v}$ *is an objective function such that* $\boldsymbol{x}_0$ *is the* $\boldsymbol{c}$*-maximum of the* $\boldsymbol{v}$*-minimal face of* $\pi(Q)$*.*

Note that a Shadow rule must come with a mechanism for choosing $\boldsymbol{v}$ given an initial vertex $\boldsymbol{x}_0$. In the probabilistic analysis of the performance of the Simplex method using Shadow rules in [16, 33, 87], the choice of $\boldsymbol{v}$ is made randomly. In contrast, our Shadow Rules provide a deterministic way of making this choice.

The following lemma shows that $\boldsymbol{c}$-coherent $\boldsymbol{v}$-monotone paths not only lead to a maximum of $\boldsymbol{v}$ but also go through a maximum of $\boldsymbol{c}$. Note that this lemma holds for all $\boldsymbol{c}$-coherent $\boldsymbol{v}$-monotone paths on any polytope regardless of the choice of $\boldsymbol{v}$.

Lemma 11. *Consider the LP* $\max\{\boldsymbol{c}^\intercal \boldsymbol{x} : \boldsymbol{x} \in Q\}$ *for any polytope* Q*. Let* $\boldsymbol{v} \in \mathbb{R}^n \setminus \{\boldsymbol{0}\}$*, and let* $\Gamma = [\boldsymbol{x}_0, \boldsymbol{x}_1, \boldsymbol{x}_2, \dots, \boldsymbol{x}_k]$ *be a* $\boldsymbol{c}$*-coherent* $\boldsymbol{v}$*-monotone path in* Q*. Then there exists some* $0 \leq i \leq k$ *such that* $\boldsymbol{x}_i$ *maximizes* $\boldsymbol{c}$ *on* Q *and such that the portion of the path from* $\boldsymbol{x}_0$ *to* $\boldsymbol{x}_i$ *is both* $\boldsymbol{c}$*-monotone and* $\boldsymbol{v}$*-monotone.*

Proof. Let $\pi : Q \to \mathbb{R}^2$ be defined by $\pi(\boldsymbol{x}) = (\boldsymbol{v}^\intercal \boldsymbol{x}, \boldsymbol{c}^\intercal \boldsymbol{x})$. Equivalently, we have $\boldsymbol{e}_1^\intercal \pi(\boldsymbol{x}) = \boldsymbol{v}^\intercal \boldsymbol{x}$ and $\boldsymbol{e}_2^\intercal \pi(\boldsymbol{x}) = \boldsymbol{c}^\intercal \boldsymbol{x}$. Since Γ is a $\boldsymbol{c}$-coherent $\boldsymbol{v}$-monotone path, $\pi(\Gamma)$ follows the upper path in $\pi(Q)$. Let $\boldsymbol{u}^0$ and $\boldsymbol{u}^1$ denote $\pi(\boldsymbol{x}_0)$ and $\pi(\boldsymbol{x}_k)$, the first and final vertices of the upper path $\pi(\Gamma)$, respectively. As in the definition of an upper path, define $L : \mathbb{R} \to \mathbb{R}$ to be the equation of the line passing through $\boldsymbol{u}^0$ and $\boldsymbol{u}^1$.

Let $\boldsymbol{q}^*$ be an $\boldsymbol{e}_2$-maximal vertex of $\pi(Q)$. Note that $\boldsymbol{u}^0$ and $\boldsymbol{u}^1$ are $\boldsymbol{e}_1$-minimal and $\boldsymbol{e}_1$-maximal, respectively due to being the first and last vertices of the upper path. Hence, $\boldsymbol{e}_1^\intercal \boldsymbol{u}^0 \leq \boldsymbol{e}_1^\intercal \boldsymbol{q}^* \leq \boldsymbol{e}_1^\intercal \boldsymbol{u}^1$. It follows that the point $\boldsymbol{q} = (\boldsymbol{e}_1^\intercal \boldsymbol{q}^*, L(\boldsymbol{e}_1^\intercal \boldsymbol{q}^*))$ lies on the line segment from $\boldsymbol{u}^0$ to $\boldsymbol{u}^1$ and is therefore contained in $\pi(Q)$ by convexity. Since $\boldsymbol{q}^*$ is $\boldsymbol{e}_2$-maximal by assumption, $\boldsymbol{e}_2^\intercal \boldsymbol{q}^* \geq \boldsymbol{e}_2^\intercal \boldsymbol{q} = L(\boldsymbol{e}_1^\intercal \boldsymbol{q}^*)$. Thus, by definition, $\boldsymbol{q}^*$ is a vertex in the upper path.

Thus, all $\boldsymbol{e}_2$-maximal vertices are in the upper path of $\pi(Q)$. By our assumption of coherence, there is a vertex in the coherent monotone path Γ that projects to an $\boldsymbol{e}_2$-maximal vertex in $\pi(Q)$. Then, since $\boldsymbol{c}^\intercal \boldsymbol{x} = \boldsymbol{e}_2^\intercal \pi(\boldsymbol{x})$ for each $\boldsymbol{x}$ in Q, the $\boldsymbol{c}$-maximum of Q is attained at a point $\boldsymbol{x} \in Q$ exactly when $\pi(\boldsymbol{x})$ is $\boldsymbol{e}_2$-maximal in $\pi(Q)$. Therefore, the $\boldsymbol{c}$-maximum is attained on Γ.

By convexity, the slope from $\pi(\boldsymbol{x}_i)$ to $\pi(\boldsymbol{x}_{i+1})$ is at least the slope from $\pi(\boldsymbol{x}_{i+1})$ to $\pi(\boldsymbol{x}_{i+2})$ for each $0 \leq i \leq k-2$. Hence, the upper path is strictly $\boldsymbol{e}_2$-monotone exactly until it reaches an $\boldsymbol{e}_2$-maximum of $\pi(Q)$. Recall again that $\boldsymbol{e}_2^\intercal \pi(\boldsymbol{x}_i) = \boldsymbol{c}^\intercal \boldsymbol{x}_i$ for all $0 \leq i \leq k$. Thus, Γ is $\boldsymbol{c}$-monotone until it reaches a $\boldsymbol{c}$-maximum on Q, meaning that Γ is both $\boldsymbol{v}$-monotone and $\boldsymbol{c}$-monotone until it reaches a $\boldsymbol{c}$-maximum, as desired. □

To prove Theorems 3 and 4, the key idea is to choose the auxiliary vector $\boldsymbol{v}$ carefully so that the corresponding shadow paths are always short. The following corollary provides a bound on the lengths of the paths.

Corollary 4. *Let $Q \subseteq \mathbb{R}^n$ be a polytope, and let $\boldsymbol{v} \in \mathbb{R}^n \setminus \{\boldsymbol{0}\}$. Denote the set of vertices of Q by V, and let F be the face of Q minimized by $\boldsymbol{v}$. Then for any objective function $\boldsymbol{c}$, the $\boldsymbol{c}$-coherent $\boldsymbol{v}$-monotone path from a $\boldsymbol{c}$-maximum of F to a $\boldsymbol{c}$-maximum of Q is of length at most $|\{\, \boldsymbol{v}^\intercal \boldsymbol{u} : \boldsymbol{u} \in V \,\}| - 1$.*

Proof. Since the path is strictly $\boldsymbol{v}$-monotone, any two vertices $\boldsymbol{x}$ and $\boldsymbol{y}$ in the path satisfy $\boldsymbol{v}^\intercal \boldsymbol{x} \neq \boldsymbol{v}^\intercal \boldsymbol{y}$. It follows that the length of the path is at most $|\{\, \boldsymbol{v}^\intercal \boldsymbol{u} : \boldsymbol{u} \in V \,\}| - 1$. Furthermore, since the path is $\boldsymbol{c}$-coherent it reaches a maximum of $\boldsymbol{c}$ by Lemma 11. □

For the original Shadow pivot rule in [16], the choice of $\boldsymbol{v}$ is taken to be a random vector such that $\boldsymbol{v}^\intercal \boldsymbol{x}$ is minimized uniquely at the starting vertex. For our Shadow rules, we instead take advantage of the structure of 0/1 polytopes to make this choice of $\boldsymbol{v}$ explicitly to guarantee that $\boldsymbol{v}^\intercal \boldsymbol{x}$ always takes on few values, which by Corollary 4, yields short paths. In essence, we try to make $\boldsymbol{v}$ as degenerate as possible in place of Borgwardt's generic choice.

Finally, we observe that any Shadow rule can be computed in polynomial time.

Lemma 12. *Let (Q) be an LP with feasible region Q and objective function $\mathbf{c}^\intercal \boldsymbol{x}$, let $\boldsymbol{x}_0$ be a starting vertex of Q, and let $\boldsymbol{v}$ be chosen according to a particular Shadow rule. Then that Shadow rule can be computed in polynomial time on (Q).*

Proof. Let $\boldsymbol{x}_0, \ldots, \boldsymbol{x}_i$ be the path of vertices in Q followed by the Shadow rule up to the i^{th} step, and assume that $\boldsymbol{x}_i$ is not an optimal solution. Let $\boldsymbol{z}'$ be the edge-direction chosen by the Shadow rule at $\boldsymbol{x}_i$. Since $\boldsymbol{v}^\intercal \boldsymbol{z}' > 0$, assume without loss of generality that $\boldsymbol{v}^\intercal \boldsymbol{z}' = 1$. We will show that $\boldsymbol{z}'$ is the optimal solution to the LP $\mathcal{Q}$ defined by

$$\begin{aligned}
&\max \boldsymbol{c}^\intercal \boldsymbol{z} \\
&\text{s.t.} \\
&\boldsymbol{v}^\intercal \boldsymbol{z} = 1 \\
&\boldsymbol{z} \in \mathcal{C}(\boldsymbol{x}_i).
\end{aligned}$$

We will first show that for all elements $\boldsymbol{z} \in \mathcal{C}(\boldsymbol{x}_i)$, if $\boldsymbol{c}^\intercal \boldsymbol{z} > 0$, then $\boldsymbol{v}^\intercal \boldsymbol{z} > 0$. First, suppose that $i = 0$. Since $\boldsymbol{x}_0$ is the $\boldsymbol{c}$-maximum on the $\boldsymbol{v}$-minimal face, for any feasible direction $\boldsymbol{z} \in \mathcal{C}(\boldsymbol{x}_0)$ with $\boldsymbol{c}^\intercal \boldsymbol{z} > 0$, we have that $\boldsymbol{v}^\intercal \boldsymbol{z} > 0$, as desired.

Now suppose that $i \geq 1$, and let $\boldsymbol{y} = \boldsymbol{x}_i - \boldsymbol{x}_{i-1}$, the edge-direction taken in the previous step. We know that $\boldsymbol{v}^\intercal \boldsymbol{z}' > 0$ and $\boldsymbol{c}^\intercal \boldsymbol{z}' > 0$. Then we have that the feasible cone $\mathcal{C}(\boldsymbol{x}_i)$ contains the directions $\boldsymbol{z}'$ and $-\boldsymbol{y}$. This implies that the projection $\pi(\mathcal{C}(\boldsymbol{x}_i))$ contains an element whose pre-image satisfies $\boldsymbol{v}^\intercal \boldsymbol{z} < 0$ and $\boldsymbol{c}^\intercal \boldsymbol{z} < 0$ (namely, $\pi(-\boldsymbol{y})$), and a direction satisfying $\boldsymbol{v}^\intercal \boldsymbol{z} > 0$ and $\boldsymbol{c}^\intercal \boldsymbol{z} > 0$ (namely, $\pi(\boldsymbol{z}')$). Recall that, under the projection π, $\boldsymbol{v}$ and $\boldsymbol{c}$ act as coordinate vectors for the first and second coordinate of space, respectively. Then by convexity, the cone $\pi(\mathcal{C}(\boldsymbol{x}_i))$ does not contain any element of the orthant of $\mathbb{R}^2$ defined by $\boldsymbol{v}^\intercal \boldsymbol{z} \leq 0$, $\boldsymbol{c}^\intercal \boldsymbol{z} \geq 0$. Therefore, by the definition of π, $\mathcal{C}(\boldsymbol{x}_i)$ does not contain any element satisfying both $\boldsymbol{v}^\intercal \boldsymbol{z} \leq 0$ and $\boldsymbol{c}^\intercal \boldsymbol{z} \geq 0$. Then for all elements $\boldsymbol{z} \in \mathcal{C}(\boldsymbol{x}_i)$, if $\boldsymbol{c}^\intercal \boldsymbol{z} > 0$, then $\boldsymbol{v}^\intercal \boldsymbol{z} > 0$, as desired.

We now observe that $\mathcal{Q}$ is not unbounded. Since $\boldsymbol{v}^\intercal \boldsymbol{z} > 0$ for all $\boldsymbol{z} \in \mathcal{C}(\boldsymbol{x}_i)$ satisfying $\boldsymbol{c}^\intercal \boldsymbol{z} > 0$, we have that the set $Z = \{\, \boldsymbol{z} \in \mathcal{C}(\boldsymbol{x}_i) : \boldsymbol{v}^\intercal \boldsymbol{z} \leq 1,\ \boldsymbol{c}^\intercal \boldsymbol{z} > 0 \,\}$ is a bounded set. The set of feasible solutions to $\mathcal{Q}$ with positive objective value is contained in the set Z, and so it is also a bounded set. Therefore, $\mathcal{Q}$ is not unbounded. Then since the feasible region of $\mathcal{Q}$ is the feasible cone at $\boldsymbol{x}_i$ intersected with a single half-space, the non-zero extreme point solutions of $\mathcal{Q}$ correspond to edge-directions at $\boldsymbol{x}_i$. Let $\boldsymbol{z}''$ be any other $\boldsymbol{v}$-increasing edge-direction at $\boldsymbol{x}_i$ normalized so that $\boldsymbol{v}^\intercal \boldsymbol{z}'' = 1$. Then by definition,

$$\boldsymbol{c}^\intercal \boldsymbol{z}' = \frac{\boldsymbol{c}^\intercal \boldsymbol{z}'}{\boldsymbol{v}^\intercal \boldsymbol{z}'} \geq \frac{\boldsymbol{c}^\intercal \boldsymbol{z}''}{\boldsymbol{v}^\intercal \boldsymbol{z}''} = \boldsymbol{c}^\intercal \boldsymbol{z}'',$$

and therefore $\boldsymbol{z}'$ is the optimal solution to $\mathcal{Q}$. It follows that $\boldsymbol{z}'$ can be computed in polynomial time, as desired. □

4.3.2 The Slim Shadow Rule

We now return to the case of 0/1-LPs of the form $\max\{\, \boldsymbol{c}^\intercal \boldsymbol{x} : \boldsymbol{x} \in P \,\}$ where $P = \{\boldsymbol{x} \in \mathbb{R}^n : A\boldsymbol{x} = \boldsymbol{b},\, D\boldsymbol{x} \leq \boldsymbol{d}\}$, and the feasible region is a 0/1 polytope of dimension d. The Slim Shadow rule is given by the following Shadow rule:

Definition 11. *Given a 0/1-LP of the form (1.1) with feasible region P, let $\boldsymbol{x}_0$ be any initial vertex of P. Let $\boldsymbol{v} = \mathbf{1} - 2\boldsymbol{x}_0$. At a vertex $\boldsymbol{x}_i$ of P, the Slim Shadow rule moves to a neighbor*

$$\boldsymbol{x}_{i+1} = \operatorname*{argmax}_{\boldsymbol{u} \in N_{\boldsymbol{v}}(\boldsymbol{x}_i)} \left(\frac{\boldsymbol{c}^\intercal (\boldsymbol{u} - \boldsymbol{x}_i)}{\boldsymbol{v}^\intercal (\boldsymbol{u} - \boldsymbol{x}_i)} \right).$$

Recall that, by Lemma 11, the maximum in this definition is always attained at a neighbor $\boldsymbol{u}$ satisfying $\boldsymbol{c}^\intercal \boldsymbol{u} > \boldsymbol{c}^\intercal \boldsymbol{x}_i$ whenever $\boldsymbol{x}_i$ is not $\boldsymbol{c}$-maximal.

Note that, although the Slim Shadow rule defines $\boldsymbol{v}$ *similarly* to the way it is defined for the True Steepest-Edge pivot rule, they are not precisely the same. While for the True Steepest-Edge pivot rule we change $\boldsymbol{v}$ at each new extreme point, for the Slim Shadow rule, $\boldsymbol{v}$ never changes. However, they *are* defined identically at the *initial* extreme point solution, and as we will see in Section 5.3 when we extend this rule to a pivot rule, the vector $\boldsymbol{v}$ also plays a similar role to the one it plays in the analysis of the True Steepest-Edge rule.

As discussed earlier, we chose this $\boldsymbol{v}$ because $\boldsymbol{v}^\intercal \boldsymbol{x}$ takes on very few distinct values in 0/1-LPs. For example, consider the case where P is the 0/1 cube $[0,1]^n$. Then for $\boldsymbol{v} = \mathbf{1}$, $\boldsymbol{v}^\intercal \boldsymbol{x}$ takes on precisely $n+1$ values at vertices of P, given by the possible numbers of nonzero coordinates in each vertex of the cube. For the Slim Shadow rule on the cube, if we choose $\mathbf{0}$ as our starting point, we have $\boldsymbol{v} = \mathbf{1} - 2(\mathbf{0}) = \mathbf{1}$. Thus, Corollary 4 tells us that the length of a monotone path chosen by the Slim Shadow rule on the cube starting at the point $\mathbf{0}$ is at most n. We generalize this bound to all 0/1 polytopes.

Theorem 18. *On any* $0/1$*-LP of the form (1.1), the Slim Shadow rule reaches an optimal solution by performing at most* n *steps.*

Proof. Let the LP be $\max\{\boldsymbol{c}^\intercal \boldsymbol{x} : \boldsymbol{x} \in P\}$ where $P = \{\boldsymbol{x} \in \mathbb{R}^n : A\boldsymbol{x} = \boldsymbol{b},\, D\boldsymbol{x} \leq \boldsymbol{d}\}$ is a $0/1$ polytope. Let $\boldsymbol{x}_0$ be an initial extreme point solution. Let $S = \{s \in [n] : \boldsymbol{x}_0(s) = 1\}$. Note that in the cube $[0,1]^n$, $\boldsymbol{x}_0$ is the unique minimizer of the linear function $\boldsymbol{v}^\intercal \boldsymbol{x}$ where $\boldsymbol{v} = \mathbf{1} - 2\boldsymbol{x}_0$. Hence, $\boldsymbol{x}_0$ is the unique $\boldsymbol{v}$-minimizer on P, since all vertices of P are vertices of the cube.

By Corollary 4, the $\boldsymbol{c}$-coherent $\boldsymbol{v}$-monotone path reaches the maximum of $\boldsymbol{c}$ from $\boldsymbol{x}_0$ in at most $|\{\boldsymbol{v}^\intercal \boldsymbol{x} : \boldsymbol{x} \in V\}| - 1$ steps. Let $\bar{S} = [n] \setminus S$. Then $-|S| \leq \boldsymbol{v}^\intercal \boldsymbol{x} \leq |\bar{S}|$ for all $\boldsymbol{x} \in \{0,1\}^n$, so $|\{\boldsymbol{v}^\intercal \boldsymbol{x} : \boldsymbol{x} \in V\}| \leq |S| + |\bar{S}| + 1 = n + 1$. Therefore, the length of the path is at most n. □

For a few special cases, we may tighten the bounds on the lengths of paths found by the Slim Shadow rule.

Lemma 13. *On any* $0/1$*-LP of the form (1.1) in which the number of nonzero coordinates among all vertices is a constant* M*, the Slim Shadow rule takes at most* M *steps.*

Proof. Let the LP be $\max\{\boldsymbol{c}^\intercal \boldsymbol{x} : \boldsymbol{x} \in P\}$ where $P = \{\boldsymbol{x} \in \mathbb{R}^n : A\boldsymbol{x} = \boldsymbol{b},\, D\boldsymbol{x} \leq \boldsymbol{d}\}$ is a $0/1$ polytope. Let $\boldsymbol{x}_0$ be an initial vertex, and let $S = \{s \in [n] : \boldsymbol{x}_0(s) = 1\}$. Then by assumption, $|S| = M$, and the linear function $-(\boldsymbol{x}_0)^\intercal \boldsymbol{x}$ takes on at most $M+1$ distinct values on P given by the different possible sizes of subsets of S. Furthermore, by assumption, $\mathbf{1}^\intercal \boldsymbol{x}$ always yields the same value when applied to any vertex on P. Hence, $(\mathbf{1} - 2\boldsymbol{x}_0)^\intercal \boldsymbol{x}$ takes on at most $M+1$ distinct values on vertices of P. Thus, by Corollary 4, the length of the path used by the Slim Shadow rule starting at $\boldsymbol{x}_0$ is at most M. □

Note that this bound is tight when the number of nonzero coordinates k is less than $n/2$, since the monotone diameter of the hyper-simplex $\Delta(n,k)$ in that case is easily verified to be k. For $0/1$-LPs containing $\mathbf{0}$ as an extreme point solution, we may improve this bound in an analogous manner.

Lemma 14. *On any* $0/1$*-LP of the form (1.1) in which* $\mathbf{0}$ *is a vertex of the feasible region* P *and in which each vertex has at most* M *nonzero entries, the Slim Shadow rule starting at* $\mathbf{0}$ *takes at most* M *steps.*

Proof. Let the LP be $\max\{\boldsymbol{c}^\intercal \boldsymbol{x} : \boldsymbol{x} \in P\}$ where $P = \{\boldsymbol{x} \in \mathbb{R}^n : A\boldsymbol{x} = \boldsymbol{b},\, D\boldsymbol{x} \leq \boldsymbol{d}\}$ is a $0/1$ polytope. Since we are starting at $\mathbf{0}$, we have that $\boldsymbol{v} = \mathbf{1}$. By assumption, $\mathbf{1}^\intercal \boldsymbol{x}$ takes on at most $M+1$ distinct values on vertices of P. Hence, the Slim Shadow rule will take at most M steps. □

4.3.3 The Ordered Shadow Rule

The bound given for the Slim Shadow rule is given in terms of the number of variables n. However, from [74], we know that the diameter of a 0/1 polytope is at most its dimension d. To attain a bound of at most d steps, we introduce the *Ordered Shadow rule*.

Definition 12. *Given a 0/1-LP of the form (1.1) with feasible region P, let $\boldsymbol{x}_0$ be an initial extreme point in P. Let $c^* = ||\boldsymbol{c}||_1 + 2$. Define $\boldsymbol{v} \in \mathbb{R}^n$ by $\boldsymbol{v}(k) = (-1)^{\boldsymbol{x}_0(k)}(c^*)^k$. At a vertex $\boldsymbol{x}_i$ of P, the Ordered Shadow rule moves to a neighbor*

$$\boldsymbol{x}_{i+1} = \underset{\boldsymbol{u} \in N_{\boldsymbol{v}}(\boldsymbol{x}_i)}{\operatorname{argmax}} \frac{\boldsymbol{c}^\intercal(\boldsymbol{u} - \boldsymbol{x}_i)}{\boldsymbol{v}^\intercal(\boldsymbol{u} - \boldsymbol{x}_i)}.$$

Like the Slim Shadow rule, the Ordered Shadow rule follows a shadow path found by choosing an auxiliary vector carefully. Note also that Definition 12 implicitly assumes an ordering on the coordinates. Namely, we chose $|\boldsymbol{v}(k)| = (c^*)^k$ but could also choose $|\boldsymbol{v}(k)| = (c^*)^{\sigma(k)}$ for any permutation $\sigma : [n] \to [n]$, giving a different order of the variables. Different orderings of variables can yield paths of different lengths. However, for any choice of σ, $\boldsymbol{v}$ is minimized uniquely at $\boldsymbol{x}_0$ and the length of the path will still be at most d by the same proof we provide here. To prove that the path is always short, we may no longer directly apply Corollary 4. Instead, we show that the path followed satisfies another equivalent characterization for which the length of the path is easier to analyze. We first bound the length of a path satisfying this alternative characterization.

Lemma 15. *Consider a 0/1-LP of the form (1.1) with d-dimensional feasible region P. Let $\boldsymbol{x}_0$ be an initial vertex in P. We build a monotone path Γ on P as follows: Define $f : \mathbb{R}^n \to \{0, 1, \ldots, n\}$ by $f(\boldsymbol{u}) = \max(\{k : \boldsymbol{u}(k) - \boldsymbol{x}_0(k) \neq 0\} \cup \{0\})$. Given the i-th vertex $\boldsymbol{x}_i$ of the path, let $N_{\min}(\boldsymbol{x}_i)$ be the set of f-minimal $\boldsymbol{c}$-improving neighbors of $\boldsymbol{x}_i$. Select the next extreme point of the path $\boldsymbol{x}_{i+1}$ as the $\boldsymbol{c}$-maximum of $N_{\min}(\boldsymbol{x}_i)$. The length of the path Γ constructed in this way is at most d.*

Proof. Let the LP be $\max\{\,\boldsymbol{c}^\intercal\boldsymbol{x} : \boldsymbol{x} \in P\,\}$ where $P = \{\boldsymbol{x} \in \mathbb{R}^n : A\boldsymbol{x} = \boldsymbol{b},\, D\boldsymbol{x} \leq \boldsymbol{d}\}$ is a 0/1 polytope, and let $\Gamma = [\boldsymbol{x}_0, \boldsymbol{x}_1, \ldots]$ be the path followed by the rule described in the statement. Define $H_k = \{\boldsymbol{x} \in \mathbb{R}^d : \boldsymbol{x}(a) = \boldsymbol{x}_0(a) \text{ for all } a \geq k+1\}$ to be the plane given by fixing the last $n-k$ coordinates of a vector to agree with $\boldsymbol{x}_0$. Let $F_k = H_k \cap P$. Note that $[0,1]^n \cap H_k$ is a face of the n-cube, so since P is a 0/1 polytope, F_k is also a face of P for all $0 \leq k \leq n$. We equivalently have that $F_k = \{\boldsymbol{x} \in P : f(\boldsymbol{x}) \leq k\}$.

Observe that $F_0 = \boldsymbol{x}_0$, $F_n = P$, and $F_k \subseteq F_{k+1}$ for all $k \in [n]$. Consider $\boldsymbol{x}_i$, the ith vertex in the path. Suppose that, for each choice of $i \geq 0$, $\boldsymbol{x}_i$ and $\boldsymbol{x}_{i+1}$ are $\boldsymbol{c}$-maxima of some $F_{\alpha(i)}$ and $F_{\alpha(i+1)}$ respectively for some function $\alpha : \mathbb{N} \to [n]$. Then each vertex in the path is associated to a face. Since the path is $\boldsymbol{c}$-monotone, $F_{\alpha(i)}$ is a proper face of $F_{\alpha(i+1)}$. Therefore, because the dimension of each associated face must strictly increase, the length of the path is at most d. Thus, to finish the proof, it suffices to show that $\boldsymbol{x}_i$ and $\boldsymbol{x}_{i+1}$ are $\boldsymbol{c}$-maxima of $F_{f(\boldsymbol{x}_{i+1})-1}$ and $F_{f(\boldsymbol{x}_{i+1})}$ respectively.

Suppose for the sake of contradiction that $\boldsymbol{x}_i$ is not a $\boldsymbol{c}$-maximum on the face $F_{f(\boldsymbol{x}_{i+1})-1}$. Then there exists a vertex $\boldsymbol{u}$ of $F_{f(\boldsymbol{x}_{i+1})-1}$ adjacent to $\boldsymbol{x}_i$ with $\boldsymbol{c}^\intercal \boldsymbol{u} > \boldsymbol{c}^\intercal \boldsymbol{x}_i$. Thus, we must have $f(\boldsymbol{u}) \leq f(\boldsymbol{x}_{i+1}) - 1 < f(\boldsymbol{x}_{i+1})$. However, by the definition of f, $f(\boldsymbol{x}_{i+1})$ is minimal among all $\boldsymbol{c}$-improving neighbors of $\boldsymbol{x}_i$, a contradiction. Hence, $\boldsymbol{x}_i$ is a $\boldsymbol{c}$-maximum of $F_{f(\boldsymbol{x}_{i+1})-1}$.

Consider the LP $\max\{\boldsymbol{c}^\intercal \boldsymbol{x} : \boldsymbol{x} \in F_{f(\boldsymbol{x}_{i+1})}\}$. It remains to show that $\boldsymbol{x}_{i+1}$ is an optimal solution to this LP. We may take $\boldsymbol{x}_i$ to be our initial point and argue that from this starting point, this LP may be solved by following a path of one step by the Shadow rule with auxiliary vector $\boldsymbol{e}_{f(\boldsymbol{x}_{i+1})}$ when $\boldsymbol{x}_0(f(\boldsymbol{x}_{i+1})) = 0$ or $-\boldsymbol{e}_{f(\boldsymbol{x}_{i+1})}$ when $\boldsymbol{x}_0(f(\boldsymbol{x}_{i+1})) = 1$. We then argue that $\boldsymbol{x}_{i+1}$ is a valid choice for this Shadow rule.

In what remains, we shall assume that $\boldsymbol{x}_0(f(\boldsymbol{x}_{i+1})) = 0$, but a completely analogous argument follows when $\boldsymbol{x}_0(f(\boldsymbol{x}_{i+1})) = 1$. Since $\boldsymbol{x}_0(f(\boldsymbol{x}_{i+1})) = 0$, $F_{f(\boldsymbol{x}_{i+1})-1}$ is the $\boldsymbol{e}_{f(\boldsymbol{x}_{i+1})}$-minimal face of $F_{f(\boldsymbol{x}_{i+1})}$. We have already shown that $\boldsymbol{x}_i$ is a $\boldsymbol{c}$-maximum of that face. Thus, $\boldsymbol{x}_i$ is a valid starting point for the Shadow rule with auxiliary vector $\boldsymbol{e}_{f(\boldsymbol{x}_{i+1})}$ for that LP.

Note that P is 0/1, so $\boldsymbol{e}_{f(\boldsymbol{x}_{i+1})}^\intercal \boldsymbol{x} = \boldsymbol{x}(f(\boldsymbol{x}_{i+1}))$ takes on at most two values on vertices of P: 0 or 1. In particular, $\boldsymbol{e}_{f(\boldsymbol{x}_{i+1})}^\intercal \boldsymbol{x}$ takes on at most two values on the vertices of the face $F_{f(\boldsymbol{x}_{i+1})}$. Hence, by Corollary 4, a $\boldsymbol{c}$-coherent $\boldsymbol{e}_{f(\boldsymbol{x}_{i+1})}$-monotone path from a $\boldsymbol{c}$-maximum of the $\boldsymbol{e}_{f(\boldsymbol{x}_{i+1})}$-minimal face of $F_{f(\boldsymbol{x}_{i+1})}$ is of length at most $2 - 1 = 1$. Hence, a $\boldsymbol{c}$-coherent $\boldsymbol{e}_{f(\boldsymbol{x}_{i+1})}$-monotone path on $F_{f(\boldsymbol{x}_{i+1})}$ starting at $\boldsymbol{x}_i$ is of length at most one.

Thus, $\boldsymbol{x}_i$ is either equal to or adjacent to a $\boldsymbol{c}$-maximum on $F_{f(\boldsymbol{x}_{i+1})}$. Note that $\boldsymbol{x}_i, \boldsymbol{x}_{i+1} \in F_{f(\boldsymbol{x}_{i+1})}$, and $\boldsymbol{c}^\intercal \boldsymbol{x}_{i+1} > \boldsymbol{c}^\intercal \boldsymbol{x}_i$. Hence, $\boldsymbol{x}_i$ is not a $\boldsymbol{c}$-maximum on $F_{f(\boldsymbol{x}_{i+1})}$. It follows that $\boldsymbol{x}_i$ is adjacent to such a $\boldsymbol{c}$-maximum, and in particular, any vertex chosen by the Shadow rule on $F_{f(\boldsymbol{x}_{i+1})}$ with auxiliary vector $\boldsymbol{e}_{f(\boldsymbol{x}_{i+1})}$ must be $\boldsymbol{c}$-maximal.

Equivalently, a vertex $\boldsymbol{y}^*$ is $\boldsymbol{c}$-maximal on $F_{f(\boldsymbol{x}_{i+1})}$ whenever

$$\boldsymbol{y}^* \in \operatorname*{argmax}_{\boldsymbol{u} \in N_{i+1}} \frac{\boldsymbol{c}^\intercal(\boldsymbol{u} - \boldsymbol{x}_i)}{\boldsymbol{e}_{f(\boldsymbol{x}_{i+1})}^\intercal(\boldsymbol{u} - \boldsymbol{x}_i)},$$

where N_{i+1} is the set of $\boldsymbol{e}_{f(\boldsymbol{x}_{i+1})}$-improving neighbors $\boldsymbol{u}$ of $\boldsymbol{x}_i$ in $F_{f(\boldsymbol{x}_{i+1})}$. Note that any $\boldsymbol{e}_{f(\boldsymbol{x}_{i+1})}$-improving neighbor $\boldsymbol{u}$ of $\boldsymbol{x}_i$ must satisfy $\boldsymbol{e}_{f(\boldsymbol{x}_{i+1})}^\intercal(\boldsymbol{u} - \boldsymbol{x}_i) = 1 - 0 = 1$, since P is 0/1.

Furthermore, observe that $\boldsymbol{x}_{i+1}$ must be f-minimal among all $\boldsymbol{c}$-improving neighbors of $\boldsymbol{x}_i$, since $\boldsymbol{x}_i$ is the $\boldsymbol{c}$-maximum of $F_{f(\boldsymbol{x}_{i+1})-1}$. Thus, because $F_{f(\boldsymbol{x}_{i+1})} = \{\boldsymbol{x} : f(\boldsymbol{x}) \leq f(\boldsymbol{x}_{i+1})\}$, all $\boldsymbol{c}$-improving neighbors of $\boldsymbol{x}_i$ in $F_{f(\boldsymbol{x}_{i+1})}$ must be f-minimal. Hence, the set of $\boldsymbol{c}$-improving neighbors of $\boldsymbol{x}_i$ in $F_{f(\boldsymbol{x}_{i+1})}$ is exactly $N_{\min}(\boldsymbol{x}_i)$. It follows that

$$\boldsymbol{x}_{i+1} \in \underset{\boldsymbol{u}\in N_{\min}(\boldsymbol{x}_i)}{\operatorname{argmax}} (\boldsymbol{c}^\intercal \boldsymbol{u}) = \underset{\boldsymbol{u}= N_{\min}(\boldsymbol{x}_i)}{\operatorname{argmax}} (\boldsymbol{c}^\intercal (\boldsymbol{u}-\boldsymbol{x}_i)) = \underset{\boldsymbol{u}\in N_{i+1}}{\operatorname{argmax}} \frac{\boldsymbol{c}^\intercal(\boldsymbol{u}-\boldsymbol{x}_i)}{\mathbf{e}_i^\intercal(\boldsymbol{u}-\boldsymbol{x}_i)}.$$

Therefore, $\boldsymbol{x}_{i+1}$ is a $\boldsymbol{c}$-maximum of $F_{f(\boldsymbol{x}_{i+1})}$ as desired. □

See Figure 4.3 for an example of the construction of Lemma 15.

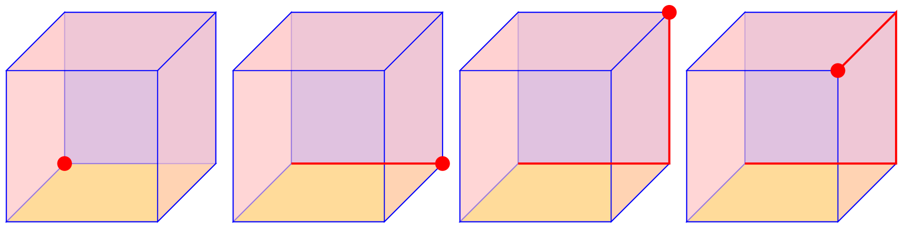

Figure 4.3: The construction of Lemma 15 yields the displayed path for maximizing $\boldsymbol{c}^\intercal = (1, 2, 3)$ on the cube $[0, 1]^3$ starting at $\mathbf{0}$. Observe that $\mathbf{0}$ is trivially the $\boldsymbol{c}$-maximum of the face in which all coordinates are fixed to be 0. Then the path moves to $(1, 0, 0)$, the $\boldsymbol{c}$-maximum of the edge given by fixing the final two coordinates to equal 0. The next step lands at $(1, 1, 0)$, the $\boldsymbol{c}$-maximum on the face in which the final coordinate fixed at 0. Finally, the path ends at $(1, 1, 1)$, the $\boldsymbol{c}$-maximum on $[0, 1]^3$.

We may use the bound on the path constructed in Lemma 15 to bound the length of the path followed by the Ordered Shadow rule.

Theorem 19. *Consider a 0/1-LP of the form (1.1) with d-dimensional feasible region P. The Ordered Shadow rule follows a path Γ as described in the statement of Lemma 15. Hence, the number of steps taken by the Ordered Shadow rule to arrive at an optimal solution is at most d.*

Proof. Let $\Gamma = [\boldsymbol{x}_0, \boldsymbol{x}_1, \ldots, \boldsymbol{x}_k]$ be the path followed by the Ordered Shadow rule. In particular, we have

$$\boldsymbol{x}_{i+1} \in \underset{\boldsymbol{u}\in N_{\boldsymbol{v}}(\boldsymbol{x}_i)}{\operatorname{argmax}} \frac{\boldsymbol{c}^\intercal(\boldsymbol{u}-\boldsymbol{x}_i)}{\boldsymbol{v}^\intercal(\boldsymbol{u}-\boldsymbol{x}_i)}.$$

Our goal is to show that $\boldsymbol{x}_{i+1}$ is a $\boldsymbol{c}$-maximal element of $N_{\min}(\boldsymbol{x}_i)$. To prove this, we will show the following three claims hold.

Claim 1: Let $\boldsymbol{u}$ be a $\boldsymbol{c}$-improving neighbor of $\boldsymbol{x}_i$. Then $\boldsymbol{u}$ is also $\boldsymbol{v}$-improving.

Claim 2: Let $\boldsymbol{u}^0$ and $\boldsymbol{u}^1$ be neighbors of $\boldsymbol{x}_i$ that are both $\boldsymbol{c}$-improving and $\boldsymbol{v}$-improving. Suppose that $b = f(\boldsymbol{u}^1) > f(\boldsymbol{u}^0) = a$. Then we have

$$\frac{\boldsymbol{c}^\intercal(\boldsymbol{u}^0 - \boldsymbol{x}_i)}{\boldsymbol{v}^\intercal(\boldsymbol{u}^0 - \boldsymbol{x}_i)} > \frac{\boldsymbol{c}^\intercal(\boldsymbol{u}^1 - \boldsymbol{x}_i)}{\boldsymbol{v}^\intercal(\boldsymbol{u}^1 - \boldsymbol{x}_i)}.$$

Claim 3: Let $\boldsymbol{u}^0$ and $\boldsymbol{u}^1$ be neighbors of $\boldsymbol{x}_i$ that are both $\boldsymbol{c}$-improving and $\boldsymbol{v}$-improving. Suppose that $b = f(\boldsymbol{u}^1) = f(\boldsymbol{u}^0)$ and $\boldsymbol{c}^\intercal(\boldsymbol{u}^0) > \boldsymbol{c}^\intercal(\boldsymbol{u}^1)$. Then we have

$$\frac{\boldsymbol{c}^\intercal(\boldsymbol{u}^0 - \boldsymbol{x}_i)}{\boldsymbol{v}^\intercal(\boldsymbol{u}^0 - \boldsymbol{x}_i)} > \frac{\boldsymbol{c}^\intercal(\boldsymbol{u}^1 - \boldsymbol{x}_i)}{\boldsymbol{v}^\intercal(\boldsymbol{u}^1 - \boldsymbol{x}_i)}.$$

Suppose that Claims 1-3 are true. By Claim 1, all $\boldsymbol{c}$-improving neighbors of $\boldsymbol{x}_i$ are $\boldsymbol{v}$-improving, so the results of Claims 2 and 3 hold under the weaker assumption that $\boldsymbol{u}^0$ and $\boldsymbol{u}^1$ are any $\boldsymbol{c}$-improving neighbors. Claim 2 shows us that when $f(\boldsymbol{u}^1) > f(\boldsymbol{u}^0)$, $\boldsymbol{u}^0$ would be chosen over the $\boldsymbol{u}^1$ by the Ordered Shadow rule. Thus, $\boldsymbol{x}_{i+1}$ must be f-minimal among all neighbors that are $\boldsymbol{c}$-improving. That is, $\boldsymbol{x}_{i+1} \in N_{\min}(\boldsymbol{x}_i)$. Similarly, Claim 3 shows that the Ordered Shadow rule will always choose a neighbor with larger $\boldsymbol{c}$-value among two f-minimal neighbors. That is, $\boldsymbol{x}_{i+1}$ is a $\boldsymbol{c}$-maximal element of $N_{\min}(\boldsymbol{x}_i)$, which yields the result. Therefore, to prove the theorem, it suffices to prove Claims 1-3.

Note that each claim assumes that we start at a point $\boldsymbol{x}_i$ in the path. By induction, we may assume that the path up to $\boldsymbol{x}_i$ is of the desired type with base case satisfied for $i = 0$ by hypothesis. Equivalently, by the proof of Lemma 15, we are able to assume that $\boldsymbol{x}_i$ is the $\boldsymbol{c}$-maximum of the face $F_{f(\boldsymbol{x}_i)} = \{\boldsymbol{x} : f(\boldsymbol{x}) \leq f(\boldsymbol{x}_i)\}$. As a result, we can apply a key assumption that all $\boldsymbol{c}$-improving neighbors of $\boldsymbol{x}_i$ have larger f-value.

We may also assume without loss of generality that $\boldsymbol{x}_0 = \boldsymbol{0}$, which may be accomplished by a change of coordinates. In that case, $\boldsymbol{v} = (c^*, (c^*)^2, \ldots, (c^*)^n)$, and the following final claim will simplify our arguments for the proofs of Claims 1-3.

Claim 4: Let $\boldsymbol{u}^0$ and $\boldsymbol{u}^1$ be neighbors of $\boldsymbol{x}_i$ that are both $\boldsymbol{c}$-improving and $\boldsymbol{v}$-improving. Let $b = \max\{f(\boldsymbol{u}^0), f(\boldsymbol{u}^1)\}$, and let

$$\kappa_j = \boldsymbol{c}^\intercal(\boldsymbol{u}^0 - \boldsymbol{x}_i)(\boldsymbol{u}^1(j) - \boldsymbol{x}_i(j)) - \boldsymbol{c}^\intercal(\boldsymbol{u}^1 - \boldsymbol{x}_i)(\boldsymbol{u}^0(j) - \boldsymbol{x}_i(j)).$$

Then we have $\frac{\boldsymbol{c}^\intercal(\boldsymbol{u}^0 - \boldsymbol{x}_i)}{\boldsymbol{v}^\intercal(\boldsymbol{u}^0 - \boldsymbol{x}_i)} > \frac{\boldsymbol{c}^\intercal(\boldsymbol{u}^1 - \boldsymbol{x}_i)}{\boldsymbol{v}^\intercal(\boldsymbol{u}^1 - \boldsymbol{x}_i)}$ if and only if

$$\kappa_b(c^*)^b > \sum_{j=1}^{b-1} -\kappa_j(c^*)^j.$$

Proof of Claim 4: Since $\boldsymbol{u}^0$ and $\boldsymbol{u}^1$ are both $\boldsymbol{c}$-improving and $\boldsymbol{v}$-improving neighbors, the numerators and denominators of each term of the first inequality are all positive. By rearranging, we arrive at the following equivalent inequality:

$$\boldsymbol{c}^\intercal(\boldsymbol{u}^0 - \boldsymbol{x}_i)\boldsymbol{v}^\intercal(\boldsymbol{u}^1 - \boldsymbol{x}_i) > \boldsymbol{c}^\intercal(\boldsymbol{u}^1 - \boldsymbol{x}_i)\boldsymbol{v}^\intercal(\boldsymbol{u}^0 - \boldsymbol{x}_i).$$

Note that $\boldsymbol{v} = (c^*, (c^*)^2, \ldots, (c^*)^n)$, so by evaluating $\boldsymbol{v}^\intercal(\boldsymbol{u}^0 - \boldsymbol{x}_i)$ and $\boldsymbol{v}^\intercal(\boldsymbol{u}^1 - \boldsymbol{x}_i)$ we find the inequality is equivalent to the following:

$$\boldsymbol{c}^\intercal(\boldsymbol{u}^0 - \boldsymbol{x}_i)\sum_{j=1}^{n}(c^*)^j(\boldsymbol{u}^1(j) - \boldsymbol{x}_i(j)) > \boldsymbol{c}^\intercal(\boldsymbol{u}^1 - \boldsymbol{x}_i)\sum_{j=1}^{n}(c^*)^j(\boldsymbol{u}^0(j) - \boldsymbol{x}_i(j)).$$

In what remains, it is useful to view the left and right hand sides of the equations as polynomials in c^*. To simplify this expression further, we will make use of the definition of f from Lemma 15. Recall that, by induction, we may assume that all neighbors of $\boldsymbol{x}_i$ are f-improving. Under this assumption, we claim that $f(\boldsymbol{u}^1)$ and $f(\boldsymbol{u}^0)$ are exactly the indices of the highest powers of c^* in the left and right polynomials respectively. To see this, recall that $f(\boldsymbol{u}^i)$ is the highest index j for which $\boldsymbol{u}^i(j) \neq \boldsymbol{x}_0(j) = 0$.

By our inductive hypothesis, $f(\boldsymbol{u}^0) > f(\boldsymbol{x}_i)$. Equivalently, the highest index j for which $\boldsymbol{u}^0(j) \neq 0$ is greater than the highest index k for which $\boldsymbol{x}_i(k) \neq 0$. Thus, $\boldsymbol{u}^0(j) = \boldsymbol{x}_0(j) = 0$ for all $j > f(\boldsymbol{u}^0)$ by our definition of f. Similarly $\boldsymbol{u}^0(f(\boldsymbol{u}^0)) = 1$ by the definition of f. At the same time $\boldsymbol{x}_i(f(\boldsymbol{u}^0)) = 0$, since $f(\boldsymbol{u}^0) > f(\boldsymbol{x}_i)$. Thus, we have that $\boldsymbol{u}^0(j) - \boldsymbol{x}_i(j) = 0 - 0 = 0$ for $j > f(\boldsymbol{u}^0)$, and $\boldsymbol{u}^0(f(\boldsymbol{u}^0)) - \boldsymbol{x}_i(f(\boldsymbol{u}^0)) = 1 - 0 = 1$. Hence, $f(\boldsymbol{u}^0)$ is the highest nonzero coefficient of $(c^*)^j$ for the right polynomial. The same exact argument follows for $\boldsymbol{u}^1$, which establishes that it suffices to understand when the following inequality holds:

$$\boldsymbol{c}^\intercal(\boldsymbol{u}^0 - \boldsymbol{x}_i)\sum_{j=1}^{f(\boldsymbol{u}^1)}(c^*)^j(\boldsymbol{u}^1(j) - \boldsymbol{x}_i(j)) > \boldsymbol{c}^\intercal(\boldsymbol{u}^1 - \boldsymbol{x}_i)\sum_{j=1}^{f(\boldsymbol{u}^0)}(c^*)^j(\boldsymbol{u}^0(j) - \boldsymbol{x}_i(j)).$$

Recall that $b = \max\{f(\boldsymbol{u}^0), f(\boldsymbol{u}^1)\}$. Then we may reduce the expression to understanding when

$$\boldsymbol{c}^\intercal(\boldsymbol{u}^0 - \boldsymbol{x}_i)\sum_{j=1}^{b}(c^*)^j(\boldsymbol{u}^1(j) - \boldsymbol{x}_i(j)) > \boldsymbol{c}^\intercal(\boldsymbol{u}^1 - \boldsymbol{x}_i)\sum_{j=1}^{b}(c^*)^j(\boldsymbol{u}^0(j) - \boldsymbol{x}_i(j)),$$

which we will do via order of magnitude estimates. To do these estimates, we rearrange the inequality so that every term is on one side. This yields the inequality

$$\sum_{j=1}^{b}\kappa_j(c^*)^j > 0,$$

where we define

$$\kappa_j = \boldsymbol{c}^\intercal(\boldsymbol{u}^0 - \boldsymbol{x}_i)(\boldsymbol{u}^1(j) - \boldsymbol{x}_i(j)) - \boldsymbol{c}^\intercal(\boldsymbol{u}^1 - \boldsymbol{x}_i)(\boldsymbol{u}^0(j) - \boldsymbol{x}_i(j)).$$

Note that κ_j is exactly the difference of terms multiplied by $(c^*)^j$ after moving all terms to the same side. By moving all the smaller degree terms back to the other side, we finally arrive at the desired inequality:

$$\kappa_b(c^*)^b > \sum_{j=1}^{b-1} -\kappa_j(c^*)^j.$$

Proof of Claim 1: Recall that, by induction, we may assume that all $\boldsymbol{c}$-improving neighbors of $\boldsymbol{x}_i$ are also f-improving. Thus, it suffices to show that f-improving neighbors are $\boldsymbol{v}$-improving. Suppose that $f(\boldsymbol{u}) > f(\boldsymbol{x}_i)$ for some $\boldsymbol{c}$-improving neighbor $\boldsymbol{u}$ of $\boldsymbol{x}_i$. Then $\boldsymbol{u}(f(\boldsymbol{u})) = 1$ and $\boldsymbol{x}_i(f(\boldsymbol{u})) = 0$, so

$$\boldsymbol{v}^\intercal(\boldsymbol{u} - \boldsymbol{x}_i) = (c^*)^{f(\boldsymbol{u})} + \sum_{j=1}^{f(\boldsymbol{u})-1} (\boldsymbol{u}(j) - \boldsymbol{x}_i(j))(c^*)^j \geq (c^*)^{f(\boldsymbol{u})} - \sum_{j=1}^{f(\boldsymbol{u})-1} (c^*)^j.$$

Since $c^* > 2$, $(c^*)^{f(\boldsymbol{u})} > \sum_{j=1}^{f(\boldsymbol{u})-1}(c^*)^j$, so

$$\boldsymbol{v}^\intercal(\boldsymbol{u} - \boldsymbol{x}_i) \geq (c^*)^{f(\boldsymbol{u})} - \sum_{j=1}^{f(\boldsymbol{u})-1} (c^*)^j > 0.$$

Hence, $\boldsymbol{u}$ is also $\boldsymbol{v}$-improving, which completes the proof of Claim 1.

Proof of Claim 2: By the result of Claim 4, it suffices to bound κ_b from below and $|\kappa_j|$ from above to achieve our desired inequality. We already showed that $\boldsymbol{u}^1(f(\boldsymbol{u}^1)) = 1$ and $\boldsymbol{x}_1(f(\boldsymbol{u}^1)) = 0$. Furthermore, $\boldsymbol{u}^0(f(\boldsymbol{u}^1)) = 0$ as well by our assumption that $f(\boldsymbol{u}^1) > f(\boldsymbol{u}^0)$ and by our definition of f. Since $b = f(\boldsymbol{u}^1)$, we have that

$$\begin{aligned}\kappa_b &= \boldsymbol{c}^\intercal(\boldsymbol{u}^0 - \boldsymbol{x}_i)(\boldsymbol{u}^1(b) - \boldsymbol{x}_i(b)) - \boldsymbol{c}^\intercal(\boldsymbol{u}^1 - \boldsymbol{x}_i)(\boldsymbol{u}^0(b) - \boldsymbol{x}_i(b)) \\ &= \boldsymbol{c}^\intercal(\boldsymbol{u}^0 - \boldsymbol{x}_i)(1-0) - \boldsymbol{c}^\intercal(\boldsymbol{u}^1 - \boldsymbol{x}_i)(0-0) \\ &= \boldsymbol{c}^\intercal(\boldsymbol{u}^0 - \boldsymbol{x}_i).\end{aligned}$$

Since $\boldsymbol{u}^0$ is $\boldsymbol{c}$-improving, $\kappa_b = \boldsymbol{c}^\intercal(\boldsymbol{u}^0 - \boldsymbol{x}_i) > 0$. Furthermore, $\boldsymbol{c}, \boldsymbol{u}^0$, and $\boldsymbol{x}_i$ are all integer vectors, so $\kappa_b = \boldsymbol{c}^\intercal(\boldsymbol{u}^0 - \boldsymbol{x}_i) \geq 1$. Hence, $\kappa_b(c^*)^b \geq (c^*)^b$.

For the other side of the inequality, we need to bound the sizes of lower order coefficients $|\kappa_j|$ for $j \leq b-1$. To do this, we need to split into cases. Suppose first that $\boldsymbol{x}_i(j) = 0$. Then

$$\begin{aligned}\kappa_j &= \boldsymbol{c}^\intercal(\boldsymbol{u}^0 - \boldsymbol{x}_i)(\boldsymbol{u}^1(j) - \boldsymbol{x}_i(j)) - \boldsymbol{c}^\intercal(\boldsymbol{u}^1 - \boldsymbol{x}_i)(\boldsymbol{u}^0(j) - \boldsymbol{x}_i(j)) \\ &= \boldsymbol{c}^\intercal(\boldsymbol{u}^0 - \boldsymbol{x}_i)\boldsymbol{u}^1(j) - \boldsymbol{c}^\intercal(\boldsymbol{u}^1 - \boldsymbol{x}_i)\boldsymbol{u}^0(j).\end{aligned}$$

Since $\boldsymbol{u}^0(j), \boldsymbol{u}^1(j) \in \{0,1\}$, $-\boldsymbol{c}^\intercal(\boldsymbol{u}^1 - \boldsymbol{x}_i) \leq \kappa_j \leq \boldsymbol{c}^\intercal(\boldsymbol{u}^0 - \boldsymbol{x}_i)$. If $\boldsymbol{x}_i(j) = 1$, we find by similar reasoning that $-\boldsymbol{c}^\intercal(\boldsymbol{u}^0 - \boldsymbol{x}_i) \leq \kappa_j \leq \boldsymbol{c}^\intercal(\boldsymbol{u}^1 - \boldsymbol{x}_i)$. Hence, we always have that $|\kappa_j| \leq \max\{\boldsymbol{c}^\intercal(\boldsymbol{u}^0 - \boldsymbol{x}_i), \boldsymbol{c}^\intercal(\boldsymbol{u}^1 - \boldsymbol{x}_i)\}$. Because $\boldsymbol{u}^0, \boldsymbol{u}^1$,and $\boldsymbol{x}_i$ are vertices of P, they are in $\{0,1\}^n$ meaning that $\boldsymbol{u}^0 - \boldsymbol{x}_i, \boldsymbol{u}^1 - \boldsymbol{x}_i \in \{-1,0,1\}^n$. It follows that

$$|\kappa_j| \leq \max\{\, \boldsymbol{c}^\intercal(\boldsymbol{u}^0 - \boldsymbol{x}_i), \boldsymbol{c}^\intercal(\boldsymbol{u}^1 - \boldsymbol{x}_i)\,\} \leq \max_{\boldsymbol{y} \in \{-1,0,1\}^n} \boldsymbol{c}^\intercal \boldsymbol{y} = ||\boldsymbol{c}||_1.$$

Thus, we have that

$$\sum_{j=1}^{b-1} -\kappa_i (c^*)^j \leq ||\boldsymbol{c}||_1 \sum_{j=1}^{b-1} (c^*)^j.$$

To finish the proof of Claim 2, it suffices to show that $||\boldsymbol{c}||_1 \sum_{j=1}^{b-1}(c^*)^j < (c^*)^b$. From a typical geometric series estimate and our choice of c^*,

$$||\boldsymbol{c}||_1 \sum_{j=1}^{b-1} (c^*)^j = ||\boldsymbol{c}||_1 \frac{(c^*)^b - c^*}{c^* - 1} = \frac{||\boldsymbol{c}||_1}{c^* - 1}((c^*)^b - c^*) < \frac{||\boldsymbol{c}||_1}{||\boldsymbol{c}||_1}((c^*)^b - c^*) < (c^*)^b.$$

Thus, Claim 2 is true.

Proof of Claim 3: We again use the equivalent characterization shown in Claim 4. Note that we have the same upper bounds for κ_i as from the proof of Claim 2 for all $i \leq b-1$ meaning that

$$\sum_{j=1}^{b-1} -\kappa_j (c^*)^j \leq ||\boldsymbol{c}||_1 \sum_{j=1}^{b-1} (c^*)^j.$$

However, for κ_b, the situation is different. Namely, we now have $\boldsymbol{u}^0(b) = \boldsymbol{u}^1(b) = 1$, while $\boldsymbol{x}_i(b) = 0$. Thus,

$$\begin{aligned}\kappa_b &= \boldsymbol{c}^\intercal(\boldsymbol{u}^0 - \boldsymbol{x}_i)(\boldsymbol{u}^1(b) - \boldsymbol{x}_i(b)) - \boldsymbol{c}^\intercal(\boldsymbol{u}^1 - \boldsymbol{x}_i)(\boldsymbol{u}^0(b) - \boldsymbol{x}_i(b)) \\ &= \boldsymbol{c}^\intercal(\boldsymbol{u}^0 - \boldsymbol{x}_i) - \boldsymbol{c}^\intercal(\boldsymbol{u}^1 - \boldsymbol{x}_i) \\ &= \boldsymbol{c}^\intercal(\boldsymbol{u}^0 - \boldsymbol{u}^1).\end{aligned}$$

By assumption, $\boldsymbol{c}^\intercal(\boldsymbol{u}^0) > \boldsymbol{c}^\intercal(\boldsymbol{u}^1)$, so $\boldsymbol{c}^\intercal(\boldsymbol{u}^0 - \boldsymbol{u}^1) > 0$. By the same reasoning as before, since $\boldsymbol{c}, \boldsymbol{u}^0$, and $\boldsymbol{u}^1$ are integer vectors, we must then have $\kappa_b = \boldsymbol{c}^\intercal(\boldsymbol{u}^0 - \boldsymbol{u}^1) \geq 1$. Then we again have $\kappa_b(c^*)^b \geq (c^*)^b$, so by the same argument as in the proof of Claim 2,

$$\sum_{j=1}^{b-1} -\kappa_j(c^*)^j < \kappa_b(c^*)^b.$$

This completes the proof.

□

Chapter 5

Extending Edge Rules to Pivot Rules

In this chapter, we will build on the results in Chapter 4 to develop pivot rules for the Simplex method on 0/1-LPs. The objective for these pivot rules is to design them to generate the same paths as their corresponding edge rules. We remind the reader that since in this chapter we will only consider 0/1-LPs, we will always assume that an LP's inequality constraints include the non-negativity constraints $-\boldsymbol{x} \leq \mathbf{0}$.

5.1 The True Steepest Edge Pivot Rule

Definition 13. *Given a 0/1-LP (P) of the form (1.1) with feasible region P, let (P') be the LP obtained by putting (P) into standard equality form, let B be the current basis of (P'), and let $\boldsymbol{x}$ be the vertex of P associated to that basis. Let $\boldsymbol{v} = \mathbf{1} - 2\boldsymbol{x}$. The True Steepest Edge pivot rule selects a pivot direction as follows:*

- *If any generator $\boldsymbol{z}^j$ of an extreme ray of $\mathcal{C}(B)$ satisfies $\boldsymbol{v}^\intercal \boldsymbol{z}^j \leq 0$ and $\boldsymbol{c}^\intercal \boldsymbol{z}^j > 0$, then it selects $\boldsymbol{z}^j$;*
- *Otherwise, it selects the generator $\boldsymbol{z}^j$ that maximizes $\frac{\boldsymbol{c}^\intercal \boldsymbol{z}^j}{\boldsymbol{v}^\intercal \boldsymbol{z}^j}$.*

We show that if the True Steepest Edge pivot rule performs a non-degenerate pivot, $\boldsymbol{z}^j$ is the *steepest edge-direction* at $\boldsymbol{x}$ in P.

Lemma 16. *Let $\boldsymbol{x}$ be a vertex solution of a 0/1-LP of the form (1.1) and let $\boldsymbol{x}' = \boldsymbol{x} + \alpha \boldsymbol{z}^j$ be the vertex solution obtained from $\boldsymbol{x}$ after moving maximally along the direction $\boldsymbol{z}^j$ selected* *according to the True Steepest Edge pivot rule. If $\boldsymbol{x}' \neq \boldsymbol{x}$, then $\boldsymbol{z}^j$ corresponds to the steepest edge-direction at $\boldsymbol{x}$ in P.*

Proof. As in Section 4.1, consider the vector $\boldsymbol{v} = \mathbf{1} - 2\boldsymbol{x}$. We have that $\boldsymbol{v}$ is contained in the same orthant O_x of $\mathbb{R}^n$ as the cone $\mathcal{C}(\boldsymbol{x})$, where O_x is equal to the set of all $\boldsymbol{z} \in \mathbb{R}^n$ satisfying the inequalities

$$\boldsymbol{z}(i) \geq 0 \text{ for all } i \text{ such that } \boldsymbol{x}(i) = 0,$$
$$\boldsymbol{z}(i) \leq 0 \text{ for all } i \text{ such that } \boldsymbol{x}(i) = 1.$$

Then by the definition of the 1-norm, for all vectors $\boldsymbol{z}$ in the orthant O_x, $||\boldsymbol{z}||_1$ is precisely equal to $\boldsymbol{v}^\intercal \boldsymbol{z}$. In particular, this holds for all vectors in $\mathcal{C}(\boldsymbol{x})$, and so all rays in $\mathcal{C}(\boldsymbol{x})$ intersect the hyperplane H defined by $H = \{\boldsymbol{z} : \boldsymbol{v}^\intercal \boldsymbol{z} = 1\}$.

Let the d extreme rays of the basic cone $\mathcal{C}(B)$ be generated by $\boldsymbol{z}^1, \ldots, \boldsymbol{z}^d$. If the extreme ray generated by $\boldsymbol{z}^i$ intersects H, then assume without loss of generality (by possibly rescaling) that $\boldsymbol{z}^i$ is in H. Note then that for any generator $\boldsymbol{z}^i$ of an extreme ray of $\mathcal{C}(B)$, if $\boldsymbol{z}^i$ happens to also correspond to an edge-direction at $\boldsymbol{x}$ in P, we have that $\boldsymbol{z}^i \in H$.

Assume without loss of generality that the direction chosen according to the True Steepest Edge pivot rule is $\boldsymbol{z}^1$. Since $\boldsymbol{x}' \neq \boldsymbol{x}$, the direction $\boldsymbol{z}^1$ also corresponds to an edge-direction at $\boldsymbol{x}$ in P. Then $\boldsymbol{z}^1$ is in the feasible cone, and by our earlier discussion, $\boldsymbol{v}^\intercal \boldsymbol{z}^1 = ||\boldsymbol{z}^1||_1 = 1$. Now, consider the optimization problem $\mathcal{Q}$ defined by

$$\begin{aligned}
&\max \boldsymbol{c}^\intercal \boldsymbol{z} \\
&\text{s.t.} \\
&\boldsymbol{v}^\intercal \boldsymbol{z} \leq 1 && (1) \\
&\boldsymbol{z} \in \mathcal{C}(B), && (2)
\end{aligned}$$

and let $P_{\mathcal{Q}}$ denote its feasible region. Note that $P_{\mathcal{Q}}$ is a polyhedron. By construction, any generator $\boldsymbol{z}^i \in H$ is a vertex of $P_{\mathcal{Q}}$. In particular, this is true for those generators $\boldsymbol{z}^i$ which also correspond to edge-directions at $\boldsymbol{x}$ in P.

We will argue that the selected direction $\boldsymbol{z}^1$ is an optimal solution to the LP $\mathcal{Q}$. Clearly, it is feasible for $\mathcal{Q}$. First, suppose for the sake of contradiction that $\mathcal{Q}$ is unbounded. Then there exists an entire ray of $\mathcal{C}(B)$ which is contained in $P_{\mathcal{Q}}$ on which the objective function $\boldsymbol{c}$ is unbounded, and therefore there exists such a ray that is an extreme ray of $\mathcal{C}(B)$. Let this extreme ray be generated by $\boldsymbol{z}^i$.

Then $\boldsymbol{z}^i$ does not correspond to an edge-direction at $\boldsymbol{x}$ in P, and therefore corresponds to a degenerate pivot at B. Furthermore, $\boldsymbol{v}^\intercal \boldsymbol{z}^i \leq 0$, and $\boldsymbol{c}^\intercal \boldsymbol{z}^i > 0$. However, this implies that the pivot rule would have chosen the direction $\boldsymbol{z}^i$ and not $\boldsymbol{z}^1$, a contradiction. Thus, the LP $\mathcal{Q}$ is not unbounded. This implies that all generators $\boldsymbol{z}^i$ of extreme rays of $\mathcal{C}(B)$ satisfying $\boldsymbol{c}^\intercal \boldsymbol{z}^i > 0$ generate extreme rays that intersect H.

There exists an optimal vertex $\boldsymbol{y}$ of $P_{\mathcal{Q}}$, and since $\boldsymbol{c}^\intercal \boldsymbol{z}^1 > 0$, the optimal vertex is not $\mathbf{0}$. Thus, $\boldsymbol{y}$ is precisely one of the generators that lies in H. Let these generators be $\boldsymbol{z}^1, \ldots, \boldsymbol{z}^k$. By the fact that $\boldsymbol{z}^1$ was selected, we have that $\boldsymbol{z}^1$ maximizes $\boldsymbol{c}^\intercal \boldsymbol{z}$ overall $\boldsymbol{z} \in \{ \boldsymbol{z}^1, \ldots, \boldsymbol{z}^k \}$, and so $\boldsymbol{z}^1$ is an optimal solution to $\mathcal{Q}$, as desired.

We will now show that $\boldsymbol{z}^1$ corresponds to a steepest edge-direction at $\boldsymbol{x}$ in P. Let $\boldsymbol{z}'$ be any edge-direction at $\boldsymbol{x}$, and without loss of generality assume $||\boldsymbol{z}'||_1 = 1$. Then $\boldsymbol{z}'$ is a feasible solution to $\mathcal{Q}$, as $\boldsymbol{z}' \in \mathcal{C}(\boldsymbol{x}) \subseteq \mathcal{C}(B)$. Since $\boldsymbol{z}^1$ is an optimal solution to $\mathcal{Q}$, we have that

$$\frac{\boldsymbol{c}^\intercal \boldsymbol{z}^1}{\|\boldsymbol{z}^1\|_1} = \boldsymbol{c}^\intercal \boldsymbol{z}^1 \geq \boldsymbol{c}^\intercal \boldsymbol{z}' = \frac{\boldsymbol{c}^\intercal \boldsymbol{z}'}{\|\boldsymbol{z}'\|_1},$$

as desired. □

The above result shows that our pivot rule guarantees the following: whenever we perform a non-degenerate pivot, this always corresponds to moving along a steepest edge-direction at the corresponding vertex solution of the original LP. Together Lemma 17 and Theorem 15 provide a proof of Theorem 2:

Theorem 2. *On any 0/1-LP of the form (1.1), the Simplex method with a True Steepest Edge pivot rule reaches an optimal solution by performing a strongly polynomial number of non-degenerate pivots. Furthermore, it generates the same monotone path as the Steepest Edge rule.*

We now provide an example showing that the standard Steepest Edge pivot rule (using the 1-norm) does not follow the same path as the True Steepest Edge pivot rule. In particular, it is possible for the Steepest Edge pivot rule to perform a non-degenerate pivot at a vertex solution $\boldsymbol{x}$ where the edge corresponding to that pivot is not a steepest edge-direction at $\boldsymbol{x}$. First, we recall the definition of the Steepest Edge pivot rule in terms of the extreme ray of $\mathcal{C}(B)$.

Definition 14. *Given an LP (P) of the form (1.1) with feasible region P, let (P') be the LP obtained by putting (P) into standard equality form, let B be the current basis of (P'), and let $\boldsymbol{x}$ be the vertex of P associated to B. The Steepest Edge pivot rule selects as a pivot direction the generator of $\mathcal{C}(B)$ that maximizes $\frac{\boldsymbol{c}^\intercal \boldsymbol{z}}{\|\boldsymbol{z}\|_1}$ among all generators $\boldsymbol{z}$ of $\mathcal{C}(B)$.*

Now, consider the 3-dimensional 0/1 polytope given by the convex hull of the points $(0,0,0)$, $(0,1,0)$, $(1,0,0)$, $(1,1,0)$, and $(0,0,1)$. This is a 0/1 pyramid over a square. It can be easily checked that this polytope can be minimally described by the following inequalities:

$$\boldsymbol{x}(1) + \boldsymbol{x}(3) \leq 1, \quad \boldsymbol{x}(2) + \boldsymbol{x}(3) \leq 1, \quad \boldsymbol{x}(1) \geq 0, \quad \boldsymbol{x}(2) \geq 0, \quad \boldsymbol{x}(3) \geq 0.$$

Furthermore, it is clear that at the point $\boldsymbol{x}' = (0,0,1)$, all but the last inequality are tight. That is, these are the inequalities yielding the feasible cone $\mathcal{C}(\boldsymbol{x}')$. The edge-directions at $\boldsymbol{x}'$ are given by $(0,0,-1)$, $(0,1,-1)$, $(1,0,-1)$, and $(1,1,-1)$. Given the objective function $\boldsymbol{c} = (50,-1,0)$ to be maximized, it can be checked that the steepest edge-direction at $\boldsymbol{x}'$ is $\boldsymbol{z}^* = (1,0,-1)$, the edge-direction that moves to the optimal solution $(1,0,0)$. Note that $\frac{\boldsymbol{c}^\intercal \boldsymbol{z}^*}{\|\boldsymbol{z}^*\|_1} = 25$. The only other edge-direction at $\boldsymbol{x}'$ satisfying $\boldsymbol{c}^\intercal \boldsymbol{z} > 0$ is the direction $\boldsymbol{z}' = (1,1,-1)$ which moves to the point $(1,1,0)$. Note that $\frac{\boldsymbol{c}^\intercal \boldsymbol{z}'}{\|\boldsymbol{z}'\|_1} = \frac{49}{3}$, and so $\boldsymbol{z}'$ is not a steepest edge-direction at $\boldsymbol{x}'$

Now, consider a basis B associated with $\boldsymbol{x}'$ whose corresponding basic cone is given by the inequalities $\boldsymbol{x}(1) \geq 0$, $\boldsymbol{x}(1) + \boldsymbol{x}(3) \leq 1$, and $\boldsymbol{x}(2) + \boldsymbol{x}(3) \leq 1$. It can be checked that the edge-direction $\boldsymbol{z}^*$ does not generate an extreme ray of $\mathcal{C}(B)$, but the edge-direction $\boldsymbol{z}'$ does. The pivot directions at B satisfying $\boldsymbol{c}^\intercal \boldsymbol{z} > 0$ are the direction given by $\boldsymbol{z}'$ and $(0,-1,0)$. Therefore, $\boldsymbol{z}'$ is the pivot direction chosen by the Steepest Edge pivot rule. This gives a non-degenerate pivot, but as observed above, it does not correspond to a steepest edge-direction at $\boldsymbol{x}'$.

5.1.1 Implementation of the True Steepest Edge Pivot Rule

We now briefly describe the implementation of the True Steepest Edge pivot rule. Recall from Section 2.1 that in the context of the Simplex algorithm, we first put our original LP into equality form by adding slack variables.

Suppose we have some feasible basis B. For each $j \in N$, let $\boldsymbol{z}'^j$ be defined by $\boldsymbol{z}'^j(B) = A'^{-1}_{[B]} A'_{[j]}$, $\boldsymbol{z}'^j = 1$, and $\boldsymbol{z}'^j(i) = 0$ for all $i \in N \setminus \{ j \}$. These vectors give the possible pivot directions. Let $\boldsymbol{v}'$ be defined by

$$\boldsymbol{v}'(i) := \begin{cases} 1 & \text{if } \boldsymbol{x}(i) = 0 \text{ and } \boldsymbol{x}(i) \text{ is an original variable,} \\ -1 & \text{if } \boldsymbol{x}(i) = 1 \text{ and } \boldsymbol{x}(i) \text{ is an original variable,} \\ 0 & \text{otherwise.} \end{cases}$$

That is, $\boldsymbol{v}'$ is the extension $\boldsymbol{v}$ (as defined earlier in Section 5.1) to the slack variables obtained by padding it with 0's. First, if there exists $j \in N$ such that $\bar{\boldsymbol{c}}'(j) > 0$ and $\boldsymbol{v}'^{\intercal}\boldsymbol{z}'^{j} \leq 0$, we choose j to be the entering column (and we choose such a j arbitrarily if more than one element of N satisfies this condition). Otherwise, we choose

$$j = \text{argmax}_{i \in N:\, \bar{\boldsymbol{c}}'(i)>0} \left(\frac{\bar{\boldsymbol{c}}'(i)}{\boldsymbol{v}'^{\intercal}\boldsymbol{z}'^{i}} \right). \tag{5.1}$$

Again, if more than one element of N satisfies this condition we can we choose such a j arbitrarily.

It may be that the choice of ℓ in Step 4 of our earlier description of Simplex is unique. However, we may encounter degeneracy, in which case it may not be. In that situation we use a *lexicographic pivot rule* to choose the leaving variable. This is a well-known way to break ties and avoid cycling [72, 73, 91].

Note that during the algorithm we carry the extra "auxiliary cost" vector $\boldsymbol{v}'^{\intercal}$. We can think of $\boldsymbol{v}'^{\intercal}$ as an additional row of the tableau that needs to be updated. The vector $\boldsymbol{v}'^{\intercal}$ is always zero on the slack variables added and it has only $+1, -1$ entries for the indices of the original variables. After each pivot, we can easily update the entries of $\boldsymbol{v}'^{\intercal}$. The *original* variables must take 0 or 1 values because they are vertices of a 0/1 polyhedron. If after a pivot, an original variable goes from being 0-valued to being 1-valued, then we change the entry value in $\boldsymbol{v}'^{\intercal}$ from a 1 to a -1. The opposite switch occurs when we change one of the original variables from being 1-valued to 0-valued.

Finally, over the years there have been improvements on the implementations of the Simplex method. It is well-known that a lot of the steps can be performed faster by relying on sparsity of matrices and some numerical tricks, such as LU factorization, but we refer the reader to Chapter 8 of [93] for details. One special detail is that we use the 1-norm to measure how steep the edge is. In our algorithm each iteration requires knowledge of the norms $\|A_B^{-1}A_h\|_1$. It is worth remarking that while for the most common 2-norm Steepest Edge pivot rule Forrest and Goldfarb [50] showed how to update these vector in a fast way, here we do not offer a speedup.

5.2 Asymmetric Steepest Descent Pivot Rule

In this section, we extend the ASD rule to a pivot rule for the Simplex method which generates the same path.

Definition 15. *Given a 0/1-LP (P) in the standard equality form (2.2) with feasible region P, let B be the current basis of (P), and let* $\boldsymbol{x}$ *be the vertex of P associated to that basis. Let* $\boldsymbol{v} = -\boldsymbol{x}$*. The ASD pivot rule selects a pivot direction as follows:*

- *If any generator* $\boldsymbol{z}^j$ *of an extreme ray of* $\mathcal{C}(B)$ *satisfies* $\boldsymbol{v}^\intercal \boldsymbol{z}^j \leq 0$ *and* $\mathbf{c}^\intercal \boldsymbol{z}^j > 0$*, then it selects* $\boldsymbol{z}^j$*;*
- *Otherwise, it selects the generator* $\boldsymbol{z}^j$ *that maximizes* $\frac{\mathbf{c}^\intercal \boldsymbol{z}^j}{\boldsymbol{v}^\intercal \boldsymbol{z}^j}$*.*

We show that the ASD pivot rule generates the same path as the ASD edge rule. The proof of this lemma closely follows the proof of Lemma 16.

Lemma 17. *Let* $\boldsymbol{x}$ *be an extreme point solution of a 0/1-LP of the form (2.2) and let* $\boldsymbol{x}' = \boldsymbol{x} + \alpha \boldsymbol{z}^j$ *be the extreme point solution obtained from* $\boldsymbol{x}$ *after moving maximally along the direction* $\boldsymbol{z}^j$ *selected according to the ASD pivot rule. If* $\boldsymbol{x}' \neq \boldsymbol{x}$*, then* $\boldsymbol{z}^j$ *corresponds to edge direction chosen by the ASD edge rule at* $\boldsymbol{x}$*.*

Proof. Let H be the hyperplane defined by $H = \{\, \boldsymbol{z} : \boldsymbol{v}^\intercal \boldsymbol{z} = 1 \,\}$. As observed in the proof of Lemma 10, the set

$$\{\, \boldsymbol{z} \in \mathcal{C}(\boldsymbol{x}) : \boldsymbol{v}^\intercal \boldsymbol{z} \leq 1 \,\}$$

is a bounded set. That is, all rays in $\mathcal{C}(\boldsymbol{x})$ intersect H. Let the d extreme rays of the basic cone $\mathcal{C}(B)$ be generated by $\boldsymbol{z}^1, \ldots, \boldsymbol{z}^d$. If the extreme ray generated by $\boldsymbol{z}^i$ intersects H, then assume without loss of generality (by possibly rescaling) that $\boldsymbol{z}^i$ is in H. Note then that for any generator $\boldsymbol{z}^i$ of an extreme ray of $\mathcal{C}(B)$, if $\boldsymbol{z}^i$ happens to also correspond to an edge-direction at $\boldsymbol{x}$ in P, we have that $\boldsymbol{z}^i \in H$.

Assume without loss of generality that the direction chosen according to the ASD pivot rule is $\boldsymbol{z}^1$. Since $\boldsymbol{x}' \neq \boldsymbol{x}$, the direction $\boldsymbol{z}^1$ also corresponds to an edge-direction at $\boldsymbol{x}$ in P. Then $\boldsymbol{z}^1$ is in the feasible cone, and thus $\boldsymbol{v}^\intercal \boldsymbol{z}^1 = 1$. Now, consider the optimization problem $\mathcal{Q}$ defined by

$$\begin{aligned} & \max \boldsymbol{c}^\intercal \boldsymbol{z} \\ & \text{s.t.} \\ & \boldsymbol{v}^\intercal \boldsymbol{z} 1 && (1) \\ & \boldsymbol{z} \in \mathcal{C}(B), && (2) \end{aligned}$$

and let $P_\mathcal{Q}$ denote its feasible region. Note that $P_\mathcal{Q}$ is a polyhedron. By construction, any generator $\boldsymbol{z}^i \in H$ is a vertex of $P_\mathcal{Q}$. In particular, this is true for those generators $\boldsymbol{z}^i$ which also correspond to edge-directions at $\boldsymbol{x}$ in P.

We will argue that the selected direction $\boldsymbol{z}^1$ is an optimal solution to the LP $\mathcal{Q}$. Clearly, it is feasible for $\mathcal{Q}$. First, suppose for the sake of contradiction that $\mathcal{Q}$ is unbounded. Then there exists an entire ray of $\mathcal{C}(B)$ which is contained in $P_{\mathcal{Q}}$ on which the objective function $\boldsymbol{c}$ is unbounded, and therefore there exists such a ray that is an extreme ray of $\mathcal{C}(B)$. Let this extreme ray be generated by $\boldsymbol{z}^i$.

Then $\boldsymbol{z}^i$ does not correspond to an edge-direction at $\boldsymbol{x}$ in P, and therefore corresponds to a degenerate pivot at B. Furthermore, $\boldsymbol{v}^\intercal \boldsymbol{z}^i \leq 0$, and $\boldsymbol{c}^\intercal \boldsymbol{z}^i > 0$. However, this implies that the pivot rule would have chosen the direction $\boldsymbol{z}^i$ and not $\boldsymbol{z}^1$, a contradiction. Thus, the LP $\mathcal{Q}$ is not unbounded. This implies that all generators $\boldsymbol{z}^i$ of extreme rays of $\mathcal{C}(B)$ satisfying $\boldsymbol{c}^\intercal \boldsymbol{z}^i > 0$ generate extreme rays that intersect H.

There exists an optimal vertex $\boldsymbol{y}$ of $P_{\mathcal{Q}}$, and since $\boldsymbol{c}^\intercal \boldsymbol{z}^1 > 0$, the optimal vertex is not $\mathbf{0}$. Thus, $\boldsymbol{y}$ is precisely one of the generators that lies in H. Let these generators be $\boldsymbol{z}^1, \ldots, \boldsymbol{z}^k$. By the fact that $\boldsymbol{z}^1$ was selected, we have that $\boldsymbol{z}^1$ maximizes $\boldsymbol{c}^\intercal \boldsymbol{z}$ overall $\boldsymbol{z} \in \{ \boldsymbol{z}^1, \ldots, \boldsymbol{z}^k \}$, and so $\boldsymbol{z}^1$ is an optimal solution to $\mathcal{Q}$. Then by definition, $\boldsymbol{z}^1$ is also the edge-direction chosen by the ASD edge rule, as desired. □

This implies the following theorem:

Theorem 20. *On any* 0/1*-LP of the form (2.2), the Simplex method with an ASD pivot rule reaches an optimal solution by performing a strongly polynomial number of non-degenerate pivots. Furthermore, it generates the same monotone path as the ASD rule.*

Furthermore, the ASD pivot rule can be implemented in tableaux form in the same way as described in Subsection 5.1.1 by instead using the definition of $\boldsymbol{v}$ used here.

5.3 Shadow Vertex Pivot Rules

In this section, we explain how to convert any Shadow edge rule to a corresponding pivot rule that follows the same path.

Let us start with describing the precise question that we want to address in this section. We are given a general LP (not necessarily a 0/1-LP) with feasible region Q, and a feasible basis B for the corresponding LP in standard equality form. Let $\boldsymbol{x}$ be the extreme point solution of Q to which B is associated. Recall that a Shadow rule chooses an improving edge direction $\boldsymbol{z}$ in the feasible cone $\mathcal{C}(\boldsymbol{x})$ that maximizes $\frac{\boldsymbol{c}^\intercal \boldsymbol{z}}{\boldsymbol{v}^\intercal \boldsymbol{z}}$ for a chosen vector $\boldsymbol{v}$. However, if we have degenerate bases, the extreme directions of the basic cone $\mathcal{C}(B)$ associated with the basis B may not coincide with the extreme directions of the feasible cone $\mathcal{C}(\boldsymbol{x})$. Hence, we have the following question: How should we select an improving direction in $\mathcal{C}(B)$ in order to guarantee that we are following the same path on the 1-skeleton of P traced by the Shadow rule?

Our first observation is that if the initial basis B_0 satisfies a small assumption, then it is enough to consider the direction that maximizes the slope among the ones in the basic cone $\mathcal{C}(B_0)$. We first prove this, and later show that it is always possible to find a basis that satisfies the needed assumption via a sequence of degenerate pivots. Formally:

Theorem 21. *Given an LP with feasible region Q, let B^0 be an initial feasible basis for the corresponding LP in standard equality form. Let $\boldsymbol{v}$ be defined according to a given Shadow rule, and assume that B^0 satisfies the following: for each $\boldsymbol{z} \in \mathcal{C}(B^0)$, if $\boldsymbol{c}^\intercal \boldsymbol{z} > 0$ then $\boldsymbol{v}^\intercal \boldsymbol{z} > 0$.*

Then by starting from B^0 and by selecting at each basis any improving pivot direction $\boldsymbol{z} \in \mathcal{C}(B)$ that maximizes $\frac{\boldsymbol{c}^\intercal \boldsymbol{z}}{\boldsymbol{v}^\intercal \boldsymbol{z}}$, the Simplex method follows the same path on the 1-skeleton of Q as the one followed by the Shadow rule.

Proof. Suppose that after a number of Simplex iterations, we have reached a basis B which is not optimal, and assume that every basis B' visited until B satisfies the property that for each $\boldsymbol{z} \in \mathcal{C}(B')$, if $\boldsymbol{c}^\intercal \boldsymbol{z} > 0$ then $\boldsymbol{v}^\intercal \boldsymbol{z} > 0$. Note that this is true by assumption at the initial basis B^0. We claim that B satisfies the property as well.

Suppose that at the current iteration the Shadow rule would move along an edge-direction $\boldsymbol{y} \in \mathcal{C}(B)$. Note that since B is not an optimal basis with respect to $\boldsymbol{c}$, and since the path has not yet reached an optimum with respect to $\boldsymbol{c}$, Lemma 11 implies that it is also not optimal with respect to $\boldsymbol{v}$. Then Lemma 11 further implies that $\boldsymbol{v}^\intercal \boldsymbol{y} > 0$, and we may assume without loss of generality that $\boldsymbol{v}^\intercal \boldsymbol{y} = 1$. Let B' be the basis visited right before B, and $\boldsymbol{z}'$ be the pivot direction chosen at B' by our procedure. We know that $\boldsymbol{v}^\intercal \boldsymbol{z}' > 0$ and $\boldsymbol{c}^\intercal \boldsymbol{z}' > 0$. Then we have that the basic cone $\mathcal{C}(B)$ contains the directions $-\boldsymbol{z}'$ and $\boldsymbol{y}$. This implies that the projection $\pi(\mathcal{C}(B))$ contains an element whose pre-image satisfies $\boldsymbol{v}^\intercal \boldsymbol{z} < 0$ and $\boldsymbol{c}^\intercal \boldsymbol{z} < 0$ (namely, $\pi(-\boldsymbol{z}')$), and a direction satisfying $\boldsymbol{v}^\intercal \boldsymbol{z} > 0$ and $\boldsymbol{c}^\intercal \boldsymbol{z} > 0$ (namely, $\pi(\boldsymbol{y})$). Recall that, under the projection π, $\boldsymbol{v}$ and $\boldsymbol{c}$ act as coordinate vectors for the first and second coordinate of space, respectively. Then by convexity, the cone $\pi(\mathcal{C}(B))$ does not contain any element of the orthant of $\mathbb{R}^2$ defined by $\boldsymbol{v}^\intercal \boldsymbol{z} \leq 0$, $\boldsymbol{c}^\intercal \boldsymbol{z} \geq 0$. Therefore, by the definition of π, $\mathcal{C}(B)$ does not contain any element satisfying both $\boldsymbol{v}^\intercal \boldsymbol{z} \leq 0$ and $\boldsymbol{c}^\intercal \boldsymbol{z} \geq 0$. That is, for all elements $\boldsymbol{z} \in \mathcal{C}(B)$, if $\boldsymbol{c}^\intercal \boldsymbol{z} > 0$, then $\boldsymbol{v}^\intercal \boldsymbol{z} > 0$. This proves our claim.

Assume that our procedure selects $\tilde{\boldsymbol{z}}$ as pivot direction for our basis B. Since $\boldsymbol{v}^\intercal \tilde{\boldsymbol{z}} > 0$, we can assume without loss of generality that $\boldsymbol{v}^\intercal \tilde{\boldsymbol{z}} = 1$. We will show that $\tilde{\boldsymbol{z}}$ is an optimal solution to the following LP we call $\mathcal{Q}$:

$$\begin{aligned}&\max \boldsymbol{c}^\intercal \boldsymbol{z}\\ &\text{s.t.}\\ &\boldsymbol{v}^\intercal \boldsymbol{z} = 1\\ &\boldsymbol{z} \in \mathcal{C}(B).\end{aligned}$$

We first observe that $\mathcal{Q}$ is not unbounded. Since $\boldsymbol{v}^\intercal \boldsymbol{z} > 0$ for all $\boldsymbol{z} \in \mathcal{C}(B)$ satisfying $\boldsymbol{c}^\intercal \boldsymbol{z} > 0$, we have that the set $Z = \{ \boldsymbol{z} \in \mathcal{C}(B) : \boldsymbol{v}^\intercal \boldsymbol{z} \leq 1, \, \boldsymbol{c}^\intercal \boldsymbol{z} > 0 \}$ is a bounded set. The set of feasible solutions to $\mathcal{Q}$ with positive objective value is contained in the set Z, and so it is also a bounded set. Therefore, $\mathcal{Q}$ is not unbounded.

Since the feasible region of $\mathcal{Q}$ is just the basic cone at B intersected with a single hyperplane (which does not contain the unique vertex $\mathbf{0}$ of $\mathcal{C}(B)$), all extreme point solutions of $\mathcal{Q}$ correspond to generators of extreme rays of $\mathcal{C}(B)$ satisfying $\boldsymbol{v}^\intercal \boldsymbol{z} = 1$.

It follows that the optimal extreme point solution of $\mathcal{Q}$ is an extreme ray generator of $\mathcal{C}(B)$ that maximizes $\frac{\boldsymbol{c}^\intercal \boldsymbol{z}}{\boldsymbol{v}^\intercal \boldsymbol{z}}$. That is, it is the chosen pivot direction, $\tilde{\boldsymbol{z}}$.

Assume now that $\tilde{\boldsymbol{z}}$ is a non-degenerate direction. Since $\boldsymbol{y}$ is the edge-direction chosen by the Shadow rule, and since $\tilde{\boldsymbol{z}}$ in this case is also an edge-direction, it follows from Definition 11 that

$$\boldsymbol{c}^\intercal \boldsymbol{y} = \frac{\boldsymbol{c}^\intercal \boldsymbol{y}}{\boldsymbol{v}^\intercal \boldsymbol{y}} \geq \frac{\boldsymbol{c}^\intercal \tilde{\boldsymbol{z}}}{\boldsymbol{v}^\intercal \tilde{\boldsymbol{z}}} = \boldsymbol{c}^\intercal \tilde{\boldsymbol{z}}.$$

Note that this holds because the term $\left(\frac{\boldsymbol{c}^\intercal (\boldsymbol{u} - \boldsymbol{x}_i)}{\boldsymbol{v}^\intercal (\boldsymbol{u} - \boldsymbol{x}_i)}\right)$ in Definition 11 is invariant under scaling. However, since $\boldsymbol{y}$ is also feasible for $\mathcal{Q}$ and since $\tilde{\boldsymbol{z}}$ is optimal for $\mathcal{Q}$, we have that in fact $\boldsymbol{c}^\intercal \boldsymbol{y} = \boldsymbol{c}^\intercal \tilde{\boldsymbol{z}}$. That is, this non-degenerate pivot corresponds to an edge-direction that the Shadow pivot rule would choose. Thus, the Simplex method with the Shadow pivot rule follows the same path on the 1-skeleton as the Shadow rule, as desired. □

It remains to argue how to find a basis B^0 that satisfies the property needed in our previous theorem.

Lemma 18. *Given an LP with feasible region Q, let B^0 be an initial feasible basis which is not optimal for the corresponding LP in standard equality form. Let $\boldsymbol{v}$ be defined according to a given Shadow rule, and assume that B^0 does not satisfy the following: for each $\boldsymbol{z} \in \mathcal{C}(B^0)$, if $\boldsymbol{c}^\intercal \boldsymbol{z} > 0$ then $\boldsymbol{v}^\intercal \boldsymbol{z} > 0$. Then, there exists a sequence of degenerate pivots that eventually yield a basis B satisfying the above condition.*

Proof. Let $\boldsymbol{x}_0$ be the extreme point associated to B^0. Since B^0 is not an optimal basis, there exists a pivot direction $\boldsymbol{z} \in \mathcal{C}(B^0)$ such that $\boldsymbol{c}^\intercal \boldsymbol{z} > 0$, and by hypothesis we can select one such that $\boldsymbol{v}^\intercal \boldsymbol{z} \leq 0$. Since $\pi(\boldsymbol{x}_0)$ lies on the upper path of the shadow of Q, any $\boldsymbol{c}$-increasing, non-degenerate pivot direction $\boldsymbol{z}'$ at any basis corresponding to $\boldsymbol{x}_0$ satisfies $\boldsymbol{v}^\intercal \boldsymbol{z}' > 0$. As such, at any basis corresponding to $\boldsymbol{x}_0$, any $\boldsymbol{c}$-increasing pivot direction satisfying $\boldsymbol{v}^\intercal \boldsymbol{z} \leq 0$ is degenerate pivot direction.

Then suppose that we perform a series of degenerate pivots by choosing pivot directions satisfying $\boldsymbol{c}^\intercal \boldsymbol{z} > 0$ and, if possible, $\boldsymbol{v}^\intercal \boldsymbol{z} \leq 0$. Suppose that we use the lexicographic rule [72, 73, 91] to select the variable leaving the basis. The lexicographic rule ensures that we do not cycle, so we will eventually reach a basis B at which it is not possible to pick a pivot direction satisfying both $\boldsymbol{c}^\intercal \boldsymbol{z} > 0$ and $\boldsymbol{v}^\intercal \boldsymbol{z} \leq 0$, as desired. □

We now turn to the issue of cycling. In [66], Klee and Kleinschmidt provided a method to implement Shadow pivot rules in general without having to worry about degeneracy and cycling. In essence, they showed that for any sufficiently generic choice of objective function $\boldsymbol{c}^\intercal$ and any sufficiently generic choice of auxiliary vector $\boldsymbol{v}$, the Shadow pivot rule with an implementation they provide does not cycle. For implementing the Ordered Shadow and Slim Shadow pivot rules, we may choose $\boldsymbol{c}$ to be sufficiently generic by perturbing the objective function. Furthermore, for obtaining the Ordered Shadow pivot rule, we may perturb $\boldsymbol{v}$ by increasing the value of c^* in Definition 12 by any small $\varepsilon > 0$. Thus, their implementation of Simplex yields the Ordered Shadow pivot rule if we allow an additional step of perturbing $\boldsymbol{v}$. However, for the Slim Shadow rule, such a perturbation would affect our argument for bounding the length. Hence, we may not apply their implementation in that case.

As explained in [72, 73, 91], the lexicographic pivot rule provides a general technique to avoid cycling. When the entering variable is already chosen all we need is to select the leaving variable lexicographically. In particular, this can be integrated with the Shadow pivot rule and we do not need to assume non-degeneracy. The lexicographic rule is exactly what is used in our method for the True Steepest-Edge pivot rule. Namely, one may attach a lexicographic rule for choosing the outgoing variable to prevent cycling. This method applies to our case for both pivot rules, since it makes no assumption about $\boldsymbol{c}$ or $\boldsymbol{v}$. Hence, the Ordered Shadow and Slim Shadow pivot rules can be implemented correctly using the lexicographic method.

However, we close this section by introducing another implementation that is a hybrid between Klee and Kleinschmidt's implementation and lexicographic. Specifically, we now show that cycling can also be avoided if (a) one uses a lexicographic pivot rule to select the leaving variable until the first non-degenerate pivot, and (b) one imposes the assumption that the objective function $\boldsymbol{c}$ is generic. In particular, we require that given any two pivot directions $\boldsymbol{z}'$ and $\boldsymbol{z}''$ chosen by the Shadow pivot rule, $\frac{\boldsymbol{c}^\intercal \boldsymbol{z}'}{\boldsymbol{v}^\intercal \boldsymbol{z}'} \neq \frac{\boldsymbol{c}^\intercal \boldsymbol{z}''}{\boldsymbol{v}^\intercal \boldsymbol{z}''}$. Note that this only needs to hold for pivot directions with $\boldsymbol{v}^\intercal \boldsymbol{z} > 0$ since these are the only pivot directions chosen by the Shadow pivot rule.

If this assumption of genericity does not hold, it can be achieved by, as usual, randomly perturbing $\boldsymbol{c}$ by a small amount. That is, by replacing $\boldsymbol{c}$ by a new objective function $\boldsymbol{c}'$ chosen uniformly at random from the ε-ball centered at $\boldsymbol{c}$ (for a sufficiently small choice of ε). Note that there are finitely many possible pivot directions (when normalized so that they satisfy $\boldsymbol{v}^\intercal \boldsymbol{z} = 1$). If $\boldsymbol{c}^\intercal \boldsymbol{z}' = \boldsymbol{c}^\intercal \boldsymbol{z}''$, then $\boldsymbol{c}^\intercal(\boldsymbol{z}' - \boldsymbol{z}'') = 0$. As there are only finitely many possible pivot directions of the above form, there are finitely many possible vectors of the form $\boldsymbol{w} = (\boldsymbol{z}' - \boldsymbol{z}'')$ for distinct pivot directions $\boldsymbol{z}'$ and $\boldsymbol{z}''$. Then these vectors give a finite collection of $(n-1)$-dimensional linear spaces, each defined by $\boldsymbol{w}^\intercal \boldsymbol{x} = 0$ for a choice of $\boldsymbol{w}$. Since $\boldsymbol{c}$ is obtained as a random element of an n-dimensional set, we have that with probability 1, $\boldsymbol{c}$ does not lie in any of these linear spaces. That is, $\boldsymbol{c}^\intercal(\boldsymbol{z}' - \boldsymbol{z}'') \neq 0$, as desired.

Under this genericity assumption, the particular choice of the new basis (when there are ties for the leaving variable) follows the lexicographic pivot rule until we make our first non-degenerate pivot. After that, the choice of the leaving variable is arbitrary. The following lemma implies that the Shadow pivot rule with a generic objective function does not cycle:

Lemma 19. *Given an LP of the form (1.1) with a generic objective function, in each iteration of the Shadow pivot rule, the value of* $\frac{\boldsymbol{c}^\intercal \boldsymbol{z}^j}{\boldsymbol{v}^\intercal \boldsymbol{z}^j}$ *of the chosen pivot direction* $\boldsymbol{z}^j$ *is strictly less than that of the previous iteration.*

Proof. Consider an arbitrary iteration at a basis B. Assume without loss of generality that for each generator $\boldsymbol{z}^i$ of $\mathcal{C}(B)$, if $\boldsymbol{v}^\intercal \boldsymbol{z}^i > 0$, then $\boldsymbol{v}^\intercal \boldsymbol{z}^i = 1$. Note that under this assumption, we seek to show that the value of $\boldsymbol{c}^\intercal \boldsymbol{z}^j$ is strictly less than that of the previous iteration.

Recall that the pivot direction $\boldsymbol{z}^j$ chosen by the Shadow pivot rule always has the property that $\pi(\boldsymbol{z}^j)$ is an extreme ray of $\pi(\mathcal{C}(B))$, where π is the projection defined in Subsection 4.3.1. For an element $\boldsymbol{z}$ of $\mathcal{C}(B)$, $\frac{\boldsymbol{c}^\intercal \boldsymbol{z}}{\boldsymbol{v}^\intercal \boldsymbol{z}}$ is precisely the slope of the ray of $\pi(\mathcal{C}(B))$ generated by $\pi(\boldsymbol{z})$, and by the convexity of $\pi(\mathcal{C}(B))$, the slopes of consecutive pivot directions are non-increasing. By our assumption of genericity, the slopes of consecutive pivot directions are not equal, and so they are decreasing, as desired. □

Together, Lemma 18 and Lemma 19 allow us to find our initial basis and ensure that we do not cycle. This allows us to prove Theorem 3 and Theorem 4, both of which we restate here for convenience:

Theorem 3. *On any* $0/1$*-LP of the form (1.1), the Simplex method with the Slim Shadow pivot rule reaches an optimal solution by performing no more than* n *non-degenerate pivots. Furthermore, it generates the same monotone path as the Slim Shadow rule.*

Theorem 4. *On any* $0/1$*-LP of the form (1.1) whose feasible region has dimension* d*, the Ordered Shadow pivot rule reaches an optimal solution by performing no more than* d *non-degenerate pivots. Furthermore, it generates the same monotone path as the Ordered Shadow rule.*

Proof of Theorems 3 and 4. The number of non-degenerate pivots performed by the Slim (resp. Ordered) Shadow pivot rule is precisely the number of edges in the path it takes on the 1-skeleton of the feasible region. It follows from Theorem 18 (resp. Theorem 19) and Theorem 21 that the number of non-degenerate pivots is therefore at most n (resp. d). □

5.4 Connections to Classical Algorithms

We wish to remark that there are several examples of well-known combinatorial algorithms that turn out to use exactly the same choice of improving steps as the pivot rules presented in this chapter, either in general or in special cases. In fact, while Theorem 2 only shows that the True Steepest-Edge pivot rule reaches an optimal solution within a strongly-polynomial number of non-degenerate steps, one can get a more refined bound on the number of steps for some well-known classes of polytopes, by realizing that classical algorithms for famous combinatorial optimization problems can be interpreted as moving along steepest edges on the 1-skeleton of the 0/1 polytope given by the set of feasible solutions.

The first example is the *shortest augmenting path algorithm* for the maximum matching and maximum flow problems. Specifically, the seminal works of Dinic in [42], and Edmonds and Karp in [45], gave the first strongly-polynomial time algorithm for the maximum flow problem, showing that one can augment a given flow using augmenting paths of shortest possible length (i.e., with the minimum number of edges). Since then, the idea of using shortest augmenting paths has been widely used in various contexts, such as for the maximum matching problem. Note that augmenting a given matching by switching the edges along an augmenting path corresponds to moving between adjacent extreme points of the matching polytope [31]. Therefore, computing a maximum matching using the shortest augmenting path algorithm corresponds (from a polyhedral perspective) to moving along steepest edge-directions on the 1-skeleton of the matching polytope.

Another example is the *minimum mean cycle canceling algorithm* by Goldberg and Tarjan [55]. This algorithm finds a minimum cost circulation in a directed graph by pushing flow along cycles whose ratio of cost to number of edges is minimal. This corresponds to moving along a Steepest Descent circuit (after doing a conversion to a minimization format), and as proved in Section 4.1, for 0/1 polytopes this corresponds to a steepest edge-direction at a given vertex. Hence, computing a minimum cost circulation using the minimum mean cycle canceling algorithm corresponds (from a polyhedral perspective) to moving along steepest edge-directions on the 1-skeleton of the 0/1 circulation polytope.

Similarly, the paths followed by the modified Shadow pivot rules specialize to well known optimization algorithms. Consider the *greedy algorithm for optimization on matroids*. Denote by $\mathcal{I}$ be the set of independent sets of a matroid on a ground set E. Recall the 0/1 matroid polytope associated to $\mathcal{I}$ is $P_{\mathcal{I}} = \operatorname{conv}\left\{\sum_{s\in S} \boldsymbol{e}_s : S \in \mathcal{I}\right\}$.

Consider the linear program $\max(\boldsymbol{c}^\intercal \boldsymbol{x} : \boldsymbol{x} \in P_{\mathcal{I}})$ for a matroid polytope $P_{\mathcal{I}}$ on a ground set E. Let $[\boldsymbol{0} = \boldsymbol{x}_0, \boldsymbol{x}_1, \ldots, \boldsymbol{x}_k]$ the path followed by the Slim Shadow rule for this LP. Let $\emptyset = S_0 \subsetneq S_1 \subsetneq \cdots \subsetneq S_k$ be the sequence of subsets chosen by the greedy algorithm. Then our goal is to show that $S_i = \operatorname{supp}(\boldsymbol{x}_i)$ for all $0 \leq i \leq k$. At S_i, the greedy algorithm says to add the highest weight element j not in S_i such that $S_i \cup \{j\}$ is still independent. Similarly, the Slim Shadow vertex pivot rule chooses $\boldsymbol{x}_{i+1}$ as follows:

$$\boldsymbol{x}_{i+1} = \operatorname*{argmax}_{\boldsymbol{u}\in N_{\boldsymbol{1}}(\boldsymbol{x}_i)} \frac{\boldsymbol{c}^\intercal(\boldsymbol{u} - \boldsymbol{x}_i)}{\boldsymbol{1}^\intercal(\boldsymbol{u} - \boldsymbol{x}_i)} = \operatorname*{argmax}_{\boldsymbol{u}\in N_{\boldsymbol{1}}(\boldsymbol{x}_i)} \boldsymbol{c}^\intercal(\boldsymbol{u} - \boldsymbol{x}_i).$$

All $\boldsymbol{1}$-improving neighbors of $\boldsymbol{x}_i$ are given by $\boldsymbol{x}_i + \boldsymbol{e}_j$ for some $j \notin \operatorname{supp}(\boldsymbol{x}_i)$ such that $\boldsymbol{x}_i + \boldsymbol{e}_j$ is still a vertex of P. Thus, $\boldsymbol{x}_{i+1}$ is given by maximizing $\boldsymbol{c}^\intercal(\boldsymbol{u} - \boldsymbol{x}_i) = \boldsymbol{c}^\intercal \boldsymbol{e}_j = \boldsymbol{c}(j)$ over all possible choices of j, which yields the result. The greedy algorithm also reflects the path chosen by True Steepest-Edge. At a greedily chosen vertex, all improving neighbors again correspond to adding some $\boldsymbol{e}_j$. Normalizing does not change the weight, so True Steepest-Edge corresponds also to maximizing $\boldsymbol{c}(j)$ over all options for j.

Special cases of the Ordered Shadow pivot rule paths also appear in the literature. Consider the stable set polytope of the complement of a chordal graph. This 0/1 polytope is the convex hull of all 0/1 incidence vectors of the cliques $\mathcal{S}(G)$ on a chordal graph G, i.e., $P_{\mathcal{S}(G)} = \operatorname{conv}(\left\{\sum_{s\in S} \boldsymbol{e}_s : S \in \mathcal{S}(G)\right\})$ (see [53]). Note that because chordal graphs are perfect, there is a complete inequality description of $P_{\mathcal{S}}$ using only clique inequalities. It is also well-known that a graph is chordal if and only if it has a *perfect elimination ordering* of its vertices [53], namely an ordering of the vertices of the graph such that, for each vertex v, v and the neighbors of v that occur after v in the order form a clique. This can be interpreted as a sequence of cliques of increasing sizes. Thus, one can use the perfect elimination orderings to obtain a maximum-size clique of a chordal graph in polynomial-time. Furthermore, it was shown in [81] that such an ordering may be found efficiently. One can check that the sequence of vertices obtained by the perfect elimination ordering coincides with the steps taken by the Ordered Shadow rule so long as the perfect elimination ordering coincides with the ordering of the indices in the corresponding 0/1 polytope.

In [69], the authors showed that there exist two-dimensional projections of 0/1 polytopes with exponentially many vertices. Hence, the original Shadow pivot rule may take an exponential number of iterations. Since the Slim Shadow rule requires a number of steps bounded by n, the dimension of the ambient space, this suggests the question: Is there an example in which the length of the path chosen by the Slim Shadow rule is exponential in the dimension d of the feasible region? Unfortunately, yes. The example in [69] can be modified to yield an explicit set of LPs for which the Slim Shadow rule requires a number of steps exponential in d (while of course still being bounded by the number of variables).

There is no canonical, universal winner on performance between our two Shadow rules. At least when applied as we described them here, in some cases the Slim Shadow rule may actually perform better than the Ordered Shadow rule. In particular, the bounds from sparsity in Lemma 13 may be stronger than the dimension bound. To see this note that the Birkhoff polytope for $n \times n$ permutation matrices has dimension $(n-1)^2$. This is the bound on the number of steps for the Ordered Shadow rule, yet the Slim Shadow rule achieves a bound of only n steps by Lemma 13.

One can ask, how good are the bounds we obtain on the length of monotone paths compared to the optimal bounds? Let us compare the Slim Shadow rule in a few instances:

- The rule yields at most n steps on both the asymmetric and symmetric traveling salesman polytopes for n vertex graphs. The respective optimal bounds are $\lfloor n/2 \rfloor$ and $\lfloor n/3 \rfloor$ (see [78]).
- The rule yields at most n steps on the Birkhoff polytope for $n \times n$ matrices. The optimal bound is $\lfloor n/2 \rfloor$ ([77]).
- The rule yields at most $\lfloor n/2 \rfloor$ steps for the perfect matching polytope on the complete graph with n vertices. The optimal bound is $\lfloor n/4 \rfloor$ ([77]).
- The rule yields at most $\text{rank}(\mathcal{M})$ for an independent set matroid polytope of a matroid $\mathcal{M}$ (starting at $\mathbf{0}$). As we saw this matches the optimal bound.

Thus the lengths we obtain are − up to a constant − the same as the actual monotone diameter in all of the above cases. Note that the bounds we find do not require any knowledge of the combinatorics of the graphs of any of these polytopes, yet they remain not far off from the best possible bounds.

Chapter 6

Future Work

The work herein leaves many open questions which would be good avenues for future work. Of course, the main open questions are whether there exists a polynomial pivot rule for the Simplex method and whether the polynomial Hirsch conjecture is true.

Our results in Chapter 5 do not analyze the number of degenerate pivots performed by any of the rules we propose. This suggests the following question:

Question 4. *Is there a polynomial bound to the number of* degenerate *pivots performed by any of the pivot rules presented here when applied to* $0/1$*-LPs? Is there a polynomial bound for any special subclasses of* $0/1$*-LPs?*

The results of Chapter 4 and Chapter 5 do not naively translate to LPs which are not $0/1$. It would be interesting to investigate ways to achieve similar results for other well-studied classes of LPs. For example, a natural generalization of 0/1 polytopes are lattice polytopes, but even in the case of LPs defined over 2-lattice polytopes (i.e., those whose vertices are contained in $\{0, 1, 2\}^n$), similar techniques fail.

Question 5. *Is there a pivot rule for the Simplex method that guarantees a (strongly) polynomial number of non-degenerate pivots on lattice polytopes?*

Of course, there may indeed be other classes of polyhedra for which our proposed pivot rules *do* perform well. For example, another natural generalization of 0/1 polytopes are *box polytopes* – those whose vertices are a subset of the vertices of a rectangular prism. These have similar properties to 0/1 polytopes that make them promising candidates. In general, we have the following question:

Question 6. *For what other classes of LPs do our proposed pivot rules require only a polynomial number of non-degenerate pivots?*

The hardness result of Theorem 1 requires the use of degeneracy. The same theorem for non-degenerate polytopes would be a stronger result, and is left as an open question:

Question 7. *What is the computational complexity of finding a shortest monotone path to an optimal solution of a non-degenerate 0/1-LP?*

Theorem 7 gives a *weakly*-polynomial bound on the circuit diameter of polyhedra, while it has been conjectured that the Hirsch bound holds for the circuit diameter [19]. A next step in this direction is to answer the following question:

Question 8. *Does there exist a strongly-polynomial bound on the on the circuit diameter of polyhedra?*

There are still many open questions in the setting of circuit-augmentation algorithms. Of paramount importance is the following:

Question 9. *Does there exist a circuit-pivot rule that can be computed in (strongly) polynomial time without the use of an LP solver which arrives at an optimal solution of an LP in a (strongly) polynomial number of augmentations?*

Recent work of Sanità [82] shows that it is strongly-NP-Hard to compute the combinatorial diameter of a polyhedron by utilizing a characterization of the combinatorial diameter of the fractional matching polytope. The computational complexity of computing the circuit-diameter of a polyhedron is open. It is possible that the characterization of the circuits of the fractional matching polytope can be useful in determining this complexity, and resolving this question would be of significant interest.

References

[1] I. Adler, C. Papadimitriou, and A. Rubinstein. On simplex pivoting rules and complexity theory. In *Integer programming and combinatorial optimization*, volume 8494 of *Lecture Notes in Comput. Sci.*, pages 13–24. Springer, Cham, 2014.

[2] R.K. Ahuja, T.L. Magnanti, and J.B. Orlin. *Network flows.* Prentice Hall Inc., Englewood Cliffs, NJ, 1993.

[3] N. Amenta and G. M. Ziegler. Deformed products and maximal shadows of polytopes. In *Advances in discrete and computational geometry (South Hadley, MA, 1996)*, volume 223 of *Contemp. Math.*, pages 57–90. Amer. Math. Soc., Providence, RI, 1999.

[4] D. Avis and O. Friedmann. An exponential lower bound for Cunningham's rule. *Mathematical Programming*, 161(1):271–305, Jan 2017.

[5] M. L. Balinski. The Hirsch conjecture for dual transportation polyhedra. *Mathematics of Operations Research*, pages 629–633, 1984.

[6] M. L. Balinski and A. Russakoff. On the assignment polytope. *SIAM Review*, 16(4):516–525, 1974.

[7] R. E. Behrend. Fractional perfect b-matching polytopes. i: General theory. *Linear Algebra and its Applications*, 439, 01 2013.

[8] D. Bertsimas and J. Tsitsiklis. *Introduction to Linear Optimization.* Athena Scientific, 1997.

[9] L. Billera, M. Kapranov, and B. Sturmfels. Cellular strings on polytopes. *Proceedings of the American Mathematical Society*, 122(2):549–555, 1994.

[10] L. J. Billera and B. Sturmfels. Fiber polytopes. *Ann. of Math. (2)*, 135(3):527–549, 1992.

[11] A. Black and J. A. De Loera. Monotone paths on cross-polytopes. *arXiv:2102.01237*, 2021.

[12] A. Black, J. A. De Loera, S. Kafer, and L. Sanità. On the Simplex method for 0/1 polytopes. *arXiv: https://arxiv.org/abs/2111.14050*, 2021.

[13] R. G. Bland. *Complementary orthogonal subspaces of n-dimensional Euclidean space and orientability of matroids.* 1974. Thesis (Ph.D.)–Cornell University.

[14] R. G. Bland. On the generality of network flow theory. Presented at the ORSA/TIMS Joint National Meeting, Miami, Florida, 1976.

[15] R. G. Bland and J. Edmonds. Fast primal algorithms for totally unimodular linear programming. Presented at XI-th International Symposium on Mathematical Programming, Bonn, West Germany, 1982.

[16] K. H. Borgwardt. *The simplex method: a probabilistic analysis*, volume 1. Springer Science & Business Media, 1 edition, 1987.

[17] S. Borgwardt, J. A. De Loera, and E. Finhold. Edges vs circuits: a hierarchy of diameters in polyhedra. *Advances in Geometry*, 16, 09 2014.

[18] S. Borgwardt, C. Brand, A. E. Feldmann, and M. Koutecký. A note on the approximability of deepest-descent circuit steps. *Operations Research Letters*, 49:310–315, 2021.

[19] S. Borgwardt, E. Finhold, and R. Hemmecke. On the circuit diameter of dual transportation polyhedra. *SIAM Journal on Discrete Mathematics*, 29(1):113–121, 2015.

[20] S. Borgwardt, E. Finhold, and R. Hemmecke. Quadratic diameter bounds for dual network flow polyhedra. *Mathematical Programming*, 159:237–251, 2016.

[21] S. Borgwardt, J. A. De Loera, and E. Finhold. The diameters of network-flow polytopes satisfy the Hirsch conjecture. *Mathematical Programming*, 171:283–309, 2018.

[22] S. Borgwardt, T. Stephen, and T. Yusun. On the circuit diameter conjecture. *Discrete & Computational Geometry*, 60:558–587, 2018.

[23] S. Borgwardt and C. Viss. An implementation of steepest-descent augmentation for linear programs. *Operations Research Letters*, 48:323–328, 2020.

[24] S. Borgwardt and C. Viss. Constructing clustering transformations. *SIAM Journal on Discrete Mathematics*, 35(1):152–178, 2021.

[25] S. Borgwardt and C. Viss. A Polyhedral Model for Enumeration and Optimization over the Set of Circuits. *Discrete Applied Mathematics*, 308:68–83, Feb 2022.

[26] G. Brightwell, J. Heuvel, and L. Stougie. A linear bound on the diameter of the transportation polytope. *Combinatorica*, pages 133–139, 2006.

[27] T. Brunsch and H. Röglin. Finding short paths on polytopes by the shadow vertex algorithm. In F. V. Fomin, R. Freivalds, M. Kwiatkowska, and D. Peleg, editors, *Automata, Languages, and Programming*, pages 279–290, Berlin, Heidelberg, 2013. Springer Berlin Heidelberg.

[28] D. Cardoso and J. Clímaco. The generalized simplex method. *Operations Research Letters*, 12:337–348, 1992.

[29] S. Chubanov. A generalized simplex method for integer problems given by verification oracles. *SIAM J. Optim.*, 31(1):686–701, 2021.

[30] S. Chubanov. A scaling algorithm for optimizing arbitrary functions over vertices of polytopes. *Math. Program.*, 190(1-2, Ser. A):89–102, 2021.

[31] V Chvátal. On certain polytopes associated with graphs. *Journal of Combinatorial Theory, Series B*, 18(2):138–154, 1975.

[32] W. Cook, W. Cunningham, W. Pulleyblank, and A. Schrijver. *Combinatorial Optimization*. Wiley, 1997.

[33] D. Dadush and N. Hähnle. On the shadow simplex method for curved polyhedra. *Discrete & Computational Geometry*, 56(4):882–909, 2016.

[34] D. Dadush and S. Huiberts. A friendly smoothed analysis of the simplex method. In *Proceedings of the 50th Annual ACM SIGACT Symposium on Theory of Computing, STOC 2018, Los Angeles, CA, USA, June 25-29, 2018*, pages 390–403, 2018.

[35] D. Dadush, Z. K. Koh, B. Natura, and L. Végh. On circuit diameter bounds via circuit imbalances. In *Integer programming and combinatorial optimization*, volume 13265 of *Lecture Notes in Computer Science*. Springer, Cham, 2022.

[36] J. A. De Loera, R. Hemmecke, and M. Köppe. *Algebraic and geometric ideas in the theory of discrete optimization*, volume 14 of *MOS-SIAM Series on Optimization*. Society for Industrial and Applied Mathematics (SIAM), Philadelphia, PA, 2013.

[37] J. A. De Loera, R. Hemmecke, and J. Lee. On augmentation algorithms for linear and integer-linear programming: From Edmonds–Karp to Bland and beyond. *SIAM Journal on Optimization*, 25:2494–2511, 2015.

[38] J. A. De Loera, S. Kafer, and L. Sanità. Pivot rules for circuit-augmentation algorithms in linear optimization. *arXiv:1909.12863*, 2020.

[39] J. A. De Loera, B. Sturmfels, and R. R. Thomas. Gröbner bases and triangulations of the second hypersimplex. *Combinatorica*, 15(3):409–424, Sep 1995.

[40] A. Del Pia and C. Michini. Short Simplex paths in lattice polytopes. *Discrete & Computational Geometry*, 67:503–524, 2022.

[41] A. Deza and L. Pournin. Improved bounds on the diameter of lattice polytopes. *Acta Mathematica Hungarica*, 154:457–469, 2018.

[42] E. A. Dinic. Algorithm for solution of a problem of maximum flow in a network with power estimation. In *Soviet Mathematics - Doklady*, volume 11, pages 1277–1280. 1970.

[43] Y. Disser and A. V. Hopp. On Friedmann's subexponential lower bound for Zadeh's pivot rule. In A. Lodi and V. Nagarajan, editors, *Integer Programming and Combinatorial Optimization*, pages 168–180, Cham, 2019. Springer International Publishing.

[44] Y. Disser and M. Skutella. The simplex algorithm is NP-mighty. In *Proceedings of the Twenty-Sixth Annual ACM-SIAM Symposium on Discrete Algorithms*, pages 858–872. SIAM, Philadelphia, PA, 2015.

[45] J. Edmonds and R. M. Karp. Theoretical improvements in algorithmic efficiency for network flow problems. *J. ACM*, 19(2):248–264, April 1972.

[46] F. Eisenbrand, C. Hunkenschröder, K. Klein, M. Koutecký, A. Levin, and S. Onn. An algorithmic theory of integer programming, 2019.

[47] F. Eisenbrand and S. Vempala. Geometric random edge. *Mathematical Programming*, 164(1):325–339, 2017.

[48] J. Fearnley and R. Savani. The complexity of the simplex method. In *STOC'15—Proceedings of the 2015 ACM Symposium on Theory of Computing*, pages 201–208. ACM, New York, 2015.

[49] E. Finhold. *Circuit diameters and their application to transportation problems.* PhD thesis, Technische Univ. München, 2015.

[50] J. J. Forrest and D. Goldfarb. Steepest-edge simplex algorithms for linear programming. *Math. Program.*, 57:341–374, 1992.

[51] A. Frank and E. Tardos. An application of simultaneous Diophantine approximation in combinatorial optimization. *Combinatorica*, 7(1):49–65, 1987.

[52] O. Friedmann, T. D. Hansen, and U. Zwick. Subexponential lower bounds for randomized pivoting rules for the simplex algorithm. In *Proceedings of the 43rd ACM Symposium on Theory of Computing, STOC'11, San Jose, CA, USA*, pages 283–292, 2011.

[53] D. Fulkerson and O. Gross. Incidence matrices and interval graphs. *Pacific Journal of Mathematics*, 15(3):835–855, 1965.

[54] J. B. Gauthier, J. Desrosiers, and M. Lübbecke. Decomposition theorems for linear programs. *Operations Research Letters*, 42(8):553 – 557, 2014.

[55] A. V. Goldberg and R. E. Tarjan. Finding minimum-cost circulations by canceling negative cycles. *J. ACM*, 36(4):873–886, October 1989.

[56] D. Goldfarb. On the complexity of the simplex algorithm. In S. Gomez and J.P. Hennart, editors, *Advances in Optimization and Numerical Analysis - Proceedings 6th Workshop on Optimization and Numerical Analysis, Oaxaca Mexico, 1992*, pages 25–38. Kluwer Dordrecht, 1994.

[57] D. Goldfarb and W. Y. Sit. Worst case behavior of the steepest edge simplex method. *Discrete Applied Mathematics*, 1(4):277 – 285, 1979.

[58] J.E. Graver. On the foundations of linear and integer programming I. *Mathematical Programming*, 9:207–226, 1975.

[59] M. Grötschel and M. Padberg. Polyhedral theory. *The Traveling Salesman Problem: A Guided Tour of Combinatorial Optimization*, pages 251–305, 1986.

[60] T. Hansen and U. Zwick. An improved version of the random-facet pivoting rule for the simplex algorithm. In *Proceedings of the Forty-Seventh Annual ACM on Symposium on Theory of Computing, STOC 2015, Portland, OR, USA, June 14-17, 2015*, pages 209–218, 2015.

[61] R. Hemmecke, S. Onn, and R. Weismantel. A polynomial oracle-time algorithm for convex integer minimization. *Mathematical Programming*, 126:97–117, 2011.

[62] R. G. Jeroslow. The simplex algorithm with the pivot rule of maximizing criterion improvement. *Discrete Mathematics*, 4(4):367 – 377, 1973.

[63] S. Kafer, K. Pashkovich, and L. Sanità. On the circuit diameter of some combinatorial polytopes. *SIAM Journal on Discrete Mathematics*, 33(1):1–25, 2019.

[64] T. Kitahara and S. Mizuno. Klee-Minty's LP and upper bounds for Dantzig's simplex method. *Operations Research Letters*, 39(2):88–91, 2011.

[65] T. Kitahara and S. Mizuno. A bound for the number of different basic solutions generated by the simplex method. *Math. Program.*, 137(1-2):579–586, 2013.

[66] V. Klee and P. Kleinschmidt. Geometry of the Gass-Saaty parametric cost lp algorithm. *Discrete & Computational Geometry*, 5(1):13–26, 1990.

[67] V. Klee and G. J. Minty. How good is the simplex algorithm? In *Inequalities, III (Proc. Third Sympos., Univ. California, Los Angeles, Calif., 1969; dedicated to the memory of Theodore S. Motzkin)*, pages 159–175. Academic Press, New York, 1972.

[68] V. Klee and D. W. Walkup. The d-step conjecture for polyhedra of dimension d <6. *Acta Mathematica*, 117(none):53 – 78, 1967.

[69] U. Kortenkamp, J. Richter-Gebert, A. Sarangarajan, and G. M. Ziegler. Extremal properties of 0/1-polytopes. *Discret. Comput. Geom.*, 17(4):439–448, 1997.

[70] S. Müller and G. Regensburger. Elementary vectors and conformal sums in polyhedral geometry and their relevance for metabolic pathway analysis. *Frontiers in Genetics*, 7:90, 2016.

[71] K. G. Murty. Computational complexity of parametric linear programming. *Math. Programming*, 19(2):213–219, 1980.

[72] K. G. Murty. *Linear programming.* John Wiley & Sons, Inc., New York, 1983. With a foreword by George B. Dantzig.

[73] K. G. Murty. Complexity of degeneracy. In C. A. Floudas and P. M. Pardalos, editors, *Encyclopedia of Optimization*, pages 419–425. Springer US, Boston, MA, 2009.

[74] D. Naddef. The Hirsch conjecture is true for $(0, 1)$-polytopes. *Math. Program.*, 45(1):109–110, August 1989.

[75] S. Onn. *Non-Linear Discrete Optimization.* Zurich Lectures in Advanced Mathematics. European Mathematical Society, 2010.

[76] J. B. Orlin. A polynomial time primal network simplex algorithm for minimum cost flows. *Mathematical Programming*, 78(2):109–129, 1997.

[77] F. J. Rispoli. The monotonic diameter of the perfect matching and shortest path polytopes. *Operations Research Letters*, 12(1):23–27, 1992.

[78] F. J. Rispoli. The monotonic diameter of traveling salesman polytopes. *Operations Research Letters*, 22(2-3):69–73, 1998.

[79] F.J. Rispoli and S. Cosares. A bound of 4 for the diameter of the symmetric traveling salesman polytope. *SIAM Journal on Discrete Mathematics*, pages 373–380, 1998.

[80] R.T. Rockafellar. The elementary vectors of a subspace of R^N. In *Combinatorial Mathematics and its Applications (Proc. Conf., Univ. North Carolina, Chapel Hill, N.C., 1967)*, pages 104–127. Univ. North Carolina Press, Chapel Hill, N.C., 1969.

[81] D. J. Rose, R. E. Tarjan, and G. S. Lueker. Algorithmic aspects of vertex elimination on graphs. *SIAM Journal on computing*, 5(2):266–283, 1976.

[82] L. Sanità. The diameter of the fractional matching polytope and its hardness implications. In *FOCS 2018 (59th Annual IEEE Symposium on Foundations of Computer Science)*, pages 910–921, 10 2018.

[83] F. Santos. A counterexample to the Hirsch conjecture. *Annals of Mathematics*, 176(1):383–412, 2012.

[84] A. Schrijver. *Theory of linear and integer programming.* Wiley-Interscience Series in Discrete Mathematics. John Wiley & Sons Ltd., 1986. A Wiley-Interscience Publication.

[85] A. S. Schulz and R. Weismantel. The complexity of generic primal algorithms for solving general integer programs. *Mathematics of Operations Research*, 27:681–692, 2002.

[86] A. S. Schulz, R. Weismantel, and G. M. Ziegler. 0/1-integer programming: Optimization and augmentation are equivalent. In Paul Spirakis, editor, *Algorithms — ESA '95*, pages 473–483, Berlin, Heidelberg, 1995. Springer Berlin Heidelberg.

[87] D. A. Spielman and S.-H. Teng. Smoothed analysis of algorithms: why the simplex algorithm usually takes polynomial time. *J. ACM*, 51(3):385–463 (electronic), 2004.

[88] B. Sturmfels. *Gröbner bases and convex polytopes*, volume 8 of *University Lecture Series*. American Mathematical Society, Providence, RI, 1996.

[89] N. Sukegawa. Improving bounds on the diameter of a polyhedron in high dimensions. *Discrete Mathematics*, 340(9):2134–2142, 2017.

[90] N. Sukegawa. An asymptotically improved upper bound on the diameter of polyhedra. *Discrete & Computational Geometry*, 62(3):690–699, 2019.

[91] T. Terlaky. Lexicographic pivoting rules. In C. A. Floudas and P. M. Pardalos, editors, *Encyclopedia of Optimization, Second Edition*, pages 1870–1873. Springer, 2009.

[92] T. Terlaky and S. Zhang. Pivot rules for linear programming: A survey on recent theoretical developments. *Annals OR*, 46-47(1):203–233, 1993.

[93] R. J. Vanderbei. *Linear programming*. International Series in Operations Research & Management Science, 114. Springer, New York, third edition, 2008. Foundations and extensions.

[94] R. Vershynin. Beyond Hirsch conjecture: walks on random polytopes and smoothed complexity of the simplex method. *SIAM J. Comput.*, 39(2):646–678, 2009.

[95] Y. Ye. The simplex and policy-iteration methods are strongly polynomial for the Markov decision problem with a fixed discount rate. *Math. Oper. Res.*, 36(4):593–603, 2011.

[96] N. Zadeh. What is the worst case behavior of the simplex algorithm? In *Polyhedral computation*, volume 48 of *CRM Proc. Lecture Notes*, pages 131–143. Amer. Math. Soc., Providence, RI, 2009.

[97] G. M. Ziegler. *Lectures on polytopes*, volume 152 of *Graduate Texts in Mathematics*. Springer-Verlag, New York, 1995.

[98] G. M. Ziegler. Lectures on 0/1-polytopes. In *Polytopes—combinatorics and computation (Oberwolfach, 1997)*, volume 29 of *DMV Sem.*, pages 1–41. Birkhäuser, Basel, 2000.

[99] G. M. Ziegler. Typical and extremal linear programs. In *The sharpest cut*, MPS/SIAM Ser. Optim., pages 217–230. SIAM, Philadelphia, PA, 2004.