

On Mehrotra-Type Predictor-Corrector Algorithms

M. Salahi^{*†}, J. Peng[†], T. Terlaky[†]

April 7, 2005

Abstract

In this paper we discuss the polynomiality of Mehrotra-type predictor-corrector algorithms. We consider a variant of the original prototype of the algorithm that has been widely used in several IPM based optimization packages, for which no complexity result is known to date. By an example we show that in this variant the usual Mehrotra-type adaptive choice of the parameter μ might force the algorithm to take many small steps to keep the iterates in a certain neighborhood of the central path, which is essential to prove the polynomiality of the algorithm.

This example has motivated us to incorporate a safeguard in the algorithm that keeps the iterates in the prescribed neighborhood and allows us to get a reasonably large step size. This safeguard strategy is used also when the affine scaling step performs poorly that forces the algorithm to take pure centering steps. We proved that the modified algorithm, in the worst case, will terminate after at most $\mathcal{O}(n^2 \log \frac{(x^0)^T s^0}{\epsilon})$ iterations. By slightly modifying the Newton system in the corrector step, we reduce the iteration complexity to $\mathcal{O}(n \log \frac{(x^0)^T s^0}{n})$. To ensure the asymptotic convergence of the algorithm, we changed Mehrotra's heuristic slightly. The new algorithms have the same iteration complexity as the previous one, but enjoy superlinear convergence rate as well.

Keywords: Linear Optimization, Predictor-Corrector Method, Interior-Point-Methods, Mehrotra-Type Algorithm, Polynomial Complexity, Superlinear Convergence.

AMS Subject Classification: 90C05, 90C51

^{*}Department of Mathematics and Statistics, McMaster University, Hamilton, Ontario, Canada, L8S 4K1, salahim@math.mcmaster.ca

[†]Advanced Optimization Lab, Department of Computing and Software, McMaster University, Hamilton, Ontario, Canada, L8S 4L7, pengj, terlaky@mcmaster.ca,

1 Introduction

Since Karmarkar's landmark paper [6], Interior Point Methods (IPMs) became one of the most active research area that produced a large amount of research results. Moreover, several powerful IPMs based optimization packages have been developed. Predictor-Corrector methods are among the most efficient IPMs and have become the back bones of several optimization packages [22, 24]. Most implementations are based on a variant of Mehrotra's [8] predictor-corrector algorithm. The practical importance of Mehrotra's algorithm, and lack of theoretical analysis of this method motivated us to study Mehrotra-type IPMs. Before going in the details of Mehrotra's algorithm, first we briefly review the basics and unique results of IPMs and predictor-corrector IPMs.

We consider primal-dual IPMs for solving the following standard forms *Linear Optimization* (LO) problem:

$$(P) \quad \min \{c^T x : Ax = b, x \geq 0\},$$

where $A \in R^{m \times n}$ satisfies $\text{rank}(A) = m$, $b \in R^m$, $c \in R^n$, and its dual problem

$$(D) \quad \max \{b^T y : A^T y + s = c, s \geq 0\}.$$

Without loss of generality [17] we may assume that both (P) and (D) satisfy the interior point condition (IPC), i.e., there exists an (x^0, s^0, y^0) such that

$$Ax^0 = b, x^0 > 0, \quad A^T y^0 + s^0 = c, s^0 > 0.$$

Before getting to the main theme of the paper we first give a brief introduction to IPMs. Finding optimal solutions of (P) and (D) is equivalent to solving the following system:

$$\begin{aligned} Ax &= b, \quad x \geq 0, \\ A^T y + s &= c, \quad s \geq 0, \\ xs &= 0. \end{aligned} \tag{1}$$

The basic idea of primal-dual IPMs is to replace the third equation in (1) by the parameterized equation $xs = \mu e$, where e is the all one vector. This leads to the following system:

$$\begin{aligned} Ax &= b, \quad x \geq 0, \\ A^T y + s &= c, \quad s \geq 0, \\ xs &= \mu e. \end{aligned} \tag{2}$$

If the IPC holds, then for each $\mu > 0$, system (2) has a unique solution. This solution, denoted by $(x(\mu), y(\mu), s(\mu))$, is called the μ -center of the primal-dual pair (P) and (D). The set of μ -centers with all $\mu > 0$ gives the *central path* of (P) and (D) [7, 19]. It has been shown that the limit of the central path (as μ goes to zero) exists. Because the limit point satisfies the complementarity condition, it naturally yields optimal solutions for both (P) and (D), respectively [17].

Applying Newton's method to (2), gives the following linear system of equations

$$\begin{aligned} A\Delta x &= 0, \\ A^T \Delta y + \Delta s &= 0, \\ x\Delta s + s\Delta x &= \mu e - xs, \end{aligned} \tag{3}$$

where Δx , Δy , Δs give the Newton step.

Predictor-corrector algorithms use (3) with different values of μ in the predictor and corrector steps. The best known predictor-corrector algorithm is the Mizuno-Todd-Ye (MTY) algorithm for LO, that operates in two small neighborhoods of the central path [10]. In the predictor step the MTY algorithm use the so-called primal-dual affine scaling step with $\mu = 0$ in (3) and it moves to a slightly larger neighborhood. Then, in the corrector step, it uses $\mu = \mu_g = \frac{x^T s}{n}$, proportional to the duality gap to bring the iterate towards the central path, back to the smaller neighborhood. Although there are strong theoretical results behind this algorithm for LO and some other classes of conic linear optimization problems, it has not been used in developing IPMs based software packages. Several variants of MTY type predictor-corrector algorithms in both small and large neighborhoods have been proposed in the past decade, but most of them dealing only with the theoretical framework [1, 4, 5, 12, 13, 14, 15, 16, 18].

A more practical algorithm was proposed by Mehrotra [8] based on the power series algorithm of Bayer and Lagarias [2] and its primal-dual version by Monteiro, Adler and Resende [11]. Zhang and Zhang [23] proved the polynomiality of a second order variant of Mehrotra's original algorithm.

In what follows we describe in details a popular version of Mehrotra's algorithm that has been adopted in most IPM packages. A feature of this algorithm is the dynamic update of the barrier parameter. In the predictor step Mehrotra's algorithm computes the affine scaling search direction, i.e.,

$$\begin{aligned} A\Delta x^a &= 0, \\ A^T \Delta y^a + \Delta s^a &= 0, \\ s\Delta x^a + x\Delta s^a &= -xs, \end{aligned} \tag{4}$$

then it computes the maximum feasible step size that ensures

$$(x + \alpha_a \Delta x^a, s + \alpha_a \Delta s^a) \geq 0.$$

However, the algorithm does not take such a step. It uses the information from the predictor step to compute the centering direction that is defined as follows.

$$\begin{aligned} A\Delta x &= 0, \\ A^T \Delta y + \Delta s &= 0, \\ s\Delta x + x\Delta s &= \mu e - xs - \Delta x^a \Delta s^a, \end{aligned} \tag{5}$$

where μ is defined adaptively by

$$\mu = \left(\frac{g_a}{g} \right)^2 \frac{g_a}{n},$$

where $g_a = (x + \alpha_a \Delta x^a)^T (s + \alpha_a \Delta s^a)$ and $g = x^T s$. Since $(\Delta x^a)^T \Delta s^a = 0$, the previous relation implies

$$\mu = (1 - \alpha_a)^3 \mu_g. \quad (6)$$

Finally, Mehrotra's algorithm makes a step in the $(\Delta x, \Delta y, \Delta s)$ direction by an appropriate step size.

The excellent practical performance of this variant of Mehrotra's method motivated us to analyze its worst case iteration complexity. This is the first theoretical analysis of this variant.¹ *In this paper we show by an example that Mehrotra's heuristic may result in very small steps in order to keep the iterate in a certain neighborhood of the central path, which is essential to prove the polynomiality of the algorithm. To avoid this trap, we propose to incorporate a safeguard in the algorithm in order to guarantee polynomial complexity. Further, to ensure the superlinear convergence of the algorithm we changed the update scheme so that the new scheme preserves the same iteration complexity with stronger asymptotic convergence result.*

The rest of the paper is organized as follows. First, in Section 2, we present a numerical example that motivates the introduction of a safeguard in Mehrotra's heuristic. Then, in Section 3, we describe our algorithmic scheme and discuss in details the complexity analysis of our Mehrotra type algorithm with safeguard. In Section 4, we introduce a slightly modified version of our algorithm and discuss its iteration complexity. In Section 5, we further modify Mehrotra's μ -update heuristic to ensure the superlinear convergence of both algorithms. Some preliminary numerical results are reported in Section 6. Concluding remarks and future works are given in Section 7. Finally, some technical results are presented in the Appendix that will be used during the complexity analysis.

Throughout the paper $\|\cdot\|$ denotes the 2-norm of vectors. We denote by \mathcal{I} the index set $\mathcal{I} = \{1, 2, \dots, n\}$. For any two vectors x and s , xs denotes the componentwise product of the two vectors and e denotes the vector with all components equal to one. For simplicity of notation we remove the iteration index in the coming sections. We also use the notations

$$\mathcal{I}_+ = \{i \in \mathcal{I} \mid \Delta x_i^a \Delta s_i^a > 0\}, \quad \mathcal{I}_- = \{i \in \mathcal{I} \mid \Delta x_i^a \Delta s_i^a < 0\},$$

$$\mathcal{F} = \{(x, y, s) \mid (x, s) \geq 0, Ax = b, A^T y + s = c\}.$$

2 Motivation

The following example shows that using Mehrotra's algorithm, as described in the introduction, might force the algorithm to make very small steps to keep the iterate in a certain neighborhood of the central path that itself may imply the algorithm to take too many iterations to convergence. The example indicates that Mehrotra's method has to be combined with some safeguard that gives a warranted step size at each iteration.

¹In [3] the author provided an example that shows that Mehrotra-type predictor-corrector algorithms may fail to converge to an optimal solution. This paper has appeared in Optimization online when we were finalizing the paper for submission.

Let us consider the following simple LP.

$$\begin{aligned} \min \quad & -x_2 \\ \text{s.t.} \quad & 0 \leq x_1 \leq 1, \\ & 0 \leq x_2 \leq 1 + \delta x_1. \end{aligned} \tag{7}$$

If the algorithm starts from

$$x^0 = (0.28, 0.86, 0.72, 0.1568), \quad s^0 = (0.91, 0.5, 1, 1.5), \quad y^0 = (-1, -1.5), \quad \delta = 0.06,$$

where $(x^0, s^0) \in \mathcal{N}_\infty^-(0.5737)$ (see (8) for the definition of the negative infinity neighborhood), then at the first iteration it makes a very small step ($\alpha_c = \mathcal{O}(10^{-16})$) in the corrector step in order to keep the iterate in the $\mathcal{N}_\infty^-(0.5737)$ neighborhood, while the step size in the predictor step is $\alpha_a = 0.8228$. By looking at the second order terms $\Delta x^a \Delta s^a$ and $\Delta x \Delta s$ we notice the difference between these vectors that may cause such a phenomenon

$$\begin{aligned} \Delta x^a \Delta s^a &= (-0.0116, -0.0553, 0.0120, 0.0549), \\ \Delta x \Delta s &= (-0.0249, -0.0719, 0.0257, 0.0710), \\ \mu_g &= 0.41, \quad \mu = 0.0023. \end{aligned}$$

If for the same example with $\delta = 0.08$, where the algorithm starts from the point

$$\begin{aligned} x^0 &= (0.255688159275703, 0.900928060482674, 0.744311840724297, 0.119526992259382), \\ y^0 &= (-0.838967769079751, -1.41512087750413), \\ s^0 &= (0.725758098879421, 0.415120877504125, 0.838967769079751, 1.41512087750413), \end{aligned}$$

which belongs to the $\mathcal{N}_\infty^-(0.5)$ neighborhood, then at the first iteration one has $\alpha_a = 0.838997390116749$ that by $\mu_g = 0.338290146525301$ gives $\mu = 0.00141184849887486$ and $\alpha_c = 1.76986987950339e - 006$. Seeing the difference between the vectors

$$\Delta^a x \Delta^a s = (-0.01515115, -0.03752814, 0.01585476, 0.03682453)$$

and

$$\Delta x \Delta s = (-0.03207875, -0.04746870, 0.03347581, 0.04607164)$$

may explain this phenomenon. In addition, we have to mention that for the next few iterations α_c gives a decreasing sequence.

The previous example indicates that for any prescribed neighborhood, by modifying δ and the starting point, one can find an appropriate starting point from which the algorithm makes a very small step in the corrector step, while the affine scaling step size is sufficiently large. This example makes it difficult to argue about the theoretical, and as well as the practical efficiency of the algorithm based on such a updating strategy. This observation motivates us to introduce a safeguard strategy that will help us to have control on the minimal warranted step size both theoretically and practically.

Another drawback of Mehrotra's choice can be when the affine scaling step size is very small, then the algorithm might make pure centering steps that would only marginally

reduce the duality gap. This stalling phenomenon might hurt the practical efficiency of the algorithm as well.

Its worth mentioning that this phenomenon may also be observed by disabling all other heuristics that are combined with Mehrotra's one in the McIPM software package [24], and by solving some of the relatively large problems from the NETLIB LO test set. For example, for the PILOTNOV problem, due to many pure centering steps, the algorithm takes 314 iterations before it stops. However, if we have a safeguard, by looking at the maximum step size in the predictor and corrector steps, we can decide which updating strategy is better: an adaptive choice of the target value or the classical large update strategy. Therefore, in the new algorithm, first we evaluate α_a after the affine scaling search direction is calculated. If it is too small, then we apply a large update strategy, otherwise we refer to the adaptive strategy. If the adaptive strategy is not able to make a sufficiently large step, then it employs the large update safeguard. Therefore, in the worst case our algorithm does an extra backsolve compared to the original Mehrotra's one, which is a reasonable price to pay for the enhanced performance.

3 Algorithmic Scheme

In this section first we describe our algorithmic scheme, then we discuss the estimation of the maximum feasible step size for both the affine scaling and corrector steps.

Most efficient IPM solvers work in the negative infinity norm neighborhood defined by

$$\mathcal{N}_{\infty}^-(\gamma) := \{(x, y, s) \in \mathcal{F}^0 : x_i s_i \geq \gamma \mu_g \ \forall i = 1, \dots, n\}, \quad (8)$$

where $\gamma \in (0, 1)$ is a constant independent of n . In this paper, we consider algorithms that are working in $\mathcal{N}_{\infty}^-(\gamma)$ (called large neighborhood).

The following theorem gives a lower bound for the maximum feasible step size in the worst case for the predictor step.

Theorem 3.1 *Suppose the current iterate $(x, y, s) \in \mathcal{N}_{\infty}^-(\gamma)$ and $(\Delta x^a, \Delta y^a, \Delta s^a)$ be the solution of (4). Then the maximum feasible step size, α_a , so that $(x(\alpha_a), s(\alpha_a)) \geq 0$, satisfies*

$$\alpha_a \geq \frac{2\sqrt{\gamma^2 + \gamma n} - 2\gamma}{n}. \quad (9)$$

Proof: For $i \in \mathcal{I}_+$ any positive step is feasible. Since $(\Delta x^a)^T \Delta s^a = 0$, there is at least one $i \in \mathcal{I}_-$. Since $(x, s) \in \mathcal{N}_{\infty}^-(\gamma)$, then by Lemma A.2 (see the Appendix) we have

$$x_i(\alpha)s_i(\alpha) = (1 - \alpha)x_i s_i + \alpha^2 \Delta x_i^a \Delta s_i^a \geq \gamma \left(1 - \alpha - \frac{n\alpha^2}{4\gamma}\right) \mu_g. \quad (10)$$

Our aim is to ensure that $x_i(\alpha)s_i(\alpha) \geq 0$. For this it suffices to require that

$$\gamma \left(1 - \alpha - \frac{n\alpha^2}{4\gamma}\right) \mu_g \geq 0,$$

that is equivalent to

$$n\alpha^2 + 4\gamma\alpha - 4\gamma \leq 0.$$

This inequality holds when $\alpha \in [\frac{-2\sqrt{\gamma^2 + \gamma n - 2\gamma}}{n}, \frac{2\sqrt{\gamma^2 + \gamma n - 2\gamma}}{n}]$ that completes the proof of the lemma. \square

We use the maximum step size α_a that we computed in the predictor step to define the target value in the corrector step. The following technical lemma will be used in the next theorem, which estimates the maximum step size in the corrector step. This is a special safeguard step in a large update strategy that keeps the next iterate in $\mathcal{N}_\infty^-(\gamma)$.

Lemma 3.2 *Suppose that the current iterate is $(x, y, s) \in \mathcal{N}_\infty^-(\gamma)$ and let $(\Delta x, \Delta y, \Delta s)$ be the solution of (5), where $\mu \geq 0$. Then we have*

$$\|\Delta x \Delta s\| \leq 2^{\frac{-3}{2}} \left(\frac{1}{\gamma} \left(\frac{\mu}{\mu_g} \right)^2 - \left(2 - \frac{1}{2\gamma} \right) \frac{\mu}{\mu_g} + \frac{17\gamma + n}{16\gamma} \right) n\mu_g.$$

Proof: If we multiply the third equation of (5) by $(XS)^{-\frac{1}{2}}$, then by Lemma 5.3 of [20] we have

$$\begin{aligned} \|\Delta x \Delta s\| &\leq 2^{\frac{-3}{2}} \left\| \mu(XSe)^{-\frac{1}{2}} - (XSe)^{\frac{1}{2}} - (XS)^{-\frac{1}{2}} \Delta x^a \Delta s^a \right\|^2 \\ &= 2^{\frac{-3}{2}} \left(\mu^2 \sum_{i \in \mathcal{I}} \frac{1}{x_i s_i} + \sum_{i \in \mathcal{I}} x_i s_i + \sum_{i \in \mathcal{I}} \frac{(\Delta x_i^a \Delta s_i^a)^2}{x_i s_i} - 2n\mu - 2\mu \sum_{i \in \mathcal{I}} \frac{\Delta x_i^a \Delta s_i^a}{x_i s_i} \right) \\ &\leq 2^{\frac{-3}{2}} \left(\frac{n\mu^2}{\gamma\mu_g} + n\mu_g + \frac{n\mu_g}{16} + \frac{n^2\mu_g}{16\gamma} - 2n\mu + \frac{n\mu}{2\gamma} \right), \end{aligned}$$

where the last inequality follows from Lemma A.2 and the assumption that the previous iterate is in $\mathcal{N}_\infty^-(\gamma)$. By reordering and factorizing we conclude the proof of the lemma. \square

The following corollary gives an explicit upper bound for a specific value of μ .

Corollary 3.3 *If $\mu = \frac{\gamma}{1-\gamma}\mu_g$, where $\gamma \in (0, \frac{1}{2})$, then*

$$\|\Delta x \Delta s\| \leq \frac{\sqrt{2}}{\gamma} n^2 \mu_g.$$

Theorem 3.4 *Suppose that the current iterate $(x, y, s) \in \mathcal{N}_\infty^-(\gamma)$, where $\gamma \in (0, \frac{1}{2})$ and $(\Delta x, \Delta y, \Delta s)$ be the solution of (5) with*

$$\mu = \frac{\gamma}{1-\gamma}\mu_g.$$

Then the maximum step size α_c , that keeps $(x(\alpha_c), y(\alpha_c), s(\alpha_c))$ in $\mathcal{N}_\infty^-(\gamma)$, satisfies

$$\alpha_c \geq \frac{\gamma^2}{2n^2}.$$

Proof: The goal is to find the maximum α for which $x_i(\alpha)s_i(\alpha) \geq \gamma\mu_g(\alpha)$. If the previous inequality is true for $\alpha > \frac{1}{1+t}$, where

$$t = \max_{i \in \mathcal{I}_+} \left\{ \frac{\Delta x_i^a \Delta s_i^a}{x_i s_i} \right\} \quad (11)$$

and $t \leq \frac{1}{4}$ by Lemma A.1, then we are done. Otherwise, by using Corollary 3.3 we have

$$\begin{aligned} x_i(\alpha)s_i(\alpha) &= x_i s_i + \alpha(\mu - x_i s_i - \Delta x_i^a \Delta s_i^a) + \alpha^2 \Delta x_i \Delta s_i \\ &\geq (1 - \alpha)x_i s_i + \alpha\mu - \alpha t x_i s_i + \alpha^2 \Delta x_i \Delta s_i \\ &\geq (1 - (1+t)\alpha)x_i s_i + \alpha\mu - \frac{\sqrt{2}\alpha^2 n^2 \mu_g}{\gamma}, \\ &\geq (1 - (1+t)\alpha)\gamma\mu_g + \alpha\mu - \frac{\sqrt{2}\alpha^2 n^2 \mu_g}{\gamma}. \end{aligned}$$

We also have

$$\mu_g(\alpha) = \left(1 - \alpha + \alpha \frac{\mu}{\mu_g}\right) \mu_g.$$

Therefore, in order to stay in $\mathcal{N}_\infty^-(\gamma)$ we have to have

$$(1 - (1+t)\alpha)\gamma\mu_g + \alpha\mu - \frac{\sqrt{2}\alpha^2 n^2 \mu_g}{\gamma} \geq \gamma \left(1 - \alpha + \alpha \frac{\mu}{\mu_g}\right) \mu_g,$$

which is equivalent to

$$(1 - \gamma)\mu - \gamma t \mu_g \geq \frac{\sqrt{2}\alpha n^2 \mu_g}{\gamma}.$$

Using Lemma A.1, this is certainly true if

$$\alpha \leq \frac{\gamma^2}{2n^2}$$

that completes the proof. \square

Remark 3.5 One can get a similar lower bound for the maximum step size in the corrector step for a fixed fraction $\mu = \frac{\beta}{1-\beta}\mu_g$ of the μ_g value as target in the corrector step, for any β satisfying $\gamma < \beta < \frac{1}{2}$. This is an important point to notice, because the smaller γ is, the larger the neighborhood is, and then the $\mu = \frac{\gamma}{1-\gamma}\mu_g$ formula implies also a more aggressive update. Choosing a large β from the interval $(0, \frac{1}{2})$ would allow us to work with a more moderate fix update (disconnected from the size of the neighborhood).

Based on the previous discussion we can outline our new algorithm as follows.

Algorithm 1

Input:

A proximity parameters $\gamma \in (0, \frac{1}{2})$;

an accuracy parameter $\epsilon > 0$;

$(x^0, y^0, s^0) \in \mathcal{N}_\infty^-(\gamma)$.

begin

while $x^T s \geq \epsilon$ **do**

begin

Predictor Step

Solve (4) and compute the maximum step size α_a such that $(x(\alpha_a), y(\alpha_a), s(\alpha_a)) \in \mathcal{F}$;

end

begin

Corrector step

If $\alpha_a \geq 0.1$, then solve (5) with $\mu = (1 - \alpha_a)^3 \mu_g$ and compute the maximum step size α_c such that $(x(\alpha_c), y(\alpha_c), s(\alpha_c)) \in \mathcal{N}_\infty^-(\gamma)$;

If $\alpha_a < 0.1$, or $\alpha_c < \frac{\gamma^2}{2n^2}$, then solve (5) with $\mu = \frac{\gamma}{1-\gamma} \mu_g$ and compute the maximum step size α_c such that $(x(\alpha_c), y(\alpha_c), s(\alpha_c)) \in \mathcal{N}_\infty^-(\gamma)$;

Set $(x(\alpha_c), y(\alpha_c), s(\alpha_c)) = (x + \alpha_c \Delta x, y + \alpha_c \Delta y, s + \alpha_c \Delta s)$.

end

end

Remark 3.6 *In the worst case, comparing with the pure Mehrotra's strategy, the algorithm requires an extra backsolve to make a reasonably large step.*

The following theorem gives an upper bound for the maximum number of iterations in which Algorithm 1 stops with an ϵ -approximate solution.

Theorem 3.7 *Algorithm 1 stops after at most*

$$O\left(n^2 \log \frac{(x^0)^T s^0}{\epsilon}\right)$$

iterations with a solution for which $x^T s \leq \epsilon$.

Proof: Using the definition of $\mu_g(\alpha)$ we have

$$\mu_g(\alpha) = \left(1 - \alpha + \alpha \frac{\mu}{\mu_g}\right) \mu_g. \tag{12}$$

If $\alpha_a < 0.1$ or $\alpha_c < \frac{\gamma^2}{2n^2}$, then the algorithm uses the safeguard strategy, and by Theorem 3.4 one has

$$\mu_g(\alpha) \leq \left(1 - \frac{\gamma^2(1-2\gamma)}{2(1-\gamma)n^2}\right) \mu_g.$$

If $\alpha_a \geq 0.1$ and $\alpha_c \geq \frac{\gamma^2}{2n^2}$, then the algorithm uses the Mehrotra's updating strategy, and thus one has

$$\mu_g(\alpha) \leq \left(1 - \frac{\gamma^2}{8n^2}\right) \mu_g$$

that completes the proof by Theorem 3.2 of [20]. \square

4 A Modified Version of Algorithm 1

In this section we propose a slightly modified version of Algorithm 1 that reduces the iteration complexity significantly. The motivation for this variant is based on the proof of Theorem 3.1 that indicates that the corresponding upper bound can be strengthened by using the following lemma.

Lemma 4.1 *For $i \in \mathcal{I}_-$ one has*

$$-\Delta x_i^a \Delta s_i^a \leq \frac{1}{\alpha_a} \left(\frac{1}{\alpha_a} - 1 \right) x_i s_i. \quad (13)$$

Proof: The proof is a direct consequence of inequality (10) in the proof of Theorem 3.1. \square

We modify the corrector step by introducing the factor of α_a in the right hand side of the third equation of the corrector step. The new system becomes

$$\begin{aligned} A\Delta x &= 0, \\ A^T \Delta y + \Delta s &= 0, \\ s\Delta x + x\Delta s &= \mu e - xs - \alpha_a \Delta x^a \Delta s^a, \end{aligned} \quad (14)$$

where μ can be defined as in the previous section. In other words in the new algorithm one solves system (14) instead of system (5) in the corrector step. This slight modification of Algorithm 1 is based on the following observation. If the affine scaling search direction is a good choice, which typically implies that α_a is large, then the classical corrector direction should also be a good direction. If the affine scaling is not that good, then we should be more careful in the corrector step.

Analogous to Lemma 3.2 we have the following bound for $\|\Delta x \Delta s\|$, that for reasonably large α_a is much stronger than the one given in Lemma 3.2.

Lemma 4.2 *Suppose that the current iterate $(x, y, s) \in \mathcal{N}_\infty^-(\gamma)$ and $(\Delta x, \Delta y, \Delta s)$ be the solution of (14). Then we have*

$$\|\Delta x \Delta s\| \leq 2^{\frac{-3}{2}} \left(\frac{1}{\gamma} \left(\frac{\mu}{\mu_g} \right)^2 - \left(2 - \frac{\alpha_a}{2\gamma} \right) \frac{\mu}{\mu_g} + \frac{20 - 4\alpha_a + \alpha_a^2}{16} \right) n\mu_g.$$

Proof: Since $(\Delta x^a)^T \Delta s^a = 0$, both \mathcal{I}_+ and \mathcal{I}_- are nonempty. If we multiply the third equation of (14) by $(XS)^{-\frac{1}{2}}$, then by Lemma 5.3 of [20] we have

$$\begin{aligned}
\|\Delta x \Delta s\| &\leq 2^{\frac{-3}{2}} \left\| \mu(XSe)^{-\frac{1}{2}} - (XSe)^{\frac{1}{2}} - \alpha_a(XS)^{-\frac{1}{2}} \Delta x^a \Delta s^a \right\|^2 \\
&= 2^{\frac{-3}{2}} \left(\mu^2 \sum_{i \in \mathcal{I}} \frac{1}{x_i s_i} + x^T s + \alpha_a^2 \sum_{i \in \mathcal{I}} \frac{(\Delta x_i^a \Delta s_i^a)^2}{x_i s_i} - 2n\mu - 2\alpha_a \mu \sum_{i \in \mathcal{I}} \frac{\Delta x_i^a \Delta s_i^a}{x_i s_i} \right) \\
&\leq 2^{\frac{-3}{2}} \left(\frac{n\mu^2}{\gamma\mu_g} + n\mu_g + \frac{\alpha_a^2 n\mu_g}{16} - (1 - \alpha_a) \sum_{i \in \mathcal{I}_-} \Delta x_i^a \Delta s_i^a - 2n\mu + \frac{\alpha_a n\mu}{2\gamma} \right) \\
&\leq 2^{\frac{-3}{2}} \left(\frac{1}{\gamma} \left(\frac{\mu}{\mu_g} \right)^2 + 1 + \frac{\alpha_a^2}{16} + \frac{(1 - \alpha_a)}{4} - 2\frac{\mu}{\mu_g} + \frac{\alpha_a}{2\gamma} \frac{\mu}{\mu_g} \right) n\mu_g \\
&= 2^{\frac{-3}{2}} \left(\frac{1}{\gamma} \left(\frac{\mu}{\mu_g} \right)^2 - \left(2 - \frac{\alpha_a}{2\gamma} \right) \frac{\mu}{\mu_g} + \frac{20 - 4\alpha_a + \alpha_a^2}{16} \right) n\mu_g,
\end{aligned}$$

where the first inequality follows from (13), Lemma A.2, and the assumption that the previous iterate is in $\mathcal{N}_\infty^-(\gamma)$. The second inequality also follows from Lemma A.2, that completes the proof of the lemma. \square

The following corollary gives an explicit upper bound for a specific μ .

Corollary 4.3 *Let $\mu = \frac{\gamma}{1-\gamma}\mu_g$, where $\gamma \in (0, \frac{1}{2})$, then*

$$\|\Delta x \Delta s\| \leq 2n\mu_g.$$

In the following theorem we estimate the maximum step size in the corrector step of the modified algorithm, where the corrector step is defined by (14).

Theorem 4.4 *Suppose that the current iterate $(x, y, s) \in \mathcal{N}_\infty^-(\gamma)$, where $\gamma \in (0, \frac{1}{2})$, and $(\Delta x, \Delta y, \Delta s)$ is the solution of (14) with $\mu = \frac{\gamma}{1-\gamma}\mu_g$. Then the maximum step size α_c , such that $(x(\alpha_c), y(\alpha_c), s(\alpha_c)) \in \mathcal{N}_\infty^-(\gamma)$, satisfies*

$$\alpha_c \geq \frac{3\gamma}{8n}.$$

Proof: We need to estimate the maximum α for which $x_i(\alpha)s_i(\alpha) \geq \gamma\mu_g(\alpha)$. If this is true for $\alpha \geq \frac{1}{1+t}$, where t is given by (11) then we are done. Otherwise, since $(\Delta x^a)^T \Delta s^a = 0$, there exist at least an $i \in \mathcal{I}$ such that $\Delta x_i^a \Delta s_i^a > 0$. Since we have $\Delta x_i^a \Delta s_i^a \leq t x_i s_i$, where $0 \leq t \leq \frac{1}{4}$ by Lemma A.1. This implies

$$\begin{aligned}
x_i(\alpha)s_i(\alpha) &= x_i s_i + \alpha(\mu - x_i s_i - \alpha_a \Delta x_i^a \Delta s_i^a) + \alpha^2 \Delta x_i \Delta s_i \\
&\geq (1 - \alpha)x_i s_i + \alpha\mu - \alpha\alpha_a t x_i s_i + \alpha^2 \Delta x_i \Delta s_i \\
&\geq (1 - \alpha(1 + \alpha_a t)) x_i s_i + \alpha\mu - 2\alpha^2 n\mu_g,
\end{aligned}$$

where the last inequality is true when $\alpha \leq \frac{1}{1+\alpha_a t}$. Hence, we have

$$\mu_g(\alpha) = \left(1 - \alpha + \frac{\mu}{\mu_g}\alpha\right) \mu_g,$$

and then the new iterate is in $\mathcal{N}_\infty^-(\gamma)$ when

$$\gamma(1 - \alpha(1 + \alpha_a t)) + \frac{\mu}{\mu_g}\alpha - 2\alpha^2 n \geq \gamma \left(1 - \alpha + \frac{\mu}{\mu_g}\alpha\right).$$

One can easily see that this inequality holds for

$$\alpha \leq \frac{3\gamma}{8n},$$

that completes the proof. \square

In the modified algorithm the corrector step is calculated by solving system (14) instead of (5). In Remark 3.5 we observed that the complexity analysis allows a less aggressive update of μ i.e., $\mu = \frac{\beta}{1-\beta}\mu_g$ for $\gamma \leq \beta < \frac{1}{2}$. This observation holds for this modified variant too.

Based on the previous results, the following theorem gives the worst case iterations complexity of the modified version of Algorithm 1, where the corrector step is calculated by (14).

Theorem 4.5 *The modified version of Algorithm 1 stops after at most*

$$\mathcal{O}\left(n \log \frac{n}{\epsilon}\right)$$

iterations with a solution for which $x^T s \leq \epsilon$.

Proof: By the definition of $\mu_g(\alpha)$ we have

$$\mu_g(\alpha) = \left(1 - \alpha + \frac{\mu}{\mu_g}\alpha\right) \mu_g.$$

If $\alpha_a < 0.1$ or $\alpha_c < \frac{3\gamma}{8n}$, then the algorithm uses the safeguard strategy, and by Theorem 4.4 one has

$$\mu_g(\alpha) \leq \left(1 - \frac{3\gamma(1-2\gamma)}{8(1-\gamma)n}\right) \mu_g.$$

If $\alpha_a \geq 0.1$ and $\alpha_c \geq \frac{3\gamma}{8n}$, then the algorithm uses the Mehrotra's updating strategy, and thus one has

$$\mu_g(\alpha) \leq \left(1 - \frac{\gamma}{10n}\right) \mu_g$$

that completes the proof by Theorem 3.2 of [20]. \square

5 Superlinear Convergence

In this section we analyze the asymptotic behavior of the previous algorithms. We observe that one needs to change Mehrotra's heuristic in order to ensure superlinear convergence. The new adaptive updating strategy is defined by

$$\mu = \frac{\gamma t + \gamma(1 - \alpha_a)}{1 - \gamma} \mu_g, \quad (15)$$

where t is given by (11) and $0 < \gamma < \frac{1}{2}$. Changing Mehrotra's heuristic to the new updating strategy in Algorithm 1, while preserving the safeguard, do not change the polynomial complexity of this variant. One can analogously prove the polynomiality of these algorithms, and for simplicity those complexity proofs are omitted here.

We note that in case of monotone LCPs, Ye and Anstreicher [21] have proved that for sufficiently small μ_g the relation

$$|\Delta x_i^a \Delta s_i^a| = \mathcal{O}(\mu_g^2), \quad i = 1, \dots, n \quad (16)$$

holds that implies

$$\begin{aligned} x_i(\alpha) s_i(\alpha) &= (1 - \alpha) x_i s_i + \alpha^2 \Delta x_i^a \Delta s_i^a \\ &\geq (1 - \alpha) x_i s_i - \mathcal{O}(\mu_g^2) \alpha^2 \geq 0. \end{aligned} \quad (17)$$

Using Lemmas II.64 and II.65 of [17], one can prove that $\alpha_a \geq 1 - \mathcal{O}(\mu_g)$. In the following theorem we prove the superlinear convergence of Algorithm 1 and its modified version.

Theorem 5.1 *Let the iterate (x^k, y^k, s^k) generated by Algorithm 1 or its modified version, where μ is given by (15). When μ_g is sufficiently small, Algorithm 1 and its modified version are superlinearly convergent in the sense that $\mu_g^{k+1} = \mathcal{O}((\mu_g^k)^{1+r})$, where $r \in (0, 1)$.*

Proof: Since $|(\Delta x_i^a)^k (\Delta s_i^a)^k| = \mathcal{O}((\mu_g^k)^2)$, we have

$$|(\Delta x_i^a)^k (\Delta s_i^a)^k| \leq \mathcal{O}((\mu_g^k)^2).$$

By Mehrotra's heuristic the iterate is in $\mathcal{N}_\infty^-(\gamma)$ if

$$(1 - \alpha) x_i^k s_i^k + (1 - \gamma) \alpha \mathcal{O}((\mu_g^k)^4) \pm \alpha \mathcal{O}((\mu_g^k)^2) \pm \alpha^2 \mathcal{O}((\mu_g^k)^2) \geq \gamma (1 - \alpha) \mu_g^k.$$

We see that if both \pm are $-$ and $x_i^k s_i^k = \gamma \mu_g^k$ for $i \in \mathcal{I}$, then the previous inequality can't hold, and there is no warrantee that this can't happen, but by using the new choice of μ^k , the previous inequality certainly holds, because it is equivalent to

$$(1 - \alpha_a) \mu_g^k - \alpha \mathcal{O}((\mu_g^k)^2) \geq 0,$$

that holds by the definition of μ for $\alpha_c^k \geq 1 - \mathcal{O}(\mu_g^r)$ where $r \in (0, 1)$. Therefor, by the definition of $\mu_g^k(\alpha_c^k)$ one has

$$\mu_g^k(\alpha_c^k) = (1 - \alpha_c^k(1 - \mathcal{O}(\mu_g^k))) \mu_g^k \leq (1 - (1 - \mathcal{O}((\mu_g^k)^r))(1 - \mathcal{O}(\mu_g^k))) \leq \mathcal{O}((\mu_g^k)^{1+r}).$$

This gives the superlinear convergence of Algorithm 1 with the new choice of the parameter μ . The superlinear convergence of the modified version can also be proved analogously, that completes the proof. \square

Table 1: Comparison of Iteration Number

Problem	PMMcIPM	NMcIPM I	NMcIPM II	HMMcIPM
cycle	>150	40	40	40
80bau3b	>150	42	42	42
degen3	>150	14	14	14
ganges	26	21	20	20
perold	105	46	43	43
pilot	57	52	50	50
pilotja	>150	43	42	43
pilotwe	>150	42	42	42
pilot4	>150	35	35	37
pilot87	>150	73	68	78
pilotnov	>150	27	27	26
tuff	117	20	20	19

6 Numerical Results

In this section we report some preliminary numerical result based on our new update of the parameter μ . The results are obtained by modifying some of the subroutines of the McIPM, a Matlab based software package [24], that we ran on pentium 4, 2.53 GHZ, and 512 MB ram. In Table 1 we include some of the relatively big problems of the NETLIB test set. At the first column we have the problems name, columns 2 and 5 contain the number of iterations for the McIPM package with the original Mehrotra's predictor-corrector strategy (without any heuristics (PMMcIPM) and with the standard heuristics (HMMcIPM)); while columns 3 and 4 contain the number of iterations of the modified McIPM by our algorithms. We have tested those algorithms for various values of γ . The best result that we reported are for the case $\gamma = 10^{-4}$, while $\mu = \frac{\gamma^{\frac{1}{4}}}{1 - \gamma^{\frac{1}{4}}} \mu_g$ or $\beta = \gamma^{\frac{1}{4}}$.

We have ran our algorithm for all the NETLIB test set for LO and some test sets from Mészáros's BPMPD package web page [9]. Our limited computational results indicate the efficiency and competitiveness of our new Mehrotra type variants with the classical Mehrotra type algorithms. Extensive numerical testing is needed to assess full power of the new algorithms.

7 Final Remarks

We discussed the polynomiality of Mehrotra-type predictor-corrector algorithms. By some examples we showed that Mehrotra's heuristic might lead to an inefficient algorithm in practice. This motivated us to combine his idea with a large update strategy as a safeguard that allows us to get a lower bound for its worst case step size. Further, by slightly changing the Newton system the iteration complexity is significantly reduced. To ensure the superlinear convergence of the algorithm we changed Mehrotra's heuristic so that the new algorithm preserves the iteration complexity and exhibits stronger asymptotic convergence properties.

Further extensions of this approach to other classes of optimization problems, such as SDO, SOCO and convex nonlinear optimization, are left for the interested reader.

Acknowledgments: The research of the first and last authors is supported by the NSERC Discovery Grant DG:5-48923, the Canada Research Chair program, and by MITACS. The research of the second author is supported by the NSERC Discovery Grant DG:249635-02, a PREA award, and by MITACS.

References

- [1] K.M. Anstreicher and R.A. Bosch (1995), A new infinity-norm path following algorithm for linear programming, *SIAM Journal on Optimization*, **5**(2), pp. 236-246.
- [2] D.A. Bayer and J.C. Lagarias (1991), Karmarkar's linear programming algorithm and Newton's method, *Mathematical Programming*, **50**, pp. 291-330.
- [3] C. Cartis (2005), Some disadvantages of a Mehrotra-type primal-dual corrector interior point algorithm for linear programming, <http://www.optimization-online.org>.
- [4] C.C. Gonzaga (1999), Complexity of predictor-corrector algorithms for LCP based on a large neighborhood of the central path, *SIAM Journal on Optimization*, **10**, pp. 183-194.
- [5] P. Hung and Y. Ye (1996), An asymptotically $O(\sqrt{n}L)$ -iteration path-following linear programming algorithm that uses long steps, *SIAM Journal on Optimization*, **6**, pp. 570-586.
- [6] N.K. Karmarkar (1984), A new polynomial-time algorithm for linear programming, *Combinatorica*, **4**, pp. 373-395.
- [7] N. Megiddo (1989), Pathways to the optimal set in linear programming, In: N. Megiddo, editor, *Progress in Mathematical Programming: Interior Point and Related Methods*, pp. 131-158. Springer Verlag, New York. Identical version in: *Proceedings of the 6th Mathematical Programming Symposium of Japan, Nagoya, Japan*, 1986, pp. 1-35.

- [8] S. Mehrotra (1992), On the implementation of a (primal-dual) interior point method, *SIAM Journal on Optimization*, **2**, pp. 575-601.
- [9] C. Mészáros (1997), Linear and Quadratic Programming Test Problems, <http://www.sztaki.hu/~meszaros/bpmpd>.
- [10] S. Mizuno, M.J. Todd, and Y. Ye (1993), On adaptive step primal-dual interior-point algorithms for linear programming, *Mathematics of Operations Research*, **18**, pp. 964-981.
- [11] R.D.C. Monteiro, I. Adler, and M.G.C. Resende (1990), A polynomial-time primal-dual affine scaling algorithm for linear and convex quadratic programming and its power series extensions, *Mathematics of Operations Research*, **15**, pp. 191-214.
- [12] J. Peng, T. Terlaky, and Y.B. Zhao (2005), A predictor-corrector algorithm for linear optimization based on a specific self-regular proximity function, To appear in *SIAM Journal on Optimization*.
- [13] J. Ji, F.A. Potra, and S. Huang (1995), A predictor-corrector method for linear complementarity problems with polynomial complexity and superlinear convergence, *Journal of Optimization Theory and Applications*, **84(1)**, pp. 187-199.
- [14] F.A. Potra (2002), The Mizuno-Todd-Ye algorithm in a large neighborhood of the central path, *European Journal of Operation Research*, **143**, pp. 257-267.
- [15] F.A. Potra (2004), A superlinearly convergent predictor-corrector method for degenerate LCP in a wide neighborhood of the central path, *Mathematical Programming*, **100(2)**, pp. 317 - 337.
- [16] F.A. Potra and X. Liu (2003), Predictor-corrector methods for sufficient linear complementarity problems in a wide neighborhood of the central path, *Technical Report, Department of Mathematics and Statistics, University of Maryland, USA*.
- [17] C. Roos, T. Terlaky, and J.-Ph. Vial (1997). *Theory and Algorithms for Linear Optimization: An Interior Point Approach*, John Wiley & Sons, Chichester, UK.
- [18] M. Salahi and T. Terlaky (2004), Adaptive large neighborhood self-regular predictor-corrector IPMs for LO, *Technical Report 2004/7, Advanced Optimization Lab. Department of Computing and Software, McMaster University*, <http://www.cas.mcmaster.ca/~oplab/publication>. To appear in *Journal of Optimization Theory and Applications*.
- [19] G. Sonnevend (1985), An "analytic center" for polyhedrons and new classes of global algorithms for linear (smooth, convex) programming, In: A. Prékopa, J. Szelecsán, and B. Strazicky, editors, *System Modeling and Optimization: Proceeding of the 12th IFIP Conference*, Budapest, Hungary, September 1985, Volume 84, Lecture Notes in Control and Information Sciences, pp. 866-876. Springer Verlag, Berlin, 1986.
- [20] S.J. Wright (1997), *Primal-Dual Interior-Point Methods*, SIAM, Philadelphia, USA.
- [21] Y. Ye and K.M. Anstreicher (1993), On quadratic and $O(\sqrt{n}L)$ convergence of a predictor-corrector algorithm for LCP, *Mathematical Programming*, **62**, pp. 537-551.

- [22] Y. Zhang (1999), Solving large-scale linear programs by interior point methods under the Matlab environment, *Optimization Methods and Software*, **10**, pp.1-31.
- [23] Y. Zhang and D. Zhang (1995), On the polynomiality of the Mehrotra-type predictor-corrector interior point algorithms, *Mathematical Programming*, **68(3)**, pp. 303-318.
- [24] X. Zhu, J. Peng, T. Terlaky, and G. Zhang (2003). On implementing self-regular proximity based feasible IPMs, *Technical Report 2003/2, Advanced Optimization Lab. Department of Computing and Software, McMaster University*, <http://www.cas.mcmaster.ca/~oplab/publication>.

A Appendix

In this section we prove two technical lemmas that will be used frequently during the analysis.

Lemma A.1 *Let $(\Delta x^a, \Delta y^a, \Delta s^a)$ be the solution of (4), then*

$$\Delta x_i^a \Delta s_i^a \leq \frac{x_i s_i}{4}, \quad \forall i \in \mathcal{I}_+.$$

Proof: By equation (4) for $i \in \mathcal{I}_+$ we have

$$s_i \Delta x_i^a + x_i \Delta s_i^a = -x_i s_i.$$

If we divide this equation by $x_i s_i$ we get

$$\frac{\Delta x_i^a}{x_i} + \frac{\Delta s_i^a}{s_i} = -1.$$

Since $\Delta x_i^a \Delta s_i^a > 0$, this equality implies that both $\Delta x_i^a < 0$ and $\Delta s_i^a < 0$. Then, from

$$0 \leq \left(\frac{\Delta x_i^a}{x_i} - \frac{\Delta s_i^a}{s_i} \right)^2 = \left(\frac{\Delta x_i^a}{x_i} \right)^2 + \left(\frac{\Delta s_i^a}{s_i} \right)^2 - 2 \frac{\Delta x_i^a \Delta s_i^a}{x_i s_i} = 1 - 4 \frac{\Delta x_i^a \Delta s_i^a}{x_i s_i}$$

we get

$$\Delta x_i^a \Delta s_i^a \leq \frac{x_i s_i}{4}.$$

□

Lemma A.2 *Let $(\Delta x^a, \Delta y^a, \Delta s^a)$ be the solution of (4), then we have*

$$\sum_{i \in \mathcal{I}_+} \Delta x_i^a \Delta s_i^a = \sum_{i \in \mathcal{I}_-} |\Delta x_i^a \Delta s_i^a| \leq \frac{1}{4} \sum_{i \in \mathcal{I}_+} x_i s_i \leq \frac{x^T s}{4}.$$

Proof: Since $(\Delta x^a)^T \Delta s^a = 0$, the proof is a direct consequence of the previous lemma. □