Bulletin of the Iranian Mathematical Society Vol. 36 *No. 2 (2010), pp 99-108.*

NEW BARRIER PARAMETER UPDATING TECHNIQUE IN MEHROTRA-TYPE ALGORITHM

M. SALAHI

Communicated by Nezam Mahdavi-Amiri

Abstract. We introduce a new adaptive updating technique of the barrier parameter in the celebrated Mehrotra's predictor-corrector algorithm for linear optimization. Our new technique enables us to prove the polynomial iteration complexity of Mehrotra's algorithm without any safeguards. Encouraging computational results using Lipsol software package are reported.

1. Introduction

Variations of celebrated Mehrotra's [3, 4] predictor-corrector algorithm have been implemented in most Interior-Point Methods (IPMs) based software packages [1, 2, 10]. In [7], the authors have shown by a numerical example that a feasible version of the algorithm may be forced to make many small steps to reach optimality. Thus, they introduced certain safeguards, which allowed them to prove polynomial iteration complexity while keeping practical efficiency. In [6], the authors further analyzed this algorithm by postponing the choice of barrier parameter and proved results analogous to [7]. In this paper, we introduce a new adaptive updating technique for the barrier parameter, which allows us to prove the polynomial iteration complexity of Mehrotra's algorithm

MSC(2010): Primary: 90C05, 90C51.

Keywords: Linear optimization, interior-point methods, Mehrotra-type algorithm, polynomial complexity.

Received: 16 December 2008, Accepted: 19 August 2009.

without employing any safeguards, improving the results in [6, 7]. Our computational experiments show that the new algorithm is competitive with the heuristic based implementation in Lipsol [10].

Before going into the details of the algorithm, we give brief introduction to IPMs. Throughout the paper, we deal with the standard form of the linear optimization problem:

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

where $A \in R^{m\times n}$ with $\text{rank}(A) = m,\ b \in R^m,\ c \in R^n$. The dual of (p) is:

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

Without loss of generality [5], we may assume that both (P) and (D) satisfy the interior point condition (IPC); i.e., there exists an (x^0, y^0, s^0) such that

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

Finding optimal solutions of (P) and (D) is equivalent to solving the following system:

$$\begin{aligned} Ax &= b, \quad x \geq 0, \\ A^T y + s &= c, \quad s \geq 0, \\ xs &= 0, \end{aligned} \tag{1.1}$$

where xs denotes the componentwise (Hadamard) product of the vectors x and s. The basic idea of primal-dual IPMs is to replace the third equation in (1.1) by the parameterized equation $xs = \mu e$, where e is the all one vector. This leads to the following system:

$$\begin{aligned} Ax &= b, \quad x \geq 0, \\ A^T y + s &= c, \quad s \geq 0, \\ xs &= \mu e. \end{aligned} \tag{1.2}$$

If the IPC holds, then system (1.2) has a unique solution for each $\mu > 0$. 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 gives *the central path* of (P) and (D) [8]. 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 [5].

Applying Newton's method to (1.2) from a given strictly feasible solution gives the following linear system of equations[1]:

$$\begin{aligned} A\Delta x &= 0, \\ A^T\Delta y + \Delta s &= 0, \\ x\Delta s + s\Delta x &= \mu e - xs, \end{aligned} \tag{1.3}$$

where $(\Delta x, \Delta y, \Delta s)$ is the Newton direction. For detailed information about classical IPMs and their iterations complexity one may consult [5, 9], and the references therein.

Now, let us briefly discuss a feasible version of Mehrotra's original algorithm, which is the focus of our work here. In the predictor step, it solves the following system of equations, called affine scaling system:

$$\begin{aligned} A\Delta x^{\mathrm{a}} &= 0, \\ A^T\Delta y^{\mathrm{a}} + \Delta s^{\mathrm{a}} &= 0, \\ s\Delta x^{\mathrm{a}} + x\Delta s^{\mathrm{a}} &= -xs. \end{aligned} \tag{1.4}$$

Then, the maximum feasible step size in this direction is computed; i.e., the largest α_{a} for which $(x + \alpha_{\mathrm{a}}\Delta x^{\mathrm{a}}, s + \alpha_{\mathrm{a}}\Delta s^{\mathrm{a}}) \geq 0$. However, the algorithm does not make such a step right away. Using this information, it computes the corrector direction by solving the following system:

$$\begin{aligned} A\Delta x &= 0, \\ A^T\Delta y + \Delta s &= 0, \\ s\Delta x + x\Delta s &= \mu e - xs - \Delta x^{\mathrm{a}}\Delta s^{\mathrm{a}}, \end{aligned} \tag{1.5}$$

where μ is defined adaptively as $\mu = \left(\frac{g_{\mathrm{a}}}{g}\right)^2 \frac{g_{\mathrm{a}}}{n}$ with $g_{\mathrm{a}} = (x+\alpha_{\mathrm{a}}\Delta x^{\mathrm{a}})^T(s+\alpha_{\mathrm{a}}\Delta s^{\mathrm{a}})$ and $g = x^T s$. Since $(\Delta x^{\mathrm{a}})^T\Delta s^{\mathrm{a}} = 0$, the previous relation can be further simplified to

$$\mu = (1 - \alpha_{\mathrm{a}})^3\mu_g, \tag{1.6}$$

where $\mu_g := \frac{x^T s}{n}$.

The major feature of this algorithm, as compared to the existing ones, is that it uses one coefficient matrix in both the predictor and corrector steps. This leads to a significant computating savings for large scale problems. Moreover, in the corrector step, it uses some information

[1]We assume that one has a feasible starting point for the given problem, which can be obtained by using the self-dual embedding model [1, 5]. The infeasible case also can be carried out analogously [9].

from the predictor step, namely $\alpha_{\rm a}$ and $\Delta x^{\rm a}\Delta s^{\rm a}$, something that is not practiced in classical algorithms.

2. **Adaptive choice of barrier parameter**

Here, we introduce an adaptive way of updating the barrier parameter rather than using (1.6). To do so, we use the classical logarithmic barrier proximity measure to measure the distance from the central path and define the barrier parameter:

$$\Phi(x,s,\mu) := \frac{x^T s}{2\mu} - \frac{n}{2} + \frac{n}{2}\log\mu - \frac{1}{2}\sum_{i=1}^{n}\log(x_i s_i). \tag{2.1}$$

It is obvious that the global minimum of (2.1) as a function of μ is attain at $\mu = \mu_g$. For notational convenience, the geometric mean of the vector xs is denoted by μ_h; i.e.,

$$\mu_h = (x_1 s_1 \cdots x_n s_n)^{\frac{1}{n}}.$$

Obviously, $\mu_h \leq \mu_g$. We define the target barrier parameter as the smaller positive root of the equation

$$\Phi(x,s,\mu) = \frac{(\tau-1)n}{2},$$

where $\tau > 1$ is a given constant, denoted by μ_t. This is equivalent to:

$$\frac{\mu_g}{\mu} + \log\frac{\mu}{\mu_h} - \tau = 0. \tag{2.2}$$

In the next lemma, we give a condition under which, equation (2.2) is solvable.

Lemma 2.1. *For all $(x,s) \in R^n_{++} \times R^n_{++}$, for which $\mu_g \leq \tau\mu_h$, equation (2.2) has two positive roots, one is smaller than μ_g and the other one of which is greater than μ_g.*

Proof. If $\mu_g = \tau\mu_h$, then from equation (2.2) one has $\mu_t = \mu_h$ which is obviously less than μ_g. Moreover, since Φ is a strictly decreasing function of μ, for $\mu < \mu_g$, thus $\Phi(x,s,\mu_g) < \frac{(\tau-1)n}{2}$. Furthermore, since Φ is a strictly increasing function of μ for $\mu > \mu_g$, then (2.2) has another root that is greater than μ_g. Now, let us assume that $\mu_g = \tau_1\mu_h$, where

$1 \leq \tau_1 < \tau$. Then, we have

$$\Phi(x, s, \mu_h) = \frac{(\tau_1 - 1)n}{2} < \frac{(\tau - 1)n}{2}.$$

We also know that the value of the proximity measure goes to infinity when μ approaches zero. All these together imply that (2.2) has a solution which is strictly less than μ_h. Similar to the previous case, another root which is greater than μ_g must also exist. □

The following technical lemma plays a crucial role in our future analysis.

Lemma 2.2. *For any* $(x, s) \in R^n_{++} \times R^n_{++}$, *for which* $\mu_g \leq \tau\mu_h$, *one has* $\tau \leq \frac{\mu_g}{\mu_t} \leq 2\tau$.

Proof. Since $\mu_g \leq \tau\mu_h$, then $\Phi(x, s, \mu_h) \leq \frac{(\tau-1)n}{2}$. This together with the fact that Φ is strictly decreasing for $\mu < \mu_g$imply that $\mu_t \leq \mu_h$, and from (2.2) we have $\mu_g \geq \tau\mu_t$. Now, for the right hand side inequality, let $\mu_g = \tau_1\mu_h$, where $1 \leq \tau_1 \leq \tau$. Then, (2.2) becomes $\frac{\mu_g}{\mu_t} + \log\tau_1 - \log\frac{\mu_g}{\mu_t} - \tau = 0$, which obviously implies the right hand side inequality. □

Remark 2.3. The new adaptive choice of barrier parameter always guarantees a large update algorithm.

To discuss the polynomial iteration complexity of the algorithm using the new technique, we keep the iterates of the algorithm in the widely used negative infinity norm neighborhood as follows:

$$\mathcal{N}^-_\infty(\gamma) := \{(x, y, s) \in \mathcal{F}^0 : \ x_i s_i \geq \gamma\mu_g \ \forall i \in \mathcal{I}\}, \tag{2.3}$$

where $\gamma \in (0, 1)$ is a constant independent of n, $\mathcal{F}^0$ denotes the interior of the primal and dual feasible regions and $\mathcal{I} = \{1, \cdots, n\}$. For our analysis, we use $\gamma = \frac{1}{\tau}$. Now, we outline the algorithm using our new adaptive updating strategy.

Algorithm 1

Input:
a neighborhood parameter $\tau > 4$;
an accuracy parameter $\epsilon > 0$;
$(x^0, y^0, s^0) \in \mathcal{N}^-_\infty(\gamma)$ with $\gamma = \frac{1}{\tau}$.
While $x^T s \geq \epsilon$

Predictor Step
Solve (1.4).
Corrector step
Solve (1.5) with $\mu = \mu_t$, the smaller positive root
of (2.2) 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

In the following lemma, we show that for any iterate of Algorithm 1, equation (2.2) always has two positive roots.

Lemma 2.4. *Let (x, y, z), the current iterate of Algorithm 1, be in $\mathcal{N}_\infty^-(\gamma)$. Then,*

$$\mu_g \leq \tau \mu_h.$$

Proof. For any $(x, y, s) \in \mathcal{N}_\infty^-(\gamma)$, we have $x_i s_i \geq \gamma \mu_g$, $\forall i \in \mathcal{I}$. This implies $\mu_h \geq \gamma \mu_g$. Now, since $\tau = \frac{1}{\gamma}$, then we have $\mu_g \leq \tau \mu_h$. □

Corollary 2.5. *For all (x, y, s) generated by Algorithm 1, equation (2.2) has two positive roots.*

Proof. This follows from lemmas 2.1 and 2.4. □

The following corollary follows from Lemma 3.1 in [7] and is used in the next theorem.

Corollary 2.6. *Let μ_t be the smaller positive root of (2.2) for $(x, y, s) \in \mathcal{N}_\infty^-(\gamma)$. Then,*

$$||\Delta x \Delta s|| \leq \frac{\tau n^2}{4} \mu_g.$$

Theorem 2.7. *Suppose that (x, y, s), the current iterate of Algorithm 1, belong to $\mathcal{N}_\infty^-(\gamma)$ with $\gamma = \frac{1}{\tau}, \tau > 4$ and $(\Delta x, \Delta y, \Delta s)$ being the solution of (1.5) with $\mu = \mu_t$ as the smaller positive root of (2.2). Then, the maximum step size α_c, that keeps $(x(\alpha_c), y(\alpha_c), s(\alpha_c))$ in $\mathcal{N}_\infty^-(\gamma)$,*

satisfies

$$\alpha_{\mathrm{c}} \geq \frac{1}{2\tau^2 n^2}.$$

Proof. The goal is to find the maximum nonnegative α for which, $x_i(\alpha)s_i(\alpha) \geq \gamma\mu_g(\alpha), \quad \forall i \in \mathcal{I}$. To do so, first define

$$t = \max_{i \in \mathcal{I}_+} \{\frac{\Delta x_i^a \Delta s_i^a}{x_i s_i}\}, \tag{2.4}$$

where $\mathcal{I}_+ = \{i \in \mathcal{I} | \Delta x_i^a \Delta s_i^a > 0\}$. Since $(\Delta x^a)^T \Delta s^a = 0$, then $\mathcal{I}_+ \neq \emptyset$. The worst case may happen when $\Delta x_i^a \Delta s_i^a > 0$. Therefore we have

$$\begin{aligned} x_i(\alpha)s_i(\alpha) &= x_i s_i + \alpha(\mu_t - x_i s_i - \Delta x^a{}_i \Delta s^a{}_i) + \alpha^2 \Delta x_i \Delta s_i \\ &\geq (1-\alpha)x_i s_i + \alpha\mu_t - \alpha t x_i s_i - \alpha^2 \frac{\tau n^2}{4}\mu_g \\ &= (1-(1+t)\alpha)x_i s_i + \alpha\mu_t - \alpha^2 \frac{\tau n^2}{4}\mu_g, \\ &\geq (1-(1+t)\alpha)\gamma\mu_g + \alpha\mu_t - \alpha^2 \frac{\tau n^2}{4}\mu_g, \end{aligned}$$

where the first inequality follows from $\alpha \geq 0$, Corollary 2.6, definition of t given by (2.4) and the last inequality holds for $0 \leq \alpha \leq \frac{4}{5}$ since $\frac{1}{1+t} \geq \frac{4}{5}$, by Lemma A.1 in [7]. Moreover,

$$\mu_g(\alpha) = \left(1 - \alpha + \alpha\frac{\mu_t}{\mu_g}\right)\mu_g. \tag{2.5}$$

In order to keep the next iterate in $\mathcal{N}_\infty^-(\gamma)$, one has to have

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

which is equivalent to:

$$(1-\gamma)\mu_t - \gamma t\mu_g \geq \alpha\frac{\tau n^2}{4}\mu_g. \tag{2.6}$$

Moreover, by Lemma 2.2, one has

$$(1-\gamma)\mu_t - \gamma t\mu_g \geq \frac{\mu_g}{8\tau}.$$

Therefore, inequality (2.6) holds if

$$\frac{\mu_g}{8\tau} \geq \alpha\frac{\tau n^2}{4}\mu_g,$$

or in other words, if $\alpha \leq \frac{1}{2\tau^2 n^2}$, then (2.6) holds. Finally, we can conclude that

$$\alpha_c \geq \min\left(\frac{4}{5}, \frac{1}{2\tau^2 n^2}\right) = \frac{1}{2\tau^2 n^2},$$

which completes the proof. □

Theorem 2.8. *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. After each iteration in the direction generated by system (1.5), one has

$$\mu_g(\alpha_c) = \left(1 - \alpha_c - \alpha_c \frac{\mu_t}{\mu_g}\right) \mu_g.$$

Now using Lemma 2.2 it follows that

$$\mu_g(\alpha_c) \leq \left(1 - \alpha_c - \alpha_c \frac{1}{2\tau}\right) \mu_g = \left(1 - \alpha_c(\frac{2\tau + 1}{\tau})\right) \mu_g,$$

which completes the proof by Theorem 3.2 of [9]. □

Analogous to the Theorem 4.5 in [7], if we use the following modified Newton system for the corrector step

$$\begin{aligned} A\Delta x &= 0, \\ A^T \Delta y + \Delta s &= 0, \\ s\Delta x + x\Delta s &= \mu e - xs - \alpha_{\mathrm{a}} \Delta x^{\mathrm{a}} \Delta s^{\mathrm{a}}, \end{aligned} \tag{2.7}$$

then the iteration complexity of the new algorithm can be reduced to $O\left(n \log \frac{(x^0)^T s^0}{\epsilon}\right)$.

3. **Computational experiments**

Here we present some preliminary numerical results using Lipsol software package [10]. Lipsol is taking advantages of several heuristics in determining the barrier parameter at each iteration. To show the performance of our updating scheme, we disabled all heuristics and simply followed what we discussed in the previous section. For our approach,

TABLE 1. Comparison of iteration Numbers for some Netlib Test Problems

Problem	MLIPSOL	LIPSOL	Problem	MLIPSOL	LIPSOL
25fv47	24	25	pilotwe	37	37
80bau3b	42	39	pilot	29	31
afiro	8	8	capri	19	20
blend	12	12	brandy	17	17
bnl1	27	26	scfxm1	19	19
bnl2	33	31	scfxm2	21	21
boeing1	21	21	scfxm3	22	21
boeing2	20	19	truus	19	19
cycle	27	24	tuff	17	20
e226	20	21	woodw	29	28

we use $\gamma = \frac{1}{\tau}$ with $\tau = 100$ for all the test problems. Test problems are taken from Netlib. The results obtained by modified version of Lipsol is denoted by MLipsol and are summarized in Table 1.

As we see, our preliminary computational experiments show that our updating technique is competitive with various heuristics that are implemented in the LIPSOL software package.

4. **Concluding remarks**

We have introduced a new adaptive updating of the barrier parameter in the celebrated Mehrotra-type predictor-corrector algorithm. Our new strategy allow us to prove the polynomial iteration complexity of the algorithm without employing any safeguards. This improved on authors previous results in [6, 7]. Encouraging preliminary numerical results using Lipsol were reported.

Acknowledgments. The author thanks the referee for his/her useful comments.

REFERENCES

[1] E. D. Andersen and K. D. Andersen, The MOSEK interior point optimizer for linear programming: an implementation of the homogeneous algorithm, in Frenk H, Roos K, Terlaky T, and Zhang S, editors, High Performance Optimization, Kluwer Academic Publishers, (2000), pp. 197–232.

[2] CPLEX: ILOG Optimization, `http://www.ilog.com`.

[3] S. Mehrotra, On finding a vertex solution using inteior-point methods, *Linear Algebra and its Applications* **152** (1991) 233–253.
[4] S. Mehrotra, On the implementation of a (primal-dual) interior point method, *SIAM Journal on Optimization* **2** (1992) 575–601.
[5] C. Roos, T. Terlaky, and J. P. Vial, Interior Point Algorithms for Linear Optimization, Second ed., Springer Science, (2005).
[6] M. Salahi and T. Terlaky, Postponing the choice of the barrier parameter in Mehrotra-type predictor-corrector algorithms, *European Journal of Operational Researchs* **182** (2007) 502-513
[7] M. Salahi, J. Peng, T. Terlaky, On Mehrotra-type predictor corrector algorithms, *SIAM Journal on Optimimization* **18(4)** (2007) 1377–1397.
[8] G. Sonnevend, An " analytic center" for polyhedrons and new classes of global algorithms for linear (smooth, convex) programming, in: Prékopa, A. Szelezsán, J. and Strazicky, B. editors, *System Modeling and Optimization: Proceeding of the 12th IFIP Conference*, Budapest, Hungary, September 84, 1985. Lecture Notes in Control and Information Sciences, 866–876. Springer Verlag, Berlin, (1986).
[9] S. J. Wright, Primal-Dual Interior-Point Methods, SIAM, Philadelphia, USA, (1997).
[10] Y. Zhang, Solving large-scale linear programms by inteior point methods under the MATLAB enviroment, *Optimization Methods and Software* **10** (1999) 1–31.

M. Salahi
Department of Mathematics, University of Guilan, P.O. Box 1914, Rasht, Iran.
Email: salahim@guilan.ac.ir and salahi.maziar@gmail.com