A new warmstarting strategy for the primal-dual column generation method

Jacek Gondzio* Pablo González-Brevis†

School of Mathematics, University of Edinburgh
The King's Buildings, Edinburgh, EH9 3JZ, UK
Technical Report ERGO 12-007,
June 22, 2012.

Keywords: interior point methods, warmstarting, column generation, linear programming, cutting stock problem

Abstract

This paper presents a new warmstarting technique in the context of a primal-dual column generation method applied to solve a particular class of combinatorial optimization problems. The technique relies on calculating an initial point and on solving auxiliary linear optimization problems to determine the step direction needed to fully restore primal and dual feasibilities after new columns arrive. Conditions on the maximum size of the cuts and on a suitable initial point are discussed. Additionally, the strategy ensures that the duality gap of the warmstart is bounded by the old duality gap multiplied with a (small) constant, which depends on the relation between the old and modified problems. Computational experiments demonstrate the gains achieved when compared to a coldstart approach.

1 Introduction

Starting with the seminal work of Karmarkar [16] the interest and developments in interior point methods (IPMs) have continued over the last decades. Among these methods, a family of primal-dual ones have proven to be the most important and widely used [23]. Despite two decades of successful research concerning this class of methods, there still exists an open question when solving consecutive (closely) related problems: how to efficiently warmstart?

The need of solving subsequent and (closely) related linear programming problems arises in many situations. In numerous applications the instance data is subject to variations and therefore, taking advantage of the previously solved problem could lead to savings in solving the modified problem. Some examples can be found in marketing plans with changing demands or customer preferences or in engineering design with modifying product specifications. Other examples of such situation arise when column generation, cutting plane methods or branching techniques are used in the context of integer programming. Each of these aforementioned situations is based on solving related subproblems iteratively until some conditions hold. The differences between two subsequent problems under these schemes are the addition or elimination of variables/constraints. Note that under any of these schemes the size of the

*School of Mathematics, University of Edinburgh, Scotland, United Kingdom. Email: J.Gondzio@ed.ac.uk.
†School of Mathematics, University of Edinburgh, Scotland, United Kingdom. Email: P.Gonzalez-Brevis@sms.ed.ac.uk.
Supported by Facultad de Ingeniería, Universidad del Desarrollo and Beca Presidente de la República, Chile.

problem (number of variables/constraints) is modified unlike the first examples in which only the data is modified but the dimension of the problem remains the same.

In this paper a new warmstarting technique applicable in the context of primal-dual column generation method is presented. A thorough theoretical analysis of the new approach is performed and a computational experience is given to provide evidence of its good behaviour in practice.

Warmstarting is understood as the use of previous information gathered in the solution process of a given problem in order to solve a subsequent related problem. The aim of a warmstarting strategy is to solve the modified problem more efficiently than when no prior information is considered, known as the coldstart approach. The difficulty of interior point methods to re-optimize when compared to active set methods is a well-known issue [2]. It results from a completely different way in which IPMs approach optimality. To guarantee fast convergence IPMs traverse the interior point of the feasible set (by exploiting the notion of central path and forcing the iterates to stay in its neighbourhood) and approach the boundary of the feasible region only close to termination. Operating in the interior of the feasible set is the great advantage of IPMs responsible for their spectacular efficiency [23]. However it becomes a curse if one tries a naive warmstarting of IPMs. In general IPMs should not warmstart from the *optimal* solution of the previous problem because such point is too close to the boundary of the feasible set and is very likely to be far away from the new central path (leading to the optimal solution of the new problem) [10]. To overcome these difficulties, several strategies have been proposed in the literature.

Freund [6] has proposed a shifted barrier method for solving linear programs in standard-form from an infeasible warmstart. He has shown that under suitable assumptions his potential function reduction algorithm runs in polynomial time. In the same spirit as Freund, Benson and Shanno [2] have presented an exact primal-dual penalty approach. The authors have introduced a method which relaxes the non-negativity constraints of primal and dual problems and penalizes for any violation of them. Their exact method accounts for small perturbations and requires the setting of some penalization parameters which may be a non-trivial task. A similar approach has been presented in [5]. The authors have reformulated the primal-dual pair using a slacked form (without penalization terms). Theoretical conditions under which warmstarted IPMs perform better than coldstart IPMs are presented. The authors have tested their approach when the original data is perturbed (linear programming) and in a cutting plane context (combinatorial optimization). One of the main advantages of the proposed strategy is that it does not require extensive setting of parameters and is readily applicable to a wide variety of problems. Note that the methods presented in [2, 5, 6] modify the original problem adding auxiliary variables. These methods aim to drive these auxiliary variables to zero obtaining the optimal solution of the original problem. Another class of warmstarting methods calculates a warmstarting point and re-optimizes the modified problem from this point.

Mitchell [20] has studied different strategies to obtain an interior point when adding cutting planes and variables to a linear problem. His methods rely on calculating directions based on the projection onto the null space of the constraint matrix. A positive step length is always available so the methods recover feasibility in one step. The methods have been implemented using the primal projective standard-form variant of Karmarkar's algorithm for linear programming within a cutting plane method applied to matching problems [20, 21]. Additionally, Mitchell and Borchers [19] have studied a different and practical warmstarting strategy using the primal-dual barrier method applied to solve linear ordering problems.

Yildirim and Wright [24] have analysed different criteria to determine a well-suited warmstarting point. They have compared several different ways of direction searching (*i.e.,* (weighted) least-square approach and Newton step) and provided complexity and convergence results for each of these methods after perturbing the instance data. The authors have determined the size of the perturbation that can be

absorbed by each of these methods. Numerical experiments and further developments in this direction have been presented by John and Yildirim in [15].

A different approach has been presented by Gondzio [10] in a primal-dual cutting plane method context. The author has proposed to fix the initial values of the new components and then restore independently primal and dual feasibilities. To calculate the warmstarting point, modified Newton steps have to be performed. The author has distinguished between deep and shallow cuts adjusting his strategy accordingly. Although a successful implementation has been developed, no theoretical guarantee for its performance has been provided.

In [12], Gondzio and Grothey have introduced a primal-dual interior point method that relies on multiple-centrality corrector techniques to find a warmstarting solution. From this point, their method seeks feasibility. The authors have shown under which conditions their short and long step path following methods can absorb data perturbations in one step or in few steps (for larger perturbations). They have extended their analysis to problems with special structures (primal and dual block angular structures). The same authors have studied an unblocking strategy based on sensitivity analysis [13]. The main idea is to overcome the blocking issue that any advanced starting point may suffer from by increasing the size of the step allowed in the Newton direction. First, the proposed strategy identifies which components the blocking originates from. Then, the method aims to remove the blocking components using sensitivity analysis. Conditions that an unblocking direction must satisfy and proofs of the existence of such direction have been provided. The authors have observed that a large value of the duality measure can absorb more infeasibility in one step than a small value and well-centred point can allow an IPM to absorb more infeasibility.

A very different warmstarting approach has recently been proposed by Skaaja et al. [22] who employ the self-dual embedding linear programming model which facilitates taking any point as a warmstarting candidate, including points closed to the boundary of the feasible region.

In [5, 10] numerical experiments have been carried out to demonstrate how efficient these methods are when compared to coldstart in a cutting plane framework and encouraging results have been provided.

In this paper we will focus on the case when warmstarting is applied in column generation, specifically in the context of the primal-dual column generation method (PDCGM) described in [11]. We address a particularly important application of the column generation method which arises when solving combinatorial optimization problems. Under suitable assumptions, the algorithm restores primal and dual feasibilities after the addition of new columns in the primal space. The method relies on two linear optimization problems to calculate the direction which recovers primal and dual feasibilities in one step. The direction is determined so that the small components at a particular solution are not largely modified. The strategy allows to have control over the new duality gap by expanding the neighbourhood of the central path. The analysis is performed for the symmetric neighbourhood of the central path. Conditions are derived for a suitable warmstarting point such that for arbitrarily deep cuts in the dual space (columns in the primal) primal and dual feasibilities are restored in one step. An extra care is taken of the "centrality" of the new point and conditions which guarantee that the new iterate belongs to an expanded symmetric neighbourhood of the central path for the new problem are established.

The structure of the remaining sections in this paper is the following. In Section 2, we review the fundamentals of primal-dual methods. In Section 3, we briefly describe column generation methods and its primal-dual version. In Section 4, we present our specialized warmstarting idea providing a theoretical analysis which demonstrates how the algorithm deals with infeasibilities and the proximity to the central path. Computational experiments for solving the root node of the cutting stock problem after applying Dantzig-Wolfe decomposition comparing the proposed strategy with a coldstart strategy are presented in Section 5. In Section 6 the main contributions of this study are summarized.

2 Primal-dual interior point methods

In this section we introduce the notation used along the paper and the fundamental ideas behind primal-dual interior point methods. Let us consider a linear programming problem represented by the following primal-dual pair

$$\mathcal{P}_0 := \min \quad c^T x, \quad \text{s.t.} \;\; Ax = b, \;\; x \geq 0, \tag{2.1a}$$

$$\mathcal{D}_0 := \max \quad b^T y, \quad \text{s.t.} \;\; A^T y + s = c, \;\; s \geq 0, \tag{2.1b}$$

where $x \in \mathbb{R}^n$ is the vector of primal variables, $y \in \mathbb{R}^m$ and $s \in \mathbb{R}^n$ are the vectors of dual variables. $A \in \mathbb{R}^{m \times n}$ represents the coefficient matrix, where $rank(A) = m \leq n$, and $c \in \mathbb{R}^n$ and $b \in \mathbb{R}^m$ are vectors of parameters. Let us define the μ-perturbed KKT system associated to the primal-dual pair (2.1) as

$$\begin{aligned} Ax &= b && \text{(2.2a)} \\ A^T y + s &= c && \text{(2.2b)} \\ XSe &= \mu e, && \text{(2.2c)} \\ (x, s) &> 0, && \text{(2.2d)} \end{aligned}$$

where $X = diag\{x_1, x_2, ..., x_n\}$, $S = diag\{s_1, s_2, ..., s_n\}$, $e = \{1, 1, ..., 1\}$ in appropriate dimension and μ is defined as the barrier parameter which defines the central path. Equations (2.2a) and (2.2b) are the linear constraints corresponding to primal and dual feasibility, respectively, and equations (2.2c) are the perturbed complementarity conditions, which are mildly non-linear.

In a primal-dual interior point method, an approximate solution to the perturbed KKT system (2.2) is obtained at each iteration by using a (damped) step of a Newton-like system of equations. The step direction is defined by the vector $(\Delta x, \Delta y, \Delta s)$ which is obtained by solving the following system of equations

$$\begin{bmatrix} A & 0 & 0 \\ 0 & A^T & I \\ S^l & 0 & X^l \end{bmatrix} \begin{bmatrix} \Delta x^l \\ \Delta y^l \\ \Delta s^l \end{bmatrix} = \begin{bmatrix} \xi_p^l \\ \xi_d^l \\ \tau \mu_l e - X^l S^l e \end{bmatrix}, \tag{2.3}$$

where the primal-dual solution at iteration l is $(x, y, s) = (x^l, y^l, s^l)$, and $\xi_p^l = b - Ax^l$ and $\xi_d^l = c - A^T y^l - s^l$ are the primal and dual residuals at iteration l, respectively and $\tau \in (0, 1)$ is the centering parameter.

Once direction $(\Delta x^l, \Delta y^l, \Delta s^l)$ is determined from (2.3) and since the next iterate is defined as $(x^{l+1}, y^{l+1}, s^{l+1}) = (x^l + \alpha_P \Delta x^l, y^l + \alpha_D \Delta y^l, s^l + \alpha_D \Delta s^l)$ suitable values for $\alpha_P \in [0, 1]$ and $\alpha_D \in [0, 1]$ are calculated bearing in mind that $(x^{l+1}, s^{l+1}) > 0$. After this, μ_{l+1} is updated (usually reduced) and the primal-dual method continues iterating until a relative duality gap drops below a prescribed optimality tolerance ε. The stopping criterion is usually defined as

$$\frac{c^T x - b^T y}{1 + |c^T x|} \leq \varepsilon. \tag{2.4}$$

One of the most remarkable characteristics of primal-dual methods is that the distance to optimality, called the duality gap, can be calculated if some conditions hold. The analytic μ-centre, $(x(\mu), y(\mu), s(\mu))$, is defined as the unique point satisfying conditions (2.2a)-(2.2c) and $(x(\mu), s(\mu)) > 0$ for a particular $\mu > 0$. The duality gap can be calculated as

$$c^T x - b^T y = c^T x - (Ax)^T y = x^T (c - A^T y) = x^T s = n\mu.$$

3 Column generation

In this section we briefly describe the ideas behind column generation. For a more detailed explanation of the method, see [17]. The aim of a column generation method is to solve a problem called the master problem (MP) through a restricted version of it (*e.g.*, a problem with fewer variables). Initial columns are appended to construct the restricted master problem (RMP) usually via a heuristic procedure, a known feasible solution or using artificial columns. Then, the RMP is solved and dual solutions of the RMP are used in the oracle. The oracle, also known as the subproblem, is a problem which usually hides the complexity of the original problem (*i.e.*, integrality). After solving the oracle new columns with negative reduced costs are obtained and added to the RMP, if any. This process continues until the stopping criterion is satisfied. The solution of the last RMP provides the optimal solution of the MP. Note that from one iteration to another we would like to take advantage of the solution process of the previous RMP in order to speed up the solution process of the new RMP after adding new columns.

Column generation is widely applied in the context of mixed-integer programming. In this paper we analyse this method after relaxing a mixed-integer programming problem using Dantzig-Wolfe decomposition (DWD). Loosely speaking, DWD aims to represent the feasible set of an integer problem using its convex hull and optimizes over this set. This convex hull is described by a linear convex combination of extreme points and extreme rays. We are interested in the case when the instances have a non-empty feasible region and therefore, only extreme points are needed to describe the convex hull of the original problem. Also, we consider a relaxation of the integer variables of the original problem (*i.e.*, root problem in a branch-and-bound context) and thus, we have a sequence of linear optimization problems to solve. Since the number of extreme points is likely to be large and we only need to describe the convex hull in the region close to the optimal solution, one should rely on methods that exploit this fact. Column generation is such method.

Let us assume that we are in an intermediate iteration of the column generation process. The current RMP (2.1) characterized by A, b, c has been solved and the oracle generates k new columns with parameters $(\bar{A}, \bar{c})$. Note that $\bar{c}_j - \bar{A}_j^T y < 0$ for all $j \in K = \{1, \ldots, k\}$ for a dual solution y of the current RMP. The modified primal-dual pair is

$$\mathcal{P}_1 := \min \quad c^T x + \bar{c}^T \bar{x}, \quad \text{s.t.} \;\; Ax + \bar{A}\bar{x} = b, \;\; x \geq 0, \;\; \bar{x} \geq 0, \tag{3.1a}$$

$$\mathcal{D}_1 := \max \quad b^T y, \quad \text{s.t.} \;\; A^T y + s = c, \;\; \bar{A}^T y + \bar{s} = \bar{c}, \;\; s \geq 0, \;\; \bar{s} \geq 0, \tag{3.1b}$$

where k is the number of columns (variables) added to the original problem (2.1), $\bar{x} \in \mathbb{R}^k$ is the vector of new primal variables and $\bar{s} \in \mathbb{R}^k$ is the vector of new dual slack variables. $\bar{A} \in \mathbb{R}^{m \times k}$ represents the coefficient matrix for the new variables and $\bar{c} \in \mathbb{R}^k$ the vector of objective function coefficients. Note that new columns in the primal space are associated to new constraints in the dual space. Constraints in the dual space can be interpreted as the cuts which restrict the dual localization set. From now on we will use both, columns and cuts where the former term refers to the primal space while the latter to the dual. After dropping the iteration index l and similar to (2.3), the new Newton-like system of equations is

$$\begin{bmatrix} A & \bar{A} & 0 & 0 & 0 \\ 0 & 0 & A^T & I & 0 \\ 0 & 0 & \bar{A}^T & 0 & I \\ S & 0 & 0 & X & 0 \\ 0 & \bar{S} & 0 & 0 & \bar{X} \end{bmatrix} \begin{bmatrix} \Delta x \\ \Delta \bar{x} \\ \Delta y \\ \Delta s \\ \Delta \bar{s} \end{bmatrix} = \begin{bmatrix} \xi_{\bar{b}} \\ \xi_c \\ \xi_{\bar{c}} \\ \tau\mu e - XSe \\ \tau\mu e - \bar{X}\bar{S}e \end{bmatrix}, \tag{3.2}$$

where $\xi_{\bar{b}} = b - Ax - \bar{A}\bar{x}$, $\xi_{\bar{c}} = \bar{c} - \bar{A}^T y - \bar{s}$, $\bar{X} = diag\{\bar{x}_1, \ldots, \bar{x}_k\}$, $\bar{S} = diag\{\bar{s}_1, \ldots, \bar{s}_k\}$, $\mu = (x^T s +$

$\bar{x}^T\bar{s})/(n+k)$ and $\tau \in (0,1)$ is the centering parameter.

The primal-dual method under which we design our warmstarting strategy is the one described in [11]. This technique relies on well-centred close-to-optimal solutions of the RMPs which are obtained by a primal-dual interior point method. This method takes into account that solving the RMP to optimality is not needed in the early stages of the column generation procedure as it was also pointed out in [8, 18, 21]. Loosely speaking, a RMP has to be solved to optimality only at the last iteration of the column generation. An important feature of this method is that even that in one iteration the algorithm may not provide any new column, it guarantees the progress by decreasing the required distance to optimality. Termination is achieved when there are no more columns to append and the relative gap drops below a prescribed optimality tolerance. The method has shown encouraging results in solving the root node of several combinatorial optimization problems [11]. Despite the fact the warmstarting strategy presented in this paper was proposed to improve the performance in this specific context, it could be easily extended to any situation where a linear optimization problem is solved using a column generation or cutting plane framework with a primal-dual interior point method.

4 Warmstarting strategy

In this paper we introduce a specialized warmstarting strategy applicable within the primal-dual column generation scheme [11]. Specialized because we take advantage of some structure/properties which are often present when solving combinatorial optimization problems. We have observed that in many applications all the elements in A and $\bar{A}$ are non-negative. For instance, columns in the primal space describing cutting patterns (cutting stock problem), routes (vehicle routing problem) or production plans (lot sizing problem) by definition contain only non-negative entries. In our developments we do consider this observation and therefore, the analysis and results should be understood in this context.

The process of finding a warmstarting point has been divided into two stages. In the first, we need to find a point from a list of stored iterates which satisfies some properties. In the other, an adjustment $(\Delta x, \Delta y, \Delta s)$ has to be computed taking this point as a reference. After this, the step is taken in this direction and the old point is updated in the new dimensions to produce a full-dimension warmstarting point. From this point we continue iterating and solving system (3.2). Our aim is to find a starting point which reduces the number of iterations required to solve problem (3.1) when compared to a coldstart approach. To develop such a two-stage approach we need to deal with two problems: (a) feasibility in the primal and dual spaces and (b) centrality of the new warmstarting point. We address both issues later.

Despite the fact that our developments follow the idea of Gondzio in [10] of treating primal and dual infeasibilities separately, our approach differs in two key aspects. Firstly, in the choice of $\bar{x}$ and $\bar{s}$. The aforementioned paper sets $\bar{s}_j = \max(|\bar{c}_j - \bar{A}_j^T y|, \mu^{1/2})$ for every added cut j. Since it is likely to have (deep) cuts where $\bar{c}_j - \bar{A}_j^T y \ll 0$, $\bar{s}_j$ usually takes large values. Furthermore, since the complementarity products of the new variables are set to $\bar{x}_j\bar{s}_j = \mu$, for every $j = 1, 2, \ldots, k$, such strategy is likely to produce small values of $\bar{x}_j$. This goes against the expectations that the new variables corresponding to the recently appended (deep) cuts are likely to take nonzero values at the optimal solution. In our strategy we do not choose $\bar{x}_j$ as a function of $\bar{s}_j$. Instead, we choose these values bearing in mind their impact on centrality.

A second difference between the strategy of [10] and ours is in the definition of the search direction to find the new warmstarting point. In the method proposed in [10], the search direction is calculated using a variation of system (3.2) via target t, which is a consequence of the choice of $\bar{x}_j$ and $\bar{s}_j$. In the primal space, feasibility is restored by choosing small $\bar{x}_j$'s. In the dual space, the method ensures that after a

full step is taken in this direction, dual feasibility is recovered in the new components. However, this method does not guarantee the same for the old components. It is accepted that the complementarity products for the old variables may get significantly worse as a price to pay for recovering primal and dual feasibilities. In our proposed strategy, we aim to restore primal and dual feasibilities using auxiliary linear optimization problems. The principal aim of both, primal and dual auxiliary problems, is to seek feasibility while minimizing changes in old variables which are small in order to avoid large changes in the complementarity products. Our approach follows a variation of a weighted least-square strategy proposed in [24]. It differs from [24] in the sense that we only consider small variables and that we have extra constraints to satisfy in both, primal and dual spaces.

We have bounded the depth of the cut to ensure that primal and dual feasibilities are recovered in the old and new components and to prevent the complementarity products from becoming excessively large. In what follows of this Section, we analyse restoring primal and dual feasibilities, effects in complementarity products and describe the complete algorithm.

Let (x^0, y^0, s^0) be a feasible solution of the primal-dual pair (2.1). Also, let us define the primal-dual strictly feasible set and the central path neighbourhood as follows

$$\mathcal{F}^0 = \{(x, y, s) : Ax = b; A^T y + s = c; (x, s) > 0\},$$
$$\mathcal{N}_s(\gamma) = \{(x, y, s) \in \mathcal{F}^0 : \gamma\mu \leq x_j s_j \leq \frac{1}{\gamma}\mu; \forall j = 1, 2, ..., n\}.$$

We are interested in $(x^0, y^0, s^0) \in \mathcal{N}_s(\gamma)$ for a fixed $\gamma \in (0, 1)$. The reader must note that our definition of the neighbourhood is a slight modification of the wide neighbourhood $\mathcal{N}_{-\infty}(\gamma)$ in [23] proposed in [12]. The aim of our specialized warmstarting algorithm is to find an initial point from where to start solving the modified problem (3.1). This initial point is obtained from information gathered when solving problem (2.1) via system of equations (2.3) with $\xi_b = \xi_c = 0$.

Before continuing, let $N = \{1, \ldots, n\}$, $M = \{1, \ldots, m\}$ and $K = \{1, \ldots, k\}$ be index sets. Additionally, let us state the following general assumptions.

(A.1) All elements of $A, \bar{A}$ are non-negative and $\mathcal{O}(1)$. Additionally, $\sigma_{max}(A)$ and $\sigma_{min}(A)$ denote the maximum and minimum singular value of matrix A respectively, which also are $\mathcal{O}(1)$.

(A.2) Both, the old and new RMPs, are bounded. This means that $\exists\, \mathcal{U}_v : ||x, s||_\infty \leq \mathcal{U}_v$.

(A.3) $\mathcal{U}_v^2 > 1$.

(A.4) We are able to store a list of iterates which are strictly feasible and well-centred (*i.e.*, $(x^0, y^0, s^0) \in \mathcal{N}_s(\gamma)$).

Using our definition of the central path neighbourhood for $(x^0, y^0, s^0) \in \mathcal{N}_s(\gamma)$, we have

$$\gamma\mu_0 \leq x_j^0 s_j^0 \leq \frac{1}{\gamma}\mu_0 \quad \forall j \in N. \tag{4.1}$$

Both, this definition and the upper bound $\mathcal{U}_v$, allow us to bound x_j^0 and s_j^0, for every $j \in N$, as follows

$$\frac{\gamma\mu_0}{\mathcal{U}_v} \leq \quad x_j^0 \quad \leq \quad \mathcal{U}_v, \tag{4.2a}$$
$$\frac{\gamma\mu_0}{\mathcal{U}_v} \leq \quad s_j^0 \quad \leq \quad \mathcal{U}_v. \tag{4.2b}$$

The similarities between the initial RMP and the modified RMP are one of the key elements to take into consideration when a warmstarting strategy is designed for an interior point method. If there is no

similarity between the initial and modified problems, we can expect that a warmstarting strategy will not lead to any substantial improvement when compared with a coldstart approach. This could be the case when the new RMP has a completely different feasible region around the optimal solution and a close-to-optimality solution deeply violates the new constraints. Therefore, any information previously gathered close to the optimal solution will not help to speed up the solution process of the modified problem. Then, there is a need to understand the relation between the added cuts and the previous RMP. Let us define the depth of cut $j \in K$ as $|\bar{c}_j - \bar{A}_j{}^T y^0|$, where y^0 is known and $\bar{A}_j$ represents the j-th column of matrix $\bar{A}$. A sensible way to proceed is to measure the size of the newly added cuts in terms of the current penalty parameter, μ_0. Let the inequality $|\bar{c}_j - \bar{A}_j{}^T y^0| \leq f(\mu_0)$ be used to determine the depth of the cuts, where $f(\mu_0)$ is an increasing function of μ_0. Note that by using this idea of associating the depth of the cuts to μ_0 and having a list of iterates, we could retreat far enough in the iteration process to make the cuts relatively shallower. In other words, for all the cuts we could choose a suitable large enough μ_0 to decrease its relative depth.

As mentioned earlier, the choices of $\bar{x}$ and $\bar{s}$ have important consequences in the primal-dual infeasibilities and in the complementarity conditions. We aim to find a warmstart which: (a) is feasible in the primal and dual space; and (b) keeps the complementarity products reasonably small and inside a slightly modified neighbourhood if the cuts satisfy some desirable properties. We expect the duality gap to increase since we are adding variables/constraints to the primal/dual problem. However, we would like to keep this duality measure relatively close to the old one.

Now, let us define some sets and parameters used throughout the paper.

Definition 4.1 *Let $\mathcal{B}_0$ be the set containing all indices j such that $x_j^0 \geq s_j^0$, where $j \in N$. We call this set the "basic" partition at solution (x^0, y^0, s^0). Conversely, the "non-basic" partition at solution (x^0, y^0, s^0) is defined as $\mathcal{N}_0$ and contains all indices j such that $x_j^0 < s_j^0$, where $j \in N$.*

Definition 4.2 *Considering $A \in \mathbb{R}_+^{m \times n}$, let us define*

$$A_{min} := \min_{i \in M, j \in N: A_{ij} > 0} \{A_{ij}\},$$

as the minimum non-zero element of matrix A. Similarly,

$$a_{min} := \min_{i \in M: a_i > 0} \{a_i\},$$

is the minimum non-zero element of vector $a \in \mathbb{R}_+^m$.

Additionally, let us define

$$\sigma_{max} = \max\{\sigma_{max}(A), \sigma_{max}(\bar{A})\},$$

where $\sigma_{max}(H)$ denotes the largest singular value of H.

In the following three sections we will prove the main results of this paper. Namely, we will provide the methodology to choose a good warmstart solution such that the primal and dual feasibilities may be restored in one Newton step and will show (in Section 4.3) that this can be achieved without significantly affecting the proximity of the new iterate to the central path.

4.1 Dual feasibility

In order to minimize the impact of restoring dual feasibility and to measure its effect on complementarity products, we have defined an auxiliary linear optimization problem. Taking the second and third

equations of (3.2) and considering $(x^0, y^0, s^0) \in \mathcal{F}^0$, we have

$$A^T \Delta y + \Delta s = 0, \tag{4.3}$$
$$\bar{A}^T \Delta y + \Delta \bar{s} = \bar{c} - \bar{A}^T y^0 - \bar{s}. \tag{4.4}$$

Such a formulation allows for a considerable freedom in the choice of Δy. This system of equations is likely to have multiple solutions.

The following auxiliary linear optimization problem minimizes the relative change of variables in the "basic" partition (*i.e.*, corresponding to small s_j^0) when a solution of problem (2.1) is available.

$$\begin{aligned}
\text{minimize} \quad & \sum_{j \in \mathcal{B}_0} \frac{\Delta s_j}{s_j^0}, && && \text{(4.5a)}\\
\text{subject to} \quad & \sum_{i \in M} \bar{A}_{ij} \Delta y_i + \Delta \bar{s}_j = \bar{c}_j - \sum_{i \in M} \bar{A}_{ij} y_i^0 - \bar{s}_j, && \forall j \in K, && \text{(4.5b)}\\
& - \sum_{i \in M} A_{ij} \Delta y_i = \Delta s_j, && \forall j \in \mathcal{B}_0, && \text{(4.5c)}\\
& \Delta y_i \leq 0, && \forall i \in M, && \text{(4.5d)}\\
& \bar{s}_j^w \geq \gamma \mu_0 \frac{\bar{A}_{min}}{\sqrt{m}\, \mathcal{U}_v\, \sigma_{max}}, && \forall j \in K, && \text{(4.5e)}
\end{aligned}$$

where $\bar{s}_j^w = \Delta \bar{s}_j + \bar{s}_j$, for every $j \in K$. Observe that by solving this linear problem, feasibility for the new components is achieved via constraints (4.5b) (the third equation of system (3.2)). Note that the aim of this linear program is to minimize the relative change of small slacks of the dual problem while a full step is feasible for all components. The motivation behind this objective is that we would like to perturb the previous solution as little as possible. The linear problem (4.5) can be simplified by the following steps. First, let us eliminate Δs_j for every $j \in \mathcal{B}_0$ by using constraints (4.5c) and substituting $\Delta y = -\Delta \bar{y}$. Also, let us introduce a new parameter f_i for every $i \in \mathcal{M}$ such that $f_i = \sum_{j \in \mathcal{B}_0} \frac{A_{ij}}{s_j^0}$ for every $j \in K$. Additionally, let us set

$$\bar{s}_j = \gamma \mu_0 \frac{\bar{A}_{min}}{\sqrt{m}\, \mathcal{U}_v\, \sigma_{max}}, \quad \forall j \in K. \tag{4.6}$$

Now, we can rewrite problem (4.5) as

$$\begin{aligned}
\mathcal{D}_{aux} := \text{minimize} \quad & \sum_{i \in M} f_i \Delta \bar{y}_i, && && \text{(4.7a)}\\
\text{subject to} \quad & \sum_{i \in M} \bar{A}_{ij} \Delta \bar{y}_i - \Delta \bar{s}_j = -(\bar{c}_j - \sum_{i \in M} \bar{A}_{ij} y_i^0 - \bar{s}_j), && \forall j \in K, && \text{(4.7b)}\\
& \Delta \bar{y}_i \geq 0, && \forall i \in M, && \text{(4.7c)}\\
& \Delta \bar{s}_j \geq 0, && \forall j \in K. && \text{(4.7d)}
\end{aligned}$$

To avoid unbounded solutions, if $\mathcal{B}_0 = \emptyset$, we define $f_i = 1$ for every $i \in M$. Observe that by assumption (A.1), $f_i \geq 0$ for every $i \in M$. Therefore, the problem (4.7) can be interpreted as finding the minimum adjustment $\Delta \bar{y}$ of dual variables y such that at the new point $y + \Delta y$ $(= y - \Delta \bar{y})$ all dual feasibility constraints in (3.1b), including the ones corresponding to new deep cuts, are satisfied. The objective function (4.7a) promotes such changes $\Delta \bar{y}$ which do not alter the "basic" components of s^0 (the components corresponding to small values s_j^0).

Lemma 4.3 *Given $\bar{c}_j - \bar{A}_j^T y^0 < 0$ for every $j \in K$, where $\bar{A}_j$ represents the j-th column of matrix $\bar{A}$, $\bar{A}_{ij} \geq 0$ for every $i \in M, j \in K$, problem (4.7) has a bounded solution.*

Proof. From (4.7b) we have

$$\bar{A}^T \Delta\bar{y} - \Delta\bar{s} = -(\bar{c} - \bar{A}^T y^0 - \bar{s}) = d. \tag{4.8}$$

Since $\bar{c}_j - \bar{A}_j^T y^0 < 0$ for every $j \in K$ and from definition (4.6), clearly $d_j > 0$ for every $j \in K$. Note that we could determine a feasible solution to problem (4.7) in the following way. For every $j \in K$, calculate $\Upsilon_j = d_j/(\bar{A}_j)_{min}$. Then, define $\Delta\bar{y}_i$ for every $i \in M$ as

$$\Delta\bar{y}_i = \max_{j \in K}\{\Upsilon_j\}. \tag{4.9}$$

This solution is feasible since it will produce $\Delta\bar{s}_j \geq 0$ for every $j \in K$. Also, observe that any value of $\Delta\bar{y}_i$ exceeding (4.9) will not improve the objective function and therefore, equation (4.9) determines an upper bound for the $\Delta\bar{y}$ components. In other words, and for every $i \in M$, $\Delta\bar{y}_i^\star \leq \max_{j \in K}\{\Upsilon_j\}$, where $\Delta\bar{y}_i^\star$ is the optimal solution of problem (4.7). Hence, we can conclude that the problem (4.7) has a bounded solution.

□

The next lemma states that by calculating direction $\Delta\bar{y}$ from (4.7) infeasibilities are absorbed in the old components and a dual feasible point for the new problem is obtained.

Lemma 4.4 *Let $(x^0, y^0, s^0) \in \mathcal{F}^0$. If $\Delta\bar{y}$ is chosen by solving (4.7) then a full step in the old components is feasible. Therefore, dual feasibility is restored in one step.*

Proof. From the definition of problem (4.7), we have $\Delta\bar{y}_i \geq 0$ for every $i \in M$ and therefore, $\Delta y_i \leq 0$ for every $i \in M$. Since $A_{ij} \geq 0$ for every $j \in N$ and $i \in M$ and using (4.3), we have:

$$\Delta s_j = -\sum_{i \in M} A_{ij} \Delta y_i \geq 0, \quad \forall j \in N.$$

Recalling that $s_j^0 > 0$, $s_j^0 + \Delta s_j > 0, \forall j \in N$ as required.

□

Even though our objective is to minimize the variation of the dual elements in the "basic" partition, we cannot guarantee this variation to be small. This is an unavoidable consequence of the fact that there is no control of the depth of the new cuts and a large variation of some components Δs is expected. Despite the lack of control, we can still determine an upper bound for Δs in the old components. Note that from (4.3) and $\Delta y = -\Delta\bar{y}$, we have

$$\Delta s = A^T \Delta\bar{y}, \tag{4.10}$$

Also, from (4.9) we have the following bound on $\Delta\bar{y}_i$ for every $i \in M$

$$\Delta\bar{y}_i \leq \frac{d_{max}}{\bar{A}_{min}}, \tag{4.11}$$

where d_{max} is defined as

$$d_{max} = \max_{j \in K} |\bar{c}_j - \sum_{i \in M} \bar{A}_{ij} y_i^0 - \bar{s}_j|.$$

Applying l_∞ norm to (4.10) and considering the upper bound of (4.11), we obtain

$$||\Delta s||_\infty \quad \leq \quad ||A^T||_\infty ||\Delta\bar{y}||_\infty.$$

$$\leq \frac{\sqrt{m}\ \sigma_{max}(A)\ d_{max}}{\bar{A}_{min}}.$$

Thus,

$$\Delta s_j \leq \frac{\sqrt{m}\ \sigma_{max}\ d_{max}}{\bar{A}_{min}}, \quad \forall j \in N. \tag{4.12}$$

Additionally, considering (4.8) and $\bar{c}_j - \bar{A}_j^T y^0 < 0$ for every $j \in K$, we get

$$\bar{s}_j + \Delta\bar{s}_j \leq \frac{\sqrt{m}\ \sigma_{max}\ d_{max}}{\bar{A}_{min}}, \quad \forall j \in K. \tag{4.13}$$

4.2 Primal feasibility

Similarly to recovering dual feasibility, we aim to restore feasibility in the primal space by solving an auxiliary linear optimization problem and using the notion of "basic" and "non-basic" partition. Considering the first equation of system (3.2) and $(x^0, y^0, s^0) \in \mathcal{F}^0$, we have

$$A\Delta x = -\bar{A}(\bar{x} + \Delta\bar{x}) \tag{4.14}$$

To simplify the notation, let us define $\bar{x}_j^w = \bar{x}_j + \Delta\bar{x}_j$ for every $j \in K$. Primal feasibility could be easily restored if we could set $\bar{x}^w = 0$. Since in interior point methods this is not possible, we need to fix or determine a positive value for $\bar{x}^w$. In practice, primal feasibility could still be easily achieved by setting $\bar{x}^w$ sufficiently small. We could apply for example the primal feasibility restoration direction proposed in [10], $\Delta x = -\Theta A^T (A\Theta A^T)^{-1} \bar{A}\bar{x}^w$ for non-degenerate systems where $\Theta = XS^{-1}$ is a diagonal scaling matrix. This is a generalization of Mitchell's direction presented in [20] and applied in the primal projective algorithm to handle multiple cuts. Setting a small value for $\bar{x}^w$ seems to be sensible since primal infeasibility depends on this value. We have designed a slightly different strategy which takes into account setting x_j^0 small but also considering centrality aspects. For now, it is enough to say that our choice of $\bar{x}^w$ is the maximum possible value in order to ensure that: (a) a full step in the primal direction is possible, and (b) primal feasibility is restored. Similar to the dual feasibility restoration, we have defined the following auxiliary linear optimization problem.

$$\begin{aligned}
\mathcal{P}_{aux} := \quad & \text{minimize} && \sum_{j \in \mathcal{N}_0} \frac{\Delta x_j^+ + \Delta x_j^-}{x_j^0}, && && \text{(4.15a)}\\
& \text{subject to} && \sum_{j \in N} A_{ij}\Delta x_j = -\sum_{j \in K} \bar{A}_{ij}\bar{x_j}^w, && \forall i \in M, && \text{(4.15b)}\\
& && \Delta x_j \geq x_j^0(\delta_l - 1), && \forall j \in N, && \text{(4.15c)}\\
& && \Delta x_j = \Delta x_j^+ - \Delta x_j^-, && \forall j \in \mathcal{N}_0, && \text{(4.15d)}\\
& && \Delta x_j^+ \geq 0, && \forall j \in \mathcal{N}_0, && \text{(4.15e)}\\
& && \Delta x_j^- \geq 0, && \forall j \in \mathcal{N}_0, && \text{(4.15f)}
\end{aligned}$$

where δ_l is a given parameter which satisfies $0 < \delta_l < 1$ and its meaning will be explained later in the paper. In the primal case and similar to what we did for the "basic" variables in the dual case, we minimize the relative change of the "non-basic" variables (corresponding to small x_j^0). Note that in this case we allow positive and negative directions and therefore we minimize the absolute value of such directions. Constraints (4.15b) guarantee that by taking a full step in direction Δx, infeasibility in the primal space is completely restored. Constraints (4.15c) ensure that if we take a full step in direction Δx, the new iterate will remain positive. Constraints (4.15d)-(4.15f) are additional requirements that help

to calculate the absolute value of every "non-basic" Δx_j. If $\Delta x_j > 0$, then $\Delta x_j^+ > 0$ and $\Delta x_j^- = 0$. If $\Delta x_j < 0$, then $\Delta x_j^- > 0$ and $\Delta x_j^+ = 0$. Finally, if $\Delta x_j = 0$, then both Δx_j^+ and Δx_j^- are zero. Note that ensuring a non-empty feasible set in $\mathcal{P}_{aux}$ for general A and $\bar{A}$ is a non-trivial task. For instance, large values on $\bar{x}^w$ may lead to large negative values in some Δx_j and therefore, satisfying constraint (4.15c) may not be possible. Later we will define conditions to ensure that primal infeasibility is completely absorbed in one step.

Following [24], let us use the following QR factorization of A^T,

$$A^T = Q \begin{bmatrix} R \\ 0 \end{bmatrix} = [Q_1, Q_2] \begin{bmatrix} R \\ 0 \end{bmatrix} = Q_1 R, \tag{4.16}$$

where Q_1 is an $n \times m$ matrix, Q_2 is an $n \times (n-m)$ matrix, $[Q_1, Q_2]$ is a matrix with orthogonal columns, and R is an $m \times m$ upper triangular matrix. It is easy to check that for the aforementioned factorization and given $\bar{x}^w$

$$\Delta x \;\; = \;\; -Q_1 R^{-T} \bar{A} \bar{x}^w, \tag{4.17}$$

satisfies equation (4.14). If $\mathcal{N}_0 = \emptyset$, problem (4.15) becomes a feasibility problem and it only requires to find a solution which solves (4.15b) and (4.15c). Such a solution can be obtained from (4.17) given suitable value of $\bar{x}_j^w$ for every $j \in K$, as we will show later. It is known as the minimum norm solution. However, this is not the only solution Δx which satisfies (4.14). Applying l_2 norm to (4.17), we have

$$||\Delta x|| \leq ||Q_1||\ ||R^{-T}||||\bar{A}\bar{x}^w||.$$

Since $||Q_1|| = 1$, $||R^{-T}|| = \sigma_{min}(R)^{-1}$ and $\sigma_{min}(R) = \sigma_{min}(A)$, it follows that

$$\begin{aligned} ||\Delta x|| \;\; &\leq \;\; \frac{||\bar{A}\bar{x}^w||}{\sigma_{min}(A)}. \\ &\leq \;\; \frac{\sqrt{k}\ \sigma_{max}(\bar{A})\ \bar{x}^w_{max}}{\sigma_{min}(A)}. \end{aligned} \tag{4.18}$$

From (4.18), and using our definition of σ_{max}, we can derive loose upper and lower bounds for every Δx_j, with $j \in N$. These bounds are

$$-\frac{\sqrt{k}\ \sigma_{max}\ \bar{x}^w_{max}}{\sigma_{min}(A)} \leq \Delta x_j \leq \frac{\sqrt{k}\ \sigma_{max}\ \bar{x}^w_{max}}{\sigma_{min}(A)}, \tag{4.19}$$

and problem (4.15) has a bounded solution.

Let us define the feasible set of Δx associated to $\mathcal{P}_{aux}$ as $\mathcal{S}_\mathcal{P} = \{\Delta x : (4.15b) - (4.15c) \text{ are satisfied}\}$. In addition, let us define a closely related set $\mathcal{S}_\mathcal{Q} = \{\Delta x : (4.15b) \text{ are satisfied}\}$. Observe that Δx from (4.17) satisfies (4.15b) hence $\Delta x \in \mathcal{S}_\mathcal{Q}$. However, it does not have to satisfy (4.15c) and therefore in general $\Delta x \notin \mathcal{S}_\mathcal{P}$. The next lemma states that given a particular choice of $\bar{x}^w$ and the correct choice of μ_0, we can guarantee that $\Delta x \in \mathcal{S}_\mathcal{P}$.

Lemma 4.5 *Given $(x^0, y^0, s^0) \in \mathcal{N}_s(\gamma)$ for $\gamma \in (0,1)$, $A_{ij} \geq 0, \forall i \in M, \forall j \in N$, $\bar{A}_{ij} \geq 0, \forall i \in M, \forall j \in K$,*

$$\bar{x}_j^w \leq \gamma(1-\delta_l)\frac{\sigma_{min}(A)}{\sqrt{k}\ \mathcal{U}_v\ \sigma_{max}} \min\left\{\mu_0, \frac{1}{\mu_0}\right\}, \quad \forall j \in K, \tag{4.20}$$

and $0 < \delta_l < 1$, then constraint (4.15c) is satisfied.

Proof. To ensure this, we have to show that (4.15c) is satisfied for every Δx_j from (4.19). Hence, we

have to prove that

$$\Big(1-\delta_l\Big)x_j^0 \geq \frac{\sqrt{k}\ \sigma_{max}\ \bar{x}_{max}^w}{\sigma_{min}(A)}.$$

Since we need to ensure this for every $j \in N$, it suffices if we check that this inequality is satisfied for the smallest possible x_j^0. We know from (4.2a) that $x_j^0 \geq (\gamma\mu_0)/\mathcal{U}_v, \forall j \in N$. Therefore, we need to verify that

$$\Big(1-\delta_l\Big)\frac{\gamma\mu_0}{\mathcal{U}_v} \geq \frac{\sqrt{k}\ \sigma_{max}\ \bar{x}_{max}^w}{\sigma_{min}(A)}.$$

This condition is satisfied if we choose $\bar{x}_j^w$ for every $j \in K$ satisfying (4.20) and by noting that $\mu_0 \geq \min\{\mu_0, 1/\mu_0\}$.

□

Note that with our choice of $\bar{x}_j^w$ satisfying (4.20) for every $j \in K$, we can find a solution to problem (4.15). As in the dual case, we now guarantee that a full step can be taken in direction Δx.

Lemma 4.6 *Let $(x^0, y^0, s^0) \in \mathcal{N}_s(\gamma)$ with $\gamma \in (0,1)$ and $0 < \delta_l < 1$. By choosing Δx from (4.15) and setting*

$$\bar{x}_j^w = \gamma(1-\delta_l)\frac{\sigma_{min}(A)}{\sqrt{k}\ \mathcal{U}_v\ \sigma_{max}} \min\left\{\mu_0, \frac{1}{\mu_0}\right\}, \quad \forall j \in K, \tag{4.21}$$

we have $x_j^0 + \Delta x_j \geq 0$ and a full step in the primal space is feasible. Therefore, primal feasibility is restored in one step.

Proof. Similarly to Lemma 4.4, from constraints (4.15c) and since $x_j^0 > 0$ and $0 < \delta_l < 1$, we deduce $x_j^0 + \Delta x_j > 0$ as required. Since this solution satisfies condition (4.15b), we conclude that the warmstart is primal feasible.

□

Additionally, observe that by definition using this choice of $\bar{x}_j^w$ for every $j \in K$ and since $0 < \delta_l < 1$, we can guarantee

$$-(1-\delta_l) \leq \frac{\Delta x_j}{x_j^0} \leq (1-\delta_l). \tag{4.22}$$

Note that our choice of $\bar{x}_j^w$ is independent of $\bar{s}_j^w$ for every $j \in K$. Moreover, and as described in Section 4.1, $\bar{s}_j^w$ for every $j \in K$ is not fixed and must be calculated. However, and as we will see in the next section, both values are related via other parameters (*i.e.*, μ_0 and γ) in order to guarantee that complementarity products are still inside a slightly modified neighbourhood and that the new duality gap is also bounded.

Summarizing our findings so far, we have shown that by the use of $\mathcal{P}_{aux}$ and $\mathcal{D}_{aux}$ and choosing carefully $\bar{s}_j$ and $\bar{x}_j^w$, for every $j \in K$, we could take a full step in direction $(\Delta x, \Delta y, \Delta s)$ recovering primal and dual feasibilities. Now, we will analyse the consequences that such warmstarting approach has in the complementarity products.

4.3 Centrality

The main motivation why we have chosen to minimize a variation of the weighted-least squares approach in our linear optimization problems is to avoid having large variations $(\Delta x, \Delta s)$ on small components. We would like to keep the terms $|\Delta x_j|/|x_j^0|$ and $|\Delta s_j|/|s_j^0|$ for every $j \in N$ bounded by some constant so that we could have some control on the centrality of the warmstarting point and therefore, a control on the new duality gap.

To analyse what the effect of our warmstarting strategy is considering centrality of the new iterate, let us first determine the depth of the cut for which our analysis holds. As stated before, the depth of cut j is defined by $|\bar{c}_j - \bar{A}_j^T y^0|$ for every $j \in K$. It is not surprising that if the depth of the cuts is large, we will need to backtrack to an earlier iterate. This is considered in our analysis by the notion of μ_0 which is a reference barrier term that measures the depth of the cuts. When cuts are deep, μ_0 is large so we may need to backtrack far from optimality, but when the cuts get shallower, μ_0 gets smaller and therefore, we could choose an iterate close to optimality. Now, let us state the relation between the depth of the cuts and μ_0 using the following expression

$$|\bar{c}_j - \bar{A}_j{}^T y^0| \leq \gamma \mu_0 \frac{\bar{A}_{min}}{\sqrt{m}\, \mathcal{U}_v\, \sigma_{max}}, \quad \forall j \in K. \tag{4.23}$$

Note that we have some control with regards to the size of the cut. This means that we could retreat further back in the list of saved iterates to find a suitable large enough μ_0 and the corresponding (x^0, y^0, s^0) solution for which this condition is satisfied.

Now, using definition (4.23) and the definition of $\bar{s}_j$ in (4.6), from (4.8) we deduce for every $j \in K$ that

$$\begin{aligned} d_j &= -(\bar{c}_j - \bar{A}_j^T y^0) + \bar{s}_j, \\ &= |\bar{c}_j - \bar{A}_j^T y^0| + \bar{s}_j, \\ &\leq 2\gamma \mu_0 \frac{\bar{A}_{min}}{\sqrt{m}\, \mathcal{U}_v\, \sigma_{max}}. \end{aligned}$$

Hence,

$$d_{max} = 2\gamma \mu_0 \frac{\bar{A}_{min}}{\sqrt{m}\, \mathcal{U}_v\, \sigma_{max}}. \tag{4.24}$$

Substituting d_{max} in (4.12) and (4.13) we obtain

$$\Delta s_j \leq 2\gamma \mu_0 \frac{1}{\mathcal{U}_v}, \quad \forall j \in N, \tag{4.25}$$

and

$$\bar{s}_j + \Delta \bar{s}_j \leq 2\gamma \mu_0 \frac{1}{\mathcal{U}_v}, \quad \forall j \in K, \tag{4.26}$$

respectively.

Now, let us analyse the consequences of our choices of $\bar{s}^w$ and $\bar{x}^w$ and their impact in the complementarity products of the warmstarted iterate. To do so, first let us define δ_l as

$$\delta_l := \min \left\{ \frac{\mathcal{C}}{\mu_0 + \mathcal{C}}, \frac{\mathcal{C}}{\frac{1}{\mu_0} + \mathcal{C}} \right\}. \tag{4.27}$$

where,

$$\mathcal{C} := \frac{\gamma \sigma_{min}(A) \bar{A}_{min}}{\sqrt{k}\, \sqrt{m}\, \mathcal{U}_v^2\, \sigma_{max}^2}. \tag{4.28}$$

From (4.27) it is clear that $0 < \delta_l < 1$. Additionally, let us define δ_u as

$$\delta_u := 6. \tag{4.29}$$

These values represent the coefficients used to expand the neighbourhood from below and above, respectively. As mentioned earlier, we expect the new warmstarting point to be inside a modified neighbourhood. Note that the parameter δ_l contains valuable information regarding the old problem and the new columns appended to it, through matrices A and $\bar{A}$. Also, note that δ_u is problem independent. As we will discuss later, this is one of the key results of the analysis since it guarantees that regardless the size of the cut, and if some conditions hold, the new duality gap will be bounded by the old duality gap and a constant.

The following theorem states conditions and analyses complementarity products when our specialized warmstarting strategy is used.

Theorem 4.7 *Let assumptions (A.1)-(A.4) hold and let* $(x^0, y^0, s^0) \in \mathcal{N}_s(\gamma)$ *with* $\gamma \in (0,1)$. *Additionally, let* δ_l *and* δ_u *be defined by* (4.27) *and* (4.29), *respectively. Also, let us set*

$$\bar{x}_j^w = \gamma(1-\delta_l)\frac{\sigma_{min}(A)}{\sqrt{k}\,\mathcal{U}_v\,\sigma_{max}}\min\left\{\mu_0, \frac{1}{\mu_0}\right\}, \quad \forall j \in K,$$

and

$$\bar{s}_j = \gamma\mu_0\frac{\bar{A}_{min}}{\sqrt{m}\,\mathcal{U}_v\,\sigma_{max}}, \quad \forall j \in K.$$

Having $\bar{c}_j - \bar{A}_j{}^T y^0 < 0$ *and* $|\bar{c}_j - \bar{A}_j{}^T y^0|$ *satisfying* (4.23), *for every* $j \in K$, *and choosing* Δx *from* (4.15) *and* $(\Delta y, \Delta s)$ *from* (4.7) *where* $\Delta y = -\Delta\bar{y}$, *we have that*

$$(x^w, \bar{x}^w, y^w, s^w, \bar{s}^w) \in \overline{\mathcal{N}}_s(\gamma, \delta_l, \delta_u),$$

where

$$\overline{\mathcal{N}}_s(\gamma,\delta_l,\delta_u) = \{(x^w,\bar{x}^w,y^w,s^w,\bar{s}^w) \in \overline{\mathcal{F}^0} : \delta_l\gamma\mu_0 \le x_j^w s_j^w \le \delta_u\frac{1}{\gamma}\mu_0, \forall j \in N; \delta_l\gamma\mu_0 \le \bar{x}_k^w\bar{s}_k^w \le \delta_u\frac{1}{\gamma}\mu_0, \forall k \in K\},$$

and

$$\overline{\mathcal{F}^0} = \{(x^w, \bar{x}^w, y^w, s^w, \bar{s}^w) : Ax^w + \bar{A}\bar{x}^w = b; A^T y^w + s^w = c; \bar{A}^T y^w + \bar{s}^w = \bar{c}; (x^w, s^w) > 0; (\bar{x}^w, \bar{s}^w) > 0\}.$$

Proof. It is not difficult to show that by choosing δ_l from (4.27) and δ_u from (4.29), and using our previous choices for $\bar{x}_j^w$ (see (4.21)), and $\bar{s}_j^w = \bar{s}_j + \Delta\bar{s}_j$ (see (4.6) and (4.26)), the complementarity products for the new components, namely $\bar{x}_j^w\bar{s}_j^w$ for every $j \in K$, are inside the modified neighbourhood, $\overline{\mathcal{N}}_s(\gamma, \delta_l, \delta_u)$. Let us prove first that the upper bound holds. For every $j \in K$, we have that

$$\begin{aligned}\bar{x}_j^w\bar{s}_j^w &= \bar{x}_j^w(\bar{s}_j + \Delta\bar{s}_j),\\ &\le \gamma(1-\delta_l)\frac{\sigma_{min}(A)}{\sqrt{k}\,\mathcal{U}_v\,\sigma_{max}}\min\left\{\mu_0, \frac{1}{\mu_0}\right\}\left(2\gamma\mu_0\frac{1}{\mathcal{U}_v}\right).\end{aligned}$$

Since $0 < \gamma < 1$, $0 < \delta_l < 1$, $\mathcal{U}_v^2 > 1$, $k \ge 1$ and $\min\{\mu_0, 1/\mu_0\} \le 1$, it is then clear that

$$\bar{x}_j^w\bar{s}_j^w < \delta_u\mu_0, \quad \forall j \in K, \tag{4.30}$$

holds. If we now consider the lower bound for the complementarity products of the new components,

we have that for every $j \in K$

$$\begin{aligned}\bar{x}_j^w \bar{s}_j^w &= \bar{x}_j^w(\bar{s}_j + \Delta\bar{s}_j),\\ &\geq \gamma(1-\delta_l)\frac{\sigma_{min}(A)}{\sqrt{k}\,\mathcal{U}_v\,\sigma_{max}}\min\left\{\mu_0, \frac{1}{\mu_0}\right\}\left(\gamma\mu_0\frac{\bar{A}_{min}}{\sqrt{m}\,\mathcal{U}_v\sigma_{max}}\right).\end{aligned}$$

Using (4.28), we get

$$\bar{x}_j^w \bar{s}_j^w \geq \gamma\mathcal{C}(1-\delta_l)\mu_0\min\left\{\mu_0, \frac{1}{\mu_0}\right\}, \quad \forall j \in K.$$

Then, by choosing δ_l from (4.27), we have

$$\bar{x}_j^w \bar{s}_j^w \geq \delta_l\gamma\mu_0, \quad \forall j \in K, \tag{4.31}$$

Hence, and considering $\gamma \in (0,1)$, (4.30) and (4.31) the following result holds

$$\delta_l\gamma\mu_0 \leq \bar{x}_j^w \bar{s}_j^w \leq \delta_u\frac{1}{\gamma}\mu_0, \quad \forall j \in K, \tag{4.32}$$

which completes the first of the prove.

Now, we need to prove that the complementarity products of the old components, namely $x_j^w s_j^w$ for every $j \in N$, are inside the modified neighbourhood, $\overline{\mathcal{N}}_s(\gamma, \delta_l, \delta_u)$. Firstly, let us prove that the upper bound holds for every $j \in N$. By definition and conditions (4.2) and (4.22), we have that for every $j \in N$

$$\begin{aligned}x_j^w s_j^w &= x_j^0 s_j^0\left(1+\frac{\Delta x_j}{x_j^0}\right)\left(1+\frac{\Delta s_j}{s_j^0}\right),\\ &\leq x_j^0 s_j^0\Big(2\Big)\left(1+\Delta s_j\frac{\mathcal{U}_v}{\gamma\mu_0}\right).\end{aligned}$$

Hence, and by using (4.25), we get

$$x_j^w s_j^w \leq \delta_u\frac{1}{\gamma}\mu_0, \quad \forall j \in N.$$

To prove that the lower bound holds we recall constraints (4.15c) which guarantee

$$\Delta x_j \geq x_j^0(\delta_l - 1) \quad \forall j \in N. \tag{4.33}$$

Also, from (4.5), $\Delta s_j \geq 0$. Hence for every $j \in N$

$$\begin{aligned}x_j^w s_j^w &= x_j^0 s_j^0\left(1+\frac{\Delta x_j}{x_j^0}\right)\left(1+\frac{\Delta s_j}{s_j^0}\right),\\ &\geq x_j^0 s_j^0\left(1+\frac{\Delta x_j}{x_j^0}\right).\end{aligned} \tag{4.34}$$

From (4.33) and since $x_j^0 > 0$, we know that

$$1+\frac{\Delta x_j}{x_j^0} \geq \delta_l, \quad \forall j \in N.$$

Replacing this in (4.34) gives

$$x_j^w s_j^w \geq \delta_l x_j^0 s_j^0, \quad \forall j \in N,$$

which completes the proof.

□

4.4 Algorithm

Now we are in position to describe the algorithm proposed to find a warmstarting point after cuts of type $\bar{c}_j - \bar{A}_j^T y < 0$ are appended to the RMP. Note that this algorithm is embedded inside a major algorithm which is the primal-dual column generation method (PDCGM for short) [11]. Let us define $T = \{1, 2, \ldots, h\}$ as the set of indexes of iterates which are strictly feasible and well-centred in the initial problem. Observe that T, h and their corresponding list of stored solutions vary at each outer iteration of PDCGM. The list is created in ascending order so the last stored iterate and the closest-to-optimality solution is denoted by (x^h, y^h, s^h). Algorithm 1 summarizes the principal steps of our specialized warmstarting strategy.

Algorithm 1 Specialized warmstarting strategy for the primal-dual column generation method

Step 1 : Input A, b, c, $\bar{c}$, $\bar{A}$, $\mathcal{U}_v$, n, k, m, $\gamma \in (0,1)$, list of $(x^t, y^t, s^t) \in \mathcal{N}_s(\gamma)$ and $\mu_t = (x^t)^T s^t / n$, where $t \in T$.

Step 2 : If no column is returned by the oracle, set $(x^w, y^w, s^w) = (x^h, y^h, s^h)$, and then go to Step 9. Otherwise, go to Step 3.

Step 3 : Calculate the smallest μ from the list of stored iterates such that

$$\max_{j \in K} |\bar{c}_j - \bar{A}_j^T y^0| \leq \gamma\mu \frac{\bar{A}_{min}}{\sqrt{m}\, \mathcal{U}_v\, \sigma_{max}},$$

is satisfied. If there exists such μ, define $\mu = \mu_0$, denote its associated solution as (x^0, y^0, s^0), and go to Step 4. If not, use coldstart and go to Step 9.

Step 4 : Define

$$\delta_l := \min\{\frac{\mathcal{C}}{\mu_0 + \mathcal{C}}, \frac{\mathcal{C}}{\frac{1}{\mu_0} + \mathcal{C}}\}, \quad \text{where} \quad \mathcal{C} := \frac{\gamma \bar{A}_{min} \sigma_{min}(A)}{\sqrt{k}\, \sqrt{m}\, \mathcal{U}_v^2\, \sigma_{max}^2}.$$

Step 5 : Set for every $j \in K$

$$\bar{s}_j = \gamma\mu_0 \frac{\bar{A}_{min}}{\sqrt{m}\, \mathcal{U}_v\, \sigma_{max}},$$

$$\bar{x}_j^w = \gamma(1 - \delta_l) \frac{\sigma_{min}(A)}{\sqrt{k}\, \mathcal{U}_v\, \sigma_{max}} \min\left\{\mu_0, \frac{1}{\mu_0}\right\},$$

and define $\mathcal{B}_0$ and $\mathcal{N}_0$.

Step 6 : Solve $\mathcal{D}_{aux}$. Output: $(\Delta y, \Delta s)$.

Step 7 : Solve $\mathcal{P}_{aux}$. Output: Δx.

Step 8 : Define $(x^w, \bar{x}^w, y^w, s^w, \bar{s}^w) = (x^0 + \Delta x, \bar{x}^w, y^0 + \Delta y, s^0 + \Delta s, \bar{s}^w)$.

Step 9 : Continue with the usual primal-dual interior point method by solving the Newton system of equations in the old/new dimensions.

Note that Step 2 accounts for the case in which no columns with negative reduced costs are obtained from the oracle. This is a particular feature of the PDCGM. There is no point of using a warmstarting strategy since the current iterate is already well-centred and strictly feasible and therefore starting from it is the best we could do. Also, it is important to note that every cut $\bar{c}_j - \bar{A}_j^T y^h < 0$ is generated using (x^h, y^h, s^h). Hence, for any iterate (x^t, y^t, s^t), where $t \in T$, the cuts are still valid. The following theorem summarizes the main discussion and results of this section.

Theorem 4.8 *Using Algorithm 1 and given that a suitable (x^0, y^0, s^0) is available from the list of iterates, the solution $(x^w, \bar{x}^w, y^w, s^w, \bar{s}^w) \in \overline{\mathcal{N}_s}(\gamma, \delta_l, \delta_u)$ and the new duality gap is bounded by*

$$(x^w)^T s^w + (\bar{x}^w)^T \bar{s}^w \leq 6\big(n+k\big)\frac{1}{\gamma}\mu_0. \tag{4.35}$$

Proof. The proof follows from Lemmas 4.4 and 4.6 and Theorem 4.7.

□

In summary, our strategy aims to restore primal and dual feasibilities separately by means of auxiliary linear optimization problems. We can ensure that primal and dual feasibilities are fully recovered if suitable a μ can be found. Note that Theorem 4.8 provides a guarantee that the new duality gap is bounded by the old duality gap multiplied with a constant which depends on the number of columns added to the old problem. Our analysis takes into account special classes of problems arising in combinatorial optimization with non-negative technological coefficients.

5 Numerical experiments

We have implemented the proposed warmstarting strategy inside the primal-dual column generation method described in [11]. This column generation implementation has a built-in procedure which allows to store an advanced iterate in order to use it for future warmstart. We have taken advantage of this to implement our strategy. We have tested three different strategies to solve the relaxation of the one-dimensional cutting stock problem (CSP) after applying a Dantzig-Wolfe reformulation. This problem consists in the minimization of the number of rolls of fixed width that have to be cut in order to satisfy the demands of different pieces with different widths [7]. For a complete description of the problem and its corresponding decomposition, refer to [1, 3].

The strategies considered are:

- Coldstart (CS). It refers to solving a given RMP without considering any information of the previously solved RMP and therefore, every RMP is solved from scratch.
- Partial feasibility restoration (PR). This strategy was introduced in [10] and has been successfully applied to speed up the column generation in different applications [11, 14]. As described in Section 4, this strategy ensures that dual feasibility is fully restored in the new components, however, it does not ensure the same in the old components. Additionally, it sets the new components in the primal space to a small value so the infeasibility in this space is small.
- One step primal-dual feasibility restoration (1SPDR). This strategy is described in this paper and aims to recover primal and dual feasibilities in one step by solving two auxiliary problems.

Note that the subproblems obtained after applying the reformulation to the CSP are solved using the same source-code as described in [11]. For the three aforementioned strategies, the RMPs are initialized with the same columns however, after the first iteration every RMP may be different and therefore, we may expect that some of the strategies will require less outer iterations than others. Since this could confuse the comparison, we have omitted these results. To run the tests we have used a laptop with processor Intel Core i5 with 2.30 Ghz, 4 GB RAM and a Linux operating system. For each of the strategies, we stop the column generation procedure when the relative duality gap becomes smaller than the default accuracy $\delta = 10^{-6}$.

To analyse the performance of these three strategies, we have selected 252 instances from the literature in the one-dimensional CSP (`http://www.math.tu-dresden.de/~capad/`). The column generation is

initialised with columns generated by homogeneous cutting patterns, which corresponds to selecting only one piece per pattern, as many times as possible without violating the width of the rolls.

Before continuing, it is important to give some remarks about our implementation. Note that for advanced column generation iterations, the coefficients of the objective function, in the primal and dual auxiliary problems ($1/s_j^0$ in (4.5) and $1/x_j^0$ in (4.15)), can lead to very badly-scaled problems (some very large coefficients and others very small). Therefore, we have decided to scale these coefficients and restrict them inside a narrower range. In other words, if a given coefficient is greater/smaller than a predefined threshold, we have used this threshold as the coefficient for this specific variable. Note that by doing this, we do not affect the primal and dual feasible sets, only the scaling of that particular RMP and therefore, the strategy recovers primal and dual feasibility in one step. Additionally, since PDCGM is based on HOPDM [9], an infeasible primal-dual interior point method is used to solve every RMP which keeps the iterates inside a neighbourhood of the central path by making use of multiple centrality corrector steps [4]. We have considered this in our developments in the following way. When the column generation is close to optimality and the depth of the cuts is small, that is when μ and $|\bar{c}_j - \bar{A}_j^T y_j^0|$ are small, instead of solving the primal and dual auxiliary problems, we set the values of new components to $\bar{x}_j^w = \sqrt{\mu/\gamma}$ and $\bar{s}_j^w = \sqrt{\gamma\mu}$ for every $j \in K$. We keep the old iterates unchanged. This choice is justified since the cuts at this stage of the column generation are likely to be shallow and therefore, by choosing the new components in this way, we only generate small infeasibilities which HOPDM (and any infeasible primal-dual interior point method) can easily handle. Additionally, with these choices, we ensure that the complementarity products for the new iterates are inside the neighbourhood described in (4.1). Moreover, and after some testing, the directions, and therefore the correction steps, obtained by solving the auxiliary problems in an advanced stage of the column generation, are very short and do not change significantly the stored iterate. By performing this small change in the algorithm, we aim to reduce the CPU time of the column generation by avoiding unnecessary calls to the auxiliary problems and taking advantage of the infeasible primal-dual interior point method. Finally, to solve the primal and dual auxiliary problems we rely on the solver HOPDM.

In Table 1, we summarize our findings. In the first column we denote by k the number of columns added at each iteration of the column generation procedure. Note that by adding more columns at every iteration we are targeting to reduce the number of calls to the oracle, however in terms of warmstart this may have a big impact since the similarities between two consecutive RMPs are somehow lost. We have also grouped the instances in two classes, denoted by S and L. The former contains 106 small size instances while the latter 146 larger instances. In row *All*, we have included the average results when all the instances are considered. For each strategy we show the average number of inner iterations and the average CPU time required to solve the RMPs (RMP) and to converge to the optimal solution (total). Inner iterations refer to the total number of iterations required to solve the RMPs while the RMP time considers the total time which includes the time to solve the RMPs and the time to find the warmstated iterate. For instance, for 1SPDR strategy, it is the overall time of solving the primal and dual auxiliary problems and the time of solving each RMP.

It is clear from our results that by adding more columns to the RMP at each iteration, the number of outer iteration is reduced for all the strategies. However, the CPU time is not affected in the same way. This is explained because while adding more columns at each iteration reduces the number of calls to the oracle, it makes the task of warmstarting more difficult since the new problem is likely to be very different from the old one. Note that the best performance in terms of CPU time for each of the strategies and each group of instances is obtained when 10 columns are added at each iteration. From our results it seems that by adding 10 columns, the number of outer iterations is reduced considerably when compared to a single-column approach and at the same time, each strategy performs efficiently. Note that considering all instances, the best performance is achieved by 1SPDR when adding 10 columns.

Table 1: Average results of the strategies when solving the root node of the cutting stock problem with the primal-dual column generation method after applying Dantzig-Wolfe decomposition.

k	class	PR			CS			1SPDR		
		inner iterations	time (sec)		inner iterations	time (sec)		inner iterations	time (sec)	
			RMP	total		RMP	total		RMP	total
100	S	278.0	1.9	5.3	1180.2	13.7	20.6	440.7	3.3	6.7
	L	687.0	9.1	31.1	4172.6	98.7	296.2	1201.6	23.9	49.6
	All	515.0	6.1	20.2	2913.9	63.0	180.3	881.5	15.2	31.6
50	S	307.5	2.3	4.6	759.7	5.9	8.0	494.3	2.8	4.4
	L	728.4	8.9	20.7	3915.6	66.5	170.8	1190.7	13.0	22.8
	All	551.4	6.1	13.9	2588.1	41.0	102.3	897.8	8.7	15.1
10	S	421.0	1.7	2.6	1394.2	7.9	9.2	689.9	2.4	3.2
	L	1031.7	6.1	10.5	2583.8	15.8	28.0	1166.0	6.2	9.9
	All	774.8	4.2	7.2	2083.4	12.5	20.1	965.7	4.6	7.1
1	S	1253.9	4.4	7.0	3397.0	11.4	14.5	2102.5	5.5	7.6
	L	3148.6	13.6	24.4	6145.7	30.2	50.2	3143.5	11.0	20.2
	All	2351.6	9.7	17.1	4989.5	22.3	35.2	2705.7	8.7	14.9

It is also fair to say that the performance of PR strategy is comparable to the one proposed in this study when this number of columns is added per iteration. Both warmstarting strategies outperform a coldstart approach in both number of inner iterations and CPU times.

Now, let us compare CS and 1SPDR strategies. Since 1SPDR successfully restores primal and dual feasibility in almost all the column generation iterations and keeps the complementarity products of the warmstarted iterate inside a slightly modified neighbourhood, the number of inner iterations to solve the new problem is reduced when compared to CS. The reductions vary between 35% (class S when $k = 50$) and 71% (class L when $k = 100$). The savings in time are due to this reduction and also the efficiency of calculating the new warmstarted iterate. The gains in total CPU time by using 1SPDR strategy for the same class/number of columns range from 44% to 87%.

The differences between 1SPDR and PR are due to the nature of each of these strategies and the environment for which they were developed. It is necessary to remark that 1SPDR aims to recover primal and dual feasibility while keeping the warmstarted iterate inside a slightly modified neighbourhood, but PR aims to recover only feasibility in the new dual components and does not ensure the same for the old components. Additionally, the latter was designed to take full advantage of an infeasible primal-dual interior point method while the former has being developed considering a feasible primal-dual interior point method and only takes advantage of the infeasibility nature of the solver at the end of the column generation. In general, PR outperforms 1SPDR in number of inner iterations. This may be due to the nature of PR, which delivers close to feasible solutions but at the same time, the iterates are far enough from the boundaries. Then, only few centering steps are needed to recover feasibility and return to the neighbourhood of the central path. 1SPDR strategy ensures primal and dual feasibility restorations, however and as a consequence of such restorations, it slightly increases the neighbourhood and therefore, at some iterations the new iterate may be too close to the boundaries. This originates a sometimes excessive number of centering steps in order to return to the neighbourhood of the central path and therefore has a big impact on the number of iterations. It is important to point out that this belief is just empirical since no theoretical support is given for PR strategy with respect to complementarity after the restoration is performed. In terms of CPU time, it is not clear which method performs better however, the best performance in terms of total CPU time is obtained when using 1SPDR when 10 columns are added.

6 Conclusions

In this paper we have presented a new strategy to warmstart a primal-dual interior point method in the context of column generation. The method deals with primal and dual feasibility restorations while keeping some control in the duality gap of the new warmstart. Proofs that the method recovers primal and dual feasibility in one step after new columns are appended to the restricted master problem are provided. Also, conditions are given to guarantee that the new duality gap is bounded by a constant and the old duality gap. Additionally, computational experiments for solving a reformulation of the cutting stock problem demonstrate the benefits of using such warmstarting strategy when compared to coldstart. In general, savings between $35 \sim 71\%$ in the number of inner iterations and $44 \sim 87\%$ in CPU time can be achieved on average for different column generation scenarios. The advantages of the proposed method with respect to coldstart are consistent no matter the number of columns added to each restricted master problem at each iteration. Additionally, we have compared this strategy with the one presented in [10] which restores dual feasibility in the new components after new columns are added. Both warmstarting strategies are competitive in terms of CPU time. Further research directions are to analyse and develop a warmstarting strategy that considers primal and dual infeasibilities so we can take advantage of the infeasible primal-dual interior point method solver HOPDM. Also, we plan to investigate more efficient ways of calculating directions in the primal and dual spaces in order to reduce the RMP time and therefore, the total CPU time of the strategy.

References

[1] H. Ben Amor and J. Valério de Carvalho. Cutting stock problems. In G. Desaulniers, J. Desrosiers, and M. M. Solomon, editors, *Column Generation*, pages 131–161. Springer US, 2005.

[2] H. Y. Benson and D. F. Shanno. An exact primal-dual penalty method approach to warmstarting interior-point methods for linear programming. *Computational Optimization and Applications*, 38(3):371–399, 2007.

[3] O. Briant, C. Lemaréchal, P. Meurdesoif, S. Michel, N. Perrot, and F. Vanderbeck. Comparison of bundle and classical column generation. *Mathematical Programming*, 113:299–344, 2008.

[4] M. Colombo and J. Gondzio. Further development of multiple centrality correctors for interior point methods. *Computational Optimization and Applications*, 41(3):277–305, 2008.

[5] A. Engau, M. F. Anjos, and A. Vannelli. On interior-point warmstarts for linear and combinatorial optimization. *SIAM Journal on Optimization*, 20(4):1828–1861, 2010.

[6] R. M. Freund. A potential-function reduction algorithm for solving a linear program directly from an infeasible "warm start". *Mathematical Programming*, 52:441–466, 1991.

[7] P. C. Gilmore and R. E. Gomory. A linear programming approach to the cutting-stock problem. *Operations Research*, 9(6):849–859, 1961.

[8] J.-L. Goffin and J.-P. Vial. Convex nondifferentiable optimization: a survey focused on the analytic center cutting plane method. *Optimization Methods and Software*, 17:805–868, 2002.

[9] J. Gondzio. HOPDM (version 2.12) - a fast LP solver based on a primal-dual interior point method. *European Journal of Operational Research*, 85:221–225, 1995.

[10] J. Gondzio. Warm start of the primal-dual method applied in the cutting-plane scheme. *Mathematical Programming*, 83:125–143, 1998.

[11] J. Gondzio, P. González-Brevis, and P. Munari. New developments in the primal-dual column generation technique. Technical Report, ERGO 11-001, University of Edinburgh, School of Mathematics, 2011.

[12] J. Gondzio and A. Grothey. Reoptimization with the primal-dual interior point method. *SIAM Journal on Optimization*, 13(3):842–864, 2003.

[13] J. Gondzio and A. Grothey. A new unblocking technique to warmstart interior point methods based on sensitivity analysis. *SIAM Journal on Optimization*, 19(3):1184–1210, 2008.

[14] J. Gondzio and R. Sarkissian. Column generation with a primal-dual method. Technical Report 96.6, Logilab, 1996.

[15] E. John and E. A. Yildirim. Implementation of warm-start strategies in interior-point methods for linear programming in fixed dimension. *Computational Optimization and Applications*, 41:151–183, 2008.

[16] N. Karmarkar. A new polynomial-time algorithm for linear programming. *Combinatorica*, 4:373–395, 1984.

[17] M. E. Lübbecke and J. Desrosiers. Selected topics in column generation. *Operations Research*, 53(6):1007–1023, 2005.

[18] R. K. Martinson and J. Tind. An interior point method in Dantzig-Wolfe decomposition. *Computers and Operations Research*, 26:1195–1216, 1999.

[19] J. Mitchell and B. Borchers. Solving real-world linear ordering problems using a primal-dual interior point cutting plane method. *Annals of Operations Research*, 62:253–276, 1996.

[20] J. E. Mitchell. *Karmarkar's Algorithm and Combinatorial Optimization Problems*. PhD thesis, School of Operations Research and Industrial Engineering, Cornell University, 1988.

[21] J. E. Mitchell and M. Todd. Solving combinatorial optimization problems using Karmarkar's algorithm. *Mathematical Programming*, 56:245–284, 1992.

[22] A. Skajaa, E. Andersen, and Y. Yinyu. Warmstarting the homogeneous and self-dual interior point method for linear and conic quadratic problems. Optimization Online, 2012.

[23] S. J. Wright. *Primal-Dual Interior-Point Methods*. SIAM, 1997.

[24] E. A. Yildirim and S. J. Wright. Warm-start strategies in interior-point methods for linear programming. *SIAM Journal on Optimization*, 12(3):782–810, 2002.