Some Disadvantages of a Mehrotra-Type Primal-Dual Corrector Interior Point Algorithm for Linear Programming

Coralia Cartis*

March, 2008

Abstract

Employing a new primal-dual corrector algorithm, we investigate the impact that corrector directions may have on the convergence behaviour of predictor-corrector methods. The Primal-Dual Corrector (PDC) algorithm that we propose computes on each iteration a *corrector* direction in addition to the direction of the standard primal-dual path-following interior point method [9, 22] for Linear Programming (LP), in an attempt to improve performance. The new iterate is chosen by moving along the sum of these directions, from the current iterate. This technique is similar to the construction of Mehrotra's highly popular predictor-corrector algorithm [14]. We present examples, however, that show that the PDC algorithm may fail to converge to a solution of the LP problem, in both exact and finite arithmetic, regardless of the choice of stepsize that is employed. The cause of this bad behaviour is that the correctors exert too much influence on the direction in which the iterates move.

*Oxford University Computing Laboratory, Wolfson Building, Parks Road, Oxford, OX1 3QD, United Kingdom, (ccartis@comlab.ox.ac.uk). The author was supported through grant GR/S34472 from the Engineering and Physical Sciences Research Council of the UK.

1 Introduction

In the past fifteen years, Interior Point Methods (IPMs) have become highly successful in solving Linear Programming (LP) problems, especially large-scale ones, while enjoying good theoretical convergence and complexity properties (see [4, 6, 19, 21, 22] for comprehensive reviews of the field of IPMs for LP). Examples of IPMs that are reliable both in theory and in practice include the Primal-Dual (PD) path-following method of Kojima et al. [9] with some *long-step* linesearch procedure [22], and an *infeasible* formulation of this algorithm [8, 22]. The majority of commercial and public IPM codes implement a variant of the latter, Mehrotra's Predictor-Corrector (MPC) algorithm [14], and some of them employ in addition, Gondzio's higher-order corrections [5]. For descriptions of the MPC algorithm, see [11, 24] and Chapter 10 of [22]. Since its first implementations and testing on the standard set of LP test problems (the Netlib test set), the MPC algorithm proved to be, especially on large-scale problems, much faster than the infeasible PD algorithm, in terms of both the number of iterations and the computational time [11, 14]. Its past and present practical successes, however, have not been enhanced by equally praiseworthy theoretical guarantees of good performance: no global convergence or polynomial complexity results are known for this method. It is, in fact, acknowledged among practitioners that there are examples on which the MPC algorithm fails to converge (see [18], page 407). To our knowledge, no such examples have been published or analysed in the literature. Moreover, most implementations of the MPC algorithm do not include any safeguards to monitor convergence of the algorithm or to help the algorithm move away from troublesome situations since the generally excellent performance of the MPC algorithm seems to render them unnecessary (see [18], page 407). Presently, we construct a Mehrotra-type method, the Primal-Dual Corrector (PDC), whose behaviour we can understand and explain.

The PDC algorithm computes on each iteration, an additional direction, a corrector, to augment the direction of the PD algorithm. In this paper, we find, however, that employing these correctors may have an adverse effect on the performance of the algorithm. In particular, we show that the PDC algorithm may fail to converge to the solution of an LP example in both exact and finite arithmetic. If certain starting points are chosen for the algorithm, then we prove that the failure of the algorithm on the example problem occurs in exact arithmetic regardless of the stepsize procedure that is employed (see Section 3.1). We describe two numerical calculations that exhibit this failure (see Section 3.2). In the first numerical example, the barrier parameter is decreased by a fixed fraction on each iteration, and in the second one, it is chosen automatically by the procedure employed in the MPC algorithm [14, 15, 22]. Though the example that we present does not apply to the MPC algorithm, it throws doubt nevertheless on its convergence properties in general, due to the essential similarities between the MPC and PDC algorithms in the way the search directions and new iterates are constructed on each iteration, which is the cause of failure of the PDC algorithm on the example (see Section 4.2).

The structure of the paper is as follows. Section 2 describes the construction of the PDC algorithm. Section 3 presents the above-mentioned example of failure of the PDC to converge: section 3.1 gives the promised theoretical analysis, and section 3.2, the numerical evidence. The failure of the PDC algorithm to converge is due to the corrector exerting too much influence in the construction of the iterates, and determining the inefficient direction in which the iterates move. A way to reduce the impact of the correctors, which overcomes the failure encountered by the PDC, is addressed in Section 4.1. Section 4.2 concludes on the relevance of the failure example to the behaviour of the MPC algorithm.

2 The Primal-Dual Corrector (PDC) algorithm

Setting the framework Let the LP problem we are solving be given in the standard form

$$\min_{x\in\mathbb{R}^n} \quad c^\top x \quad \text{subject to} \quad Ax = b, \quad x \geq 0, \tag{P}$$

where $m < n$, $b \in \mathbb{R}^m$, $c \in \mathbb{R}^n$, and A is a real matrix of dimension $m \times n$. The dual problem corresponding to the primal problem (P) is

$$\max_{(y,s)\in\mathbb{R}^m\times\mathbb{R}^n} \quad b^\top y \quad \text{subject to} \quad A^\top y + s = c, \quad s \geq 0. \tag{D}$$

We assume that there exists a *primal-dual strictly feasible* point $w^0 = (x^0, y^0, s^0)$, that is

$$Ax^0 = b, \quad A^\top y^0 + s^0 = c, \quad x^0 > 0 \quad \text{and} \quad s^0 > 0, \tag{2.1}$$

and that the matrix A has full row rank. These assumptions are ubiquitous in IPM theory, and will be referred to as the **IPM conditions**. They imply that the solution set of (P) and (D) is nonempty [2, 22].

Subject to the IPM conditions, the perturbed system of optimality conditions [22] associated to (P) and (D)

$$F_\mu(w) := \begin{pmatrix} Ax - b \\ A^\top y + s - c \\ XSe - \mu e \end{pmatrix} = 0, \quad x > 0, \quad s > 0, \tag{2.2}$$

has a unique solution $w(\mu) = (x(\mu), y(\mu), s(\mu))$, for each $\mu > 0$ [22], where in (2.2), XS is the diagonal matrix with diagonal elements $x_i s_i$, $i = \overline{1, n}$, and $e := (1, 1, \ldots, 1) \in \mathbb{R}^n$. As μ

tends to zero, the points $w(\mu)$, $\mu > 0$, which form the *primal-dual central path*, converge to a solution of problems (P) and (D) [23].

Note that (2.2) with $\mu := 0$ and with $x \geq 0$, $s \geq 0$, is precisely the system of optimality conditions of (P) and (D), whose solutions coincide with those of (P) and (D).

Description of the algorithm Assume that a point $w^0 = (x^0, y^0, s^0)$ satisfying (2.1) is available as starting point of the algorithm.

The PDC algorithm attempts to follow the primal-dual central path approximately to a solution of problems (P) and (D), in a similar fashion to long-step primal-dual path-following IPMs.

At the current iterate $w^k = (x^k, y^k, s^k)$, $k \geq 0$, of the PDC algorithm, a parameter $\mu > 0$ is picked

$$\mu := \sigma^k \mu^k, \tag{2.3}$$

where $\mu^k := (x^k)^\top s^k / n$, and $\sigma^k \in (0, 1)$ is a *centring parameter* that can be fixed at the start of the algorithm or computed on each iteration by some automatic procedure. Then the Newton direction $dw^k = (dx^k, dy^k, ds^k)$ is computed from w^k for the system $F_\mu(w) = 0$ in (2.2), i.e., dw^k is the solution of the linear system

$$F'_\mu(w^k)\, dw^k = -F_\mu(w^k), \tag{2.4}$$

where $F'_\mu(w^k)$ is the Jacobian of F_μ at w^k. The system (2.4) is equivalent to

$$\begin{pmatrix} A & 0 & 0 \\ 0 & A^\top & I \\ S^k & 0 & X^k \end{pmatrix} \begin{pmatrix} dx^k \\ dy^k \\ ds^k \end{pmatrix} = - \begin{pmatrix} Ax^k - b \\ A^\top y^k + s^k - c \\ X^k S^k e - \sigma^k \mu^k e \end{pmatrix}. \tag{2.5}$$

Next, a *corrector* direction $dw^{k,c} = (dx^{k,c}, dy^{k,c}, ds^{k,c})$ is computed by solving the linear system

$$F'_\mu(w^k) dw^{k,c} = -F_\mu(w^k + dw^k). \tag{2.6}$$

The right-hand side of the system (2.6) represents the error that is introduced in the system $F_\mu(w) = 0$ of (2.2) by its linearization around w^k, and it has the explicit expression

$$F_\mu(w^k + dw^k) = \begin{pmatrix} A(x^k + dx^k) - b \\ A^\top(y^k + dy^k) + (s^k + ds^k) - c \\ (X^k + dX^k)(S^k + dS^k)e - \sigma^k \mu^k e \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \\ dX^k dS^k e \end{pmatrix}, \tag{2.7}$$

where the last equation depends on (2.5), and where dX^k and dS^k are the diagonal matrices with diagonal elements dx_i^k, $i = \overline{1, n}$, and ds_i^k, $i = \overline{1, n}$, respectively. It follows from (2.6) that the corrector direction attempts to correct this error, in order to position the new iterate closer to the primal-dual central path.

The resulting search direction $dw^{k,r} = (dx^{k,r}, dy^{k,r}, ds^{k,r})$ of the PDC algorithm is the sum

$$dw^{k,r} := dw^k + dw^{k,c}, \tag{2.8}$$

and the new iterate has the form

$$x^{k+1} := x^k + \theta_p^k dx^{k,r}, \quad y^{k+1} := y^k + \theta_d^k dy^{k,r}, \quad \text{and} \quad s^{k+1} := s^k + \theta_d^k ds^{k,r}, \tag{2.9}$$

where $\theta_p^k \in (0,1]$ and $\theta_d^k \in (0,1]$ are possibly different primal and dual stepsizes that provide the conditions

$$x^{k+1} > 0 \quad \text{and} \quad s^{k+1} > 0. \tag{2.10}$$

The strict inequalities (2.10), and those in (2.1), together with A having full row rank, imply that the Jacobian $F'_\mu(w^k)$ is nonsingular [22], and thus, the directions dw^k and $dw^{k,c}$ are well-defined, for every $k \geq 0$.

In the context of variants of Newton's method for solving nonlinear systems of equations, the construction of the search direction (2.8) and of the new iterate (2.9) when $\theta_p^k = \theta_d^k = \theta^k$ coincides with the *level-1 composite Newton direction and iterate* [20], respectively, for the nonlinear system $F_\mu(w) = 0$, starting at w^k, where $\mu := \sigma^k \mu^k$.

If $dw^{k,c} := 0$, for each $k \geq 0$, the PDC algorithm coincides with the PD algorithm (p. 8, [22]).

The PDC algorithm applied to problems (P) and (D) can be summarized as follows.

The PDC algorithm:

A point $w^0 = (x^0, y^0, s^0)$ is required that satisfies (2.1). Let $\epsilon > 0$ be a tolerance parameter. At the current iterate $w^k = (x^k, y^k, s^k)$, where $k \geq 0$, do:

Step 1: If $(x^k)^\top s^k \leq \epsilon$, STOP.

Step 2: Let $\mu^k := \frac{(x^k)^\top s^k}{n}$ and choose $\sigma^k \in (0,1)$.
Compute the direction $dw^k = (dx^k, dy^k, ds^k)$ from the linear system (2.4).
Compute the corrector direction $dw^{k,c} = (dx^{k,c}, dy^{k,c}, ds^{k,c})$ from the system (2.6).
Compute the search direction $dw^{k,r} = (dx^{k,r}, dy^{k,r}, ds^{k,r})$ from (2.8).

Step 3: Choose the stepsizes $\theta_p^k \in (0,1]$ and $\theta_d^k \in (0,1]$ along $dx^{k,r}$ and $(dy^{k,r}, ds^{k,r})$, respectively, such that the new iterate $w^{k+1} = (x^{k+1}, y^{k+1}, s^{k+1})$ defined by (2.9) satisfies (2.10).

Step 4: Let $k := k+1$. Go to Step 1. $\diamond$

It is easy to check that all the iterates w^k, $k \geq 0$, are primal-dual strictly feasible. Thus the only optimality condition that remains to be satisfied (asymptotically) by the iterates is the zero duality gap [2, 18], i.e., $(x^k)^\top s^k = c^\top x^k - b^\top y^k \to 0$ as $k \to \infty$, which explains the termination criterion in Step 1.

The details of how to perform Step 3 of the PDC algorithm are not relevant here; a comprehensive account is given in [3]. Note that ensuring condition (2.10) is the minimal requirement on the stepsize in any primal-dual IPM, for the latter to be well-defined.

3 An example of failure of the PDC algorithm

Some interesting features of the PDC algorithm are exposed by the LP problem

$$\min_{x\in\mathbb{R}^3} \quad x_1 + \alpha x_2 \quad \text{subject to} \quad x_2 + x_3 = 2, \quad x = (x_1, x_2, x_3) \geq 0, \tag{3.1}$$

which depends on a positive parameter α. Its dual problem is

$$\max_{(y,s)\in\mathbb{R}\times\mathbb{R}^3} \quad 2y \quad \text{subject to} \quad s_1 = 1,\ y + s_2 = \alpha,\ y + s_3 = 0, \quad s = (s_1, s_2, s_3) \geq 0. \tag{3.2}$$

For any $\alpha > 0$, problems (3.1) and (3.2) have the unique solution $w^* = (x^*, y^*, s^*)$, where

$$x^* = (0, 0, 2), \quad y^* = 0, \quad \text{and} \quad s^* = (1, \alpha, 0), \tag{3.3}$$

and the IPM conditions are satisfied.

3.1 Theoretical analysis of the example

Consider the behaviour in exact arithmetic of the PDC algorithm when applied to problems (3.1) and (3.2). We show that, if the centring parameter σ^k is set to the same value $\sigma \in (0, 1)$ on each iteration, then there exist starting points w^0 such that the sequence of duality gaps of the generated iterates does not converge to zero, which implies that the iterates do not converge to the solution of problems (3.1) and (3.2).

Theorem 3.1 *Let the* PDC *algorithm be applied to problems* (3.1) *and* (3.2)*, for some* $\alpha > 0$*, and let the centring parameters* σ^k *satisfy*

$$\sigma^k := \sigma \in (0, 1), \quad k \geq 0. \tag{3.4}$$

Let the starting point $w^0 = (x^0, y^0, s^0)$ *of the algorithm be any primal-dual strictly feasible point of* (3.1) *and* (3.2) *with*

$$x_2^0 \geq \xi \quad \text{and} \quad s_3^0 \leq \nu, \tag{3.5}$$

where $\xi := 2 - \sigma/2$ and $\nu := \alpha\,\sigma/8$. Then the sequence of duality gaps of the iterates generated by the algorithm is bounded away from zero, and the following bound holds

$$(x^k)^\top s^k > \xi\alpha, \quad \textit{for all } k \geq 0. \tag{3.6}$$

Thus the PDC *algorithm does not converge to the solution* (3.3) *of problems* (3.1) *and* (3.2).

To prove Theorem 3.1, we first identify conditions on the current iterate w^k of the PDC algorithm such that some of the components of the correctors $dx^{k,c}$ and $ds^{k,c}$ are greater in absolute value than their dx^k and ds^k counterparts, yielding a search direction that prevents the progress of w^k, in particular of x^k, towards the optimum.

Lemma 3.2 *Consider problems* (3.1) *and* (3.2)*, for some $\alpha > 0$. Let $w^k = (x^k, y^k, s^k)$, $k \geq 0$, be the sequence of iterates generated by the* PDC *algorithm when applied to these problems. If*

$$x_2^k \geq \xi^k := 2 - \frac{1}{2}\sigma^k \quad \textit{and} \quad s_3^k \leq \nu^k := \frac{1}{8}\alpha\,\sigma^k, \tag{3.7}$$

then

$$dx_2^{k,c} > -dx_2^k > 0 \quad \textit{and} \quad -ds_2^{k,c} > ds_2^k > 0, \tag{3.8}$$

which imply

$$dx_2^{k,r} = -dx_3^{k,r} > 0 \quad \textit{and} \quad ds_2^{k,r} = ds_3^{k,r} < 0. \tag{3.9}$$

Thus

$$x_2^{k+1} > x_2^k \quad \textit{and} \quad s_3^{k+1} < s_3^k. \tag{3.10}$$

Proof of Lemma 3.2. See Appendix A. □

The following proof shows that (3.10) may hold for all k. Then, $\{x_2^k\}$ is increasing from $x_2^0 > 0$ and cannot converge to $x_2^* = 0$ in (3.3).

Proof of Theorem 3.1. Recalling (3.4), let $\xi^k = \xi := 2 - \sigma/2$ and $\nu^k = \nu := \alpha\,\sigma/8$, $k \geq 0$, which also occur in (3.7). Due to condition (3.5), Lemma 3.2 applies with $k = 0$. Thus, from (3.10),

$$x_2^1 > x_2^0 \geq \xi \quad \text{and} \quad s_3^1 < s_3^0 \leq \nu. \tag{3.11}$$

Thus Lemma 3.2 applies again, this time for $k = 1$, and by the same argument as for $k = 0$, we deduce the analogue of the relations (3.11), where each index is increased by one. Inductively, relations (3.11) hold for $k \geq 0$. They provide the bound $x_2^k \geq \xi > 0$, $k \geq 0$, which implies, together with the feasibility condition $s_2^k > \alpha$, that the complementarity products $x_2^k s_2^k$ are bounded below by the positive constant $\xi\alpha$ for all $k \geq 0$. The bound (3.6) now follows from (2.1) and (2.10) which give $(x^k)^\top s^k > x_2^k s_2^k$, $k \geq 0$. The conclusion now holds due to the bound (3.6) and standard optimality conditions for LP [2, 22]. □

Remark The failure of the PDC to converge also occurs if $\{\sigma^k\}$ in Theorem 3.1 is non-decreasing (see Lemma 4.2 in [3]). The practical relevance of this choice is illustrated in Example 2 of Section 3.2. ◇

Note that subject to the conditions of Theorem 3.1, the (long-step) PD algorithm with a suitable choice of stepsize is guaranteed to converge to the solution of the problems (3.1) and (3.2) (see [22] for a general result). However, as we have just shown, in the case of the PDC, no such suitable stepsize technique exists that would make it convergent in the conditions of Theorem 3.1; then, as indicated by Lemma 3.2, the influence of the correctors is overpowering the Newton direction in the construction of the iterates, preventing the latter from reaching optimality.

A short-step variant of the PDC, where the iterates, including the starting point, are constrained to be (very) close to the central path, can be shown to be convergent as then, the influence of the correctors on the resulting direction is negligible [2, 3].

3.2 Numerical calculations

We now illustrate the numerical performance of the PDC algorithm when applied to problems (3.1) and (3.2), for certain values of the parameters. In Step 3 of the PDC, we use the following popular choice of stepsize [18]: compute the steps $\overline{\theta}_p^k$ and $\overline{\theta}_d^k$ to the boundaries of the primal and dual nonnegative bound constraints, that is

$$\overline{\theta}_p^k := 1/\max\,(0, -dx_i^{k,r}/x_i^k, i = \overline{1,n}), \quad \text{and} \quad \overline{\theta}_d^k := 1/\max\,(0, -ds_i^{k,r}/s_i^k, i = \overline{1,n}). \tag{3.12}$$

Then, having chosen $\tau \in (0,1)$ at the start of the algorithm, set

$$\theta_p^k := \min\,(1, \tau\overline{\theta}_p^k) \quad \text{and} \quad \theta_d^k := \min\,(1, \tau\overline{\theta}_d^k). \tag{3.13}$$

Example 1. We set the parameters of the algorithm to the values

$$\sigma^k := 0.1, \;\text{ for }\; k \geq 0, \quad \text{and} \quad \tau := 0.995, \quad \text{and} \quad \epsilon := 10^{-8}, \tag{3.14}$$

and applied the algorithm to (3.1) and (3.2) with $\alpha := 8$, starting from $w^0 = (x^0, y^0, s^0)$,

$$x^0 := (8, 1.95, 0.05), \quad y^0 := -0.1, \quad s^0 := (1, 8.1, 0.1), \tag{3.15}$$

which is a primal-dual strictly feasible point of these problems.

The conditions of Theorem 3.1 are satisfied in this case, implying that the duality gap $(x^k)^\top s^k$ of the iterates cannot be decreased to a value lower than $\xi\alpha = 1.95 \cdot 8 = 15.6$ (see (3.6)).

k	$(x_1^k, x_2^k, x_3^k)^\top$	$(dx_1^k, dx_2^k, dx_3^k)^\top$	$(dx_1^{k,c}, dx_2^{k,c}, dx_3^{k,c})^\top$	$\overline{\theta}_p^k$	$x_1^k + 8x_2^k$	$(x^k)^\top s^k$
	8.0000	-7.2067	0			
0	1.9500	-3.8122	$1.0347 \cdot 10^2$	$5.0173 \cdot 10^{-4}$	$2.3600 \cdot 10$	$2.3800 \cdot 10$
	$5.0000 \cdot 10^{-2}$	3.8122	$-1.0347 \cdot 10^2$			
	7.9964	-7.1966	0			
1	1.9997	$-5.3443 \cdot 10^2$	$7.5908 \cdot 10^8$	$3.2935 \cdot 10^{-13}$	$2.3994 \cdot 10$	$2.3995 \cdot 10$
	$2.5000 \cdot 10^{-4}$	$5.3443 \cdot 10^2$	$-7.5908 \cdot 10^8$			
	7.9964	-7.1965	0			
2	2.0000	$-1.0665 \cdot 10^5$	$6.0664 \cdot 10^{15}$	$2.0605 \cdot 10^{-22}$	$2.3996 \cdot 10$	$2.3996 \cdot 10$
	$1.25 \cdot 10^{-6}$	$1.0665 \cdot 10^5$	$-6.0664 \cdot 10^{15}$			
	7.9964	-7.1965	0			
3	2.0000	$-2.1330 \cdot 10^7$	$4.8531 \cdot 10^{22}$	$1.2878 \cdot 10^{-31}$	$2.3996 \cdot 10$	$2.3996 \cdot 10$
	$6.2500 \cdot 10^{-9}$	$2.1330 \cdot 10^7$	$-4.8531 \cdot 10^{22}$			
	7.9964	-7.1965	0			
4	2.0000	$-4.2660 \cdot 10^9$	$3.8825 \cdot 10^{29}$	$8.0490 \cdot 10^{-41}$	$2.3996 \cdot 10$	$2.3996 \cdot 10$
	$3.1250 \cdot 10^{-11}$	$4.2660 \cdot 10^9$	$-3.8825 \cdot 10^{29}$			

Table 1: The first five primal iterates of the PDC algorithm when applied to (3.1) and (3.2). The algorithm halts after 6 iterations.

The data in Table 1 shows that the pair (x_2^k, x_3^k) approach the point $(2, 0)$ rapidly, and x^4 is within ϵ distance to the nonoptimal boundaries determined by the constraints $x_2 = 2$ and $x_3 = 0$. The values in the third and fourth column of Table 1 indicate that the lengths of dx^k and $dx^{k,c}$ increase very rapidly with k, due to the length of their second and third components. Moreover, $dx_2^{k,c}$ and $dx_3^{k,c}$ are much longer in absolute value than, and have opposite signs to, dx_2^k and dx_3^k, conforming to (3.8) and (3.9). The direction dx^k 'points towards' the optimum $x^* = (0, 0, 2)$, while the second and third components of the corrector 'point away' from it. Thus these primal components of the resulting direction $dw^{k,r}$ point away from the optimum.

In the dual space, after four iterations, the dual iterates and the dual objective function are within $\epsilon = 10^{-8}$ of their optimal values (see (3.3)). The rapid increase in the lengths of both ds^k and $ds^{k,c}$ is similar in magnitude to the length of the primal directions. The primal-dual feasibility equations are satisfied to machine precision throughout.

Since $x_2^k \nearrow 2$ and $s_2^k \searrow \alpha = 8$, as k increases, the matrix of the systems (2.4) and (2.6) converges to a singular matrix. The increasing ill-conditioning ultimately stops the algorithm. For a more detailed analysis of the numerical results, see [3]. ◇

A numerical calculation with a popular choice of σ^k is given next.

Example 2. For $k \geq 0$, we compute the centring parameters $\sigma^k > 0$ in the PDC algorithm by the procedure employed in the MPC algorithm [15, 22]. Thus we let

$$\sigma^k := \left(\frac{(x^k + \overline{\theta}_p^k dx^{k,a})^\top (s^k + \overline{\theta}_d^k ds^{k,a})}{(x^k)^\top s^k} \right)^i, \tag{3.16}$$

where $dw^{k,a} = (dx^{k,a}, dy^{k,a}, ds^{k,a})$ is defined by (2.5) with $\sigma^k := 0$. The stepsizes $\overline{\theta}_p^k$ and $\overline{\theta}_d^k$ are the maximum steps from x^k and s^k, along $dx^{k,a}$ and $ds^{k,a}$, to the primal and dual constraint boundaries, respectively, and are defined by (3.12) with $dw^{k,r} := dw^{k,a}$. The index $i \in \{1, 2, 3, 4\}$ is a constant that we fix at the start of the PDC algorithm. See Chapter 10 of [22] for more explanations about this choice of σ^k. The PDC algorithm with this choice of σ^k is still distinct from the MPC, due to the different corrector directions the two algorithms employ (for precise details, see the first paragraph of Section 4.2).

Let $\alpha := 8$ in (3.1) and (3.2), and let the starting point $w^0 = (x^0, y^0, s^0)$ of the PDC algorithm with the stepsize procedure (3.13) be

$$x^0 := (8, 1.99, 0.01), \quad y^0 := -0.1, \quad s^0 := (1, 8.1, 0.1). \tag{3.17}$$

Let $\epsilon := 10^{-8}$, $\tau := 0.995$ and σ^k be computed from (3.16) with $i = 3$ for $k \geq 0$. The iterates generated by the algorithm are very similar to the ones in Table 1. For example, the (x_2^4, x_3^4) components are within 10^{-11} distance to the point $(2, 0)$, and the dual iterates (y^4, s^4) are within ϵ distance to the optimum. The numerical values of σ^k, $k = \overline{0, 6}$, are all of order 10^{-1} and are strictly increasing. This case is also covered by our theoretical results (see our remark following the proof of Theorem 3.1); in particular, Lemma 4.2 in [3] implies that the PDC does not converge to the solution of (3.1) and (3.2).

The behaviour of the algorithm is similar for any $i \in \{1, 2, 4\}$ in (3.16). $\diamond$

Our numerical experience with the PDC algorithm is not restricted to the example problems (3.1) and (3.2); the algorithm terminates at Step 1 on most LP instances tested.

4 Conclusions

4.1 Overcoming the failure: the Primal-Dual Second-Order Corrector (PDSOC) algorithm

The PDC algorithm presented here computes on each iteration an additional direction, a corrector, to augment the direction of the standard primal-dual path-following interior-point method for LP problems, in an attempt to improve performance. We found, however, that the PDC may fail to converge to the solution of problems (3.1) and (3.2) in both exact and finite arithmetic, regardless of the choice of stepsize that is employed. The cause of the bad performance of the algorithm on these problems is that the corrector direction had too much influence on the resulting search direction. Therefore in the PDSOC algorithm [2, 7, 17, 23, 25], the contribution from the corrector is the quadratic function of the steplength $\theta^k = \theta_p^k = \theta_d^k$

$$w^{k+1} := w^k + \theta^k dw^k + (\theta^k)^2 dw^{k,c}, \tag{4.1}$$

where $w^k = (x^k, y^k, s^k)$ is the current iterate of the PDSOC algorithm applied to problems (P) and (D), and the directions dw^k and $dw^{k,c}$ are computed as before, for $k \geq 0$.

The quadratic features of the linesearch (4.1) are supported by the interpretation that the new iterate w^{k+1} is chosen along the second-order Taylor approximation around w^k of a local nonlinear path that starts at w^k and ends at the point $w(\sigma^k \mu^k)$ of the primal-dual central path of the problems (see Section 5.1 of [2]).

Convergence and complexity properties of the PDSOC algorithm are given in [2, 25]. It can be shown (see Appendix C of [2]) that these results ensure that, subject to the conditions of Theorem 3.1, the PDSOC algorithm with suitable (long-step) linesearch converges in exact arithmetic to the solution of problems (3.1) and (3.2).

4.2 The relevance of the example to the MPC algorithm

Relating the construction of the PDC to that of the MPC algorithm, we find that, when applied to problems (P) and (D), the search direction generated in the MPC algorithm is also the sum of dw^k and a corrector direction. The MPC corrector, however, attempts to adjust the error generated in the system of optimality conditions of problems (P) and (D) (i.e., the system $F_\mu(w) = 0$ in (2.2) with $\mu := 0$) by its Newton direction, $dw^{k,a}$, from w^k. Thus the MPC corrector direction is defined by the system (2.6) with $\sigma^k := 0$ and $dw^k := dw^{k,a}$. We remark that the centring parameters in the MPC algorithm are computed as in (3.16).

As we already mentioned in the introductory section, the example of failure of the PDC algorithm to converge that we presented in Section 3 does not apply to the MPC algorithm. Our implementation of the MPC algorithm with the stepsize procedure (3.13) was successful in solving problems (3.1) and (3.2), for various starting points, including those defined in (3.15) or in (3.17). In the latter case, the MPC algorithm similarly generates long correctors that move the primal iterate away from the optimum on early iterations. It "recovers", however, and converges rapidly to the solution. The example throws doubt nevertheless, on the convergence properties of the MPC algorithm in general, due to the above-mentioned similarities between the two algorithms, particularly in the way the search directions and new iterates are formed by adding similar corrector directions to the standard Newton direction of primal-dual interior point methods, without any scaling of the correctors, which is the cause of failure of the PDC algorithm on the example. Based on our experience with the PDC and the similarities between the two algorithms, it seems highly unlikely that the occurrence of long corrector directions in the performance of the MPC algorithm would always have a beneficial or harmless effect. A theoretical understanding of the numerical behaviour of the MPC algorithm constitutes potential future work.

Besides its essential and strong connection to the MPC algorithm, we find the PDC algorithm to be interesting in itself, since the examples we presented emphasize the disadvantages of this particular way of constructing corrector directions and new iterates.

References

[1] T. J. Carpenter, I. J. Lustig, J. M. Mulvey, and D. F. Shanno. Higher-order predictor-corrector interior point methods with application to quadratic objectives. *SIAM J. Optim.*, 3(4):696–725, 1993.

[2] C. Cartis. On interior point methods for linear programming, 2004. PhD thesis, Department of Applied Mathematics and Theoretical Physics, University of Cambridge.

[3] C. Cartis. Some disadvantages of a Mehrotra-type primal-dual corrector interior point algorithm for linear programming. Technical Report 04/27, Numerical Analysis Group, Oxford University Computing Laboratory, 2005. Available at `http://www.optimization-online.org/DB_HTML/2005/02/1062.html`.

[4] R. M. Freund and S. Mizuno. Interior point methods: current status and future directions. In *High performance optimization*, volume 33 of *Appl. Optim.*, pages 441–466. Kluwer Acad. Publ., Dordrecht, 2000.

[5] J. Gondzio. Multiple centrality corrections in a primal-dual method for linear programming. *Comput. Optim. Appl.*, 6(2):137–156, 1996.

[6] C. C. Gonzaga. Path-following methods for linear programming. *SIAM Rev.*, 34(2), 1992.

[7] P.-F. Hung and Y. Ye. An asymptotical $\mathcal{O}(\sqrt{n})$-iteration path-following linear programming algorithm that uses wide neighbourhoods. *SIAM J. Optim.*, 6(3):570–586, 1996.

[8] M. Kojima, N. Megiddo, and S. Mizuno. A primal-dual infeasible-interior-point algorithm for linear programming. *Math. Programming*, 61(3, Ser. A):263–280, 1993.

[9] M. Kojima, S. Mizuno, and A. Yoshise. A primal-dual interior point algorithm for linear programming. In *Progress in mathematical programming (Pacific Grove, CA, 1987)*, pages 29–47. Springer, New York, 1989.

[10] I. J. Lustig, R. E. Marsten, and D. F. Shanno. Computational experience with a primal-dual interior point method for linear programming. *Linear Algebra Appl.*, 152:191–222, 1991.

[11] I. J. Lustig, R. E. Marsten, and D. F. Shanno. On implementing Mehrotra's predictor-corrector interior-point method for linear programming. *SIAM J. Optim.*, 2(3):435–449, 1992.

[12] S. Mehrotra. Higher order methods and their performance. Technical Report 90–16R1, Dept. of IE/MS, Northwestern University, 1990.

[13] S. Mehrotra. Generalized predictor-corrector methods and their performance. Technical Report 91–17, Dept. of IE/MS, Northwestern University, 1991.

[14] S. Mehrotra. On finding a vertex solution using interior point methods. *Linear Algebra Appl.*, 152:233–253, 1991.

[15] S. Mehrotra. On the implementation of a primal-dual interior point method. *SIAM J. Optim.*, 2(4):575–601, 1992.

[16] S. Mehrotra. Asymptotic convergence in a generalized predictor-corrector method. *Math. Programming*, 74(1, Ser. A):11–28, 1996.

[17] R. D. C. Monteiro, I. Adler, and M. G. C. Resende. A polynomial-time primal-dual affine scaling algorithm for linear and convex quadratic programming and its power series extension. *Math. Oper. Res.*, 15(2):191–214, 1990.

[18] J. Nocedal and S. J. Wright. *Numerical optimization*. Springer-Verlag, New York, 1999.

[19] F. A. Potra and S. J. Wright. Interior-point methods. *J. Comput. Appl. Math.*, 124(1-2):281–302, 2000.

[20] R. Tapia, Y. Zhang, M. Saltzman, and A. Weiser. The Mehrotra predictor-corrector interior-point method as a perturbed composite Newton method. *SIAM J. Optim.*, 6(1):47–56, 1996.

[21] M. H. Wright. The interior-point revolution in constrained optimization. In *High performance algorithms and software in nonlinear optimization (Ischia, 1997)*, volume 24 of *Appl. Optim.*, pages 359–381. Kluwer Acad. Publ., Dordrecht, 1998.

[22] S. J. Wright. *Primal-dual Interior-Point Methods.* Society for Industrial and Applied Mathematics (SIAM), Philadelphia, 1997.

[23] Y. Ye. *Interior Point Algorithms:Theory and Analysis.* John Wiley and Sons, New York, 1997.

[24] Y. Zhang. User's guide to LIPSOL: linear-programming interior point solvers V0.4. *Optim. Methods Softw.*, 11/12(1-4):385–396, 1999.

[25] Y. Zhang and D. T. Zhang. On polynomiality of the Mehrotra-type predictor-corrector interior-point algorithms. *Math. Programming*, 68(3, Ser. A):303–318, 1995.

A Appendix

Here, we give a proof of Lemma 3.2. Firstly, let us perform some calculations that will be used later, in the proof.

Assume that we apply the PDC algorithm to problems (3.1) and (3.2) (in exact arithmetic). Then, the strict feasibility of the iterates implies that every iterate $w^k = (x^k, y^k, s^k)$ satisfies the equations

$$x_2^k + x_3^k = 2, \quad s_1^k = 1, \quad y^k + s_2^k = \alpha, \quad y^k + s_3^k = 0, \quad s_2^k = s_3^k + \alpha, \tag{A.2}$$

and the inequalities

$$0 < x_1^k, \quad 0 < x_2^k < 2, \quad 0 < x_3^k < 2, \quad s_2^k > \alpha, \quad s_3^k > 0, \quad k \geq 0. \tag{A.3}$$

The direction $dw^k = (dx^k, dy^k, ds^k)$ defined by (2.5) has the following explicit expression

$$dx_1^k = -x_1^k + \sigma^k \mu^k, \quad dx_2^k = \frac{2\sigma^k \mu^k (1 - x_2^k) - \alpha x_2^k x_3^k}{2s_2^k - \alpha x_2^k}, \quad dx_2^k = -dx_3^k, \tag{A.4a}$$

$$ds_1^k = 0, \quad ds_2^k = \frac{\sigma^k \mu^k (2s_2^k - \alpha) - 2s_2^k s_3^k}{2s_2^k - \alpha x_2^k}, \quad ds_3^k = ds_2^k, \quad dy^k = -ds_2^k. \tag{A.4b}$$

The expression of the corrector direction $dw^{k,c} = (dx^{k,c}, dy^{k,c}, ds^{k,c})$ follows from the systems (2.6) and (2.7) and it is

$$dx_1^{k,c} = 0, \quad dx_2^{k,c} = \frac{-2ds_2^k}{2s_2^k - \alpha x_2^k} dx_2^k, \quad dx_3^{k,c} = -dx_2^{k,c}, \tag{A.5a}$$

$$ds_1^{k,c} = 0, \quad ds_2^{k,c} = \frac{\alpha dx_2^k}{2s_2^k - \alpha x_2^k} ds_2^k, \quad ds_3^{k,c} = ds_2^{k,c}, \quad dy^{k,c} = -ds_2^{k,c}. \tag{A.5b}$$

From (A.4) and (A.5), we deduce that the resulting search direction $dw^{k,r} = dw^k + dw^{k,c}$ has the components

$$dx_1^{k,r} = -x_1^k + \sigma^k \mu^k, \quad dx_2^{k,r} = -dx_3^{k,r} = dx_2^k \left(1 - \frac{2ds_2^k}{2s_2^k - \alpha x_2^k}\right), \tag{A.6a}$$

$$ds_1^{k,r} = 0, \quad ds_2^{k,r} = ds_3^{k,r} = ds_2^k \left(1 + \frac{\alpha dx_2^k}{2s_2^k - \alpha x_2^k}\right). \tag{A.6b}$$

Proof of Lemma 3.2. Throughout the proof, we drop the iteration superscript k. Firstly, note that (3.10) immediately follows from (2.9), (3.9) and $\theta_{p,d} > 0$. We remark that $\xi \in (1.5, 2)$ and $\nu > 0$, since $\sigma \in (0, 1)$ and $\alpha > 0$. Thus from (3.7), (A.2) and (A.3), we have

$$x_2 = 2 - x_3 \in [\xi, 2) \quad \text{and} \quad s_3 = s_2 - \alpha \in (0, \nu]. \tag{A.7}$$

Since $x_2 < 2$ and $s_2 > \alpha$, the denominator $2s_2 - \alpha x_2$ of expressions (A.6a) and (A.6b) is positive. Therefore it is sufficient to establish the relations

$$\frac{2ds_2}{2s_2 - \alpha x_2} > 1 \quad \text{and} \quad \frac{\alpha dx_2}{2s_2 - \alpha x_2} < -1. \tag{A.8}$$

Indeed, they imply $ds_2 > 0$ and $dx_2 < 0$. Further, (A.5a) and (A.5b) give $|dx_2^c| > |dx_2|$ and $|ds_2^c| > |ds_2|$ with the sign changes of expression (3.8). The inequalities in (3.9) follow from (2.8) and (3.8), while the equalities are the expressions (A.6).

The mean value μ of the complementarity products can be written

$$\mu = \frac{1}{3}(x_1 s_1 + x_2 s_2 + x_3 s_3) = \frac{1}{3}(x_1 + \alpha x_2 + 2s_3), \tag{A.9}$$

where (A.2) gives the second equality. We substitute (A.9) and the expression (A.4b) for ds_2 into the first part of (A.8). Then, using the feasibility relation $s_2 - s_3 = \alpha$, we obtain the following equivalent expression for the first inequality in (A.8), in terms of x_2, s_3 and x_1

$$8(3-\sigma)s_3^2 + 4\alpha[9 - \sigma - (3+\sigma)x_2]s_3 + \alpha^2[3x_2^2 - 2(6+\sigma)x_2 + 12] - 2\sigma(\alpha + 2s_3)x_1 < 0. \tag{A.10}$$

It follows from $x_1 > 0$, $s_3 > 0$, $\sigma > 0$ and $\alpha > 0$, that it is sufficient to show

$$8(3-\sigma)s_3^2 + 4\alpha[9 - \sigma - (3+\sigma)x_2]s_3 + \alpha^2[3x_2^2 - 2(6+\sigma)x_2 + 12] < 0, \tag{A.11}$$

for $x_2 \in [\xi, 2)$ and $s_3 \in (0, \nu]$. The left-hand side of (A.11) is a convex function in s_3, and therefore its supremum occurs at one of the end points of the interval $(0, \nu]$. It remains to verify that (A.11) holds for $s_3 = 0$ and for $s_3 = \nu = \alpha\sigma/8$. At $s_3 = 0$, condition (A.11) becomes

$$3x_2^2 - 2(6+\sigma)x_2 + 12 < 0. \tag{A.12}$$

In order to check that (A.12) is achieved for any $x_2 \in [\xi, 2)$, it is enough to verify that it holds at $x_2 = \xi$, since the the left-hand side of (A.12) is a decreasing function of x_2. In the case $x_2 = \xi = 2 - \sigma/2$, the left-hand side of (A.12) has the value $-4\sigma + 1.75\sigma^2$, which is negative as required due to $\sigma \in (0, 1)$. For $s_3 = \alpha\sigma/8$, condition (A.11) becomes

$$24x_2^2 - 4(\sigma^2 + 7\sigma + 24)x_2 - \sigma^3 - \sigma^2 + 36\sigma + 96 < 0, \tag{A.13}$$

whose left-hand side is also decreasing in x_2. At $x_2 = \xi$, the above condition becomes $\sigma^3 + 11\sigma^2 - 20\sigma < 0$, which holds for any $\sigma \in (0, 1)$. Thus the first inequality in (A.8) is achieved.

Similarly, substituting (A.9) and the expression of dx_2 from (A.4a) into the second inequality in (A.8), and employing the feasibility relations $x_3 = 2 - x_2$ and $s_2 = \alpha + s_3$, we deduce the following form of this inequality

$$6s_3^2 + 2\alpha[6 + \sigma - (3 + \sigma)x_2]s_3 + \alpha^2[(3 - \sigma)x_2^2 - (9 - \sigma)x_2 + 6] - \alpha\sigma(x_2 - 1)x_1 < 0. \tag{A.14}$$

Since $x_1 > 0$, $\alpha > 0$, $\sigma \in (0, 1)$, $x_2 \geq \xi > 1$, it is sufficient to establish

$$6s_3^2 + 2\alpha[6 + \sigma - (3 + \sigma)x_2]s_3 + \alpha^2[(3 - \sigma)x_2^2 - (9 - \sigma)x_2 + 6] < 0, \tag{A.15}$$

for $x_2 \in [\xi, 2)$ and $s_3 \in (0, \nu]$. As before, the left-hand side of (A.15) is convex in s_3. It is thus enough to show that (A.15) holds at $s_3 = 0$ and $s_3 = \nu = \alpha\sigma/8$. At $s_3 = 0$, condition (A.15) becomes

$$3(x_2^2 - 3x_2 + 2) + \sigma x_2(1 - x_2) < 0, \tag{A.16}$$

which holds for any $x_2 \in (1, 2)$ and $\sigma \in (0, 1)$. For $s_3 = \alpha\sigma/8$, condition (A.15) becomes

$$32(3 - \sigma)x_2^2 - 8(\sigma^2 - \sigma + 36)x_2 + 11\sigma^2 + 48\sigma + 192 < 0. \tag{A.17}$$

The left-hand side of (A.17) is convex in x_2. Substituting $x_2 = 2$ in expression (A.17) yields $-5\sigma^2 - 64\sigma < 0$. At $x_2 = \xi = 2 - \sigma/2$, the left-hand side of (A.17) is $-4\sigma^3 + 79\sigma^2 - 112\sigma$ which is negative for any $\sigma \in (0, 1)$. This proves that the second inequality in (A.8) also holds. □