ARTICLE IN PRESS
JID: EOR [m5G;April 19, 2023;13:4]

European Journal of Operational Research xxx (xxxx) xxx

Contents lists available at ScienceDirect

European Journal of Operational Research

journal homepage: www.elsevier.com/locate/ejor

Continuous Optimization

Revisiting degeneracy, strict feasibility, stability, in linear programming

Haesol Im*, Henry Wolkowicz

Department of Combinatorics and Optimization, Faculty of Mathematics, University of Waterloo, Research supported by The Natural Sciences and Engineering Research Council of Canada, Waterloo, Ontario N2L 3G1, Canada

ARTICLE INFO

Article history:
Received 11 July 2022
Accepted 16 March 2023
Available online xxx

Keywords:
Linear programming
Facial reduction
Preprocessing
Degeneracy
Implicit problem singularity

ABSTRACT

Currently, the simplex method and the interior point method are indisputably the most popular algorithms for solving linear programs, **LP**s. Unlike general conic programs, **LP**s with a finite optimal value do not require strict feasibility in order to establish strong duality. Hence strict feasibility is seldom a concern, even though strict feasibility is equivalent to stability and a compact dual optimal set. This lack of concern is also true for other types of degeneracy of basic feasible solutions in **LP**. In this paper we discuss that the specific degeneracy that arises from lack of strict feasibility necessarily causes difficulties in both simplex and interior point methods. In particular, we show that the lack of strict feasibility implies that every basic feasible solution, **BFS**, is degenerate; thus conversely, the existence of a nondegenerate **BFS** implies that strict feasibility (regularity) holds. We prove the results using facial reduction and simple linear algebra. In particular, the facially reduced system reveals the implicit non-surjectivity of the linear map of the equality constraint system. As a consequence, we emphasize that facial reduction involves two steps where, the first guarantees strict feasibility, and the second recovers full row rank of the constraint matrix. This illustrates the implicit *singularity* of problems where strict feasibility fails, and also helps in obtaining new efficient techniques for preproccessing. We include an efficient preprocessing method that can be performed as an extension of phase-I of the two-phase simplex method. We show that this can be used to avoid the loss of precision for many well known problem sets in the literature, e.g., the NETLIB problem set.

1. Introduction

The *Slater condition* (strict feasibility) is a useful property for optimization models to have. Unlike general conic programs, linear programs (**LP**s) do not require strict feasibility as a constraint qualification to guarantee strong duality, and therefore, it is often not discussed. In fact, degeneracy in general is not considered to be a serious concern in linear programming. The Goldman-Tucker Theorem (Goldman & Tucker, 1956) is related in that it guarantees a primal-dual optimal solution satisfying strict complementarity $x^* + z^* > 0$ for the standard form **LP**. However, it does not guarantee the existence of a strictly feasible primal solution $\hat{x} > 0$. The lack of strict feasibility for an **LP** does not seem to cause problems at first glance, especially when the simplex method is used. In this manuscript, we show that the failure of strict feasibility results in degeneracy problems when simplex-type methods are used. More specifically, the lack of strict feasibility inevitably renders **LP**s degenerate, i.e., *every basic feasible solution is degenerate*.[1] Note that strict feasibility along with full row rank of the linear constraint is the *Mangasarian-Fromovitz* constraint qualification (Mangasarian & Fromovitz, 1967). This is equivalent to a compact dual optimal set and is equivalent to stability with respect to perturbations of the right-hand side.

The simplex method (Dantzig, 1963) is one of the most popular and successful algorithms for solving linear programs. Degeneracy, a zero basic variable, could result in cycling and noncovergence. There are many anti-cycling rules, see e.g., (Bland, 1977; Dantzig et al., 1955; Gal, 1993; Hall & McKinnon, 2004; Terlaky & Zhang, 1993) and the references therein. However, techniques for the resolution of degeneracy often result in *stalling* (Bixby, 2002; Charnes, 1952; Megiddo, 1986; Ryan & Osborne, 1988), i.e., result in taking a large number of iterations before leaving a degenerate point and can even fail to leave with current techniques (Hall & McKinnon, 2004). Degeneracies are known to cause numerical issues when interior point methods are used, e.g., GüLer et al. (1993). For example, degeneracy can result in singularity of the Jacobian of the

* Corresponding author.
E-mail address: j5im@uwaterloo.ca (H. Im).
URL: http://www.math.uwaterloo.ca/~hwolkowi/ (H. Im)

[1] Conversely, if we can find *one* nondegenerate basic feasible solution, then strict feasibility holds.

https://doi.org/10.1016/j.ejor.2023.03.021
0377-2217/

optimality conditions, and thus also in ill-posedness and loss of accuracy (Gonzalez-Lima et al., 2009). We note that the method most often used in the literature when converting a problem that has a free variable into standard form, is to replace the free variable by the difference of two nonnegative variables. This results in an unbounded primal optimal set and strict feasibility failing for the dual problem, i.e., from our work we see that this standard approach changes a well-posed problem into an ill-posed one.

Our main results on the degeneracy arising from loss of strict feasibility are shown using the effective preprocessing tool called *facial reduction, **FR***. For a problem lacking strict feasibility, facial reduction strives to formulate an equivalent problem that has a Slater point. By examining the facially reduced system, we obtain two results. First, we show that every basic feasible solution is degenerate when strict feasibility fails. This leads to an efficient approach for eliminating variables that are fixed at 0. Second, we investigate implicit redundancies as a source of instability arising in problems where strict feasibility fails. We see that the linear map of the facially reduced system is non-surjective, i.e., the original constraints are implicitly redundant. Finally, we use these results to develop an efficient preprocessing technique to obtain strict feasibility. This technique is illustrated on instances from the NETLIB data set.

The contribution of this manuscript is threefold; (i) We provide the complete description of the facially reduced system of a linear program and introduce related notions of singularity; (ii) We show that every basic feasible solution of a standard linear program is degenerate when strict feasibility fails; (iii) We propose and illustrate an efficient preprocessing scheme that can be performed as an extension of phase-I of the two-phase simplex method. This technique allows for eliminating variables fixed at 0, and thus regularizing and simplifying the **LP**.

The manuscript is organized as follows. In Section 2 we present the background and notations. Included are the notions of degeneracy, facial reduction and three types of singularity degree. We then describe what facial reduction tries to achieve. In Section 3 we present our main result and immediate corollaries, as well as the efficient preprocessing method that can be used as an extension of phase-I of the two-phase simplex method. In addition, we relate our main result to known results in the literature, such as distance to infeasibility. In Section 4 we illustrate algorithmic performance of interior point methods and the simplex method under the lack of strict feasibility. We present our conclusions in Section 5.

2. Preliminaries

2.1. Background and notation

We let $\mathbb{R}^n, \mathbb{R}^{m\times n}$ be the standard real vector spaces of n-coordinates and m-by-n matrices, respectively. We use $\mathbb{R}^n_+$ ($\mathbb{R}^n_{++}$, resp.) to denote the n-tuple with nonnegative (positive) entries. We use $\langle \cdot,\cdot \rangle$ to denote the usual inner product. Given a vector $x\in\mathbb{R}^n$, we let $supp(x)$ to denote the index set $\{i : x_i \neq 0\}$. Given a matrix $A\in\mathbb{R}^{m\times n}$, we adopt the MATLAB notation to denote a submatrix of A. Given a subset $\mathcal{I}$ of column indices, we use $A_{\mathcal{I}}\in\mathbb{R}^{m\times|\mathcal{I}|}$ to denote the submatrix of A that contains the columns of A in $\mathcal{I}$, i.e., $A_{\mathcal{I}} = A(:,\mathcal{I})$. Given a convex set $\mathcal{C}$, $relint(\mathcal{C})$ denotes the relative interior of the set $\mathcal{C}$.

Throughout this manuscript, we work with feasible **LP**s in standard form with finite optimal value:

$$(\mathcal{P}) \qquad p^* = \min_x \left\{ c^T x \ : \ Ax = b,\ x \ge 0 \right\},$$

where $p^*\in\mathbb{R}, A\in\mathbb{R}^{m\times n}, b\in\mathbb{R}^m$ and $c\in\mathbb{R}^n$. We assume that $rank(A) = m$, i.e., there is no redundant constraint. We use $\mathcal{F}$ to denote the feasible region of $(\mathcal{P})$

$$\mathcal{F} = \{x\in\mathbb{R}^n : Ax = b,\ x\ge 0\}. \tag{2.1}$$

2.1.1. Degeneracy in LP

Given an index set $\mathcal{B}\subset\{1,\dots,n\}, |\mathcal{B}| = m$, a point $x\in\mathcal{F}$ is called a *basic feasible solution, **BFS***, if $A_{\mathcal{B}}$ is nonsingular and $x_i = 0,\ \forall i\in\{1,\dots,n\}\setminus\mathcal{B}$. It is well-known that the simplex method iterates from **BFS** to **BFS**. A basic feasible solution $x\in\mathcal{F}$ is *nondegenerate* if $x_i > 0,\ \forall i\in\mathcal{B}$; it is *degenerate* if $x_i = 0$, for some $i\in\mathcal{B}$. It is clear that every basic feasible solution has at most m positive entries.[2]

We partition the index set $\{1,\dots,n\}$ as

$$\{1,\dots,n\} = \mathcal{I}_+\cup\mathcal{I}_0, \text{ where } \mathcal{I}_0 := \{i : x_i = 0, \forall x\in\mathcal{F}\} \text{ and } \mathcal{I}_+ = \{1,\dots,n\}\backslash\mathcal{I}_0,$$

i.e., $\mathcal{I}_0$ denotes the variables *fixed at 0*. Note that fixed variables are identified during preprocessing in the literature if the upper and lower bounds are equal, e.g., Andersen & Andersen (1995); Huang (2004); Mészáros & Suhl (2003). However, the set $\mathcal{I}_0$ is not as easily identified.

There are in fact several types of degeneracy. Let $\bar{x}$ be a given **BFS** with basis $\mathcal{B}$. (Wlog $\mathcal{B} = \{1,\dots,m\}$.) We can write the equivalent canonical form representation of the feasible set using the basis at $\bar{x}$:

$$\mathcal{F} = \left\{ x = \begin{pmatrix} x_{\mathcal{B}} \\ x_{\mathcal{N}} \end{pmatrix} \ : \ x_{\mathcal{B}} = b - A_{\mathcal{B}}^{-1}A_{\mathcal{N}}x_{\mathcal{N}} \ge 0, x_{\mathcal{N}} \ge 0 \right\}. \tag{2.2}$$

In this form $x_{\mathcal{N}}\in\mathbb{R}^{n-m}_+$, we have n inequality constraints, and we see that degeneracy is equivalent to having an active set with cardinality greater than $n-m$. This divides into two types corresponding to the sets $\mathcal{I}_0, \mathcal{I}_+$, respectively: (i) inequalities that are active in every **BFS** and correspond to variables in $\mathcal{I}_0$ above; (ii) those that are not active in at least one **BFS**. The geometry of (i) is clear as there is no Slater point and $\mathcal{F}$ is a subset of a face of the nonnegative orthant. For (ii) the geometry is that some of the constraints are redundant in one of two ways, i.e., that discarding them does not change the feasible set nor the optimality conditions if $\bar{x}$ is optimal.

Remark 2.1. We note that adding redundant constraints is done in e.g., Deza et al. (2006, 2008) to show that the central path for interior point methods can follow the boundary closely, i.e., behave very poorly. These redundant constraints correspond to a positive variable in each **BFS**, i.e., to an inequality in Eq. (2.2) that is never active. Complementary slackness implies that they correspond to variables fixed at 0 in the dual problem, thus emphasizing that **FR** on the dual could avoid some of these difficulties.

2.2. Facial reduction

In this section we describe the concept of facial reduction and present the properties that are used to establish the main result. We emphasize in this paper that facial reduction for $(\mathcal{P})$ involves *two* steps: first, obtain an equivalent problem with strict feasibility; second, recover full row rank of the constraint matrix. Note that full row rank is *always* lost during the first step.

Let $K\subset\mathbb{R}^n$ be a convex set. A convex set $F\subseteq K$ is called a *face* of K, denoted $F\trianglelefteq K$, if for all $y,z\in K$ with $x = \frac{1}{2}(y+z)\in F$, we have $y,z\in F$. Given a convex set $\mathcal{C}\subseteq K$, the *minimal face* for $\mathcal{C}$ is the intersection of all faces containing the set $\mathcal{C}$.

Proposition 2.2 (Drusvyatskiy & Wolkowicz (2017, Theorem 3.1.3) (theorem of the alternative))**.** *For the feasible system of Eq. (2.1), exactly one of the following statements holds:*

1. *There exists $x\in\mathbb{R}^n_{++}$ with $Ax = b$, i.e., strict feasibility holds;*

[2] We mainly consider primal degeneracy here, though everything follows through for dual degeneracy. In fact, there are clear connections from complementary slackness between variables positive in every **BFS** and dual variables fixed at 0.

2. *There exists* $y \in \mathbb{R}^m$ *such that*

$$0 \neq z := A^T y \in \mathbb{R}^n_+, \text{ and } \langle b, y\rangle = 0. \tag{2.3}$$

Proposition 2.2 gives rise to a process called *facial reduction*. The *facial reduction, **FR***, for an **LP** is a process of identifying the minimal face of $\mathbb{R}^n_+$ containing the feasible set $\mathcal{F} = \{x \in \mathbb{R}^n_+ : Ax = b\}$. By finding the minimal face, we can work with a problem that lies in a smaller dimensional space and that statisfies strict feasibility. The **FR** process, i.e., finding the minimal face, is usually done by solving a sequence of auxiliary systems Eq. (2.3). More details on **FR** on general conic problems can be found in Borwein & Wolkowicz (1980/81, 1981); Drusvyatskiy & Wolkowicz (2017); Permenter (2017); Sremac (2019).

We now describe how the set $\mathcal{F}$ (see Eq. (2.1)) is represented after **FR**. Suppose that strict feasibility fails. Then Proposition 2.2 implies that there must exist a nonzero $y \in \mathbb{R}^m$ satisfying

$$\langle x, A^T y\rangle = \langle Ax, y\rangle = \langle b, y\rangle = 0, \ \forall x \in \mathcal{F}. \tag{2.4}$$

Hence, every $x \in \mathcal{F}$ is perpendicular to the nonnegative vector $z = A^T y$. We call this vector $z = A^T y$ an *exposing vector* for $\mathcal{F}$, and let the cardinality of its support be $s_z = |\{i : z_i > 0\}|$. Then $z = \sum_{j=1}^{s_z} z_{t_j} e_{t_j}$, where t_j is in increasing order. We now have

$$0 = \langle z, x\rangle \quad \text{and} \quad x, z \in \mathbb{R}^n_+ \ \Rightarrow x_i z_i = 0, \ \forall i,$$

i.e., the positive elements in z identify the corresponding elements in x that are *fixed at 0*. Then $x = \sum_{j=1}^{n-s_z} x_{s_j} e_{s_j}$, where s_j is in increasing order. We define the matrix with unit vectors for columns

$$V = \begin{bmatrix} e_{s_1} & e_{s_2} & \dots & e_{s_{n-s_z}} \end{bmatrix} \in \mathbb{R}^{n\times(n-s_z)}.$$

Then we have

$$\mathcal{F} = \{x \in \mathbb{R}^n_+ : Ax = b\} = \{x = Vv \in \mathbb{R}^n : AVv = b, v \in \mathbb{R}^{n-s_z}_+\}. \tag{2.5}$$

We call this matrix $V \in \mathbb{R}^{n\times(n-s_z)}$ a *facial range vector*. The facial range vector restricts the support of all feasible x. We use the identification Eq. (2.5) throughout this manuscript. This concludes the first step of **FR**, i.e., identifying all the variables that are fixed at 0.[3]

It is known that every facial reduction step results in at least one constraint being redundant, see e.g., Borwein & Wolkowicz (1981), Im & Wolkowicz (2021, Lemma 2.7), and Sremac (2019, Section 3.5). For completeness we now include a short proof tailored to **LP**, see Lemma 2.3.

Lemma 2.3. *Consider the facially reduced feasible set*

$$\mathcal{F}_r = \left\{v : AVv = b, v \in \mathbb{R}^{n-s_z}_+\right\}.$$

*Then at least one linear constraint of the **LP** is redundant.*

Proof. Let $z = A^T y$ be the exposing vector satisfying the auxiliary system Eq. (2.3). And let V be a facial range vector induced by z. Then

$$0 = V^T z = V^T A^T y = (AV)^T y = \sum_{i=1}^m y_i ((AV)^T)_i. \tag{2.6}$$

Since $y \in \mathbb{R}^m$ is a nonzero vector, the rows of AV are linearly dependent. □

We now see the result of the full two-step facial reduction process, i.e., we get a constraint matrix of full row rank:

$$\mathcal{F} = \{x \in \mathbb{R}^n_+ : Ax = b\} = \{x = Vv \in \mathbb{R}^n : P_{\bar m} AVv = P_{\bar m} b, \ v \in \mathbb{R}^{n-s_z}_+\}, \tag{2.7}$$

where $P_{\bar m} : \mathbb{R}^m \to \mathbb{R}^{\bar m}$, $\bar m = rank(AV)$, is the simple projection that chooses the linearly independent rows of AV. This concludes the second step of **FR**, i.e., guaranteeing the full rank. We include a graphical illustration of the two-step **FR** process; see Fig. 2.1.

For a general conic problem, such as semidefinite programs (**SDP**), the facial reduction iterations do not necessarily end in one iteration; see Cheung et al. (2013); Sremac (2019); Sremac et al. (2021). And there is a special name for the minimum length of **FR** iterations.

Definition 2.4 (Sturm (2000, Sect. 4))**.** Given a spectrahedron $\mathcal{S}$ in a closed convex cone $\mathcal{K}$, the *singularity degree, $SD(\mathcal{S})$* of $\mathcal{S}$ is the smallest number of facial reduction iterations for finding $face(\mathcal{S}, \mathcal{K})$, the minimal face of $\mathcal{K}$ containing $\mathcal{S}$.

It is known that **FR** for **LP**s can be done in one iteration, i.e., $SD(\mathcal{F}) \le 1$; see Drusvyatskiy & Wolkowicz (2017, Theorem 4.4.1). Proposition 2.2 and Lemma 2.3 imply that *any* solution to the system (2.3) gives rise to a strict reduction in the number of variables and the number of equality constraints. This gives rise to the following two novel notions of singularity.

Definition 2.5. Let $K \subseteq \mathbb{R}^n$ be a closed convex cone with corresponding feasible set $\mathcal{S} = \{x \in K : Ax = b\}$ and facially reduced feasible set $\{v \in PK : (PA\mathcal{V})(v) = Pb, v \in \mathbb{R}^r\}$, where $PA\mathcal{V}$ is onto $\mathbb{R}^{m_r}$ and PK is the cone defined over the smaller dimensional space. Then the *implicit problem singularity, $IPS(\mathcal{S}) = m - m_r$*. Moreover, the *max-singularity degree* of $\mathcal{S}$, denoted $maxSD(\mathcal{S})$, is the largest number of nontrivial facial reduction iterations for finding $face(\mathcal{S}, K)$.

The singularity degree is used in Sturm (2000, Sect. 4) for providing a Hölder regularity constant for semidefinite programs. This is then used in Drusvyatskiy et al. (2017) to derive a convergence rate for alternating projection methods for semidefinite programs. Note that $maxSD(\mathcal{S})$ can be a larger lower bound of $IPS(\mathcal{S})$ than $SD(\mathcal{S})$, since at least one linear constraint becomes redundant at each **FR** iteration. The effect on ill-conditioning of larger values of IPS is seen empirically in Section 4.1.5.[4]

2.2.1. Preprocessing in LP

An essential step for simplex and interior point methods is preprocessing, see e.g., Andersen & Andersen (1995); Gondzio (1997); Huang (2004); Mészáros & Suhl (2003) and the references therein. One specific preprocessing step refers to detecting a *fixed variable*. These are generally detected when the upper and lower bounds on a variable are equal. Fixed variables can also be detected when an invertible block A_{11} can be isolated $A = \begin{bmatrix} A_{11} & A_{12} = 0 \\ A_{21} & A_{22} \end{bmatrix}$, $b = \begin{pmatrix} b_1 \\ b_2 \end{pmatrix}$. With $x = \begin{pmatrix} x_1 \\ x_2 \end{pmatrix}$, we can eliminate $x_1 = A_{11}^{-1} b_1$ and discard the first block of now redundant rows, along with the first block of columns. If $b_1 = 0$ then we have trivially identified variables fixed at zero and removed redundant rows and columns. The remaining block A_{22} remains full row rank as happens in Gaussian elimination.

In general, **FR** for linear programs refers to identifying variables fixed at 0, and removing them along with corresponding columns and redundant rows. In general, this is not as simple as above, and the theorem of the alternative is needed. As a consequence of our main result, we see below that a single step of the simplex

[3] Note that this can be done in one step for linear programs, i.e., the singularity degree for **LP** is at most one; see Drusvyatskiy & Wolkowicz (2017, Section 4.4).

[4] Definition 2.5 can be used to strengthen the upper bound on the rank of **SDP** solutions in Im & Wolkowicz (2021), i.e., we get $t(r) \le m - IPS(\mathcal{S}) \le m - maxSD(\mathcal{S}) \le m - SD(\mathcal{S}) \le m$, where $t(r)$ is the triangular number of the rank r.

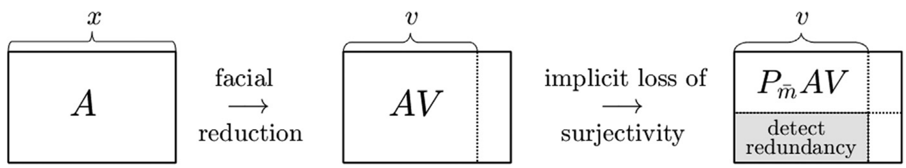

Fig. 2.1. A graphical illustration of the two-step facial reduction.

method, a phase-I part B approach, yields many of these variables that are identically zero on the feasible set.

One of the standard assumptions in linear programming is full row rank of A. As we observed in Lemma 2.3, each **FR** step results in linear dependence of the constraints. We now summarize two available methods for extracting a maximal linearly independent subset of rows of AV. The first method uses a rank-revealing QR decomposition[5]. Let $M = (AV)^T$. Let $MI(:,\pi) = QR$ be a QR factorization where π is a permutation vector, Q is a orthogonal matrix and R is an upper triangular matrix with a non-increasing diagonal in absolute value. The matrix $I(:,\pi)$ permutes the columns of M. If M has linearly dependent columns, then the matrix R contains zeros on its diagonal. Let r be the number of the nonzero diagonal entries of R. Then, $\pi(1:r)$ returns the subset of columns indices of M that are linearly independent. Another available method makes use of artificial variables Chvátal (1983, Box 8.2). It constructs $\begin{bmatrix} I & AV \end{bmatrix}$ and sets the initial basis matrix to be the first m columns. Then it performs a variant of the phase-I of the two-phase simplex method to drive the basic variables out of the basis one by one. When such an operation is not applicable, a linearly dependent row of AV is detected. Computational improvements of this method are made in Andersen (1995); Mészáros & Suhl (2003). A more recent method is the rank revealing Gaussian elimination by the maximum volume concept given in Schork & Gondzio (2020).

3. Main result and consequences

In this section we present our main result, see Theorem 3.1. We provide two proofs: one takes an algebraic approach by using the definition of the basic feasible solution; and the other takes a geometric approach by using extreme points. Both proofs rely heavily on Lemma 2.3. In Section 3.2 we present an efficient preprocessing scheme that can be used as an extension of the phase-I of the two-phase simplex method. In Section 3.3 we include immediate corollaries of the main result and interesting discussions.

3.1. Lack of strict feasibility and relations to degeneracy

Theorem 3.1. *Suppose that strict feasibility fails for $\mathcal{F}$. Then every basic feasible solution to $\mathcal{F}$ is degenerate.*

3.1.1. An algebraic proof of Theorem 3.1 via the definition of BFS

Proof. Since there is no strictly feasible point in $\mathcal{F}$, there exists a facial range vector V, and as in Eq. (2.5) we have

$$\mathcal{F} = \{ x = Vv \in \mathbb{R}^n \ : \ AVv = b, \ v \in \mathbb{R}^{n-s_z}_+ \}.$$

By Lemma 2.3, AV has at least one redundant row. By permuting the columns of A, we may assume that the matrix V is of the form

$$V = \begin{bmatrix} I_r \\ 0 \end{bmatrix} \text{ and } r = n - s_z.$$

We partition the index set $\{1,\dots,n\}$ as

$$\{1,\dots,n\} = \mathcal{I}_+ \cup \mathcal{I}_0, \text{ where } \mathcal{I}_+ = \{1,\dots,r\} \text{ and } \mathcal{I}_0 = \{r+1,\dots,n\}.$$

Then we have $A = \begin{bmatrix} A_{\mathcal{I}_+} & A_{\mathcal{I}_0} \end{bmatrix}$. Let $\bar{x} \in \mathcal{F}$ be a basic feasible solution with basic indices

$$\mathcal{B} \subset \{1,\dots,n\}, \ |\mathcal{B}| = m, \ \det(A_{\mathcal{B}}) \neq 0, \ \text{ and } \ A_{\mathcal{B}}\bar{x}(\mathcal{B}) = b.$$

Suppose $\mathcal{B} \subseteq \mathcal{I}_+$. We note, by Lemma 2.3 again, that $A_{\mathcal{I}_+} = AV$ has linearly dependent rows, i.e., $rank(A_{\mathcal{I}_+}) < m$. Hence $\bar{x}$ must include a basic variable in $\mathcal{I}_0$ and this concludes that every basic feasible solution is degenerate. □

3.1.2. A geometric proof using extreme points

We now give the second proof of our main result. Suppose that $X \in F$ with $rank(X) = r$, where F is a face of the set $\{X \in \mathbb{S}^n_+ : trace(A_i X) = b_i, \forall i = 1,\dots,m\}$. Here, $\mathbb{S}^n_+$ denotes the set of n-by-n positive semidefinite matrices. It is known that $\frac{r(r+1)}{2} \le m + \dim F$, see Pataki (1998, Theorem 2.1). We rewrite Pataki (1998, Theorem 2.1) in the language of polyhederon in Corollary 3.2. We include the proof for completeness in Section A.1.

Corollary 3.2 (Pataki (1998, Theorem 2.1))**.** *Suppose that $x \in F$, where F is a face of the set $\mathcal{F}$. Let $d = \dim F$. Then the number of nonzero entries of $x \in F$ is at most $m + d$.*

A point x in a convex set $\mathcal{C}$ is called an *extreme point* if, for all $y, z \in \mathcal{C}$, $x = \frac{1}{2}(y+z)$ implies $x = y = z$. An extreme point is itself a face and the dimension of this face is 0. Hence, we obtain Corollary 3.3 by writing Corollary 3.2 through the lens of extreme points.

Corollary 3.3. *Every extreme point $x \in \mathcal{F}$ has at most m positive entries.*

We now restate the main result of this paper Theorem 3.1 in the language of extreme points and number of rows of A.

Theorem 3.4. *Suppose that strict feasibility of $\mathcal{F}$ fails. Then every extreme point $x \in \mathcal{F}$ has at most $m-1$ positive entries.*

Proof. Since strict feasibility fails for $\mathcal{F}$, we have $\mathcal{F} = \{x = Vv \in \mathbb{R}^n : AVv = b, \ v \in \mathbb{R}^{n-s_z}_+\}$; see Eq. (2.5). From Lemma 2.3, we note that at least one equality in $AVv = b$ is redundant. Let $P_{\bar{m}}AVv = P_{\bar{m}}b$ be the system obtained after removing redundant rows of AV; see Eq. (2.7). Then, by Corollary 3.3, every extreme point of the set $\{v \in \mathbb{R}^{n-s_z}_+ : P_{\bar{m}}AVv = P_{\bar{m}}b\}$ has at most $m-1$ nonzero entries. Hence, the statement follows. □

3.1.3. Immediate consequences of main result

We first note that Theorems 3.1 and 3.4 are equivalent owing to the well-known characterization:

$$x \in \mathcal{F} \text{ is a basic feasible solution } \iff x \in \mathcal{F} \text{ is an extreme point.}$$

We now highlight that Theorems 3.1 and 3.4 do not merely imply the existence of a *single* degenerate basic feasible solution; but rather that *every* basic feasible solution is degenerate. Developing a pivot rule that prevents the simplex method from visiting degenerate points is not possible as it can never avoid degeneracies when strict feasibility fails, as we now illustrate in the following.

Example 3.5. Consider $\mathcal{F}$ with the data

$$A = \begin{bmatrix} 1 & 1 & 3 & 5 & 2 \\ 0 & 1 & 2 & -2 & 2 \end{bmatrix} \text{ and } b = \begin{pmatrix} 1 \\ 1 \end{pmatrix}.$$

[5] https://www.mathworks.com/matlabcentral/fileexchange/77437

Consider the vector $y = \begin{pmatrix} 1 \\ -1 \end{pmatrix}$. Then

$$A^T y = \begin{pmatrix} 1 & 0 & 1 & 7 & 0 \end{pmatrix}^T \text{ and } b^T y = 0.$$

Hence, Proposition 2.2 certifies that $\mathcal{F}$ does not contain a strictly feasible point. There are exactly six feasible bases in $\mathcal{F}$. The **BFS** associated with $\mathcal{B} \in \{\{1,2\},\{2,3\},\{2,4\}\}$ is $x = \begin{pmatrix} 0 & 1 & 0 & 0 & 0 \end{pmatrix}^T$; and the **BFS** associated with $\mathcal{B} \in \{\{1,5\},\{3,5\},\{4,5\}\}$ is $x = \begin{pmatrix} 0 & 0 & 0 & 0 & \frac{1}{2} \end{pmatrix}^T$. Clearly, all **BFS**s are degenerate.

Recall that strict feasibility is equivalent to the Mangasarian–Fromovitz constraint qualification, Peterson (1973). The latter is equivalent to stability with respect to perturbations of b, and to a compact dual optimal set. Therefore, the following Corollary 3.6, obtained by writing the contrapositive of Theorem 3.1, is extremely interesting and important. We provide Example 3.7 below to illustrate Corollary 3.6.

Corollary 3.6. *Suppose that there exists a nondegenerate basic feasible solution. Then there exists a strictly feasible point $\hat{x} \in \mathcal{F}$.*

Example 3.7. Consider $\mathcal{F}$ with the data

$$A = \begin{bmatrix} 1 & 0 & -2 & 3 & -4 \\ 0 & -1 & -2 & 3 & 1 \end{bmatrix} \text{ and } b = \begin{pmatrix} 1 \\ 1 \end{pmatrix}.$$

The system $\mathcal{F}$ has exactly four feasible bases; the **BFS** associated with $\mathcal{B} \in \{\{1,4\},\{2,4\},\{4,5\}\}$ is $x = \begin{pmatrix} 0 & 0 & 0 & 1/3 & 0 \end{pmatrix}^T$ and the **BFS** associated with $\mathcal{B} = \{1,5\}$ is $x = \begin{pmatrix} 5 & 0 & 0 & 0 & 1 \end{pmatrix}^T$. We note that the **BFS** associated with $\mathcal{B} = \{1,5\}$ is nondegenerate. As Corollary 3.6 states, the system $\mathcal{F}$ has a strictly feasible point, and it is verified by the point $\frac{1}{10}\begin{pmatrix} 4 & 1 & 1 & 4 & 1 \end{pmatrix}^T$.

Corollary 3.6 provides a useful check for strict feasibility when the simplex method is used, i.e., if there is any simplex iteration that yields a nondegenerate **BFS**, then it is useful to record that occurrence. We emphasize that recording the occurrence of a nondegenerate iteration is inexpensive and the occurrence gives a certificate of the stability of the **LP** instance. We revisit Corollary 3.6 in Section 3.2.1 below and present an efficient algorithm for obtaining a Slater point from a nongenerate **BFS**. But, Example 3.8 below shows that the converse of Theorems 3.1 and 3.4 is not true. In other words, strict feasibility holds and every **BFS** is degenerate.

Example 3.8.

1. Consider $\mathcal{F}$ with the data

$$A = \begin{bmatrix} 1 & 0 & 2 & 0 & -2 \\ 1 & -3 & 2 & 1 & -2 \end{bmatrix} \text{ and } b = \begin{pmatrix} 1 \\ 1 \end{pmatrix}.$$

   $\mathcal{F}$ has exactly four feasible bases and all of them are degenerate; the **BFS** associated with $\mathcal{B} \in \{\{1,2\},\{1,4\}\}$ is $x = \begin{pmatrix} 1 & 0 & 0 & 0 & 0 \end{pmatrix}^T$ and the **BFS** associated with $\mathcal{B} \in \{\{2,3\},\{3,4\}\}$ is $x = \begin{pmatrix} 0 & 0 & 1/2 & 0 & 0 \end{pmatrix}^T$. However, $\mathcal{F}$ contains a strictly feasible point $\frac{1}{10}\begin{pmatrix} 1 & 1 & 5.5 & 3 & 1 \end{pmatrix}^T$.
2. Note that the linear assignment problem (marriage problem) has a strictly feasible point but all the **BFS** are highly degenerate.[6] Therefore, $\mathcal{I}_0 = \emptyset$; the set of variables fixed at 0 is empty.

   Moreover, as an **LP**, the problem is stable with respect to perturbations in the data.

From Examples 3.5 to 3.8, we observe that there are two different types of degeneracies. One involves variables that are 0 in one **BFS** but positive in another; the second involves variables fixed at 0, i.e., that result in strict feasibility failing. Note that strict feasibility (along with A full row rank) is the Mangasarian–Fromovitz constraint qualification which is equivalent to stability with respect to right-hand side perturbations (Gauvin, 1995), which is in turn equivalent to a bounded dual optimal set.

Given a **BFS** $\bar{x} \in \mathcal{F}$, we let the *degree of degeneracy* of $\bar{x}$ denote the number of 0's among its basic variables. By exploiting the facially reduced model we can check how degenerate the **BFS**s of $\mathcal{F}$ are.

Corollary 3.9. *Suppose that strict feasibility fails for $\mathcal{F}$, and let $\mathcal{F}$ have the facial range vector representation in Eq. (2.5). Recall that the set of indices $\mathcal{I}_0 = \{i \in \{1,\dots,n\} : x_i = 0, \ \forall x \in \mathcal{F}\}$. Let $\bar{x} \in \mathcal{F}$ be a basic feasible solution with basis $\mathcal{B}$. Then, the following holds.*

1. *The basis $\mathcal{B}$ has an nonempty intersection with $\mathcal{I}_0$, i.e., $\mathcal{B} \cap \mathcal{I}_0 \neq \emptyset$.*
2. *If the degree of degeneracy of $\bar{x}$ is exactly one, with $\bar{x}_k = 0, k \in \mathcal{B}$, then $x_k, A_{:,k}$ can be discarded from the problem.*
3. *The degree of degeneracy of $\bar{x}$ is at least $m - rank(AV)$.*
4. *At least $m - rank(AV)$ number of basic indices of $\bar{x}$ are contained in $\mathcal{I}_0$.*

Proof.

1. Let $\bar{x} \in \mathcal{F}$ be a basic feasible solution and let $\mathcal{B}$ be a basis for $\bar{x}$. Item 1 follows from the proof and the definition of the set $\mathcal{I}_0$ of elements x_i that are identically zero on the feasible set.
2. The proof follows from the algebraic proof of Theorem 3.1 given in Section 3.1.1. Since every **BFS** is degenerate and the basis has a nonempty intersection with $\mathcal{I}_0$, the index k must be in $\mathcal{I}_0$.
3. For Item 3, we note that $A_\mathcal{B}$ contains linearly independent columns. Then $A_\mathcal{B}$ can contain at most $rank(AV)$ number of columns from AV. Thus, $\bar{x}(\mathcal{B})$ must contain at least $m - rank(AV)$ number of zeros.
4. Item 4 is a direct consequence of Item 1 and Item 3. □

Items 3 and 4 of Corollary 3.9 are closely related to the implicit problem singularity, *IPS*, and the max-singularity degree, *maxSD*; see Definition 2.5. In particular, $IPS(\mathcal{F})$ is a lower bound of the degree of degeneracy of every **BFS** of $\mathcal{F}$; the more implicit redundancies $\mathcal{F}$ contains, the more degenerate every **BFS** becomes. We include an alternative way to view Corollary 3.9 in Section 3.1.2.

We conclude the discussions with the following interesting observation. This again illustrates the implicit singularity of the constraints when the Slater condition fails.

Corollary 3.10. *Suppose that strict feasibility fails for $\mathcal{F}$ and that $m = 1$. Then the trivial $x^* = 0$ is an optimal solution.*

3.2. Preprocessing for facial reduction and strict feasibility

In this section we present a preprocessing method for obtaining a facially reduced system. In Section 3.2.1 we discuss obtaining a strictly feasible point using a nondegeneate **BFS** and its variant. In Section 3.2.2 we consider the general case of finding an exposing vector to obtain the facially reduced strictly feasible **LP**.

3.2.1. Towards a strictly feasible point from a nondegenerate BFS

By Corollary 3.6, the existence[7] of a nondegenerate **BFS** guarantees the existence of a strictly feasible point. We now propose a

[6] Note that this is true for the transportation and the assignment problems. Both are highly degenerate at each **BFS** but satisfy strict feasibility. For example, for the assignment problem order n, the feasible set can be considered to be the doubly stochastic matrices X. The extreme points are the permutation matrices by the Birkoff-Von Neumann theorem. Therefore, each extreme point has exactly n positive elements while there are $m = 2n - 1$ linearly independent constraints.

[7] Determining the existence of a degenerate basic feasible solution is an NP-complete problem; see Chandrasekaran et al. (1981/82).

process for acquiring a Slater point from a nondegenerate **BFS**, and include a generalization. The arguments in this section also provide a constructive proof of Corollary 3.6.

Let $\bar{x} \in \mathcal{F}$ be a nondegenerate **BFS**. Without loss of generality, we assume that the (all positive) basic variables $\bar{x}_\mathcal{B}$ of $\bar{x}$ are located at the last m entries of $\bar{x}$. We fix a scalar $\hat{\gamma} \in (0,1)$ and an index $j \in \{1,\dots,n-m\}$. For some $\alpha \ge 0$, we consider the simplex method *ratio test* type inequality

$$\hat{\gamma}\bar{x}_\mathcal{B} - \alpha (A_\mathcal{B})^{-1}A_j \ge 0. \tag{3.1}$$

Since $\bar{x}_\mathcal{B} > 0, \hat{\gamma} > 0$, there exists a *positive*α that maintains the inequality Eq. (3.1). Let

$$\alpha^* = \min\left\{1,\ \max\{\alpha \in \mathbb{R}_+ : \hat{\gamma}\bar{x}_\mathcal{B} - \alpha (A_\mathcal{B})^{-1}A_j \ge 0\}\right\}, \tag{3.2}$$

and decompose

$$\hat{\gamma}\bar{x}_\mathcal{B} = \left(\hat{\gamma}\bar{x}_\mathcal{B} - \alpha^*(A_\mathcal{B})^{-1}A_j\right) + \alpha^*(A_\mathcal{B})^{-1}A_j.$$

We observe that

$$\begin{aligned} b &= A_\mathcal{B}\bar{x}_\mathcal{B} \\ &= (1-\hat{\gamma})A_\mathcal{B}\bar{x}_\mathcal{B} + \hat{\gamma}A_\mathcal{B}\bar{x}_\mathcal{B} \\ &= (1-\hat{\gamma})A_\mathcal{B}\bar{x}_\mathcal{B} + A_\mathcal{B}\left(\hat{\gamma}\bar{x}_\mathcal{B} - \alpha^*(A_\mathcal{B})^{-1}A_j + \alpha^*(A_\mathcal{B})^{-1}A_j\right) \\ &= A_\mathcal{B}(\bar{x}_\mathcal{B} - \alpha^*(A_\mathcal{B})^{-1}A_j) + \alpha^* A_j. \end{aligned}$$

If we set $x_j = \alpha^* > 0$ and replace $\bar{x}_\mathcal{B}$ by $\bar{x}_\mathcal{B} - \alpha^*(A_\mathcal{B})^{-1}A_j$, then we have increased the cardinality of the positive entries of a solution. We note that $\bar{x}_\mathcal{B} - \alpha^*(A_\mathcal{B})^{-1}A_j$ only has strictly positive entries since it it a sum of a positive vector and a nonnegative vector;

$$\bar{x}_\mathcal{B} - \alpha^*(A_\mathcal{B})^{-1}A_j = \underbrace{(1-\hat{\gamma})\bar{x}_\mathcal{B}}_{\text{positive}} + \underbrace{\hat{\gamma}\bar{x}_\mathcal{B} - \alpha^*(A_\mathcal{B})^{-1}A_j}_{\text{nonnegative}}.$$

We can continue to increase the number of positive entries of a solution one by one for each $j \in \{1,\dots,n-m\}$. Moreover, we can achieve this by a compact vectorized operation. The main idea is that we can choose $\hat{\gamma}$ in Eq. (3.1) independently for each $j \in \{1,\dots,n-m\}$. Let γ_j be a positive real number such that $0 < \gamma := \sum_{j=1}^{n-m}\gamma_j < 1$. Then, we have

$$\bar{x}_\mathcal{B} = (1-\gamma)\bar{x}_\mathcal{B} + \gamma\bar{x}_\mathcal{B} = (1-\gamma)\bar{x}_\mathcal{B} + \sum_{j=1}^{n-m}\gamma_j\bar{x}_\mathcal{B}.$$

We set an auxiliary matrix

$$\Theta = \begin{bmatrix}\gamma_1\bar{x}_\mathcal{B} & \dots & \gamma_{n-m}\bar{x}_\mathcal{B}\end{bmatrix} - (A_\mathcal{B})^{-1}A_{1:n-m} \in \mathbb{R}^{m\times(n-m)}$$

and perform Eq. (3.2) on each column j of Θ to obtain the vector θ^*:

$$\theta_j^* := \begin{cases}\max(\Theta(:,j)) & \text{if } \max(\Theta(:,j)) \le 1, \\ 1 & \text{otherwise.}\end{cases}$$

Then the point

$$\begin{bmatrix}\theta^* \\ \bar{x}_\mathcal{B} - (A_\mathcal{B})^{-1}A_{1:n-m}\theta^*\end{bmatrix}$$

is a strictly feasible point to $\mathcal{F}$. Hence, this operation provides a constructive proof of Corollary 3.6.

We now extend the aforementioned procedure for obtaining a strictly feasible point using any feasible solution $\bar{x} \in \mathcal{F}$ such that $A_{supp(\bar{x})}$ is full row rank. We partition $\bar{x} \in \mathcal{F}$ as follows

$$\bar{x} = \begin{pmatrix}\bar{x}_{\mathcal{B}_1} \\ \bar{x}_{\mathcal{B}_2} \\ \bar{x}_\mathcal{N}\end{pmatrix},\ \text{ where } supp(\bar{x}) = \mathcal{B}_1 \cup \mathcal{B}_2,\ rank(A_{\mathcal{B}_1}) = m, \ \text{ and}$$
$$\mathcal{N} = \{1,\dots,n\}\setminus supp(\bar{x}). \tag{3.3}$$

We partition A using the same partition $\mathcal{B}_1 \cup \mathcal{B}_2 \cup \mathcal{N}$:

$$\begin{bmatrix}A_{\mathcal{B}_1} & A_{\mathcal{B}_2} & A_\mathcal{N}\end{bmatrix}\bar{x} = b \iff \begin{bmatrix}A_{\mathcal{B}_1} & A_\mathcal{N}\end{bmatrix}\begin{pmatrix}\bar{x}_{\mathcal{B}_1} \\ \bar{x}_\mathcal{N}\end{pmatrix} = \bar{b} := b - A_{\mathcal{B}_2}x_{\mathcal{B}_2}.$$

Then we can apply the aforementioned procedure to the system

$$\begin{bmatrix}A_{\mathcal{B}_1} & A_\mathcal{N}\end{bmatrix}\begin{pmatrix}\bar{x}_{\mathcal{B}_1} \\ \bar{x}_\mathcal{N}\end{pmatrix} = \bar{b}$$

and distribute positive weights to $\bar{x}_\mathcal{N}$ using $\bar{x}_{\mathcal{B}_1}$. Finally, we find a strictly feasible point to $\mathcal{F}$. This process is summarized in Algorithm 3.1. Furthermore, Algorithm 3.1 provides a constructive proof for Proposition 3.11 below.

Algorithm 3.1 Compute a slater point.

Require: Given: $A,\ \bar{x} \in \mathcal{F}$ partitioned as in (3.3).
1: Choose any $\gamma \in \mathbb{R}^{|\mathcal{N}|}_{++}$ such that $\sum_{j=1}^{|\mathcal{N}|}\gamma_j < 1$.
2: Compute
$$\Theta = \begin{bmatrix}\bar{x}_{\mathcal{B}_1} & \dots & \bar{x}_{\mathcal{B}_1}\end{bmatrix}\mathrm{Diag}(\gamma) - A_{\mathcal{B}_1}^{-1}A_\mathcal{N}.$$
3: Compute $\theta^* \in \mathbb{R}^{|\mathcal{N}|}_{++}$, where for each $j \in \{1,\dots,|\mathcal{N}|\}$,
$$\theta_j^* := \begin{cases}\max(\Theta(:,j)) & \text{if } \max(\Theta(:,j)) \le 1, \\ 1 & \text{otherwise.}\end{cases}$$
4: Set $x^\circ = \begin{pmatrix}\bar{x}_{\mathcal{B}_1} - (A_{\mathcal{B}_1})^{-1}A_\mathcal{N}\theta^* \\ \bar{x}_{\mathcal{B}_2} \\ \theta^*\end{pmatrix}$.

Proposition 3.11. *Let $x \in \mathcal{F}$ be a solution such that $rank\left(A_{supp(x)}\right) = m$. Then, $\mathcal{F}$ has a strictly feasible point.*

3.2.2. Exposing vector; phase I part B; strict feasibility testing

We now present an efficient preprocessing procedure for detecting identically 0 variables and obtaining exposing vectors in order to get the facially reduced **LP**. We do this for a given **BFS** $\bar{x}$ by solving special subproblems using the simplex method. By the end of the process, we determine one of:

1. a certificate y that produces an exposing vector A^Ty (Slater condition fails);
2. a strictly feasible point (Slater condition holds).

This process in fact has *two* applications. First, since the only requirement of this process is the **BFS**, the procedure can be considered as an extension of phase-I of the two-phase simplex method that obtains the equivalent facially reduced problem. Second, the procedure can be used as a postprocessing step. We could perform **FR** on the optimal face and find, and delete, variables fixed at zero in order to improve stability of the optimal solution.

We now describe the proposed preprocessing method. Let $\mathcal{B}$ be a degenerate initial basis of $\mathcal{F}$ with associated **BFS** $\bar{x}$. Without loss of generality, we assume that basic variables are located at the first m entries of $\bar{x}$. Let d be the degree of degeneracy of $\bar{x}$. We further assume that the degenerate basic variables are located at the first d entries of $\bar{x}$. We let $\mathcal{B}_0 := \{1,\dots,d\}$. We now test and record whether or not each $i \in \mathcal{B}_0$ is a variable fixed at 0. Let $i \in \mathcal{B}_0$, and consider the following problem:

$$p_i^* = \max\{x_i : Ax = b,\ x \ge 0\}. \tag{3.4}$$

We may assume that $i = 1$. We solve Eq. (3.4) using the simplex method from the initial **BFS** $\bar{x}$. That is, we do not need to perform the typical phase-I of the two-phase simplex method in order to find a feasible **BFS**. The optimal value p_1^* of Eq. (3.4) is clearly lower bounded by 0. We consider two cases below:

1. Suppose that $x_1 > 0$ after k iterations. Then, the variable x_1 is not an identically 0 variable, i.e., we record that $1 \in \mathcal{I}_+$.
2. Suppose that $p_1^* = 0$. Then, the variable x_1 is an identically 0 variable, i.e., we record that $1 \in \mathcal{I}_0$. Let $\mathcal{B}^*$ be an optimal basis for Eq. (3.4). Then we have

$$y^* = A_{\mathcal{B}^*}^{-T} e_1, \ \langle b, y^* \rangle = 0 \ \text{ and } A^T y^* \geq e_1, \tag{3.5}$$

where e_1 is the first unit vector of appropriate dimension. We note that the dual optimal solution y^* in Eq. (3.5) produces a solution to the auxiliary system Eq. (2.3). Therefore, we obtain a *nontrivial* exposing vector since $0 \neq A^T y^* \geq 0$.

Let $\{y^j\}$ be a collection of the certificates that are obtained from solving Eq. (3.4) with the index 1 replaced by $i \in \mathcal{B}_0$. Then $y^\circ = \sum_j y^j$ is also a certificate, i.e.,

$$A^T y^\circ = \sum_j A^T y^j \geq 0, \ A^T y^\circ \neq 0, \quad \text{and} \quad \langle b, y^\circ \rangle = \sum_j \langle b, y^j \rangle = 0,$$

and we obtain a nontrivial exposing vector $A^T y^\circ$ for the system $\mathcal{F}$. By summarizing the two cases above, we obtain an efficient preprocessing method Algorithm 3.2.

Algorithm 3.2 Preprocessing phase I part B; towards strict feasibility.

Require: A **BFS** $\bar{x}$ with corresponding basis $\mathcal{B}$; set $\mathcal{B}_0 = \{i \in \mathcal{B} : \bar{x}_i = 0\}$
Initialize: $x^\circ = \bar{x}$, $y^\circ = 0 \in \mathbb{R}^m$, $\mathcal{J}_0 = \emptyset$, $\mathcal{B}_* \leftarrow \mathcal{B}_0$
while $\mathcal{B}_0 \neq \emptyset$ and $\mathcal{B}_* \neq \emptyset$ **do**
  Pick $i \in \mathcal{B}_0$; starting from the initial **BFS** $\bar{x}$, solve for primal-dual optima x^*, y^*

$$x^* = \operatorname{argmax}_x \{x_i : Ax = b, x \geq 0\}, \quad p^* = x_i^* = b^T y^*$$

  But, if during the solve, $x_i > 0$, then stop the iterations; set x^* as the current point.
  $\mathcal{S} \leftarrow \operatorname{supp}(x^*)$
  $\mathcal{B}_* \leftarrow$ degenerate basic indices for x^*
  if $\mathcal{B}_0 \neq \emptyset$ and $\mathcal{B}_* \neq \emptyset$ **then**
    if $p^* = 0$ (strict feasibility fails) **then**
      Use dual certificate y^* to satisfy (2.3)
      $y^\circ \leftarrow y^\circ + y^*$
      $\mathcal{J}_0 \leftarrow \mathcal{J}_0 \cup (\operatorname{supp}(A^T y^*) \cap \mathcal{B})$
      $\mathcal{B}_0 \leftarrow \mathcal{B}_0 \setminus \{\mathcal{S} \cup \mathcal{J}_0\}$
    else
      $\mathcal{B}_0 \leftarrow \mathcal{B}_0 \setminus \mathcal{S}$
    end if
    Choose $\gamma \in (0,1)$ and set $x^\circ \leftarrow \gamma x^\circ + (1-\gamma) x^*$
  end if
end while
if $\mathcal{J}_0 \neq \emptyset$ **then**
  $z^\circ = A^T y^\circ$ (exposing vector)
  $\mathcal{R} \leftarrow$ redundant row indices of $A(:, \operatorname{supp}(z^\circ)^c)$
  $A \leftarrow A(\mathcal{R}^c, \operatorname{supp}(z^\circ)^c)$, $b \leftarrow b(\mathcal{R}^c)$
else
  Run Algorithm 3.1 with x° and $\det(A_{\mathcal{B}}) \neq 0$ (use x^* and $\mathcal{B}_*$, if $\mathcal{B}_* = \emptyset$)
end if

The following allows for simplifications in Algorithm 3.2.

Lemma 3.12. *Let $\mathcal{B}$ be an initial basis containing the index i for problem (3.4). Then the index i always remains in the basis throughout the iterations.*

Proof. Without loss of generality, we let $i = 1$. We argue that 1 is not chosen to leave the basis. Let $y^* = (A_{\mathcal{B}}^T)^{-1} c_{\mathcal{B}}$ and $\bar{A} = A_{\mathcal{B}}^{-1} A$. Suppose that the reduced cost at the index j is positive. Then

$$0 < \bar{c}_j = c_j - A_j^T y^* = -A_j^T y^* = -A_j^T (A_{\mathcal{B}}^T)^{-1} e_1 = -\bar{A}_{1j}.$$

Since $\bar{A}_{1j} < 0$, the index 1 is not chosen to leave the basis $\mathcal{B}$. □

The following special case is of interest. Namely, no simplex pivoting steps are required to determine strict feasibility.

Theorem 3.13 (preprocessing for degree of degeneracy 1)**.** *Given a basis $\mathcal{B}$, let $\bar{x}$ be a **BFS** with the degree of degeneracy exactly one and with $\bar{x}_i = 0, i \in \mathcal{B}$. Let $\mathcal{N} = \{1, \ldots, n\} \setminus \mathcal{B}$ and let $\bar{y} = (A_{\mathcal{B}}^T)^{-1} c_{\mathcal{B}}, c_{\mathcal{B}} = e_i$. Then strict feasibility fails if, and only if, $\bar{y}$ satisfies $A_{\mathcal{N}}^T \bar{y} \geq 0$.*

Proof. Suppose that $\bar{x}$ is a degenerate **BFS** with basis $\mathcal{B}$. Without loss of generality, we assume $1 \in \mathcal{B}$ and 1 is the degenerate index. We consider the problem

$$p_1^* = \max\{x_1 \ : \ Ax = b, \ x \geq 0\}.$$

We note that $\langle b, \bar{y} \rangle = 0$ since $\langle b, \bar{y} \rangle$ is identical to the current objective value '0'. The backward direction is clear by Proposition 2.2. Now suppose that strict feasibility fails. Suppose to the contrary that $A_{\mathcal{N}}^T \bar{y} \geq 0$ fails. Then there exists j such that $A_j^T \bar{y} < 0, \ \ j \in \mathcal{N}$. Note that, by Lemma 3.12, that 1 is not chosen to leave the basis. Thus, there is an index $k \neq 1, k \in \mathcal{B}$ that leaves the basis. Since all other basic variables are positive, we obtain a positive step length and we improve the objective value, which yields a contradiction to $p_1^* = 0$. □

Upon the termination of Algorithm 3.2, we can always determine whether the system $\mathcal{F}$ has a strictly feasible point or not. Algorithm 3.2 terminates in a finite number of iterations since we remove at least one element from the set $\mathcal{B}_0$ in each iteration. We emphasize that we do not need to solve the auxiliary **LP**s for all $i \in \{1, \ldots, n\}$. We solve Eq. (3.4) only for the degenerate basic indices of the predetermined basis $\mathcal{B}$. However, upon termination of Algorithm 3.2, it is possible that we have not obtained $face(\mathcal{F}, \mathbb{R}^n_+)$, the minimal face containing $\mathcal{F}$. Although the complete **FR** for **LP** can be completed in one iteration, one step termination is possible only when we find a solution y of Eq. (2.3) so that $A^T y$ is in the relative interior of the conjugate face of $face(\mathcal{F}, \mathbb{R}^n_+)$. In this case, we can rerun Algorithm 3.2 with the current facially reduced system. For finding an initial basis for the second trial, we may use the efficient basis recovery scheme Wright (1996, Chapter 7).

One of the nice features of Algorithm 3.2 is that we do not need to search for a new initial basis $\mathcal{B}$ for each iteration; the initial basis remains the same. Therefore, our approach can be directly employed after the standard phase-I of the two phase simplex method.

We do not need a lot of pivoting steps to determine if p_i^* is zero or positive. If $p_i^* = 0$, the initial $\mathcal{B}$ is indeed a basis that gives the optimal value. However the dual feasibility may not be obtained immediately.[8] Thus, there may be additional pivots required to obtain the dual feasibility. However, since the optimal value is obtained at $\mathcal{B}$, we do not expect that the optimal basis search to be time-consuming. For the case $p_i^* \in (0, \infty)$, the optimal value p_i^* does not need to be found. Hence once a basis that gives a positive support on i is found, we can terminate the maximization problem in Algorithm 3.2 immediately. We recall from Lemma 3.12 that the index i in Eq. (3.4) never leaves the basis. In the case of $p_i^* = \infty$, we can perform the following operation. Let $\mathcal{B}_c$ be a basis that indicates $p_i^* = \infty$ and let j be an entering variable that indicates the unboundedness. Then by setting

$$x^\circ(j) \leftarrow 1, \ x^\circ(\mathcal{B}_c) \leftarrow x_{\mathcal{B}_c} - A_{\mathcal{B}_c}^{-1} A_j \text{ and } x^\circ((\{j\} \cup \mathcal{B}_c)^c) = 0,$$

we obtain a feasible solution x° that yields a positive objective value.

[8] If we have a nondegenerate initial basis, then the dual feasibility is immediately obtained. However, our initial basis is degenerate.

We often get an exposing vector that reveals more than one element in the set $\mathcal{I}_0$ by solving Eq. (3.4). Let $p_1^* = 0$ in Eq. (3.4) and let y^* be a dual feasible solution. Suppose that $A^T y^* = e_1$, i.e., only one exposed variable is revealed. Then $y^* \in null(A(:, 2:n)^T)$. Since the data matrix A has more columns than rows, $y^* \in null(A(:, 2:n)^T)$ generally implies $y^* = 0$; this makes $A^T y^* = e_1$ impossible.

When an instance is large and have a **BFS** with a very large degree of degeneracy, one may adopt parallel computing for Algorithm 3.2 in order to reduce the total computation time. We note again that the initial basis remains the same throughout the iterations. Hence, solving Eq. (3.4) for individual $i \in \mathcal{B}_0$ can be performed independently. In fact, parallel computing can be used to obtain a strictly feasible solution in Algorithm 3.1 as well; the weight vector γ can be chosen independently for each $j \in \mathcal{N}$.

3.3. Discussions

In this section we discuss the main result in Sections 3.1 and 3.2 and make connections to new results and known results in the literature.

3.3.1. Distance to infeasibility

The *distance to infeasibility* is a measure of the smallest perturbations of the data (A, b) of a problem that renders the problem infeasible. In our setting, we can use the following simplification of the distance to infeasibility from Renegar (1994) by restricting the perturbation to b, i.e., we can force infeasibility using only perturbation in b;

$$dist(b, \mathcal{F} = \emptyset) := \inf\left\{ \|b - \tilde{b}\| \ : \ \{x \in \mathbb{R}^n : Ax = \tilde{b}, \ x \geq 0\} = \emptyset \right\}.$$

Many interesting bounds, condition numbers, are shown in Renegar (1994) under the assumption that the distance to infeasibility is positive and known. It is known that a positive distance to infeasibility of $\mathcal{F}$ implies that strict feasibility holds for $\mathcal{F}$; see e.g., Freund & Ordonez (2005); Freund & Vera (1997). The contrapositive of this statement is that, if strict feasibility fails for $\mathcal{F}$, then the distance to infeasibility is 0. We revisit this statement with the facially reduced system Eq. (2.5). We provide an elementary proof that there is an arbitrarily small perturbation for the data vector b of $\mathcal{F}$ that yields the set $\mathcal{F}$ infeasible, i.e., $dist(b, \mathcal{F} = \emptyset) = 0$. Furthermore, we provide explicit perturbations that render the set $\mathcal{F}$ empty.

Suppose that $\mathcal{F}$ fails strict feasibility. Recall the representation Eq. (2.5) for $\mathcal{F}$. Let $AV = QR$ be a QR decomposition of AV, where $Q \in \mathbb{R}^{m\times m}$ orthogonal, $R \in \mathbb{R}^{m\times(n-s_z)}$ upper triangular. We write $Q = \begin{bmatrix} Q_1 & Q_2 \end{bmatrix}$ so that $range(Q_1) = range(AV)$. Then, by the orthogonality of Q, we have

$$Ax = AVv = b \iff Q^T Ax = Rv = Q^T b.$$

Since AV is a rank deficient matrix (see Lemma 2.3), the upper triangular matrix R is of the form

$$R = \begin{bmatrix} \bar{R} \\ 0 \end{bmatrix} \in \mathbb{R}^{m\times(n-s_z)} \text{ and } \bar{R} \in \mathbb{R}^{rank(AV)\times(n-s_z)} \text{ with nonzero diagonal.} \tag{3.6}$$

Since $b \in range(AV)$, the last $m - rank(AV)$ entries of $Q^T b$ are equal to 0, i.e.,

$$Q^T b = \begin{pmatrix} Q_1^T b \\ Q_2^T b \end{pmatrix} = \begin{pmatrix} Q_1^T b \\ 0 \end{pmatrix}.$$

Consequently, the unrealized implicit non-surjectivity produces the system

$$\begin{bmatrix} \bar{R} \\ 0 \end{bmatrix} v = \begin{pmatrix} Q_1^T b \\ 0 \end{pmatrix}, \quad v \in \mathbb{R}_+^{n-s_z}. \tag{3.7}$$

Any perturbation on the last $m - rank(AV)$ equations in Eq. (3.7) that causes the system inconsistency renders the system $\mathcal{F}$ infeasible while maintaining the dimension of $relint(\mathcal{F})$. For instance, replacing the right-hand side vector in Eq. (3.7) by $\begin{pmatrix} Q_1^T b \\ Q_2^T \xi \end{pmatrix}$, where $Q_2^T \xi \neq 0$, renders Eq. (3.7) and $\mathcal{F}$ infeasible.

We now present a class of perturbations of b that maintains the feasibility of the set $\mathcal{F}$ as well as a special perturbation of b that forces $\mathcal{F}$ to be infeasible. Such perturbations can be found using linear combinations of the columns of Q_1 or Q_2, respectively. We relate this observation to the solution of the auxiliary system Eq. (2.3) in the proof of Proposition 3.14 below.

Proposition 3.14. *Suppose that strict feasibility fails for $\mathcal{F}$, and let $\mathcal{F}$ have the representation Eq. (2.5). Then the following hold.*

1. *For all $\Delta b \in range(AV)$ with sufficiently small norm, the set $\{x \in \mathbb{R}_+^n : Ax = b + \Delta b\}$ is feasible.*
2. *Let $\bar{y} \in \mathbb{R}^m$ be a solution to the auxiliary system Eq. (2.3). Then perturbing the right-hand side vector b of $\mathcal{F}$ in the direction $\bar{y}$ makes the system $\mathcal{F}$ infeasible.*

Proof. Let Δb be any perturbation in $range(AV)$. Let $QR = AV$ be a QR decomposition of AV. In particular, let R have the form Eq. (3.6) and $Q = \begin{bmatrix} Q_1 & Q_2 \end{bmatrix}$ so that $range(Q_1) = range(AV)$. Let ϵ be a sufficiently small scalar. Then

$$\begin{aligned} Ax = AVv = b + \epsilon\Delta b &\iff Rv = Q^T b + \epsilon Q^T \Delta b \\ &\iff \bar{R}v = Q_1^T b + \epsilon Q_1^T \Delta b. \end{aligned} \tag{3.8}$$

The last equivalence holds since $Ax = b$ and $\Delta b \in range(AV) = range(Q_1)$. Since the system $\bar{R}v = Q_1^T b$ satisfies the Mangasarian-Fromovitz constraint qualification, the distance to infeasibility of this system is positive. Thus, the perturbed system $\{v : \bar{R}v = Q_1^T b + \epsilon Q_1^T \Delta b, \ v \geq 0\}$ remains feasible. Therefore, by Eq. (3.8), perturbing $\mathcal{F}$ along the direction $\Delta b \in range(AV)$ maintains the feasibility and this concludes the proof for Item 1.

For Item 2 we show that perturbing b with $\Delta b = \bar{y}$ renders $\mathcal{F}$ infeasible, where $\bar{y}$ is a solution to the system Eq. (2.3). By Proposition 2.2 and Eq. (2.6), the nonzero vector $\bar{y} \in \mathbb{R}^m$ is in $null((AV)^T)$. Then we have

$$\bar{y} \in range(AV)^\perp = range(Q_2) \Rightarrow \bar{y} = Q_2\bar{u} \text{ for some nonzero } \bar{u}.$$

We recall Farkas' lemma:

$$\{y \in \mathbb{R}^m : A^T y \geq 0, \ \langle b, y\rangle < 0\} \neq \emptyset \Rightarrow \mathcal{F} = \emptyset.$$

Now, for any $\epsilon > 0$, setting $\Delta b_\epsilon = -\epsilon\bar{y}$ yields

$$A^T\bar{y} \geq 0, \ \langle b, \bar{y}\rangle = 0 \Rightarrow A^T\bar{y} \geq 0, \ \langle b + \Delta b_\epsilon, \bar{y}\rangle < 0. \tag{3.9}$$

Hence, by letting $\epsilon \to 0^+$, we see that the distance to infeasibility, $dist(b, \mathcal{F} = \emptyset)$, is equal to 0. □

We emphasize that the result

$$\mathcal{F} \text{ fails strictly feasibility } \Rightarrow dist((A, b), \mathcal{F} = \emptyset) = 0$$

gives rise to the second step Eq. (2.7) of **FR** discussed in Section 2.2. We note that the instability discussed in this section essentially originates from the observation made in Lemma 2.3, i.e., redundant equalities arise in the facially reduced system. Facially reduced system allows us to exploit the root of potential instability when the problem data A or b is perturbed. Although the distance to infeasibility is 0 in the absence of strict feasibility, Proposition 3.14 suggests that a carefully chosen perturbation of b does not have an impact on the feasibility of $\mathcal{F}$. We provide a related numerical experiment in Section 4.1.4 below.

3.3.2. Applications to known characterizations for strict feasibility

There are some known characterizations for strict feasibility of $\mathcal{F}$. Using these characterizations we can obtain extensions of Theorem 3.1, Theorem 3.4 and Corollary 3.6.

The dual $(\mathcal{D})$ of $(\mathcal{P})$ is

$$(\mathcal{D}) \qquad \max_{y,s} \left\{ b^T y \ : \ A^T y + s = c, \ s \geq 0 \right\}. \tag{3.10}$$

It is known that strict feasibility fails for $\mathcal{F}$ if, and only if, the set of optimal solutions for the dual $(\mathcal{D})$ is unbounded; see e.g., Wright (1996, Theorem 2.3) and Gauvin (1977). Then Corollary 3.15 follows.

Corollary 3.15.

1. *Suppose that the set of optimal solutions for the dual $(\mathcal{D})$ is unbounded. Then every basic feasible solution to $\mathcal{F}$ is degenerate.*
2. *Suppose that there exists a nondegenerate basic feasible solution to $\mathcal{F}$. Then the set of optimal solutions for the dual $(\mathcal{D})$ is bounded.*

It is known that strict feasibility holds for $\mathcal{F}$ if, and only if, $b \in relint(A(\mathbb{R}^n_+))$, where *relint* denotes the relative interior; see e.g., Drusvyatskiy & Wolkowicz (2017, Proposition 4.4.1). Then if one finds a set of indices $\mathcal{I} \subset \{1, \dots, n\}$ such that $A_\mathcal{I}$ is nonsingular and $A_\mathcal{I} z = b$ has a solution z with positive entries, then $b \in relint(A(\mathbb{R}^n_+))$.

3.3.3. Applications to obtain a strictly complementary primal-dual solution

In this section we present an application of Algorithm 3.1 for obtaining a strictly complementary primal-dual optimal solution.

Let (x^*, y^*, s^*) be an optimal triple for the standard primal-dual **LP** pair. Let $\mathcal{B}^* \cup \mathcal{N}^* = \{1, \dots, n\}$ be the strict complementary partition of the primal-dual optimal pair. The existence of such a partition is guaranteed by the Goldman–Tucker theorem (Goldman & Tucker, 1956) and the partition $\mathcal{B}^* \cup \mathcal{N}^*$ is unique. For the first application of Algorithm 3.1, we provide a method for obtaining a strict complementary primal-dual solution when the primal optimal solution x^* is nondegenerate or the submatrix $A(:, supp(x^*))$ of A has rank m. To elaborate, we list the two cases where Algorithm 3.1 can be used to obtain maximal complementary solutions.

1. Let x^* be a nondegenerate (optimal) basic feasible solution. Then, $supp(s^*) = \mathcal{N}^*$ and $supp(x^*)$ can be extended to complete $\mathcal{B}^*$;
2. Let x^* be an optimal solution such that $A(:, supp(x^*))$ is full row rank. Then, $supp(s^*) = \mathcal{N}^*$ and $supp(x^*)$ can be extended to complete $\mathcal{B}^*$.

Suppose that we are given a primal-dual optimal solution (x^*, y^*, s^*) of the form

$$\begin{bmatrix} A_\mathcal{B} & A_\mathcal{J} & A_\mathcal{N} \end{bmatrix} \begin{pmatrix} x_\mathcal{B} \\ x_\mathcal{J} \\ x_\mathcal{N} \end{pmatrix} = b,$$

$$\text{where } rank(A_\mathcal{B}) = m, \ \begin{pmatrix} x_\mathcal{B} \\ x_\mathcal{J} \\ x_\mathcal{N} \end{pmatrix} \begin{matrix} > \\ = \\ = \end{matrix} \begin{pmatrix} 0 \\ 0 \\ 0 \end{pmatrix} \text{ and } \begin{pmatrix} s_\mathcal{B} \\ s_\mathcal{J} \\ s_\mathcal{N} \end{pmatrix} \begin{matrix} = \\ = \\ > \end{matrix} \begin{pmatrix} 0 \\ 0 \\ 0 \end{pmatrix}. \tag{3.11}$$

We claim that $\mathcal{N}^* = supp(s^*)$. That is, the support of the current dual optimal solution s^* is maximal and hence we obtain the strict complementary partition for free. We rewrite the system $Ax = b$ of Eq. (3.11) as

$$\begin{bmatrix} A_{\mathcal{B}_1} & A_{\mathcal{B}_2} & A_\mathcal{J} \end{bmatrix} \begin{pmatrix} x_{\mathcal{B}_1} \\ x_{\mathcal{B}_2} \\ x_\mathcal{J} \end{pmatrix} = b, \quad \text{where } A_\mathcal{B} = \begin{bmatrix} A_{\mathcal{B}_1} & A_{\mathcal{B}_2} \end{bmatrix},$$

$$x_\mathcal{B} = \begin{pmatrix} x_{\mathcal{B}_1} \\ x_{\mathcal{B}_2} \end{pmatrix} \text{ and } rank(A_{\mathcal{B}_1}) = m.$$

Then, by replacing the data in Algorithm 3.1 by

$$\mathcal{N} \leftarrow \mathcal{J}, \ A \leftarrow A(:, \mathcal{B}_1 \cup \mathcal{B}_2 \cup \mathcal{N}), \ \tilde{x} \leftarrow x^*,$$

we can endow positive weights to $x_\mathcal{J}$ while maintaining the primal feasibility. Since we maintain the feasibility of the primal-dual solution without violating the complementarity, we maintain the optimality.

3.3.4. Lack of strict feasibility and interior point methods

In this section we provide a new perspective on the ill-conditioning that typically arises in interior point methods. Many interior point algorithms are derived from block Gaussian-elimination of the linearized primal $(\mathcal{P})$ and dual $(\mathcal{D})$ optimality conditions (KKT conditions). Let (x_c, y_c, s_c) be the current primal-dual pair iterate. The search direction is computed by solving the Newton equation

$$\begin{bmatrix} 0_{n\times n} & A^T & I \\ A & 0_{m\times m} & 0_{m\times n} \\ Diag(s_c) & 0_{n\times m} & Diag(x_c) \end{bmatrix} \begin{pmatrix} \Delta x \\ \Delta y \\ \Delta s \end{pmatrix} = - \begin{pmatrix} r_d \\ r_p \\ r_c \end{pmatrix}, \tag{3.12}$$

where r_d, r_p, r_c are the residuals of dual feasibility, primal feasibility and complementarity, respectively. After the block elimination, we first find the change Δy by solving the so-called normal equation, a square system,

$$AD_cA^T \Delta y = \bar{r}, \ \text{ where } D_c = Diag(x_c)Diag(s_c)^{-1}, \tag{3.13}$$

$\bar{r} \in \mathbb{R}^m$ is some residual; see e.g., Wright (1996, Chapter 11). It is known that (3.13) often becomes ill-conditioned near an optimum. The ill-conditioning of the matrix AD_cA^T under degeneracy is discussed in GüLer et al. (1993) in terms of the lack of nice positive diagonal elements of D_c. This relates to our results in the sense that all vertices that form the optimal face of $(\mathcal{P})$ are also degenerate in the absence of strict feasibility. Moreover, we show that the ill-conditioning of the matrix AD_cA^T not only originates from the columns of A chosen by D_c but also from the rows of A in the absence of strict feasibility. In particular, a large *IPS* is a good indicator for ill-conditioning.

We partition the matrix $A = \begin{bmatrix} P_{\bar{m}}AV & A_{\mathcal{I}_0} \\ R_{AV} & R_{\mathcal{I}_0} \end{bmatrix}$, where $[A_{\mathcal{I}_0}; R_{\mathcal{I}_0}]$ corresponds to the submatrix of A associated with the index set $\mathcal{I}_0$. The submatrix R_{AV} refers to the rows of A that are implicitly redundant due the lack of strict feasibility. Let (x^*, y^*, s^*) an optimal triple and let $D^* = Diag(x^*)Diag(s^*)^{-1}$. As $x_c \to x^*$, i.e., as the iterates get closer to the feasible set $\mathcal{F}$, we observe the limiting behaviour $AD_cA^T \to AD^*A^T$ below:

$$\begin{aligned} AD_cA^T \ \to \ AD^*A^T &= \begin{bmatrix} P_{\bar{m}}AV & A_{\mathcal{I}_0} \\ R_{AV} & R_{\mathcal{I}_0} \end{bmatrix} \begin{bmatrix} D^*_{AV} & 0 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} P_{\bar{m}}AV & A_{\mathcal{I}_0} \\ R_{AV} & R_{\mathcal{I}_0} \end{bmatrix}^T \\ &= \begin{bmatrix} (P_{\bar{m}}AV)D^*_{AV}(P_{\bar{m}}AV)^T & (P_{\bar{m}}AV)D^*_{AV}R^T_{AV} \\ R_{AV}D^*_{AV}(P_{\bar{m}}AV)^T & R_{AV}D^*_{AV}R^T_{AV} \end{bmatrix} \end{aligned}$$

where D^*_{AV} is the submatrix of D^* with the diagonal associated with $\mathcal{I}_+$. We recall from Lemma 2.3 that the rows of R_{AV} are linear combinations of the rows of $P_{\bar{m}}AV$. Therefore, the more implicit redundant constraints $\mathcal{F}$ has, the more '0' singular values AD^*A^T has, i.e., ill-conditioned.

The self-dual embedding (Ye et al., 1994) is a popular formulation of the primal-dual **LP** pair used for an interior point method. An attractive feature of the self-dual embedding is that a *feasible* initial iterate in the interior is analytically given. The success of the self-dual embedding technique is supported by strong performances of some solvers. However, the absence of strict feasibility results in the same type of ill-conditioning even when this reformulation is used. For instance, Ye et al. (1994, equation (17)) displays the equation as a part of computing the search direction $(d_x; d_y)$:

$$\begin{bmatrix} X^kS^k & -X^kA^T \\ AX^k & 0 \end{bmatrix}\begin{pmatrix} (X^k)^{-1}d_x \\ d_y \end{pmatrix} = \begin{pmatrix} \gamma\mu^k e - X^k s^k \\ 0 \end{pmatrix} - \begin{bmatrix} X^k c & -X^k\bar{c} \\ -b & \bar{b} \end{bmatrix}\begin{pmatrix} d_\tau \\ d_\theta \end{pmatrix}.$$

Here, $X^k = Diag(x^k)$ and $S^k = Diag(s^k)$, where x^k, s^k are the current primal-dual iterate. It then uses the back-solve steps to complete the remaining components of the search direction. For simplicity, we set the right-hand side of the system to be $\begin{pmatrix} r_1 \\ r_2 \end{pmatrix}$. By expanding the first block equation, we obtain

$$(X^kS^k)(X^k)^{-1}d_x - X^kA^Td_y = r_1 \iff (X^k)^{-1}d_x = (X^kS^k)^{-1}r_1 + (X^kS^k)^{-1}X^kA^Td_y.$$

We then substitute the equality above into the second block equation, i.e.,

$$AX^k(X^k)^{-1}d_x = r_2 \quad \iff \quad AX^k(S^k)^{-1}A^Td_y = r_2 - AX^k(X^kS^k)^{-1}r_1.$$

Finally, we obtain the normal matrix $AX^k(S^k)^{-1}A^T$ that appear in Eq. (3.13).

3.3.5. Lack of strict feasibility in the dual

Recall Remark 2.1 that redundant constraints can result in poor behaviour for interior point methods. Moreover, complementary slackness means we get dual variables fixed at 0. This is one motivation for considering **FR** on the dual $(\mathcal{D})$; see Eq. (3.10). We denote the feasible set of the dual $(\mathcal{D})$ by

$$\begin{aligned}\mathcal{G} &:= \{(y,s) \in \mathbb{R}^m \oplus \mathbb{R}^n_+ \ : \ A^Ty + s = c\} \\ &= \left\{(y,s) \in \mathbb{R}^m \oplus \mathbb{R}^n_+ \ : \ \begin{bmatrix} A^T & I \end{bmatrix}\begin{pmatrix} y \\ s \end{pmatrix} = c\right\}.\end{aligned} \tag{3.14}$$

The facial reduction arguments applied to the dual are parallel to the ones given in Section 2.2. We provide the theorem of the alternative for the dual and a short derivation for the facially reduced system for $\mathcal{G}$ in Section A.3.1. We also conclude that the absence of strict feasibility for $\mathcal{G}$ implies dual degeneracy at all **BFS**s.

A popular method for rewriting an instance with a free variable x_i into the primal standard form is to write x_i into the difference of two nonnegative variables, i.e., $x_i = x_i^+ - x_i^-$ with $x_i^+, x_i^- \geq 0$. This equivalent transformation does not seem to cause any difficulties at first glance; at least the primal simplex method does not consider both x_i^+ and x_i^- as a basic variables simultaneously in order to form a nonsingular basis matrix. However, this equivalent transformation has a significant consequence to the dual program. For any $K \geq \max\{x_i^+, x_i^-\}$, we can maintain the equality

$$x_i = x_i^+ - x_i^- = (x_i^+ + K) - (x_i^- + K).$$

Thus, the primal optimal set is unbounded. This implies that the dual feasible region of the reformulated primal does not have a strictly feasible point. Consequently, the results that we established for the primal applies to the dual; (i) this implies that all **BFS**s of the dual are degenerate; (ii) the equality system for the dual feasibility contains implicit redundancies and thus the Newton equation that appears in the interior point method (3.12) becomes very ill-conditioned near an optimum. More details for loss of strict feasibility in the dual is given in Section A.3.

4. Numerical investigation

We now provide empirical evidence that **FR** is indeed a useful preprocessing tool for reducing the size of problems as well as for improving the *conditioning*. We do this first for interior point methods and then for simplex methods. In particular, this provides empirical evidence that lack of strict feasibility is equivalent to implicit singularity. All the numerical tests are performed using MATLAB version 2021a on Dell XPS 8940 with 11th Gen Intel(R) Core(TM) i5-11400 @ 2.60 GHz 2.60 GHz with 32 Gigabyte memory. We use three different solvers in our tests: (i) *linprog* from MATLAB[9]; (ii) *SDPT3*[10]; and (iii) *MOSEK*.[11] MATLAB version 2021a is used to access all the solvers for the tests, and we use their default settings for stopping criteria. Note that MOSEK has a preprocessing option.[12]

4.1. Empirics with interior point methods

In this section we compare the behaviour for finding near-optimal points with instances that do and do not satisfy strict feasibility. More specifically, given a near optimal primal-dual point $(x^*, s^*) \in \mathbb{R}^n_{++} \oplus \mathbb{R}^n_{++}$ obtained from an interior point solver, we observe the condition number, i.e., the ratio of largest to smallest eigenvalues of the normal matrix at (x^*, s^*):

$$\kappa\left(AD^*A^T\right), \quad \text{where } D^* = Diag(x^*)Diag(s^*)^{-1}. \tag{4.1}$$

We show that instances that do not have strictly feasible points tend to have significantly larger condition numbers of the normal equation near the optimum. We also present a numerical experiment on perturbations of the right-hand side vector b.

4.1.1. Generating LPs without strict feasibility

Given $m, n, r \in \mathbb{N}$, we construct the data $A \in \mathbb{R}^{m\times n}$ and $b \in \mathbb{R}^m$ to satisfy Eq. (2.3) with r as the dimension of the relative interior of $\mathcal{F}$, $relint(\mathcal{F})$.

1. Pick any $0 \neq y \in \mathbb{R}^m$. Let
   $$\{y\}^\perp = span\{a_i\}_{i=1}^{m-1} \quad (= null(y^T)).$$
   We let $R \in \mathbb{R}^{(m-1)\times r}$ be a random matrix, and get
   $$A_1 := \begin{bmatrix} a_1 & \dots & a_{m-1} \end{bmatrix} R \in \mathbb{R}^{m\times r}, \quad A_1^Ty = 0 \in \mathbb{R}^r.$$
2. Pick any $\hat{v} \in \mathbb{R}^r_{++}$ and set $b = A_1\hat{v}$. We note that $y^TA_1 = 0$ and $\langle b, y\rangle = 0$.
3. Pick any matrix $A_2 \in \mathbb{R}^{m\times(n-r)}$ satisfying $(y^TA_2)_i \neq 0, \ \forall i$. If there exists i such that $(y^TA_2)_i < 0$, then change the sign of the i-th column of A_2 so that we conclude
   $$(A_2^Ty) \in \mathbb{R}^{n-r}_{++}.$$
4. We define the matrix $A = \begin{bmatrix} A_1 & A_2 \end{bmatrix} \in \mathbb{R}^{m\times n}$. Then $\{x \in \mathbb{R}^n_+ : Ax = b\}$ is a polyhedron with a feasible point $\hat{x} = [\hat{v}; 0]$ having r number of positives. The vector y is a solution for the system Eq. (2.3):
   $$0 \lneq z = A^Ty = \begin{pmatrix} A_1^Ty = 0 \\ A_2^Ty > 0 \end{pmatrix}, \ b^Ty = 0.$$
   We then randomly permute the columns of A to avoid the zeros always being at the bottom of the feasible variables x.

For the empirics, we construct the objective function c^Tx of $(\mathcal{P})$ as follows. We choose any $\bar{s} \in \mathbb{R}^n_{++}, \bar{y} \in \mathbb{R}^m$ and set $c = A^T\bar{y} + \bar{s}$. Then we have the data for the primal-dual pair of **LP**s and the primal *fails* strict feasibility:

$$(\mathcal{P}_{(A,b,c)}) \quad \min\{\ c^Tx : Ax = b, \ x \geq 0\ \} \quad \text{and}$$
$$(\mathcal{D}_{(A,b,c)}) \quad \max\{\ b^Ty : A^Ty + s = c, \ s \geq 0\ \}.$$

[9] https://www.mathworks.com/. Version 9.10.0.1669831 (R2021a) Update 2.
[10] https://www.math.cmu.edu/~reha/sdpt3.html, version SDPT3 4.0.
[11] https://www.mosek.com/. Version 8.0.0.60.
[12] MOSEK has a presolve with five steps that includes eliminating fixed variables. However, itis clear from the empirical evidencethat the variables fixed at 0 are not found.

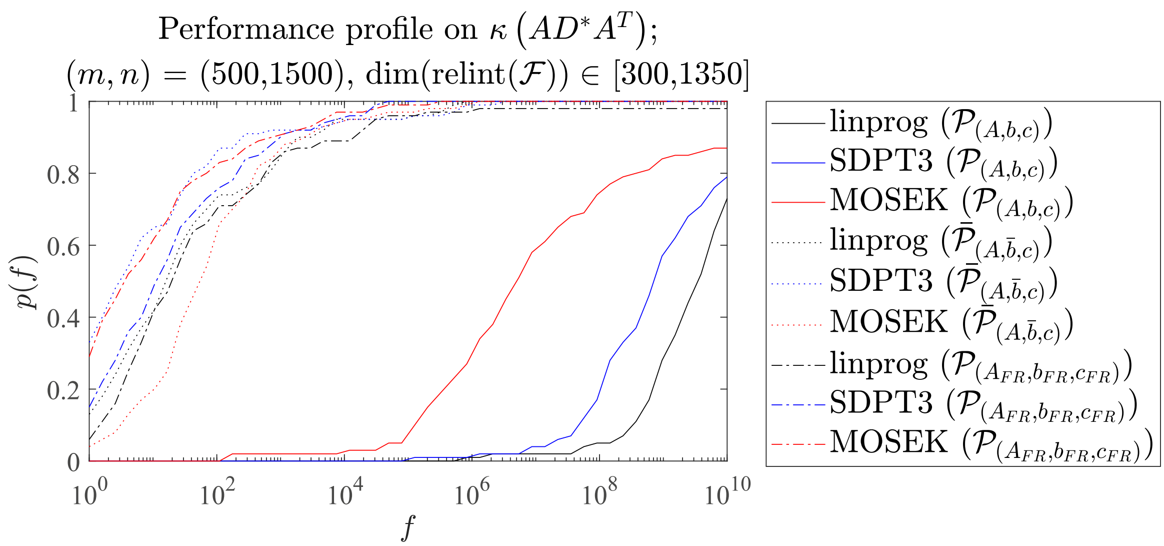

Fig. 4.1. Performance profile on $\kappa\left(AD^*A^T\right)$ with(out) strict feasibility near optimum; various solvers .

We note that by choosing $\bar{s} \in \mathbb{R}^n_{++}$, the dual problem $(\mathcal{D}_{(A,b,c)})$ has a strictly feasible point. In order to generate instances with strictly feasible points, we maintain the same data A, c used for the pair $(\mathcal{P}_{(A,b,c)})$ and $(\mathcal{D}_{(A,b,c)})$. We only redefine the right-hand side vector by $\bar{b} = Ax^\circ$, where $x^\circ \in \mathbb{R}^n_{++}$:

$$(\bar{\mathcal{P}}_{(A,\bar{b},c)}) \quad \min\{\ c^Tx : Ax = \bar{b},\ x \geq 0\ \} \quad \text{and}$$
$$(\bar{\mathcal{D}}_{(A,\bar{b},c)}) \quad \max\{\ \bar{b}^Ty : A^Ty + s = c,\ s \geq 0\ \}.$$

The facially reduced instances of $(\mathcal{P}_{(A,b,c)})$ are denoted by $(\mathcal{P}_{(A_{FR},b_{FR},c_{FR})})$. They are obtained by discarding the variables that are identically 0 in the feasible set $\mathcal{F}$ and the redundant constraints. In other words, the affine constraints of $(\mathcal{P}_{(A_{FR},b_{FR},c_{FR})})$ are of the form Eq. (2.7).

4.1.2. Condition numbers

In order to illustrate the differences in condition numbers of the normal matrices, we solve the three families of instances: (i) $(\mathcal{P}_{(A,b,c)})$, strictly feasible fails; (ii) $(\bar{\mathcal{P}}_{(A,\bar{b},c)})$, strictly feasible holds; (iii) $(\mathcal{P}_{(A_{FR},b_{FR},c_{FR})})$, facially reduced instances of $(\mathcal{P}_{(A,b,c)})$.

In Fig. 4.1 we use a *performance profile* (Dolan & Moré, 2002; Gould & Scott, 2016) to observe the overall behaviour on different families of instances using the three solvers. The performance profile provides a useful graphical comparison for solver performances. Figure 4.1 displays the performance profile on the condition numbers of the normal matrix AD^*A^T near optimal points from different solvers. We generate 100 instances for each family that have $\dim(relint(\mathcal{F})) \in [300, 1350]$. The instance sizes are fixed with $(m,n) = (500, 1500)$. The vertical axis in Fig. 4.1 represents the statistics of the performance ratio on $\kappa\left(AD^*A^T\right)$, the condition number of normal matrix near optimum (x^*, s^*); see Eq. (4.1). Roughly speaking, the vertical axis represents the probability of achieving a performance ratio within a factor of f among all methods used. We used the lower the better statistics. The details of the performance ratio are discussed in Dolan & Moré (2002); Gould & Scott (2016). The solid lines in Fig. 4.1 represent the performance of the instances $(\mathcal{P}_{(A,b,c)})$ that fail strict feasibility. They show that the condition numbers of the normal matrices near optima are significantly higher when strict feasibility fails. That is, when strict feasibility fails for $\mathcal{F}$, the matrix AD^*A^T is more ill-conditioned and it is difficult to obtain search directions of high accuracy. We also observe that facially reduced instances yield smaller condition numbers near optima. We note that the instances $(\mathcal{P}_{(A,b,c)})$ and $(\mathcal{P}_{(A_{FR},b_{FR},c_{FR})})$ are equivalent.

4.1.3. Stopping criteria

We now use the three solvers to observe the accuracy of the first-order optimality conditions (KKT conditions) and the running times, for the instances $(\mathcal{P}_{(A,b,c)})$ and $(\mathcal{P}_{(A_{FR},b_{FR},c_{FR})})$, see Table 4.1. We test the average performance of 10 instances of the size $(n,m,r) = (3000, 500, 2000)$. The headers used in Table 4.1 provide the following. Given solver outputs (x^*, y^*, s^*), the header 'KKT' exhibits the average of the triple consisting of the primal feasibility, dual feasibility and complementarity;

$$\text{KKT} = \left(\frac{\|Ax^* - b\|}{1 + \|b\|},\ \frac{\|A^Ty^* + s^* - c\|}{1 + \|c\|},\ \frac{\langle x^*, s^*\rangle}{n} \right).$$

The headers 'iter' and 'time' in Table 4.1 refer to the average of the number of iterations and the running time in seconds, respectively.

From Table 4.1 we observe that facially reduced instances provide significant improvement in first-order optimality conditions, the number of iterations and the running times for all solvers in general. We note that the instances $(\mathcal{P}_{(A,b,c)})$ and $(\mathcal{P}_{(A_{FR},b_{FR},c_{FR})})$ are equivalent. Hence, our empirics show that performing facial reduction as a preprocessing step not only improves the solver running time but also the *quality* of solutions.

4.1.4. Distance to infeasibility

In this section we present empirics that illustrate the effect of perturbations of the right-hand side b when strict feasibility fails. We recall, from Proposition 3.14, that there exists an arbitrarily small perturbation of the right-hand side vector b of $\mathcal{F}$ that renders the set $\mathcal{F}$ infeasible, i.e., $dist(b, \mathcal{F} = \emptyset) = 0$. Moreover, the vector $\Delta b = y$ that satisfies the auxiliary system Eq. (2.3) is a perturbation that makes the set $\mathcal{F}$ empty; see Eq. (3.9).

We follow the steps in Section 4.1.1 to generate instances of the order $(n,m) = (1000, 200)$ and $r = relint(\mathcal{F}) = 900$. The objective function c^Tx is chosen as presented in Section 4.1.1. For the fixed (n,m,r), we generate 10 instances and observe the average performance of these instances as we gradually increase the magnitude of the perturbation. We recall the matrix AV from Eq. (2.5). We use two types of perturbations for b;

$$\Delta b,\ \text{where } \Delta b \in range(AV)^\perp, \quad \Delta\bar{b},\ \text{where } \Delta\bar{b} \in range(AV).$$

We choose Δb to be the vector y that satisfies Eq. (2.3). For $\Delta\bar{b}$, we choose AVd, where $d \in \mathbb{R}^r$ is a randomly chosen vector. As we

Table 4.1
Average of KKT conditions, iterations and time of (non)-facially reduced problems.

		Non-facially reduced system	Facially reduced system
linprog	KKT	(2.44e−15, 2.05e−12, 3.18e−09)	(5.85e−16, 4.74e−16, 9.22e−09)
	iter	22.30	17.90
	time	2.34	0.81
SDPT3	KKT	(8.11e−10, 7.55e−12, 5.65e−02)	(1.43e−11, 3.67e−16, 4.38e−06)
	iter	25.50	19.30
	time	1.73	0.70
mosek	KKT	(7.52e−09, 1.80e−15, 3.27e−06)	(3.85e−09, 3.69e−16, 1.19e−06)
	iter	40.30	10.20
	time	1.40	0.35

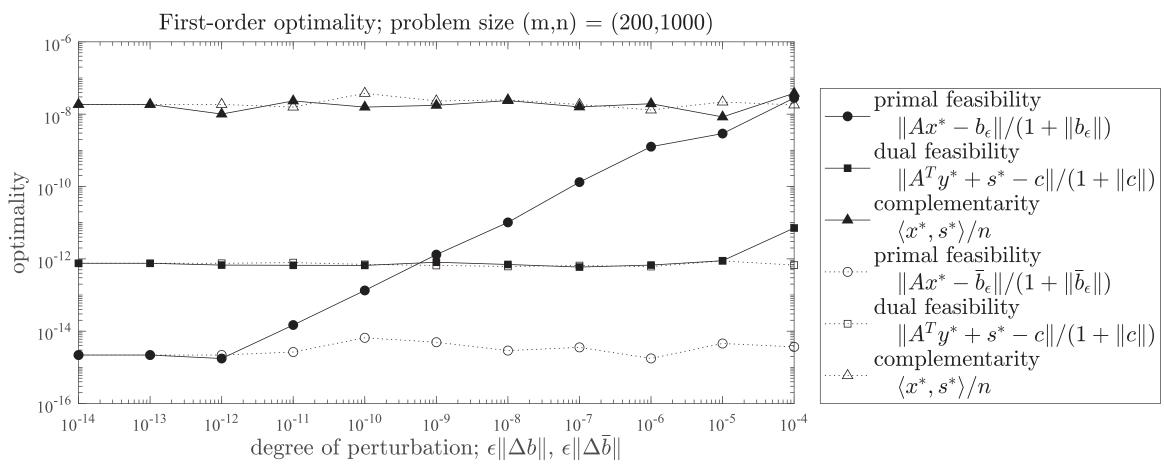

Fig. 4.2. Changes in the first-order optimality condition as the perturbation of b increases.

increase $\epsilon > 0$, we observe the performance of the two families of the systems

$$Ax = b_\epsilon := b - \epsilon \Delta b \quad \text{and} \quad Ax = \bar{b}_\epsilon := b - \epsilon \Delta \bar{b}.$$

We use the interior point method from MATLAB's linprog for the test. Figure 4.2 contains the average of the first-order optimality conditions evaluated at the solver outputs (x^*, y^*, s^*) of these instances; primal feasibility, dual feasibility and the complementarity.

The horizontal axis of Fig. 4.2 indicates the degree of the perturbation imposed on the right-hand side vector b, $\epsilon\|\Delta b\|$ and $\epsilon\|\Delta \bar{b}\|$. The vertical axis indicates the individual component of the first-order optimality. From Fig. 4.2, we observe that the KKT conditions with the perturbation $\Delta\bar{b}$ display a steady performance regardless of the perturbation degree; see the markers ○,□, △ with the dotted lines. In contrast, the markers •,■,▲ in Fig. 4.2 exhibit the performance of the instances that are perturbed with Δb and they display a different performance. In particular, we see that the relative primal feasibility $\|Ax^* - b_\epsilon\|/(1 + \|b_\epsilon\|)$, marked with •, consistently increases as the perturbation magnitude $\epsilon\|\Delta b\|$ increases when strict feasibility fails for $\mathcal{F}$.

4.1.5. Empirics on singular values and IPS

In this section we present our numerical experiment on the ill-conditioning discussed in Section 3.3.4 in terms of *maxSD* (see Definition 2.5). We generated instances with different settings for $maxSD = 1, 5$ and 10. We recall the generation for the vector y and A_2 in Section 4.1.1. For generating and instance with $maxSD > 1$, we generated $Y_c = blkdiag(y^1, \ldots, y^{IPS}) \in \mathbb{R}^{m \times maxSD}$ and $A_2 = blkdiag(A_2^1, \ldots, A_2^{maxSD})$ of appropriate dimension in order to produce the exposing vector $A_2^T \sum_{j=1}^{maxSD} Y_c(:, j) \geq 0$. Each column of Y_c serves as a vector satisfying Eq. (2.3).

Let $\sigma_{\max}(AD^*A^T)$ be the maximum singular value of AD^*A^T. We count the number of singular values of AD^*A^T that are smaller than $10^{-8} \cdot \sigma_{\max}(AD^*A^T)$. In Table 4.2 below, we report the cardinality of

$$\Sigma_0 := \{i : \sigma_i(AD^*A^T) < \sigma_{\max}(AD^*A^T)\}.$$

Table 4.2
(rel.) small singular values of AD^*A^T near optimum; average over 20 instances.

		$maxSD = 1$	$maxSD = 5$	$maxSD = 10$
linprog	$\|\Sigma_0\|$	4.10	8.65	13.10
SDPT3	$\|\Sigma_0\|$	4.75	8.00	34.65
MOSEK	$\|\Sigma_0\|$	6.45	12.35	14.50

Table 4.3
Average of the ratio of degenerate iterations.

		100% − (r/n)%				
		40	30	20	10	0
(n, m)	(1000, 250)	36.62	10.18	0.01	0.02	0.00
	(2000, 500)	39.72	18.28	0.07	0.15	0.01
	(3000, 750)	25.99	10.66	0.32	0.75	0.02
	(4000, 1000)	29.78	18.25	0.25	0.53	0.02

We test the average performance on the 20 instances of the fixed size $(n, m, r) = (3000, 500, 2000)$. We display the average number of $|\Sigma_0|$. We see from Table 4.2 a larger *maxSD* and *IPS* values produce a greater number of small singular values. When there is a significant number of redundant constraints, it is more difficult to obtain a good search direction due to a large number of relatively small singular values.

4.2. Empirics with simplex method

In this section we compare the behaviour of the dual simplex method with instances that have strictly feasible points and in-

Table 4.4
Average of the ratio of degenerate iterations.

ϵ_A	ϵ_b	(A, b)	$(P_{\bar{m}}AV, P_{\bar{m}}b)$	$(A_{\text{trans}}, b_{\text{trans}})$
1.0e−09	0	(11, 4.938e−02)	(97, 6.705e−03)	100
0	1.0e−09	(27, 2.470e−10)	(100, 2.208e−10)	100
1.0e−09	1.0e−09	(11, 1.339e−01)	(96, 8.719e−03)	100

stances that do not. We also observe the degeneracy issues that arise in the instances from NETLIB.

4.2.1. Empirics on the number of degenerate iterations

In this section we test how the lack of strict feasibility affects the performance of the dual simplex method. We provide the construction of instances that fail strict feasibility in Appendix A.3.2. We choose MOSEK for our tests since MOSEK reports the percentage of degenerate iterations as a part of the solver report. MOSEK reports the quantity ‘DEGITER(%)’, the ratio of degenerate iterations.

Given a set $\mathcal{G}$ and a point $(y, s) \in relint(\mathcal{G}) \subseteq \mathbb{R}^m \oplus \mathbb{R}^n_+$, let r be the number of positive entries of s, i.e., $r = |supp(s)|$. In our tests, we gradually increase r for fixed n, m and generate instances for $\mathcal{G}$ as described in Appendix A.3.2. We then observe the behaviour of the dual simplex method. Table 4.3 contains the results. In Table 4.3, a smaller value for the header $(r/n)\%$ means that there are more entries of s that are identically 0 in the set $\mathcal{G}$; and the value 0% means that strict feasibility holds. For each triple (n, m, r), we generated 10 instances and we report the average of ‘DEGITER(%)’ of these instances.

We recall Theorem 3.1: lack of strict feasibility implies that all basic feasible solutions are degenerate. However, we observe more, i.e., from Table 4.3, the frequency of degenerate iterations increases as r decreases. In other words, higher degeneracy of the set $\mathcal{G}$ yields more degenerate iterations when the dual simplex method is used.

4.2.2. NETLIB problems; perturbations; stability

We now illustrate the lack of strict feasibility on instances from the NETLIB data set. We used the following 67 instances that are in standard form at this link:

25fv47	adlittle*	afiro	agg*	agg2*	agg3*	bandm*	beaconfd*	blend	bnl1*
bnl2*	brandy*	cre_a*	cre_b*	cre_c*	cre_d*	d2q06c*	degen2*	degen3*	e226*
fffff800*	israel	lotfi	maros_r7	nug05	nug06	nug07	nug08	nug12	nug15
nug20	osa_07*	osa_14*	qap12	qap15	qap8	sc105*	sc205*	sc50a*	sc50b*
scagr25	scagr7	scfxm1*	scfxm2*	scfxm3*	scorpion*	scrs8*	scsd1	scsd6	scsd8
sctap1	sctap2	sctap3	share1b	share2b	ship04l*	ship04s*	ship08l*	ship08s*	ship12l*
ship12s*	stocfor1	stocfor2	stocfor3	truss	wood1p*	woodw*			

We removed redundant rows to guarantee full row rank of A.

Surprisingly, the Slater condition fails for 37 out of these 67 instances.[13] This has interesting implications for both interior point and simplex methods. The standard interior point method stopping criteria is complicated by the unbounded dual optimal set. For the primal simplex method, every iteration is at a degenerate **BFS** and *stalling* generally occurs. Therefore preprocessing to eliminate the variables fixed at 0 is important. In addition, in order to motivate robust optimization, it is shown in e.g., Ben-Tal et al. (2009); Ben-Tal & Nemirovski (1999) that optimal solutions of many of the NETLIB instances are extremely sensitive to perturbations in the data. We now see this to be the case, and we show that **FR** regularizes the problem and avoids this instability.

We first use the instance degen3 in order to illustrate the consequence of lack of strict feasibility. The data matrix A after removing two redundant rows is 1501-by-2604. After **FR**, we obtain the constraint matrix $P_{\bar{m}}AV$ of size 1226-by-1648. This implies that $2604 - 1648 = 956$ number of variables are identically 0 on the feasible set. Furthermore, $IPS(\mathcal{F}) = 275$ equality constraints are implicitly redundant. By Item 3 of Corollary 3.9, without **FR**, the degree of degeneracy of every **BFS** is at least 275. Namely, the length of the basis is 1501 and every basis contains at least 275 degenerate indices.

We now illustrate that **FR** gives a more robust model with respect to data perturbations using the instance brandy. Let (A, b) be the data after removing the redundant equality constraints. Let $(P_{\bar{m}}AV, P_{\bar{m}}b)$ be the data for the facially reduced system. The data matrices A and $P_{\bar{m}}AV$ have sizes 193-by-303 and 155-by-260, respectively.[14] Set the perturbation scalars $\epsilon_A = \epsilon_b = 10^{-9}$. We construct a random perturbation matrix Φ, $\|\Phi\|_F = \|A\|_F + 1$, and random perturbation vector ϕ, $\|\phi\|_2 = \|b\|_2 + 1$. We then solve the problem

$$\tilde{p}^* = \max\{\langle c, x\rangle : (A + \epsilon_A \Phi)x = b + \epsilon_b \phi, \; x \geq 0\}.$$

For the facially reduced system, we used the identical perturbation data Φ, ϕ and discard the rows and columns of (A, b) found from **FR**. That is, we use the perturbations $P_{\bar{m}}\Phi V$ and $P_{\bar{m}}\phi$ to the facially reduced system after the scaling $\|P_{\bar{m}}\Phi V\|_F = \|P_{\bar{m}}AV\|_F + 1$ and $\|P_{\bar{m}}\phi\|_2 = \|P_{\bar{m}}b\|_2 + 1$. We then solve

$$\max\{\langle V^T c, v\rangle : (P_{\bar{m}}AV + \epsilon_A P_{\bar{m}}\Phi V)v = P_{\bar{m}}b + \epsilon_b P_{\bar{m}}\phi, \; v \geq 0\}.$$

In this way, we maintain the identical perturbation structure for the original system and the facially reduced system. We also generate a transportation problem and use the aforementioned perturbations. We note that the transportation problems have Slater points but are known to be highly degenerate. The size of the data generated is 49-by-600.

In the experiment, we tested the instances using 100 different perturbation settings. We randomly generated perturbations Φ, ϕ with density set at 0.1. We used MOSEK simplex with the setting ‘MSK_OPTIMIZER_FREE_SIMPLEX’. In Table 4.4, the headers ϵ_A and ϵ_b refer to the scalars used for perturbations as described above. The headers (A, b), $(P_{\bar{m}}AV, P_{\bar{m}}b)$ and $(A_{\text{trans}}, b_{\text{trans}})$ refer to the non-facially reduced system, the facially reduced system and the transportation problems, with the perturbations. The

[13] The instances that fail strict feasibility are marked with an asterisk $*$ in the list above.

[14] This also means that, without **FR**, every **BFS** has at least 38 degenerate basic variables. At least 19.69 percent of basic variables are always degenerate.

integral values in the table indicate the number of times that the solver outputs PRIMAL_AND_DUAL_FEASIBLE. Let p^* be the optimal value for the unperturbed instance brandy, and let $\tilde{p}^*$ be the optimal value of a perturbed instance of brandy. The non-integral values in the table indicate the average relative difference in the optimal values between p^* and $\tilde{p}^*$. The relative difference is computed using the formula $\frac{|p^*-\tilde{p}^*|}{2|p^*+\tilde{p}^*|}$. For example, the first entry 11 in Table 4.4 means that $100-11$ out of 100 perturbed instances yield infeasibility or unknown status, i.e., only 11 solved successfully. The entry 4.938e−02 next to 11 indicates the average of $\frac{|p^*-\tilde{p}^*|}{2|p^*+\tilde{p}^*|}$ on those 11 instances.

The columns (A, b) and $(P_{\bar{m}}AV, P_{\bar{m}}b)$ in Table 4.4 demonstrate that the facially reduced problems are more immune to data perturbations; the number of successfully solved perturbed instances are significantly larger and the optimal values under the perturbations are less influenced. The last column indicates that although the instance may have many degenerate **BFS**s, having a strictly feasible point is important in terms of perturbations in data, i.e., this emphasizes the difference between the two types of degeneracy.

5. Conclusion

We have addressed the impact, for both theoretical and computational reasons, of loss of strict feasibility in **LP**, distinguishing one type of degeneracy at a **BFS**. For our numerics we illustrated this using the accuracy of optimality conditions as well as the effect of perturbations, for the two most popular classes of algorithms, i.e., simplex and interior point methods. For the theory, we proved, using the two-step facial reduction, that if strict feasibility fails for a linear program, then every **BFS** is degenerate. In addition, we showed that facial reduction can be implemented efficiently to obtain a smaller simpler problem with strict feasibility, and that this improves stability. This was illustrated on random problems, as well as instances from the NETLIB data set.

An essential step for almost all algorithms for linear programming is preprocessing. One part of preprocessing is identifying *fixed variables*. However, identifying variables fixed at 0, facial reduction, has not been done due to expense and accuracy problems. In this paper we have shown that not eliminating these variables, i.e., lack of strict feasibility, is equivalent to implicit singularity and this helps explain the numerical difficulties that arise. We have further provided an efficient preprocessing step for facial reduction, i.e., we continue on phase I of the simplex method that eliminates all the artificial variables, and eliminate the variables fixed at 0. We observed that a variable that is basic (positive) in every **BFS** corresponds to a redundant constraint and, by complementary slackness, corresponds to a variable fixed at 0 in the dual. And redundant constraints have been shown in the literature to poorly affect algorithms (Deza et al., 2006). Moreover, identifying redundant constraints is a nontrivial operation e.g., Caron et al. (1997). This motivates doing **FR** on both the primal and the dual problems. (It is still unclear whether or not we have to repeat **FR** on the primal again.)

In the literature, in particular in textbooks on **LP**, the method most often used to handle a free variable x_i is to replace it by two nonnegative variables $x_i \leftarrow x_i^+ - x_i^-$. This means that the optimal solution is unbounded as one can add an arbitrary positive constant to both new variables. But then strict feasibility fails for the dual, i.e., stable problems are transformed into ill-conditioned problems. One can speculate that this may account for the large number of instances in the NETLIB set where strict feasibility fails and numerical accuracy is difficult to maintain.

We have presented various numerical experiments that convey the importance of preprocessing for strict feasibility for linear programs, Section 4. For interior point methods, we illustrated the importance of strict feasibility using condition numbers and relationships with *nearness to infeasibility*. We also shed light on the main difficulties that arose with the implicit redundant constraints and used the QR decomposition to show how these difficulties come into play. This also relates to the implicit problem singularity, *IPS*. A larger *IPS* means that there is a higher chance of inducing an infeasible problem under perturbations. A large number of degenerate **BFS**s is believed to cause difficulties for the simplex method. We have shown that the settings for having many identically 0 variables in the dual program yield many degenerate iterations in the simplex method. We also have shown that many NETLIB instances fail strict feasibility and used selected instances to show the effect of this degeneracy. Moreover, the facially reduced problems are seen to be more robust with respect to data perturbations. In addition, an essential element of solving an **LP** is *postoptimal analysis*, this becomes difficult when strict feasibility fails and perturbations of b can lead to infeasibility. These facts further emphasize that ensuring strict feasibility should be part of preprocessing for linear programming.

Our results can easily extend to other forms of **LP**s and to more general problems where degeneracies arise, such as the active set method for quadratic programs (Forsgren et al., 2015; Wolfe, 1959). We are currently extending the efficient **FR** technique to semidefinite programs.

Acknowledgements

This research is supported by the National Sciences and Engineering Research Council (NSERC) of Canada, Grant # No. 50503-10827.

Appendix A. Technical proofs, supplementary materials

A1. Proof of Corollary 3.2

Proof. Let $x \in F$ and let r be the number of positive entries in x. Let $\bar{x} \in \mathbb{R}^r$ be the vector obtained by discarding the 0 entries in x. This is readily given by the following matrix-vector multiplication $\bar{x} = I(supp(x), :)x$, where $supp(x)$ is the support of x, the set of indices $\{i : x_i > 0\}$. Let $\bar{A} \in \mathbb{R}^{m\times r}$ be the matrix after removing the columns of A that are not in the support of x, i.e., $\bar{A} = A_{supp(x)}$. We note that $\bar{x}$ is a particular solution to the system $\bar{A}z = b$ and $\bar{x} > 0$.

Suppose to the contrary that $r > m + d$. Since $r - m > d$, there exists at least $d + 1$ linearly independent vectors, say $v_1, \dots, v_{d+1} \in \mathbb{R}^r$, satisfying $\bar{A}v_i = 0, \ \forall i = 1, \dots, d + 1$. For each $i \in \{1, \dots, d + 1\}$ and for $\epsilon \in \mathbb{R}$, we define

$$\begin{array}{ll} v_{i,+} := \bar{x} + \epsilon v_i, & v_{i,-} := \bar{x} - \epsilon v_i, \\ x_{i,+} := I(:, supp(x))(\bar{x} + \epsilon v_i), & x_{i,-} := I(:, supp(x))(\bar{x} - \epsilon v_i). \end{array}$$

For a sufficiently small ϵ, we have $x_{i,+}, x_{i,-} \in \mathcal{F}$. We note that $x = \frac{1}{2}(x_{i,+} + x_{i,-}), \ \forall i$. Hence, by the definition of face, $x_{i,+} \in F, \ \forall i$. Therefore, F contains vectors $\{x_{i,+}\}_{i=1,\dots,d+1} \cup \{x\}$ that are affinely independent and hence $\dim(F) \geq d + 1$. □

A2. A condition measure using degeneracy

Although degeneracy is a well-known subject, to the best of our knowledge, the relationships between degeneracy and stability are rarely discussed. We now show that the degree of degeneracy at a **BFS** provides useful information on the robustness of the **LP**; the least degenerate **BFS** provides an upper bound on the number of implicitly redundant equalities of the set $\mathcal{F}$. We note that an $\mathcal{F}$ that contains a large number of implicit redundancies is a more *ill-conditioned* set. (This is comparable to a linear system $Ax = b$ with more redundant rows having the error in the solution being more susceptible to perturbations of b.)

The arguments used in the proof of Corollary 3.9 are rather algebraic. The geometric argument used in the proof of Theorem 3.4 provides two useful estimates. For any extreme point $x \in \mathcal{F}$, the number of nonzero elements of x, $|supp(x)|$, satisfies

$$|supp(x)| \leq m - IPS(\mathcal{F}) \Rightarrow IPS(\mathcal{F}) \leq m - |supp(x)|.$$

Since this holds for all extreme points of $\mathcal{F}$, we get the following:

$$\begin{aligned} SD(\mathcal{F}) &\leq maxSD(\mathcal{F}) \leq IPS(\mathcal{F}) \\ &\leq \hat{d} := \min_{\textbf{BFS}\, x \in \mathcal{F}} \{\text{degree of degeneracy of } x\}. \end{aligned} \tag{A.1}$$

The shortest **FR** steps for $\mathcal{F}$, $SD(\mathcal{F})$, is at most 1, thus the inequality $SD(\mathcal{F}) \leq \hat{d}$ does not provide useful information. However, the relation (A.1) provides two meaningful corollaries related to $maxSD(\mathcal{F})$ and $IPS(\mathcal{F})$:

1. The inequality $maxSD(\mathcal{F}) \leq \hat{d}$ implies that the number of nontrivial **FR** steps cannot exceed the degree of degeneracy of a least degenerate **BFS** of $\mathcal{F}$;
2. The inequality $IPS(\mathcal{F}) \leq \hat{d}$ shows that it is useful to record the minimum degree of degeneracy observed throughout the simplex iterations. This gives an estimate for the number of implicitly redundant equalities of $\mathcal{F}$.

If $\mathcal{F}$ contains a nondegenerate **BFS**, we get $\hat{d} = 0$. Hence $SD(\mathcal{F}) = maxSD(\mathcal{F}) = IPS(\mathcal{F}) = 0$ and it provides an alternative way to view Corollary 3.6. We comment that evaluating and recording the degree of degeneracy of a **BFS** are not expensive operations.

A3. Dual degeneracy in the absence of strict feasibility

A3.1. Implicit redundancies in the dual

The following Lemma A.1 provides the corresponding dual form of the theorem of the alternative for set $\mathcal{G}$ in Eq. (3.14).

Lemma A.1 (theorem of the alternative in dual form, Cheung (2013, Theorem 3.3.10))**.** *Let $\mathcal{G} \neq \emptyset$ in Eq. (3.14). Then, exactly one of the following statements holds:*

1. *There exists $(y, s) \in \mathbb{R}^m \oplus \mathbb{R}^n_{++}$ with $A^T y + s = c$, i.e., strict feasibility holds for $\mathcal{G}$;*
2. *There exists $w \in \mathbb{R}^n$ such that*

$$0 \neq w \in \mathbb{R}^n_+, \; Aw = 0 \;\text{ and }\; \langle c, w\rangle = 0. \tag{A.2}$$

We recall that the vector $A^T y$ in Eq. (2.4) provides an exposing vector to the set $\mathcal{F}$. Similarly, a solution w to the auxiliary system Eq. (A.2) provides an exposing vector for $\mathcal{G}$:

$$(y, s) \in \mathcal{G} \Rightarrow \left\{\langle w, s\rangle = \langle w, c - A^T y\rangle = \langle c, w\rangle - \langle Aw, y\rangle = 0 - \langle 0, y\rangle = 0\right\}.$$

We let

$$\mathcal{I}_w = \{1, \ldots, n\} \setminus supp(w), \; U = I(:, \mathcal{I}_w) \;\text{ and }\; s_w = |supp(w)|.$$

Then, the facially reduced system of $\mathcal{G}$ is given by

$$\left\{(y, u) \in \mathbb{R}^m \oplus \mathbb{R}^{n-s_w}_+ \;:\; \begin{bmatrix} A^T & U \end{bmatrix} \begin{pmatrix} y \\ u \end{pmatrix} = c \right\}. \tag{A.3}$$

The notion of degeneracy in Section 2.1 naturally extends to an arbitrary polyhedron, e.g., see Bertsimas & Tsitsiklis (1997, Section 2). For a general polyhedron $P \subseteq \mathbb{R}^n$, a point p in P is called a *basic solution* if there are n linearly independent active constraints at p. In addition, if there are more than n active constraints at the point $p \in P$, then the point p is called *degenerate*. Using this definition of degeneracy, we now show that the absence of strict feasibility for $\mathcal{G}$ implies that every basic feasible solution of $\mathcal{G}$ is degenerate.

First, note that the facially reduced system in Eq. (A.3) contains a redundant constraint, i.e., let w be an exposing vector for $\mathcal{G}$ from the system Eq. (A.2). Then we have

$$\begin{bmatrix} A \\ U^T \end{bmatrix} w = \begin{bmatrix} Aw \\ U^T w \end{bmatrix} = \begin{bmatrix} 0_m \\ 0_{n-s_w} \end{bmatrix}.$$

In other words, there is a nontrivial row combination of $\begin{bmatrix} A^T & U \end{bmatrix}$ that yields the 0 vector implying the existence of a redundant row and a redundant constraint in the facially reduced system. The redundancy immediately implies the dual degeneracy; for any basic solution of $\mathcal{G}$, there always exists an redundant equality in $\begin{bmatrix} A^T & I \end{bmatrix} \begin{pmatrix} y \\ s \end{pmatrix} = c$.

A3.2. Construction of dual LPs without strict feasibility

We first show how to generate an instance for the dual feasible set $\mathcal{G}$ that fails strict feasibility. The construction is similar to the one in Section 4.1.1. We generate a degenerate problem by finding a feasible auxiliary system Eq. (A.2). Given $m, n, r \in \mathbb{N}$, we construct $A \in \mathbb{R}^{m \times n}$ and $c \in \mathbb{R}^n$ that satisfy Eq. (A.2) with $\dim(relint(\mathcal{G})) = m + r$.

1. Pick any $0 \neq w \in \mathbb{R}^n_+$ with $|supp(w)| = n - r$. Let
$$\{w\}^\perp = span\{d_i\}_{i=1}^{n-1} \subset \mathbb{R}^n \quad \left(= null(w^T)\right).$$
We let $D \in \mathbb{R}^{(n-1) \times n}$ be the matrix where its rows consist of $\{d_i^T\}_{i=1}^{n-1}$. We let $R \in \mathbb{R}^{m \times (n-1)}$ be a random matrix and we set $A = RD$. We note that $Aw = 0$.
2. Pick $s \in \mathbb{R}^n_+$ so that
$$s_i = \begin{cases} 0 & \text{if } i \in supp(w) \\ \text{positive} & \text{if } i \notin supp(w). \end{cases}$$
We note that $\langle w, s\rangle = 0$ holds.
3. Pick $y \in \mathbb{R}^m$ and set $c = A^T y + s$. We note that $\langle c, w\rangle = 0$ holds.

For the empirics, we construct the objective function $b^T y$ of $(\mathcal{D})$ by choosing a vector $\hat{x} \in \mathbb{R}^n_{++}$ and setting $b = A\hat{x}$.

References

Andersen, E. (1995). Finding all linearly dependent rows in large-scale linear programming. *Optimization Methods and Software, 6*(3), 219–227.
Andersen, E., & Andersen, K. (1995). Presolving in linear programming. *Mathematical Programming, 71*(2), 221–245.
Ben-Tal, A., El Ghaoui, L., & Nemirovski, A. (2009). Robust optimization. *Princeton series in applied mathematics*. Princeton University Press, Princeton, NJ. https://doi.org/10.1515/9781400831050.
Ben-Tal, A., & Nemirovski, A. (1999). Robust solutions of uncertain linear programs. *Operations Research Letters, 25*(1), 1–13. https://doi.org/10.1016/S0167-6377(99)00016-4.
Bertsimas, D., & Tsitsiklis, J. (1997). *Introduction to linear optimization*. Belmont, MA: Athena Scientific.
Bixby, R. (2002). Solving real-world linear programs: A decade and more of progress. *Operations Research, 50*(1), 3–15. 50th anniversary issue of Operations Research
Bland, R. G. (1977). New finite pivoting rules for the simplex method. *Mathematics of Operations Research, 2*(2), 103–107. https://doi.org/10.1287/moor.2.2.103.
Borwein, J., & Wolkowicz, H. (1980/81). Facial reduction for a cone-convex programming problem. *Journal of the Australian Mathematical Society: Series A, 30*(3), 369–380.
Borwein, J., & Wolkowicz, H. (1981). Regularizing the abstract convex program. *Journal of Mathematical Analysis and Applications, 83*(2), 495–530.
Caron, R., Boneh, A., & Boneh, S. (1997). Redundancy. In *Advances in sensitivity analysis and parametric programming*. In *Internat. ser. oper. res. management sci.: vol. 6* (pp. 13.1–13.41). Boston, MA: Kluwer Acad. Publ..
Chandrasekaran, R., Kabadi, S. N., & Murty, K. G. (1981/82). Some NP-complete problems in linear programming. *Operations Research Letters, 1*(3), 101–104. https://doi.org/10.1016/0167-6377(82)90006-2.
Charnes, A. (1952). Optimality and degeneracy in linear programming. *Econometrica, 20*, 160–170.
Cheung, Y.-L. (2013). *Preprocessing and reduction for semidefinite programming via facial reduction: Theory and practice*. University of Waterloo Ph.D. thesis..

Cheung, Y.-L., Schurr, S., & Wolkowicz, H. (2013). Preprocessing and regularization for degenerate semidefinite programs. In D. Bailey, H. Bauschke, P. Borwein, F. Garvan, M. Thera, J. Vanderwerff, & H. Wolkowicz (Eds.), *Computational and analytical mathematics, in honor of Jonathan Borwein's 60th birthday*. In *Springer proceedings in mathematics & statistics: vol. 50* (pp. 225–276). Springer. http://www.optimization-online.org/DB_HTML/2011/02/2929.html

Chvátal, V. (1983). Linear programming. *A series of books in the mathematical sciences*. W. H. Freeman and Company, New York.

Dantzig, G. (1963). *Linear programming and extensions*. Princeton, New Jersey: Princeton University Press.

Dantzig, G., Orden, A., & Wolfe, P. (1955). The generalized simplex method for minimizing a linear form under linear inequality restraints. *Pacific Journal of Mathematics, 5*, 183–195.

Deza, A., Nematollahi, E., Peyghami, R., & Terlaky, T. (2006). The central path visits all the vertices of the Klee–Minty cube. *Optimization Methods and Software, 21*(5), 851–865. https://doi.org/10.1080/10556780500407725.

Deza, A., Nematollahi, E., & Terlaky, T. (2008). How good are interior point methods? Klee–Minty cubes tighten iteration-complexity bounds. *Mathematical Programming, 113*(1, Ser. A), 1–14. https://doi.org/10.1007/s10107-006-0044-x.

Dolan, E., & Moré, J. (2002). Benchmarking optimization software with performance profiles. *Mathematical Programming, 91*(2, Ser. A), 201–213.

Drusvyatskiy, D., Li, G., & Wolkowicz, H. (2017). A note on alternating projections for ill-posed semidefinite feasibility problems. *Mathematical Programming, 162*(1–2, Ser. A), 537–548. https://doi.org/10.1007/s10107-016-1048-9.

Drusvyatskiy, D., & Wolkowicz, H. (2017). The many faces of degeneracy in conic optimization. *Foundations and Trends® in Optimization, 3*(2), 77–170. https://doi.org/10.1561/2400000011.

Forsgren, A., Gill, P., & Wong, E. (2015). Primal and dual active-set methods for convex quadratic programming. *Mathematical Programming, 159*(1–2), 469–508.

Freund, R., & Ordonez, F. (2005). On an extension of condition number theory to nonconic convex optimization. *Mathematics of Operations Research, 30*(1), 173–194.

Freund, R., & Vera, J. (1997). Some characterizations and properties of the "distance to ill-posedness" and the condition measure of a conic linear system. *Technical report*. Cambridge, MA: MIT.

(1993). In T. Gal (Ed.), *Degeneracy in optimization problems*. Baltzer Science Publishers BV, Bussum. Ann. Oper. Res. **4**6/47 (1993), no. 1–4

Gauvin, J. (1977). A necessary and sufficient regularity condition to have bounded multipliers in nonconvex programming. *Mathematical Programming, 12*(1), 136–138.

Gauvin, J. (1995). Degeneracy, normality, stability in mathematical programming. In *Recent developments in optimization (DIJON, 1994)*. In *Lecture notes in econom. and math. systems: vol. 429* (pp. 136–141). Berlin: Springer.

Goldman, A., & Tucker, A. (1956). Theory of linear programming. In *Linear inequalities and related systems* (pp. 53–97). Princeton, N.J.: Princeton University Press. Annals of mathematics studies, no. 38

Gondzio, J. (1997). Presolve analysis of linear programs prior to applying an interior point method. *INFORMS Journal on Computing, 9*(1), 73–91.

Gonzalez-Lima, M., Wei, H., & Wolkowicz, H. (2009). A stable primal-dual approach for linear programming under nondegeneracy assumptions. *Computational Optimization and Applications, 44*(2), 213–247. https://doi.org/10.1007/s10589-007-9157-2.

Gould, N., & Scott, J. (2016). A note on performance profiles for benchmarking software. *ACM Transactions on Mathematical Software, 43*(2), 1–5.

GüLer, O., Den Hertog, D., Roos, C., Terlaky, T., & Tsuchiya, T. (1993). Degeneracy in interior point methods for linear programming: A survey. *Annals of Operations Research, 46/47*(1–4), 107–138. Degeneracy in optimization problems

Hall, J., & McKinnon, K. (2004). The simplest examples where the simplex method cycles and conditions where EXPAND fails to prevent cycling. *Mathematical Programming, 100*(1, Ser. B), 133–150. https://doi.org/10.1007/s10107-003-0488-1.

Huang, X. (2004). Preprocessing and postprocessing in linear optimization. Master's thesis. McMaster University.

Im, J., & Wolkowicz, H. (2021). A strengthened Barvinok–Pataki bound on SDP rank. *Operations Research Letters, 49*(6), 837–841. 11 pages, accepted Aug. 2021

Mangasarian, O. L., & Fromovitz, S. (1967). The Fritz John necessary optimality conditions in the presence of equality and inequality constraints. *Journal of Mathematical Analysis and Applications, 17*, 37–47.

Megiddo, N. (1986). A note on degeneracy in linear programming. *Mathematical Programming, 35*(3), 365–367. https://doi.org/10.1007/BF01580886.

Mészáros, C., & Suhl, U. (2003). Advanced preprocessing techniques for linear and quadratic programming. *OR Spectrum, 25*(4), 575–595. https://doi.org/10.1007/s00291-003-0130-x.

Mészáros, C., & Suhl, U. (2003). Advanced preprocessing techniques for linear and quadratic programming. *OR Spectrum, 25*, 575–595. https://doi.org/10.1007/s00291-003-0130-x.

Pataki, G. (1998). On the rank of extreme matrices in semidefinite programs and the multiplicity of optimal eigenvalues. *Mathematics of Operations Research, 23*(2), 339–358.

Permenter, F. (2017). *Reduction methods in semidefinite and conic optimization*. Massachusetts Institute of Technology Ph.D. thesis..

Peterson, D. (1973). A review of constraint qualifications in finite-dimensional spaces. *SIAM Review, 15*, 639–654. https://doi.org/10.1137/1015075.

Renegar, J. (1994). Some perturbation theory for linear programming. *Mathematical Programming, 65*(1, Ser. A), 73–91.

Ryan, D. M., & Osborne, M. R. (1988). On the solution of highly degenerate linear programmes. *Mathematical Programming, 41*, 385–392.

Schork, L., & Gondzio, J. (2020). Rank revealing Gaussian elimination by the maximum volume concept. *Linear Algebra and Its Applications, 592*, 1–19. https://doi.org/10.1016/j.laa.2019.12.037.

Sremac, S. (2019). *Error bounds and singularity degree in semidefinite programming*. University of Waterloo Ph.D. thesis.. http://www.math.uwaterloo.ca/~hwolkowi/henry/reports/stefanphdthesisdec.pdf

Sremac, S., Woerdeman, H., & Wolkowicz, H. (2021). Error bounds and singularity degree in semidefinite programming. *SIAM Journal on Optimization, 31*(1), 812–836. https://doi.org/10.1137/19M1289327.

Sturm, J. (2000). Error bounds for linear matrix inequalities. *SIAM Journal on Optimization, 10*(4), 1228–1248. https://doi.org/10.1137/S1052623498338606. (electronic)

Terlaky, T., & Zhang, S. (1993). Pivot rules for linear programming: A survey on recent theoretical developments. *Annals of Operations Research, 46/47*(1–4), 203–233. https://doi.org/10.1007/BF02096264. Degeneracy in optimization problems

Wolfe, P. (1959). The simplex method for quadratic programming. *Econometrica, 27*(3), 382–398.

Wright, S. (1996). *Primal-dual interior-point methods*. Philadelphia, PA: Society for Industrial and Applied Mathematics (SIAM).

Ye, Y., Todd, M., & Mizuno, S. (1994). An $\mathcal{O}(\sqrt{n}L)$–iteration homogeneous and self–dual linear programming algorithm. *Mathematics of Operations Research, 19*, 53–67.